

Áramkörök elmélete és számítása

(Előzetes jegyzet, 2003 december 28, kéziratként kezelendő)

1. Áramkörök általános jellemzői

Jelek: feszültség (uV, mV, V, kV), áram (pA, nA, mA), idő (ps, ns, us, ms), töltés, fluxus, teljesítmény.
Az áramkör méretei jóval kisebbek, mint az aktuális elektromágneses jel hullámhossza, ezért koncentrált (lumped) paraméterekkel számolunk.

Fénysebesség: 3,3ps / 1mm. 2,5kHz – 12 km, 2,5 MHz – 12m, 2,5GHz – 12 cm, 25 GHz -12mm

Építőelemek: feszültség és áramgenerátor (rajzok), RLC elemek, vezérelt generátorok, kapcsolók

Kapacitás: $i(t)=C dU(t) / dt$; Induktivitás: $u(t)=L di(t) / dt$; Ellenállás: $u(t) = R i(t)$

Ha nemlineáris: $u(t) = R ((i(t)))$; Vezetés: $G = 1/R$ nemlineáris elem pl. dióda

Kirchoff áramtörvénye: a csomópontokban ki/befolyó áramok összege zérus.

Kirchoff feszültségtörvénye: zárt hurokban a feszültségek összege zérus.

Egykapu: kapocspár...referencia irányok...! Hárompólus (3-terminal element): rajz; n-terminal element: rajz

Transzformátor és gráfja; irányított gráfok

Gauss felület (zárt térrész, amelynek felületén a diff.hányadosok folytonosak)

Vágat: ágak halmaza, amelyek mindegyikét elvágva a gráf szétesik

Incidencia mátrix (összeköttetés mátrix): sorai a csomópontok, oszlopai az ágak. Mátrix eleme zérus,

ha az illető csomópont nem érintkezik az ággal., ha igen, akkor iránytól függően +1 vagy -1.

Kirchoff feszültségtörvénye mátrixalakban: $u=M*e$, ahol $u=[u_1, u_2, ..u_b]^T$ az ágfeszültségek vektora,

$e=[e_1, e_2, ..e_{n-1}]^T$ a csomóponti feszültségek vektora

Fa: részgráf, amely minden csomópontot tartalmaz, de nem tartalmaz hurkot.

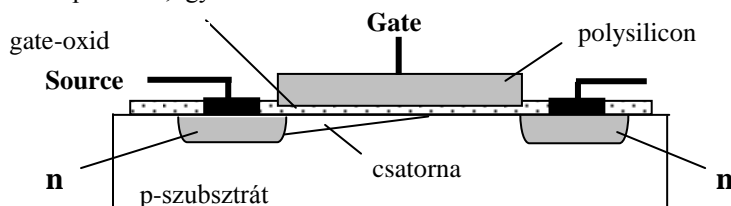
Tellegen tétel: a hálózat minden elemére nézve az azon levő feszültség és az azon átfolyó pillanatnyi áram szorzata (vagyis a pillanatnyi teljesítmény) összege zérus, másszóval az aktív (teljesítményt leadó) és passzív (teljesítményt felvevő, disszipáló) elemek egyensúlyban vannak. Egyes elemek (mint pl. kapacitások, induktivitások) időszakosan lehetnek aktívak és passzívok is (lásd energiátárolás kapacitásban).

2. Áramköri elemek

Passzív áramköri elemek, R, C, L (nehezen megvalósítható). Aktív elemek: térvézrelt (MOS) és bipoláris tranzisztor.

MOS tranzisztorok működésének alapjai. A mikroelektronika (és ezen keresztül, mondhatni, az egész elektronika) alap-eszköze a MOS tranzisztor, amelynek keresztmetszetét egy n-csatornás eszközre az ábra mutatja be. Az eszköz működésének lényege a vékony szigetelőréteg (gate-oxid) alatt létrejövő, az U_{GS} gate-feszültséggel vezérelhető csatorna. Ez a csatorna hoz létre áramvezető utat az n-adalékolt (n+) source- és a drain rétegek között. A tranzisztor küszöbfeszültsége alatt az eszköz lezár, áram nem folyik. Ezen érték felett a draináramot statikusan az $I_D = f(U_{GB}, U_{SB}, U_{DB})$ függvény adja meg, ahol a zárójelben mennyiségek rendre a gate-, a source-, a valamint a drain réteg feszültsége a szubsztrát (bulk) réteghez képest.

A MOS tranzisztorok legfontosabb jellemzői a küszöbfeszültség, a draináram értékét meghatározó áramkonstans, a parazita kapacitások értéke, a szivárgások és természetesen a feszültségbírási. Mivel VLSI áramköröknél egy chipen akár több száz millió tranzisztor is elhelyezkedhet, ezért ezek fogyasztása alapvető fontosságú. Logikai áramkörök esetében a fogyasztás (a ma használatos áramkörök esetében) alapvetően a kapacitások feltöltéséből adódó disszipációból, így ezek értéke rendkívül kritikus.



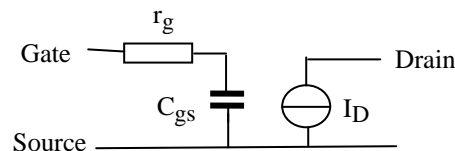
A drain-áramot az alábbi kifejezések adják meg:

$$I_D = K \left[(U_{GS} - V_T) \cdot U_{DS} - \frac{U_{DS}^2}{2} \right] \quad \text{ha: } U_{DS} \leq (U_{GS} - V_T)$$

$$I_D = \frac{K}{2} (U_{GS} - V_T)^2 \quad \text{ha: } U_{DS} \geq (U_{GS} - V_T)$$

ahol az első kifejezés az ún. triódatartomány, a második a telítéses (vagy elzáródásos) tartományra vonatkozik, továbbá V_T a küszöbfeszültség. Az összefüggések láthatóan nem veszik figyelembe a szubsztrát-feszültség hatását, és érvényességük alapfeltétele, hogy az U_{GS} gate-feszültség a küszöbfeszültségnél nagyobb legyen. A telítéses tartományban – mivel az áram független az U_{DS} drain-feszültségtől – lényegében egy áramgenerátort kapunk, aminek közelítő jellege azonnal kiderül, ha egy CMOS inverter kimeneti feszültségének statikus meghatározását kísérelnénk meg.

MOS tranzisztor helyettesítőképe.



Planár technológia. A technológia lépései: fotoreziszt anyaggal való bevonás, fotomaszk, megvilágítás, ablak a rezisztanyagban, oxid-marással ablakok létrehozása, ablakon keresztül adalékolás (p- vagy n).. Ezután megmaradt oxid eltávolítása, új reziszt, stb.

MOS technológiai fejlesztések. A mikroelektronikai kutatások és fejlesztések célja egyre nagyobb komplexitású és egyre gyorsabb áramkörök előállítása. Ennek egyik útja az elemek méretének csökkenése (scale down), ami azt jelenti, hogy az eszközök (tranzisztorok) minden mérete az eszközfizikai összefüggések által megszabott kereteken belül lecsökken. A méretcsökkentés legfőbb korlátja az ábrának a maszkolási műveletek során történő fotolitográfiai leképezése ill. ennek felbontása, de kritikusak egyes technológiai műveletek (rétegépítések, rétegmarások) felbontása is.

A méretek csökkenésével kisebb értékű terhelőkapacitások jönnek létre, ami alapvető fontosságú a sebesség szempontjából, hiszen alapvetően ezek határozzák meg az egyes kapuk, tárolók, stb. késleltetési idejét. A kapacitások értékének lecsökkenése elvben jótékony hatással van az áramkörök által disszipált teljesítményre is, ezt azonban általában nem érzékeljük közvetlenül, hiszen az alkalmazók a lehetséges legnagyobb sebességgel kívánják működtetni az áramkört, ami viszont teljesítmény-növekedést eredményez. Így lényegében azt mondhatjuk, hogy az évtizedek óta tartó scale-down nemhogy csökkentette volna, hanem inkább drasztikusan megnövelte a chippek által felvett teljesítményt – igaz, a nyújtott áramköri komplexitás ugyancsak drasztikus megnövekedése mellett.

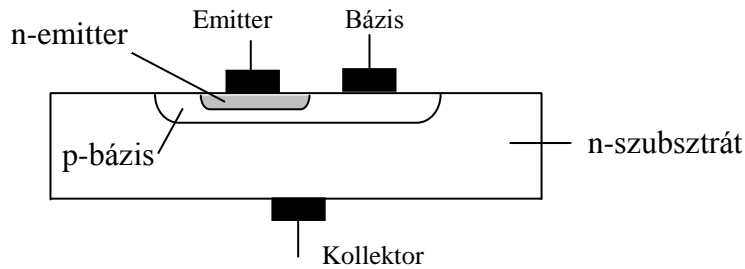
A méretek csökkentése a chip felületén elhelyezhető tranzisztorok számát is lényegesen megnövelte. Az évtizedekkel felállított ún. Gordon-Moore törvény azt jósolta, hogy a technológiai fejlődés üteme aolyan mértékű lesz, hogy az egy chipen elhelyezkedő elemek száma (amit majd a memória-áramkörökön mérnek) évente megduplázódik.. Ez a növekedési sebesség, ha valamivel lelassult formában is, de ma is érvényben van (jóideje két évente háromszorozódik az elemszám).

A méretek csökkentésével párhuzamosan csökken az alkalmazott tápfeszültség is, és rövidesen a $V_{cc}=1V$ alatti áramkörök lesznek általánosak. Ez egyrészt négyzetes arányban csökkenti a disszipációs teljesítményt, másrészt ellene dolgozik annak, hogy a kis méretek miatt nagy elektromos térerősség-érték lépjenek fel a eszközben, ami a sor paraméternél döntő fontosságú. A korábbi jóslatok, amelyek a feszültség-csökkentés hatására bekövetkező zavar-érzékenységtől tartottak, nem igazolódtak be, elsősorban az áramkörök mikroméretei következtében.

A sebességben rendkívül igénybevett processzorok által disszipált hő elvezetése ma a mikroelektronika egyik legégetőbb problémája. Egyre közelebb kerül a vízűtés általános bevezetése, de szó van különleges esetekben a folyékony nitrogénnel való hűtésről is.

A bipoláris tranzisztor. A MOS tranzisztorok kis árama igen előnyös a komplex áramkörök nagy elemszáma szempontjából, viszont előnytelen ott, ahol nagy terhelőkapacitásokat kell rövid idő alatt feltölteni vagy kisütetni. Ilyen esetekben előnyösen alkalmazható a bipoláris tranzisztor, amelynek vertikálisan folyó árama igen nagyértékű lehet. A

diszkrét bipoláris tranzisztor keresztmetszetét az ábra mutatja be – ennek integrált formája annyiban más, hogy a kollektor egy további réteggel el van szigetelve a szubsztráttól és kivezetése is fent, a felületen van.



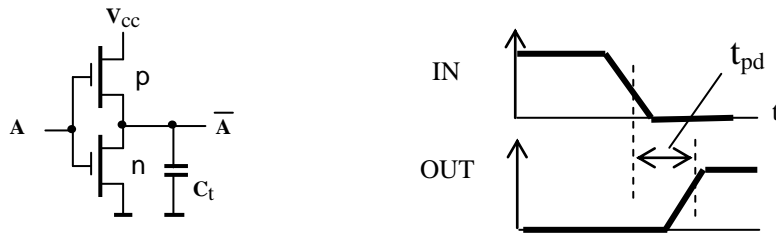
3. Logikai áramkörök

Áramkörök típusai a) logikai, b) analóg, c) mixed-mode („analógikai”).

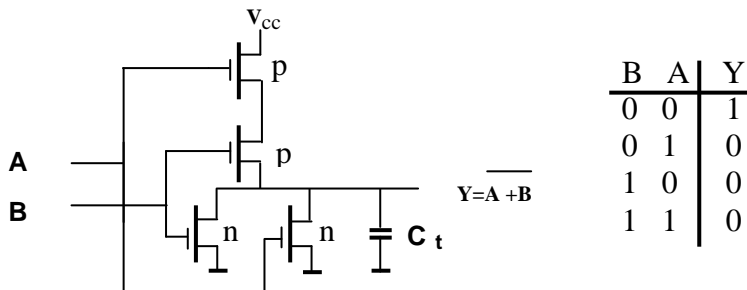
A ma használatos VLSI áramkörök általában az alábbi öt logikai rendszer alapján épülhetnek fel: statikus CMOS logika, dinamikus CMOS logika, transzfer gate-s logika, bipoláris emittercsatolt (ECL) logika, mint az előzőek kiegészítője, BiCMOS logika. A korszerű áramkörök ezen rendszerek bármelyikét alkalmazhatják, a feladattól függően a legkedvezőbb megoldást kiválasztva ill. ezek kombinációját.

Tervezésük: időtartományban, SPICE programmal: elemi időlépésekre bontva: $C \Delta u = I \Delta t$. Worst case vizsgálatok. A frekvencia-tartományba transzformálás: Fourier transzformáció.

Inverter. A logikai kapcsolások alapeleme az inverter. A lehúzó (pull-down, driver) tranzisztorral szemben a munkaellenállás (load, pull-up) húzza fel a kimenetet a telep felé, amely a statikus logikában egy a bemenetről vezérelt pMOS tranzisztor. Dinamikus logikáknál egy előtöltési fázisjellel vezérelt pMOS helyettesíti a munkaellenállást.



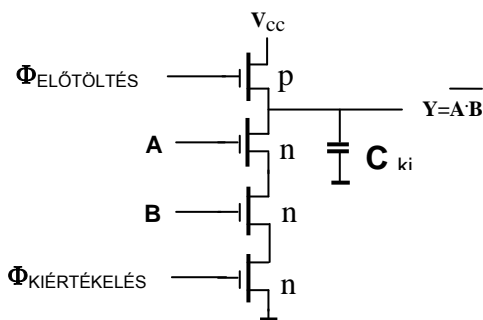
Statikus CMOS logika. A logikai áramkörök uralkodó technológiája napjainkban a CMOS technika, ezen belül is a statikus CMOS logika. Előnye a jó tervezhetőség és a időfüggetlen működésből adódó alkalmasság arra, hogy nagy és univerzálisan felhasználható cellakönyvtárat hozzunk létre. Hátránya viszont, hogy a terhelő-ellenállás helyén az nMOS elemekkel megvalósított logikai függvény teljes duál hálózatát meg kell valósítani ahhoz, hogy semmilyen kombinációban se folyék keresztirányú áram. Emiatt a tranzisztorszám megduplázódik, ami a megnövekedett helyfoglalás mellett sebességsökkenéssel is jár, mivel minden bemenet két gate-kapacitás terhelés jelent a meghajtás számára. Ha nincs logikai szintváltás, akkor nem folyik áram, nincs teljesítmény-felvétel. Késleltetés (propagation delay).



*Statikus CMOS logika,
2-bemenetű NOR*

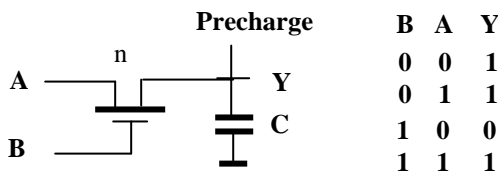
Dinamikus CMOS logika. A logikai áramkörök megvalósításának egy másik lehetősége a dinamikus logika. Ezeknek közös jellemzője, hogy a működés két fázisra oszlik, amelyet a fázisjelek vezérelnek. Az első fázis az előtöltés, amelynek során a belső (lényegében parazita eredetű) C_l kapacitást egy pMOS tranzisztor a V_{cc} tápfeszültségre tölti fel, majd a pMOS lezárása után az ide betöltött töltés, mint logikai UH szint, magára marad, lebeg. Nem sokkal ezután egy másik fázisjel megkezdja a logikai függvény kiértékelését (a 1.3.2 ábrán egy kétbemenetű ES kapcsolatot mutatunk be), amelynek során a C_{ki} kapacitás vagy teljes egészében kisül a föld felé, vagy feszültsége

közeliítőleg változatlan marad (amennyiben eltekintünk a továbbiakban tárgyalt ún. töltésmegosztás, charge-sharing jelenségtől).

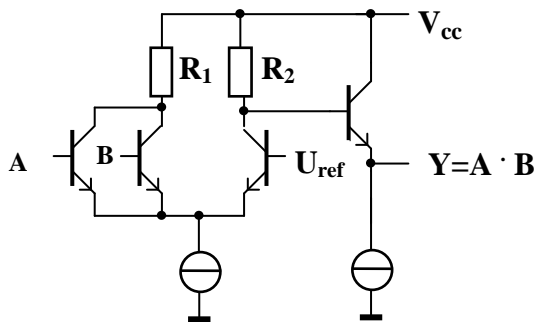


*Dinamikus CMOS logika,
2-bemenetű NAND kapu*

Transzfer-gates logika. A logikai függvények megvalósításának egy további formája az ún. transzfer-gates logika, amelyet az ábra mutat be. Könnyű észrevenni, hogy itt azáltal, hogy a source-ban is történik vezérlés, a tranzisztorok sokkal jobban kihasználhatók, ami a lecsökkentett tranzisztorszámban jelentkezik. Ennek az ára viszont az, hogy a korábbi két megoldással ellentétben itt a vezérléshez áram kell (mivel a source-ön is van jel), ami a meghajtás oldalán jelent többlet nehézséget.



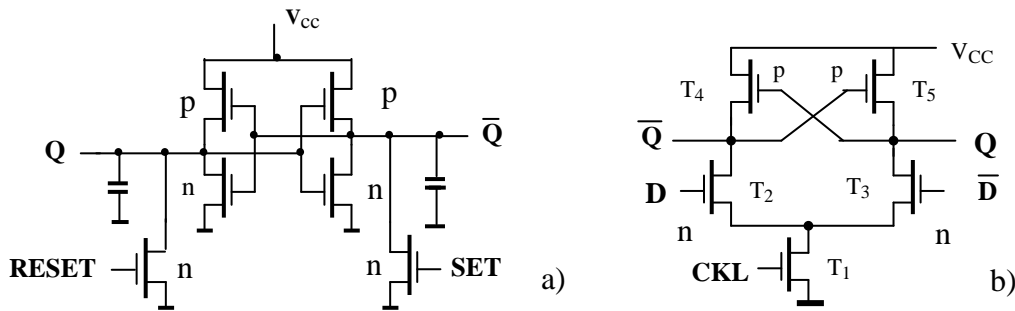
Emittercsatolt logika. Az emittercsatolt logika bipoláris tranzisztorokból épül fel a 2.4 ábrán látható séma szerint, amely egy 2-bemenetű NOR kaput mutat be. Az elrendezés nagyon emlékeztet egy differenciálerősítőre, a lényeges eltérés azonban, hogy itt az áramgenerátor teljes árama mindig csak az egyik ágban folyik, míg a másik ág lezár. A jelen esetben az *A* ill. *B* bemenetre logikai "1" szintet adva az áram a baloldali ágban fog folyni, a referencia-ág lezárva az R_2 munkaellenálláson feszültség nem esik s így az emitterkövető bázis-feszültsége V_{cc} lesz. A rendszer egy ún. "nem telítéses" bipoláris rendszer, abból adódóan, hogy a munkaellenállások helyes megválasztásával egyik tranzisztor kollektor-bázis átmenete sem nyithat ki (ami a telítéses üzemmód és ezzel a jelentős többlet-késleltetés okozója). A logikai szint-tartomány (swing) növelése érdekében a kollektor-bázis feszültséget kismértékben (0,3-0,4Vértékig) engedik nyitóirányba fordulni, de itt számottevő nyitóirányú áram az átmeneten nem folyik s így a telítési hatás elhanyagolható. Ezzel együtt a rendszer logikai szint-tartománya rendkívül alacsony, általában 0,6-0,9V körül van, de ez a mai technológiákkal megvalósítható nagyon kis paraméter-szórásoknak köszönhetően nem jelent problémát. Az emitterkövetőre csak a jelszintek helyreállítása végett (kompatibilitás a következő fokozat bemenetével) van szükség. A kapcsolás láthatóan nyugalmi állapotban is áramot vesz fel, fogyasztása tehát jelentős.



Emittercsatolt logika (ECL)

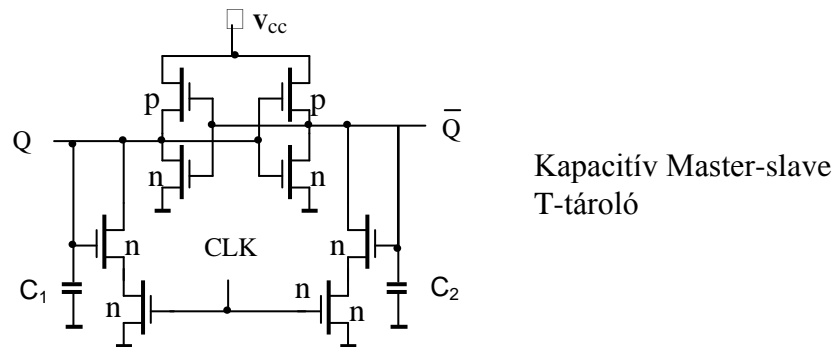
Az emittercsatolt logikák fő alkalmazási területe VLSI áramkörök egyes olyan kiemelt áramköri részletei, ahol a sebesség rendkívül kritikus, de ugyanakkor a bonyolultság nem túl nagy ahhoz, hogy a kapcsolás fogyasztása elviselhetetlenül megnöjjön.

Statikus RS-tároló. Statikus inverterekből felépített RS-tárolót mutat be a 2.3.1 ábra, ahol a Set és Reset állapotokba billentő lehúzó tranzisztorokat időfüggetlenül kapcsoljuk rá a CMOS flip-flop-ra. A két jel értelemszerűen nem lehet egyidőben logika "1", mivel ez határozatlan állapotot eredményez. A kapcsolás átbillentése nyers erővel (brute force) történik, ami annyit jelent, hogy a billentés kezdő szakaszában (pl. RESET alkalmazásakor) a C1 kapacitást egyidőben sűti ki T1, míg a flip-flop felső pMOS eleme pedig felfelé húzva tölti azt. Megfelelően nagyáramú T1 esetén az átbillentés megtörténik, de közben jelentős idő- és teljesítmény-vesztés lép fel.



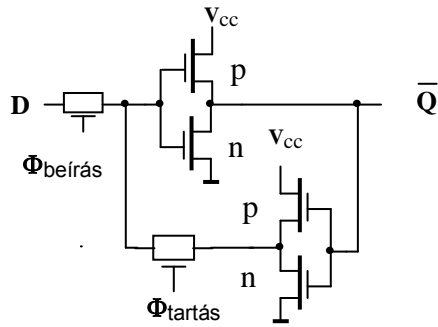
a) statikus RS-tároló, b) órajellel vezérelt statikus D-tároló

Kapacitív tárolású T-tároló. Master-slave T- (Toggle) tárolót mutat be a 2.3.2 ábra, amelynél a slave-funkciót a C1 és C2 parazita kapacitások valósítják meg. A tároló minden CLK órajel impulzusra állapotot vált. Mint ismeretes, a master-slave tároló lényege, hogy a master (amely itt egy szokványos CMOS flip-flop) billentése alatt a fenti két kapacitás tartja a flip-flop előző állapotát, és csak az órajel visszafutása után íródik be az új állapot. A flip-flop mindig abba az irányba billen (sűtődik ki zérusra), amely oldalon a tároló kapacitások logikai "1"-ben van (ez a korábbi állapot) és ezáltal nyitja a kisütő ágban levő nMOS tranzisztort.



Kapacitív Master-slave T-tároló

Kvázi-statikusan D-tároló. Transzfer gate-ekkel megvalósított kvázi-statikusan D-tárolót mutat be a 2.3.3 ábra, amely lényegében két statikus CMOS inverterből áll. A tartás fázisában a két inverter egy szokványos statikus flip-flop-ot képez, amely mindaddig tárolja az információt, ameddig $\Phi_{tartás} = 1$. A beírás alatt a tartás megszűnik (a két fázisjel tehát nem lapolja át egymást) és az új adat (D) "nyers erőszak" nélkül tárolódik be a C1 kapacitásba. Az egyszerűség végett a $\Phi_{tartás}$ jel általában a $\Phi_{beírás}$ jel negáltja, de ilyenkor a negálást végző inverter késleltetése miatt a tartás megszűnése késik a beírás bekapcsolásához képest. Ebből következően, ha az új adat ellentétje a korábbinak, akkor a C1 kapacitásra a két transzfer gate-en keresztül két ellentétes jel kényszerítődik. Ennek kivédésére a flip-flop második inverterét kisebb áramúra méretezik, hogy az új adat ellenhatás nélkül íródjék be. Ez az egymás ellen dolgozás természetesen azonnal megszűnik, amint a tartást végző transzfer gate lezár, de értelemszerűen a gyors működés érdekében a fázisjelek hosszát minél rövidebbre kell választani és ezért még ez a kezdeti átlapolásból adódó problémát is célszerű kiküszöbölni.

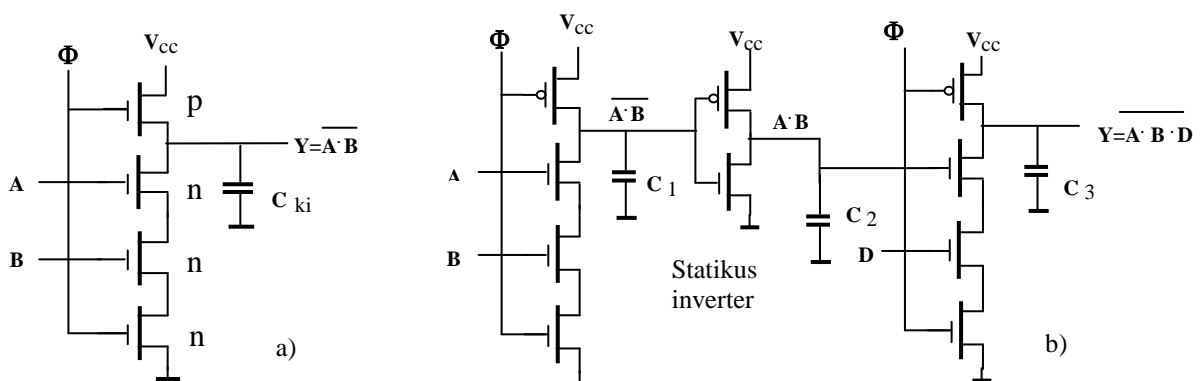


Kvázi-státikus D-tároló

Dinamikus, egyfázisú logika. Mint később látni fogjuk, a fázisjelek előállítása és szétosztása a chipen sok problémát vet fel. Éppen ezért célszerű a lehetőségekhez mérten a fázisjelek számát alacsonyan tartani. Ezen cél érdekében valósult meg az egyfázisú CMOS logika, amelynek elvi vázlatát egy 2-bemenetű NAND kapura a 2.4.1 ábra mutatja be. A Φ fázisjel alacsony szintjén a pMOS tranzisztor előtölt a Cki (általában csupán parazita) kapacitást (Precharge), majd a fázisjel magas értékénél történik az A és B logikai változók kiértékelése (Evaluate). Ezek értékétől függően Cki kisül zérusra vagy feltöltve marad (rail-to-rail működés).

A kimeneti magas szint kapcsán azonban szót kell ejteni két olyan másodlagos effektusról, amely a dinamikus működésű áramkörök közös problémája. Mindkettő abból adódik, hogy az előtöltés befejeztével a feltöltött Cki kapacitás magára van hagyva, lényegében lebeg, tehát feszültségének utólagos korrekciójára már nincsen lehetőség.

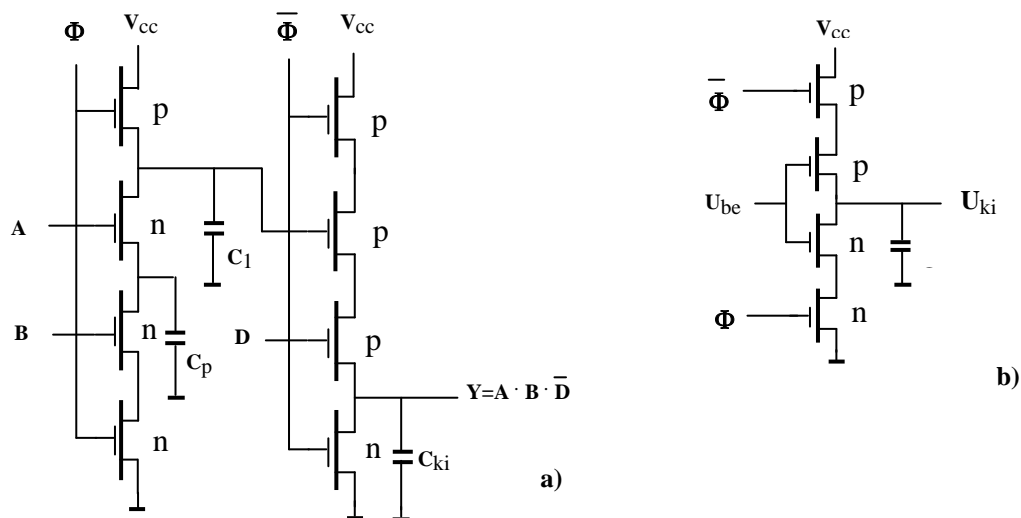
Az első effektus a *töltésmegosztás* (charge sharing), amely akkor lép fel, ha $A=1$ és $B=0$, továbbá, ha a kiértékelés kezdetén a C_p parazita kapacitás ki van sűtve. Ilyenkor a V_{cc} -re feltöltött Cki kapacitás és C_p között töltésmegosztás jön létre, aminek hatására a kimeneti magas szint lecsökken, aminek mértékét a kapacitások aránya határozza meg. Ne feledjük azonban, hogy a valóságban egy bonyolult logikai áramkör esetén (az ábrán látható két-tranzisztoros kapcsolással szemben) igen sok olyan parazita kapacitás adódik, amelyre a töltésmegosztásra vonatkozó feltételek teljesülnek, és amelyek kapacitásának összege nagy valószínűséggel összemérhető lesz Cki-vel és ennek eredményeképpen a kimeneti jelszint lényegesen kisebb lesz, mint V_{cc} . Ez érthető módon korlátozza az ide beilleszthető logikai hálózat bonyolultságát és mindezeneken felül egy rendkívül alapos tranziens szimulációs vizsgálatot igényel, figyelembe véve az összes *worst case* kombinációt, az adott lépést *megelőző* logikai állapottal együtt.



Domino CMOS logikák a) egyfokozatú 2-bemenetű NAND kapu, b) többfokozatú inverteres kapu

A másik effektus a *feszültség-beccsátolás* által okozott változás (charge corruption), amely a Cki kapacitáshoz közel haladó vezetéseken (pl. órajel-vezetékeken) előálló 1-0 átmenetek által becsátolt feszültség ugrás. Ha a chip layout-ján a két pont közötti kapacitás a Cki értékhez viszonyítva nem elhanyagolható, akkor ezen kapacitív osztó által meghatározott ΔU feszültséggel csökken a kimenet. Ez igen veszélyes akkor, ha a követő (meghajtott) fokozat éppen ekkor olvassa ki a kimenetet, és ezen nem segít az, hogy a zavarójel visszafutásakor a 0-1 átmenet helyreállítja a kimenet feszültségét.

Többfokozatú egyfázisú dinamikus logikák. Az ábrán látható fokozat kimenetére elvben csatlakoztathatunk egy másik, ugyanazzal a fázisjellel meghajtott fokozatot s ílymódon látszólag többfokozatú áramköröket építhetünk fel. A valóság azonban az, hogy az ilyen közvetlen összekapcsolás nem megoldható a kiértékelés kezdetén fellépő tranzien্স állapot miatt. Ha ugyanis a második fokozat elötöltött C_{ki2} kapacitását a logikai függvény szerint nem sűtjük ki, viszont



a) Alternáló” típusú Domino CMOS logika, b) C^2 MOS latch tároló

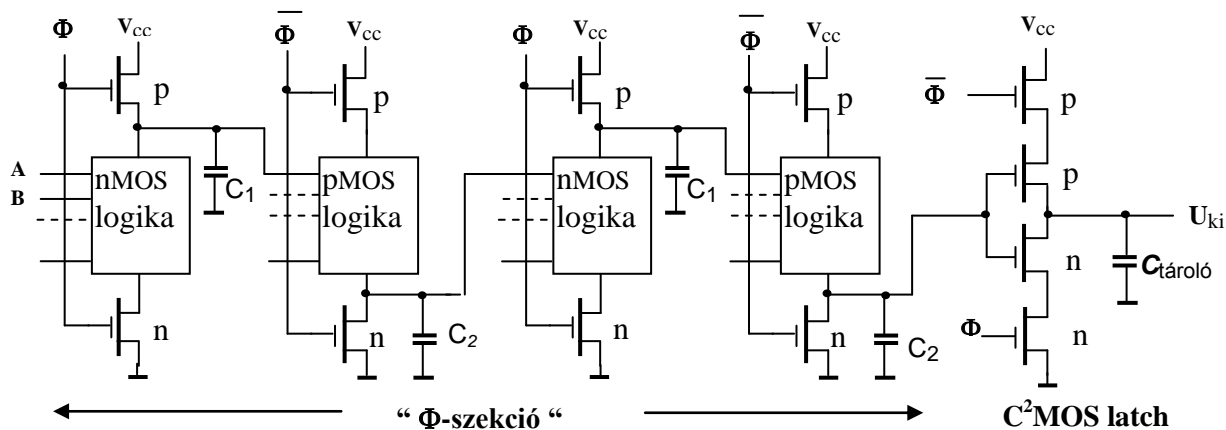
az előző fokozatét igen, akkor a kiértékelés kezdetén (a fázisjel 0-1 átmenetét követően) a második fokozat nMOS tranzisztora (amely az első fokozat C_{ki} kapacitására csatlakozik) átmenetileg nyitófeszültséget kap. Ezen idő alatt C_{ki2} sűl ki, mindaddig, míg C_{ki} feszültsége nem csökken le a küszöbfeszültség alá, és ezen idő alatt C_{ki2} feszültsége helyre nem hozható módon megsérül. Ez a látszólag nagyon kecsgetető közvetlen összekapcsolást lehetetlenné teszi.

A probléma egyik lehetséges megoldása az, hogy pl. statikus invertereket iktatunk be a két fokozat köz, amit az ábra mutat be. Ez a megoldás lényegében a statikus és dinamikus rendszer kombinálása, ami egy igen biztonságos megoldást eredményez,, viszont megnöveli a késleltetési időt.

Alternáló fokozatok alkalmazása. A statikus invertereknél lényegesen kedvezőbb megoldás az ún. alternáló fokozatok alkalmazása, amelyet a 2.4.3 ábra mutat be. Itt aztal, hogy a második fokozat p-csatornás logikát tartalmaz, a kiértékelés kezdetén (C_1 -en kötelezően) fellépő logikai “1” a pMOS számára zárófeszültség, tehát C_{ki} átmeneti kisűtése nem indul meg.

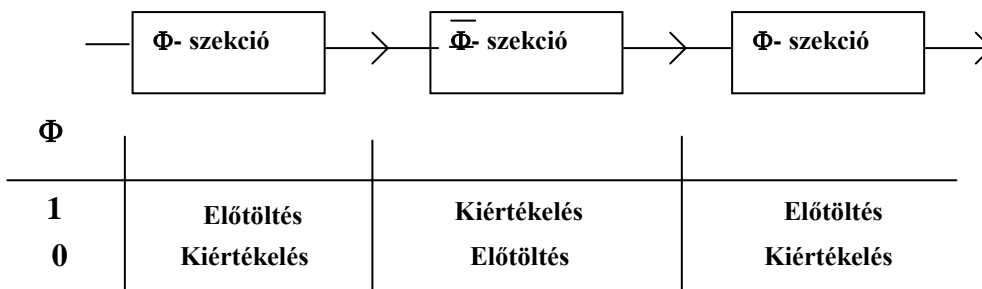
Domino CMOS hálózat. Képzeljük el ezek után több, azonos fázisjellel vezérelt fokozat megfelelő összekapcsolását (2.4.4 ábra). Ez végül is egy olyan aszinkron hálózatot eredményez, amelyben az egyes fokozatok (“oszlopok”) kiértékelése rendre történik, és mindegyik oszlop kiértékelésére csak a megelőző oszlop érvényes kimeneti adatainak birtokában kerül sor. Innen adódik az áramkör Domino CMOS elnevezése. A teljes hálózat késleltetési idejét az aszinkron működésből következően a fokozatok késleltetésének összege szabja meg. Az áramkör tehát gyors, de természetesen előfeltétel, hogy minden bemeneten az adott időben érvényes jelnek kell lennie.

C^2 MOS latch. A fent részletezett aszinkron áramkörnek egy szinkron hálózatba való beillesztéséhez egy átmeneti tárolóra van szükség. Ennek egy igen népszerű megvalósítását, a Clocked CMOS tárolót mutatja be a 2.4.5 ábra. A kapcsolás lényegében egy statikus CMOS inverter, amelyet egy beágyazott transzfer gate fog közre. A $\Phi=0$ esetben az inverter föld- és tápvezetéke meg van szakítva, a leválasztott $C_{tároló}$ tehát tartja a beírt adatot. A $\Phi=1$ esetben az inverter működésbe lép és a bemeneti adat negáltját a tároló kapacitásba írja.

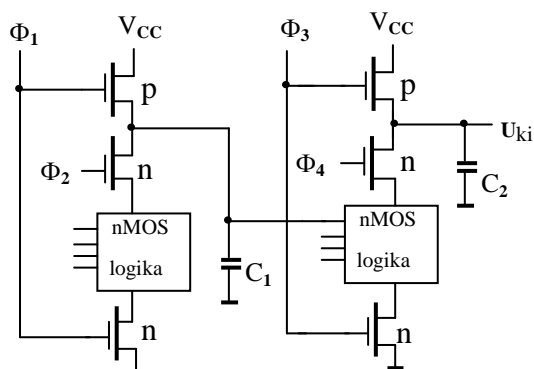


Alternáló Domino CMOS logika kimeneti latch-el.

Pipeline Domino CMOS logika. Az ábrán látható többfokozatú áramkört (Φ -szekciót) a Φ -jel negáltjával vezérelve egy olyan áramkörhöz jutunk, amelyben időben éppen fordítva történik meg az előtöltés ill. a kiértékelés. Ez a tény lehetőséget ad a két áramkör sorbakapcsolásával pipeline hálózatok létrehozására, amit 2.4.6 ábra mutat be. Az egyes szekciók felváltva követik egymást és a táblázatnak megfelelően felváltva hajtják végre az előtöltési ill. kiértékelési fázist. Mindegyik kimenetén egy C^2 MOS latch van, amelyben az aszinkron módon érkező adatok bevárják a következő fázis kezdetét és csak akkor íródnak át a következő szekcióba.



A klasszikus négyfázisú logika. A fázisok számának további növelésével olyan logikai áramkörök hozhatók létre, amelyeknél lényegében minden másodlagos, töltésmegosztást eredményező effektus kiküszöbölhető. Ennek egy sokrétűen használt formája a klasszikus négyfázisú logika (2.4.10. ábra). Ennek lényege legjobban azzal jellemezhető, hogy a kiértékelő ágakba helyezett közbenső, kiegészítő fázisjelekkel vezérelt tranzisztorok tökletes elválasztást biztosítanak a következő fokozat felé, amíg az átmenetileg még érvénytelen jelek nemkívánt töltés-átrendeződést eredményeznének.



Klasszikus négyfázisú logika

A kapcsolásnál alkalmazott fázisjelek időbeli lefutását az ábra mutatja be. Mint látható, a jobb időkihasználás végett itt egyes fázisjelek átlapolják egymást. Az átlapolás tényének és formájának egyértelmű leírására vezeték be a minor-minor, minor-major, major-minor ill. major-major meghatározásokat, annak megfelelően, hogy a rövidebb (minor) és hosszabb (major) fázisjelek egymáshoz hogyan illeszkednek.

Az ábrán láthatóan a fázisjelek optimális megválasztása folytán egy pipeline-szerű működés áll elő, amelynél egy-egy fokozat előtöltése megkezdődik, amikor az előtte levő fokozatban a kiértékelési fázis még tart. Az ábrán ELŐ az előtöltést és KIÉ a kiértékelési fázist jelöli,

A fogyasztás csökkentésének lehetőségei. Méretcsökkentés. Az elem-méreték csökkentése értelemszerűen csökkenti a parazita kapacitások értékét és ezzel a töltőáramokat is. Ennek ellenére, a várható trend, nevezetesen, hogy a scale-down segítségével a chip-ek fogyasztása csökkenni fog, nem vált be, sőt a melegezés még tovább fokozódott, a méretcsökkentésből adódó nagyobb működési sebesség kihasználása folytán. Mindezek mellett a chip méretének növekedése is rendkívül lelassult (szinte megállt), ami szintén rontja a hűtési viszonyok javulási esélyeit. Az irodalom ezért a méretcsökkentésből magából adódó fogyasztás-csökkenéssel egyáltalán nem számol és ezért más utakon keresi a megoldást.

Tápfeszültség (V_{dd}) lecsökkentés. Ez módszer lényegében hasonló az előzőhöz, de itt a kapuk lassítását ill. ezzel együtt a fogyasztás csökkenését egy további, második tápfeszültség alkalmazásával érjük el. Hogy az ebből adódó logikai-szint problémák minél kevésbé legyenek zavaróak, ezért ezt a beavatkozást nagyobb egységekre érdemes elvégezni, amelyek kimenetén a szintek helyreállításáról természetesen gondoskodni kell.

Szigetelő-anyagú hordozó. A parazita-kapacitások lényeges lecsökkenését eredményezi a zafír-alapú (SOI) áramkörök konstrukciója, amely azonban jelenlegi drága előállítási költsége miatt most elsősorban csak professzionális (katonai, űrkutatási) alkalmazásokra korlátozódik, de az előrejelzések szerint 2005 után egyre szélesebb körben jelenik majd meg a közfogyasztású elektronikában is.

4. Aritmetikai és tároló áramkörök.

Összeadók. Általános összefüggések. Az összeadók alapeleme a kétbites, átvitel is kezelő kapcsolás, amely az összeg és átvitel értékét állítja elő a következő összefüggés alapján:

$$S_{i+1} = A_i \oplus B_i \oplus C_i = A_i B_i C_i + \bar{A}_i B_i C_i + A_i \bar{B}_i C_i + \bar{A}_i \bar{B}_i C_i$$

$$C_{i+1} = A_i B_i + (A_i + B_i) \cdot C_i = G_i + P_i C_i$$

ahol A_i és B_i az összeadandók, C_i az átvitel, továbbá $G_i = A_i B_i$ és $P_i = A_i + B_i$ az ismert generate és propagate mennyiségek. A fenti összefüggéseket átalakítva azonban számos felírási módhoz jutunk. Ezek közül az invertáló logikai rendszerekkel megvalósított kapcsolások eleve más kifejezésekből indulnak ki, hogy a szükségszerűen jelenlevő inverziót beépíthessék. Egy további szempont, hogy az elemszámot, a minimális végrehajtási időt, vagy a minimális teljesítmény-felvétel legyen a cél.

Domino CMOS logikával megvalósított összeadó. A 3.1.2. ábrán látható igen gyakran alkalmazott kapcsolás az ismert Domino elv alapján épül fel. Érdekessége, hogy az eső lépésben az átvitel új értékét számolja ki és ennek segítségével határozza meg a összeg értékét. Az összeg és ay átvitel negált értékének kifejezésére átalakítás után kapjuk:

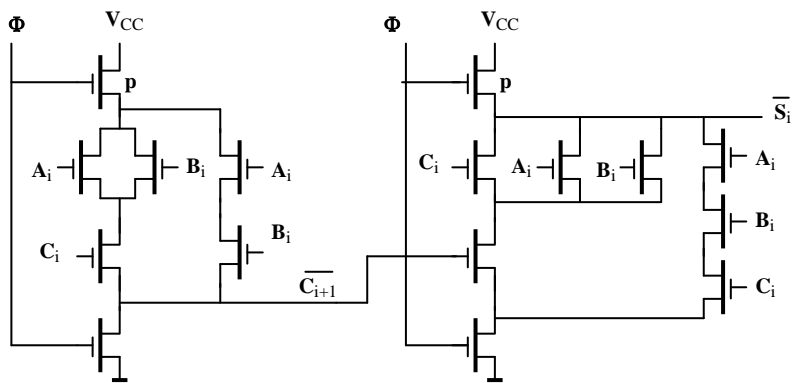
$$S_{i+1} = (A_i + B_i + C_i) \cdot [\bar{A}_i \bar{B}_i + (\bar{A}_i + \bar{B}_i) \cdot \bar{C}_i] + A_i B_i C_i$$

$$\bar{C}_{i+1} = \bar{A}_i \bar{B}_i + (\bar{A}_i + \bar{B}_i) \cdot \bar{C}_i$$

Amiből az összegre a következő egyszerű kifejezés adódik, ha felhasználjuk az új átvitelt:

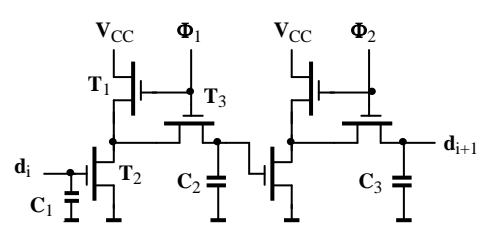
$$S_{i+1} = (A_i + B_i + C_i) \cdot \bar{C}_{i+1} + A_i B_i C_i$$

Az ábrán bemutatott kapcsolás láthatóan ezt függvényt valósítja meg.



Domino CMOS teljes összeadó

Kétfázisú dinamikus léptető regiszter. Az ábrán látható shift regiszter működése egyszerűen követhető a fázisjelek bekapcsolásával és a jel továbbterjedésével az egyes kapacitások között.

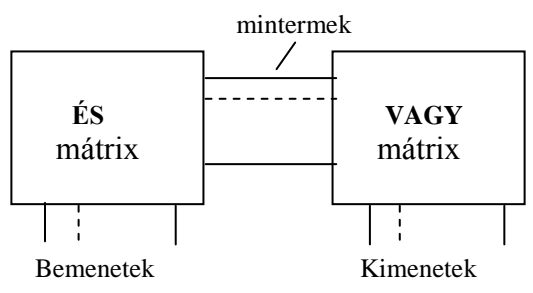


Kétfázisú dinamikus léptető regiszter

PLA áramkörök. Programozott hálók elvi működése. A bonyolultabb logikai függvények megvalósításának legáltalánosabb megoldása a programozott logikai háló (Programmed Logic Array, PLA), amelynek elvi felépítését a 3.4.1 ábra mutatja be. Belátható, hogy az ÉS, valamint VAGY mátrixokból felépített áramkörrel minden olyan függvény előállítható, amely a szorzatok összegeként írható fel, tehát

$$f = x_1 x_2 x_3 + x_4 x_5$$

ahol $x_1..x_5$ a bemeneti változók. A két mátrix közötti vonalak az ún. minterm-ek, amelyek jelen esetben az egyes szorzatokat tartalmazzák. Annak a feltétele, hogy egy adott logikai feladat a PLA-val megoldható legyen, mind a bemenetek, mind a kimenetek, mind pedig a mintermek számának elegendőnek kell lennie.



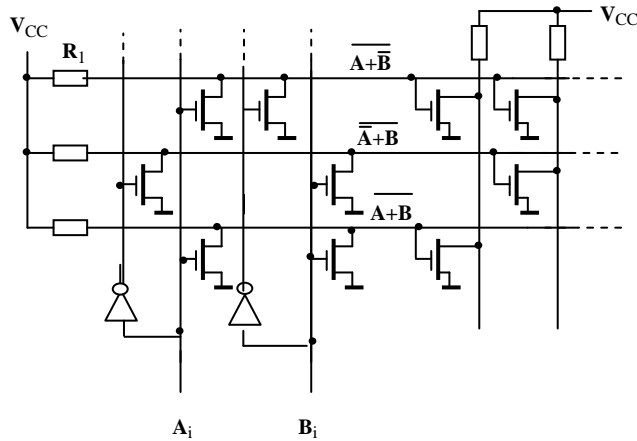
Programozott logikai hálók (PLA) általános felépítése

A PLA mátrixai lényegében közös vezérlésű ÉS ill. VAGY kapuk sokasága, és így érvényesek rá mindazok, amik a alapáramkörökre. Ennek értelmében a túl sok elemből álló ÉS-kapú késleltetési ideje rendkívül megnő (a sok sorbakapcsolódó soros ellenállás miatt) és arány-típusú logikáknál megnő a logikai "0" szint is. A VAGY-kapuknál a korlátozó tényezők kedvezőbbek, ezért is általában (amint azt a következőkben látni fogjuk) az ilyen típusú mátrixokat alkalmazzák. Itt is fellép viszont az, hogy a sok párhuzamosan kapcsolt tranzisztort meg az összekötő vezeték parazita kapacitása olyan nagy értékűre adódhat, ami rendkívül lelassítja a működést. Ezért célszerű ilyenkor a függvényt több egyszerűbb felépítésűre szétbontani és több kisebb mátrix-szal megoldani. Erre is fogunk látni példát a ASIC áramkörök kapcsán.

A PLA-áramkörökkel máshol is gyakorta találkozhatunk. A ROM memóriák pl. szintén lényegében programozott logikai hálók, ahol az első mátrix fix tartalma maga a sordekóder, a mintermekkel megcímezett sorok

tartalma pedig maga a tárolt memória-tartalom. Tulajdonképpen ilyennek tekinthető egy RAM is, azzal a különbséggel, hogy itt a második mátrix tartalmát a működés során változtatni tudjuk.

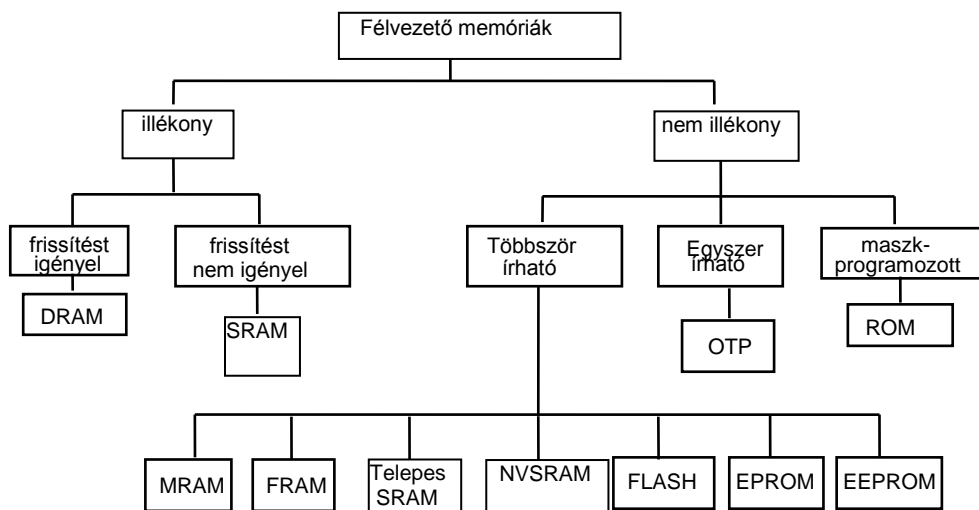
A fentiekből látható, hogy tehát az egyes mátrixok szerkezetük szerint lehetnek fix felépítésűek, amiknél tehát a mátrixpontok levő tranzisztorok aktív vagy passzív volta a gyártás követően nem változtató, vagy lehetnek programozhatóak, amit pl. a felhasználó programoz be akár hosszútávra, akár a működés során át konfigurálható módon. Ez a kérdés lényegében a különböző memória-típusok tárgyalásához vezet.



NOR-kapus PLA
áramkör

5. Memóriák.

Memóriák osztályozása és szerkezeti felépítése. A memória-áramkörök sokféleképpen osztályozhatók; az alábbi felosztás az illékonyásra helyezi a hangsúlyt, mivel a tápfeszültség kikapcsolása alkalmával is fennmaradó tárolókapesség ma már az alkalmazások zömében alapvető követelmény. Ezt mutatja be a 4.1.1. ábra, amelyben a baloldali ágba helyezkedik el a két legáltalánosabb memóriatípus, a dinamikus és a statikus memória, vagy másképpen DRAM és SRAM. Létük meghatározó a számítástechnika szempontjából és bár fejlődésük a majd 30-éves híres „Gordon Moore törvény” szerinti évenkénti bitszám duplázódáshoz képest valamelyest lelassult, az utóbbi években mérhető 1,5/év bitszám növekedés még mindig biztató.



A jobb oldali ágba a nem illékony (non-volatile, NV) áramkörök szerepelnek aszerint felosztva, hogy a memória tartalma hány alkalommal írható át a felhasználó által. A maszkprogramozott ROM (Read Only Memory) tartalmát a félvezetőgyár írja be egy technológiai lépés során, így ennek későbbi módosítására nincs lehetőség és költségét tekintve csak igen nagy darabszámoknál célszerű az alkalmazása.

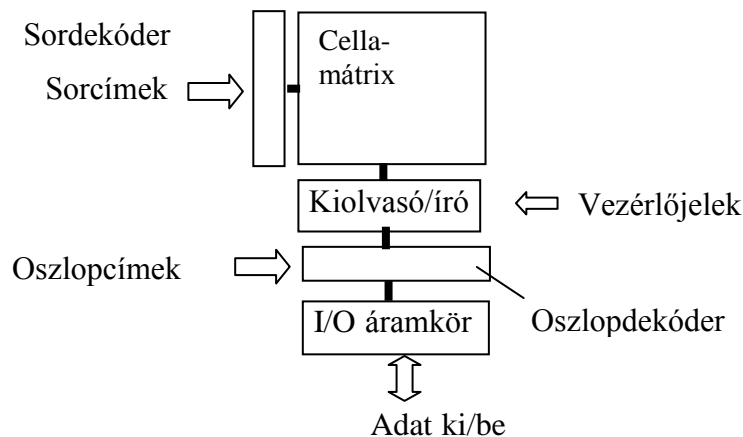
Az egyszer írható (One Time Programmable, OTP) ROM-ok lebegő-gates konstrukciók (lásd később), amelyeknél azonban a tokozáson nincs ablak az UV-törléshez, s így az egyszer beírt tartalom nem törölhető. Közepes darabszámoknál használják, mivel a készülék működésének software-oldali módosításakor egyszerűen csak új tartalmat kell beírni a ROM-okba.

A legflexibilisebb lehetőséget értelem szerűen a többször írható ROM-ok nyújtják. Ezek közül a mágneses (MRAM) és ferromágneses (FRAM) megoldásokkal, mivel nem a klasszikus szilícium-technológiával készülnek, nem foglalkozunk. A telepes SRAM nem egy speciális félvezető konstrukció, hanem egyszerűen a „szünetmentes tápegység” megvalósítása egy tokon belül, a RAM chip fölé egy kis akkumulátort szerelve. Az folyamatosan tért vesztő NVRAM egy speciális, kettős dielektrikummal rendelkező struktúra, amelyről a későbbiekben még szó lesz. Ugyancsak a kiszoruló típusok közé tartozik az ablakos kivitelű EPROM (Elektronically Programmable ROM, újabban UV-EPROM elnevezéssel). A legdinamikusabban fejlődő típusok a sorban egyértelműen az EEPROM (Elektronically Erasable ROM) és a Flash memóriák, amelyeket viszonylag nagy beírási idejük miatt RAM-ként használni nem lehet, de program- és adattárolásra kiválóan alkalmasak.

Az ábra alapvetően a számítástechnikában felhasznált memóriákat csoportosítja és nem tartalmazza a multimédia speciális szervezésű memóriáit. Ezek alapvetően ugyanazon működésű cellákból épülnek fel, csak éppen a tárolt adat felhasználásának jellegétől függően más konstrukcióval rendelkeznek. A legnagyobb csoport ezek közül a soros memóriák (Serial Access memory, SAM), amely a képmegjelenítés soros technikájához igazodik; ide tartoznak a video RAM-ok (VRAM).

Memóriák szerkezeti felépítése. A memóriák leggyakoribb formája a véletlen-hozzáférésű típus (nem egészen találóan lefordítva a *random* szóból), amelynél egyetlen ciklusban bármelyik cella (memória-rekesz) írható, vagy olvasható. Ezek szerkezeti felépítését (architektúráját) az ábra mutatja be, amelyet elsődlegesen a késleltetés szempontjából vizsgálunk meg. A késleltetés ennél az áramkörnél egyértelműen a hozzáférési időt jelenti (access time), vagyis az, hogy mennyi idő alatt olvasható (vagy írható) a memória.

A közel négyzetes cellamátrix elemeit a dekóderek címezik meg, amelyek közül a sordekóder az első áramköri egység a jelútban és ennél fogva késleltetési ideje közvetlen módon részét képezi a memória teljes késleltetésének. Az oszlopdekóder esetében kiolvasáskor a helyzet kedvezőbb, hiszen ennek csak a cellák kiolvasása után kell az érvényes értéket felvennie. Mint látható, az ábra szerinti alap-elrendezésben címzéskor egyszerre aktiválódik az egy sorban levő összes cella, amelyek közül a számunkra érdekeseket az oszlopdekóder továbbítja, a majd később látható módon egyesével, szavanként, programozható szélességű adatonként, vagy akár egy átmeneti tárolás közbeiktatásával.



Véletlen hozzáférésű memóriák szerkezeti felépítése

A lábszám minimalizálása memóriáknál is (hasonlóan más áramkörökhöz) egy igen fontos kérdés. Könnyen belátható módon azonos cellaszám mellett az egy bites szervezés kedvezőbb. Egy másik lehetséges megoldás a multiplex címzési mód, amelynél ugyanaz a láb (pin) először egy sor-, majd egy oszlop kiválasztására szolgál. Ekkor azonban (általános esetben) két további jel alkalmazására van szükség, amelyek ezen két ütem megkülönböztetésére szolgálnak. Ezek a RAS (Row Address Signal) és a CAS (Column Address Signal) jelek, amelyek természetesen két további kivezetést igényelnek.

A kiolvasó erősítő a dinamikus memóriáknál érdemel figyelmet, mivel ott egy igen kis jel megbízható érzékelése a feladat. A gyors működés érdekében azonban a statikus működésű memóriáknál is számos különleges

kapcsolási megoldás született, amelyekről a későbbiekben szó lesz. Az adatot beíró, valamint az input/output áramkörök kapcsolástechnikailag kevésbé kritikus egységek.

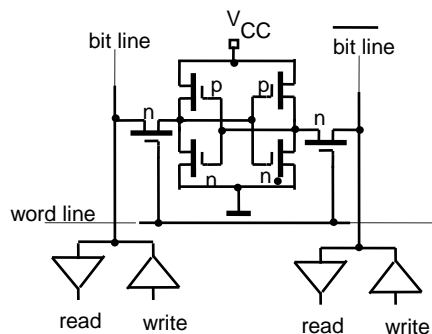
A memória legfontosabb egysége a cella; ennek mérete, késleltetése, zavarérzékenység alapvetően meghatározza a memória tulajdonságait. Ezért ne csodálkozzunk, hogy ezen hihetetlenül kicsi méretű kapcsolás fejlesztésével, tervezésével jóval többen és jóval több ideig foglalkoznak, mint a memória összes többi részével együttvéve. A dolog viszont érthető, ha arra gondolunk, hogy itt pl. a helyfoglalás szempontjából minden tízed-mikron megtakarítás akár milliószorosával jelenik meg a chipen.

A 6-tranzisztoros statikus CMOS cella. A 4.2.1 ábrán látható 6-tranzisztoros cella az elektronika egyik legismertebb áramköre. Magja az ún. keresztbe-csatolt flip-flop, amelynek két CMOS inverter egymást vezérli. Ez a tároló ennél fogva magára hagyva is mindaddig tárolja az adatot, amíg elegendő tápfeszültség van jelen – de ennek néhányszor tíz ms ideig történő eltűnésekor az adat elvész, a cella tehát *illékony* (volatile).

A két statikus CMOS inverteren nyugalmi helyzetben (statikusan) áram nem folyik, így fogyasztása ilyenkor gyakorlatilag zérus. Ezt használják ki a teleppel működő ún. *back-up* megoldások, amelyeknél pl. a hálózati kimaradás időszakára a tápfeszültséget egy alig terhelt beforrasztott akkumulátor szolgáltatja. Érdeemes megemlíteni, hogy ez a gondolat már korábban, még a CMOS cellák kidolgozása előtt is megjelent. Akkor ezt a feladatot a tápfeszültség *pulzálásával* oldották meg, ahol a tápfeszültség-impulzusok közötti szünet hosszát az szabta meg, hogy a magára hagyott flip-flop két driver- tranzisztorának gate-kapacitása mennyi ideig képes tartani a töltését (pontosabban a küszöbfeszültség feletti értéket). Ez volt ugyanis szükséges ahhoz, hogy a tápfeszült pulzusának megjelenésekor a két kapacitás közül a feltöltött még nyissa a drivert és ezzel a flip-flopot ismét a kijelölt stabil állapotába billentse.

A cella beírása ill. kiolvasása a szóvonallal (word line) működtetett két kapuzó nMOS-on keresztül történik, mégpedig a bit-vonalról ill. annak negáltjáról. Beírásakor egy adatot kényszerítünk a vonalra, ami a kapuzó tranzisztorokon keresztül bebillenti a flip-flopot a kijelölt állapotba (ha korábban az ellentétes állapotban volt). Kiolvasáskor a bitvonalakra egy-egy kiolvasó erősítő kapcsolódik, amelyek érzékelik a flip-flop állapotát (és annak negáltját).

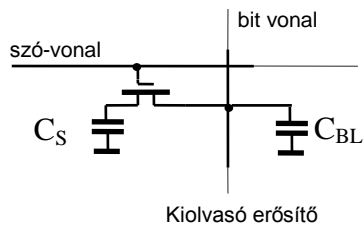
A működés láthatóan teljesen szimmetrikus. A kiolvasás nem destruktív, hiszen a tároló eredeti állapotában marad. A kiolvasó erősítő a szó igazi értelmében nem egy igazi erősítő, hiszen a flip-flop kimenetén egy jól meghatározott logikai szint jelenik meg. Az viszont tagadhatatlan, hogy kiolvasáskor az átbillentésnél nyers erőt kell alkalmazni mindaddig, amíg a két aktuálisan nyitott tranzisztor (az átlósan szemben levő nMOS és pMOS) le nem zár, mivel addig a billentésnek ellene dolgozik (ui. ez a feladata).



6-tranzisztoros statikus CMOS RAM cella

A cellamátrix a memória-chip felületének döntő részét foglalja el, és egy-egy sorban ill. oszlopban igen nagyszámú cella helyezkedik el. Pl. egy 64Mbit-es memóriában, ha a mátrix négyzetes alakú, akkor egy-egy oszlopra elvben 8192 cella lehet felírván, ami egy igen nagy kapacitív terhelést eredményez. Ezt a nagy kapacitást kell a viszonylag kis áramra tervezett cellának rövid idő alatt kisütnie ill. feltöltenie. Ez a nagy bitszámú memóriák elsőrendű problémája.

Az 1-tranzisztoros dinamikus cella. A memóriáknál, mint szó volt róla, a chip felületének majd egész részét a cellamátrix foglalja el. A cella egyszerűbb felépítésével s így kisebb helyfoglalásával a chipen integrálható bitszám növelhető. Ez az utat járják a dinamikus memóriák (DRAM-ok) azzal, hogy a 6-tranzisztoros statikus cella helyett egy kapacitásban tárolják (adott ideig) az adatot. Ennek megoldását az ábra mutat be.



1-tranzistoros dinamikus RAM cella

A beírás során a adatot a bit-vonalra (bit-line) tesszük és a szóvonal (word line) aktiválásával a tranzisztor a bit-vonal feszültségét a Cs tároló-kapacitásra kapcsolja. A tranzisztor zárásával a kapacitás töltése tárolja a információt mindaddig, amíg a töltés onnan különböző okok miatt el nem szivárog. Mivel a kapacitással párhuzamosan egy lezárt dióda is van, ezért ennek szivárgási árama kb. 100ms alatt olyan mértékben lecsökkenti a feszültséget, hogy a cella frissítésre szorul. Ez a dinamikus tárolás egyik nehézsége, ami a folyamatos működés időnkénti megszakítását igényli a frissítés miatt. Az újabb technológiai megoldásoknál ezt a tárolási időt sikerült több másodperc hosszúságú időtartamra megnyújtani, ami egyes speciális analóg áramköröknél egészen különleges megoldásokra ad lehetőséget. Ezekről az ún. *analogic* kapcsolásoknál szólnak bővebben.

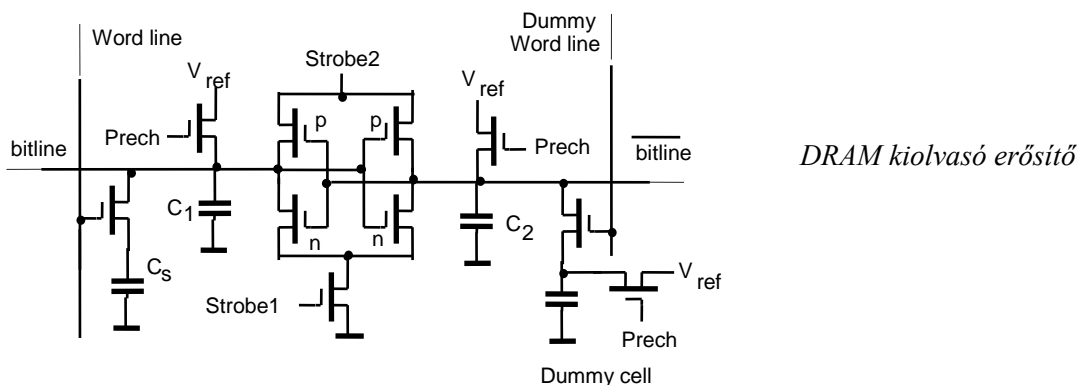
A dinamikus tárolásnál egy további probléma a nagyenergiájú (MeV nagyságrendű) elemi részecskék becsapódása, amelyek töltéshordozó párokat generálva tárolt töltés egy részét semlegesíthetik. Ez különleges óvintézkedéseket tesz szükségessé, mint pl. a speciális tokozás, a struktúrában olyan belső elektromos erők létrehozása, amelyek ezeket a töltéshordozókat eltávolítják a tároló kapacitástól, és nem utolsó sorban redundáns, hibajavító memória-blokkok (vagy akár chipkek) létrehozása, amelyek ezt a véletlenszerűen fellépő hibát korrigálják.

A Cs tárolókapacitás értéke nem lehet túl nagy, hiszen ez a geometriai méretek, s ezzel a chipfelület megnövekedését vonná maga után. A nagybonyolultságú memóriák megvalósításának viszont éppen egyik alapfeltétele, hogy a memória-cella méretét minél alacsonyabban tartsuk. A kis kapacitás-érték viszont értelemszerűen érzékenyebb lesz a fenti zavarokra. Az évek során kialakult, hogy a tárolókapacitás optimális értéke 50-80 fF körül van.

A tárolókapacitás fenti igen kis értéke kiolvasásakor jelent problémát, hiszen ilyenkor a kapcsoló-tranzisztor nyitásával ezt a nála minimálisan 20-szor nagyobb C_{BL} bit-line kapacitásra kapcsoljuk és annak ΔU_{BL} feszültségváltozását kell érzékelnünk. Ennek értékét a töltéskiegyenlítődből levezetett

$$\Delta U_{BL} = (U_S - U_{ref}) \cdot \frac{C_S}{C_{BL} + C_S}$$

összefüggés adja meg, ahol U_{ref} a bitline előtöltött feszültsége. Láthatóan a kapacitás-arányok miatt az érzékelendő feszültségváltozás igen kicsi, és itt már szó sincs klasszikus értelemben vett logikai szintekről, hanem az analóg technika tartományába eső jelszintjeiről van szó. Ez tette szükségessé a rendkívül szellemes kiolvasó erősítő áramkör kidolgozását. Ennek alapelvét a 4.3.2. ábra mutatja be.



A kapcsolás alapötlete, hogy központi CMOS flip-flopot a tápfeszültség és a földvezeték megszakításával egy lebegő állapotba hozzuk, majd ezek ezt újra visszakapcsolva, ezen metastabil állapotból az általunk ráadott *kijelölő feszültség* irányába fog bebillenni. Mivel ez a kijelölő feszültség értéke jóval kisebb, mint a bebillenés utáni logikai szint, ezért nevezzük kiolvasó erősítőnek, holott láthatóan nem egy erősítő.

A működés a valóságban a következőképpen zajlik le. A Strobe1 és Strobe2 fázisjelekkel a flip-flopot lebegő állapotba hozzuk, majd annak két karját előtöltjük a PRECH fázisjel segítségével az U_{ref} feszültségre. Ezen felül az a kiolvasandó cellával ellentétes oldalon egy dummy-cella kapacitását is előtöltjük a tárolókapacitás logikai "0" és "1"

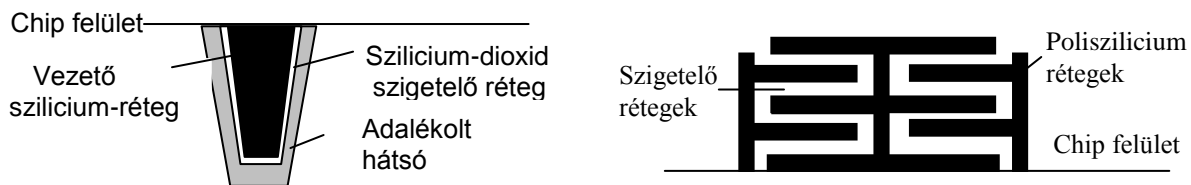
szintjeinek középértékére, mint referencia-értékre. Ezután következik a két cella szóvonalaának megcímzése, amelynek során mind a kiolvasandó cella, mind a dummy-cella a benne levő feszültséget (töltést) a most még lebegő flip-flop két karjára kapcsolja. Az ennek során kialakuló ΔU feszültség, mint kijelölő feszültség, a Strobe-jelek visszakapcsoláskor a flip-flopot a kívánt irányba billenti, tehát ha a Cs-ben logikai "1" volt, akkor a flip-flop baloldali nMOS drivere nyit, stb.

A kiolvasás során kb. $\Delta U=50-100\text{mV}$ feszültség lép fel a flip-flop bemenetein, ami nyilvánvalóan egy tökéletesen szimmetrikus felépítést követel meg. A dummy cella igen fontos része a kapcsolásnak, hiszen a szimmetrián túlmenően egyrészt a rákapcsolásnál is kiegyenlítetttséget biztosít, sőt, a kapcsoló-tranzisztor által becsatolt gate-feszültség (vigyázat, ez sokszorosa a ΔU feszültségnek!) által okozott hatást is szimmetrizálja. Mindez alapvető fontosságú a kapcsolás megbízható működése szempontjából.

A kiolvasás befejeztével, miután a flip-flop végső helyzetébe bebillent, ez a most már szokványos logikai szint íródik vissza a cella kapacitásába. Hogy a tartási idő és a zavarérzékletlenség még nagyobb legyen, egy utolsó lépésként (a korábban már tárgyalt utánhúzási technika felhasználásával) egy megemelt feszültséget írnak vissza a cellába.

A flip-flop mindkét karján azonos számú (pl. 2048 db.) cella van, továbbá egy dummy-cella. Kiolvasáskor a kiválasztott cella szóvonala aktiválódik rákapcsolva a cellát a bitvonalon keresztül a hozzátartozó kiolvasó erősítőre, és ugyanakkor bekapcsolja a másik oldalon levő dummy-cellát. Kicsit zavaró szokott lenni, hogy a kiolvasó erősítő szokásos ábrázolásánál a cella és a vonalak éppen 90 fokkal el vannak forgatva, de remélhetőleg ez a működés megértését nem zavarja.

Térbeli elhelyezkedésű tárolókapacitások. A memóriák bit-számának növelésekor alapvető probléma a tárolókapacitások helyfoglalása. Mivel ezek felületét a korábban tárgyalt okok miatt csökkenteni nem lehet, új megoldásokat kellett keresni ezek elhelyezésére. Az egyik lehetőség az ún. *árok* (trench) kapacitás, amelynél a szubsztrátban még korlátlanul rendelkezésre álló helyet használják ki. A struktúra keresztmetszetének elvét a 4.3.3. ábra mutatja be.



Térbeli tároló-kapacitások, a) árok kapacitás, b) stack-kapacitás

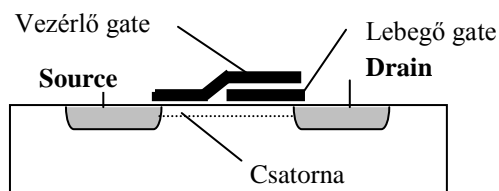
Az árok-kapacitás technológiájának során a szilícium chip kijelölt helyein egy 4-6 μm mély árkot marnak, ennek szélét (oldalfalát) először adalékolják (ez adja külső fegyverzetet), majd eloxidálva egy szigetelőréteget hoznak létre, végül a megmaradó részt poliszilíciummal kitöltve (filling) létrehozzák a kapacitás belső fegyverzetét. A kapcsoló-tranzisztor korábban (a szokásos módon) a chip felületén helyezkedett el, ma már egyes konstrukcióknál ez is fokozatosan az árok oldalfalába "csúszik be", hogy ezzel is csökken az elfoglalt chip felület.

Egy másik megoldás szerint a kapacitás felépítése nem befelé, a szubsztrát felé, hanem felfelé, a felület fölé irányul. Ennek megoldását mutatja be a 4.3.4. ábra.

Maszk-programozott ROM áramkörök. A memóriák közül a legegyszerűbb felépítésű típusok a maszk-programozott (tehát már a gyártás során a tartalommal beültetett) memóriák. Ezeknél a mátrix egy-egy pontjában attól függően van logikai "0" vagy "1", hogy az adott pontban van-e egy aktív "pull-down" tranzisztor (vagyis, ahol a szóvonál, mint gate alatt vékony gate-oxid van), vagy nincs (a szóvonál alatt vastagoxid van, csatorna nem tud kialakulni). Ezen elrendezésből következően a helykihasználás maximális, vagyis a szokásos chip-méretek mellett igen nagy elemsűrűség valósítható meg. Ez a nagy előnye. Hátránya viszont, hogy a beírt programon utólag változtatni nem lehet, csak a maszk cseréjével, ami költséges. Elsősorban ott alkalmazzák, ahol igen nagy sorozatokat állítanak elő, pl. gyerekjátékoknál, karakter-kódok konverziójánál, stb.

UV-EPROM áramkörök. A csak-olvasó memóriák közül még igen széles körben használják az ultraviola fényvel törölhető áramköröket. Működésük alapja (akárcsak a későbbiekben tárgyalt további típusoknál) a lebegő "floating" gate, amelyet töltéshordozókkal feltölve, mint feltöltött kapacitás tárolja a töltést és eközben maga alatt egy

csatornát hoz létre. Ennek a csatornának a létét (áramát) ellenőrizve tudjuk meg, hogy fel lett-e töltve a lebegő gate, vagy nem. Ez az adat kiolvasása. A hosszúidejű tárolás feltétele, hogy a kapacitás (pontosabban a lebegő gate-et körülvevő szigetelőréteg) szivárgása praktikusán zérus legyen. Ennek kidolgozása egy sok-sok éven át tartó kemény technológiai fejlesztés eredménye, de végül is sikerült megoldani. A ma gyártott UV-EPROM áramkörök tárolási ideje "garantáltan" (hiszen még nem igazoltan) közel 100 év.



UV-EPROM cella
keresztmetszete

A cella keresztmetszetét a 4.4.2. ábra mutatja be. A lebegő gate feltöltése elektronokkal a régebbi típusoknál a lavina-letöréssel előállított nagyenergiájú töltéshordozókkal történik (FAMOS, Floating Avalanche Gate MOS), amelyek egy része nagy sebessége folytán képes átjutni a nem túlzottan vékony (tehát nem tunnel-oxid típusú) szigetelőn és feltölteni a gate-et. Ez a töltés a továbbiakban mint gate-feszültség működik és ilyenformán szabályozza az alatta levő csatorna kialakulását: a feljutott elektronok negatív potenciálja lezárja a csatornát.

A lebegő gate alatti csatorna azonban még nem eredményez áramvezetést, hiszen nem ér el a source-ig. Ennek a csatorna-szakasznak a létrehozásáért a vezérlő gate (control gate) a felelős. Erre egy pozitív impulzust adva, ez a csatornaszakasz is felépül és ezzel áramvezetés jöhet létre a source és drain között (a drain-re természetesen pozitív feszültséget kell adnunk). Mint az ábrán is látható, a két gate-nek fizikailag igen közel lennie egymáshoz, hogy a két csatornaszakasz "összeérjen". Ennek a fizikai közelségnek a következménye, hogy a vezérlő gate-re adott impulzus kapacitív úton (nyitóirányú) feszültséget csatol át lebegő gate-re is. Ezt a hatás természetesen a potenciálviszonyok tervezése során figyelembe veszik.

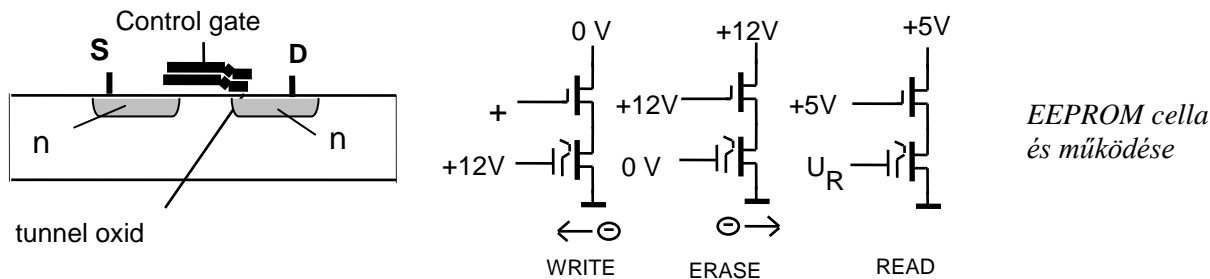
A töltések eltávolítása a lebegő gate-ről ultraviola fényel történik, amely elegendő energiával gerjeszti az elektronokat, hogy azok visszatérjenek a drain-re. (fizikailag ez azt jelenti, hogy a UV energia nagyobb, mint szilícium-szilíciumdioxid potenciállal magassága, amelyet a töltéshordozónak meg kell ugrani).

A lavina-letörés nem igazán optimális eljárás a töltéshordozók feljuttatására, egyrészt mivel hosszú ideig tart (tekintve, hogy azoknak csak egy rész jut át az oxidon), másrészt az alkalmazandó nagy lavina-feszültség a disszipáció is magas. Ezért az újabb típusoknál egy jóval vékonyabb, ún. tunnel-oxidot alkalmaznak, amelyen alagúthatás révén jutnak fel a töltéshordozók a lebegő gate-re. Erről a megoldásról a későbbiekben részletesen szólunk.

Az UV-EPROM áramkörök történelmi szerepe óriási, velük egy új korszak kezdődött, hiszen az elektronika fejlődése során első ízben nyílt lehetőség arra, hogy a villamosmérnök a saját laborjában (tehát a félvezetőgyárral való együttműködés nélkül) tudjon mikroprogramot beültetni és módosítani a mikroszámítógépében. Ez óriási lendületet adott a fejlődésnek, hiszen a mikroprogramozott elektronikai fejlesztések hihetetlen módon felgyorsultak, ami végül is elvezetett a mai, elektronizált (informatikai) társadalomhoz.

Alagút-hatáson alapuló EEPROM memóriák. Az UV-sugárral való törlés kényelmetlen voltán túl azzal a hátránnyal is jár, hogy a memóriát ki kell emelni az áramkörből, vagyis az ún. *in-circuit* újra átírás nem valósítható meg. A döntő áttörés ezen a téren az alagút-hatás alkalmazásával történt meg, amikor olyan vékony (de ugyanakkor tökéletesen szigetelő, feszültségbíró és töltésmentes) oxidréteget tudtak előállítani, amin az elektronok az alkalmazott feszültség polaritásától függően mindkét irányból át tudnak „tunnelezni”. Ezzel megvalósult a lebegő gate pusztán elektromos jellel történő törlése (*erase*), ezzel pedig az a lehetőség, hogy a memória az áramkörön belül is újra programozható. Ily módon az eszköz programozása egy belső áramköri műveletté vált.

Az ábra a klasszikus EEPROM cella keresztmetszetét mutatja be, amelynél a lebegő gate-t a drain-tartománytól egy rendkívül vékony oxidréteg (tunneloxid) szigeteli el. A *control gate*-re adott feszültség



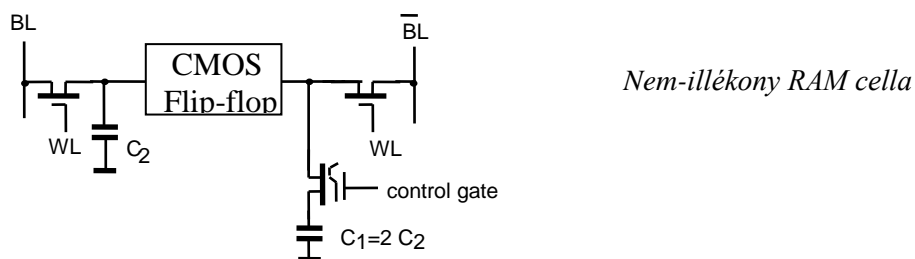
Az ábra az EEPROM cella három üzemmódját tünteti fel. A cella beírásakor az elválasztó T_1 tranzisztort nyitva a cella drain-jére $U_D=0$ feszültséget kapcsolunk, míg $U_{CG}=+12V$, aminek hatására az elektronok lebegő gate-re tunneleznek; ezzel a eszköz „felprogramozódott”. A törlés során $U_D=+12V$ és $U_{CG}=0$ értékeket állítunk be és ezzel éppen a fordított folyamat játszódik le: az elektronok „vissza”-tunneleznek a lebegő gate-ről a drain-re. Végül a normál működés (kiolvasás) esetén a T_1 tranzisztort, mint kapuzó elemet nyitva, valamint egy $U_{CG} = U_R$ feszültséget kapcsolva az eszköz attól függően fog vezetni, hogy a lebegő gate-en vannak-e elektronok, amelyek a küszöbfeszültséget olymértékben eltörlják pozitív irányban, hogy az alkalmazott U_R feszültség nem tud létrehozni vezető csatornát.

Az EEPROM memóriák alkalmazása során kiderült, hogy többségükben nincs szükség a cellánkénti/byte-onkénti/ szavankénti újra programozásra, hanem elegendő lenne az egész memóriát egyszerre, egy lépésben (egyetlen elektromos pulzussal) törölni. Ennek megvalósítására születtek a Flash memóriák (amely nevében utal arra, hogy itt mint egy egyetlen megvilágításra törlődik az egész chip, mint az UV-megvilágításra, de itt persze elektromosan).

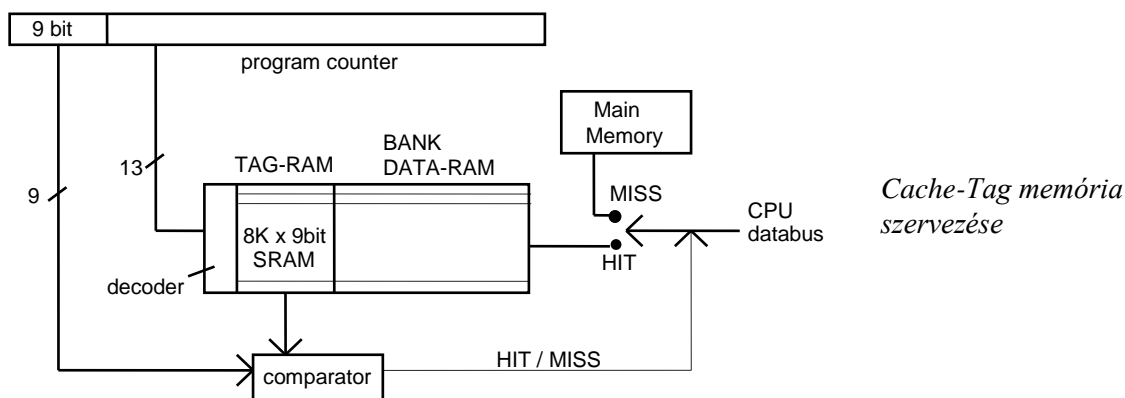
Nem illékony SRAM memóriák. Az elektronikus tervező számára a memóriák szempontjából az optimális megoldás nyilvánvalóan az lenne, ha ha egyfajta memória létezne, és az RAM-ként lenne használható (vagyis ns-os sebességgel lenne írható és olvasható, de ugyanakkor a tápfeszültség kikapcsolása után megtartaná az információt. Ez a lényegében “mindentudó” memória még ma sem létezik eszköz-szinten, bár áramkörti szinten vannak megoldások, amelyek ezt a tulajdonságot jól-rosszul közelítik.

Mint korábban is említettük, a nem-illékony RAM egy igen nagy könnyebbséget jelent a felhasználó számára, már csak a nemkívánatos feszültség-kimaradások kivédésére is. Ennek egyik legrégebbi formája az MNOS (Metal-Oxid-Nitrid-Semiconductor) struktúra, amelyet annak idején a pénztárgépek számára fejlesztettek ki. A struktúra technológiai problémái miatt nem tudott elterjedni (bár ilyen irányú fejlesztések még ma is folynak), és alkalmazási területét teljesen az EEPROM-bázisú megoldások foglalták el.

Kapacitív billentésű, nem-illékony SRAM cella. A cella kapcsolását az ábra mutatja be. A tápfeszültség adott szint alá csökkenésekor egy (a szünetmentes tápegységekben alkalmazott kapcsoláshoz hasonló) vezérlőegység a control gate-feszültség alkalmazásával az EEPROM-ba a flipflop aktuális értékét írja be. Bekapcsoláskor a) ha az EEPROM lezárt állapotban van, akkor C_1 leválasztódik és a flipflop a C_2 irányába billen be ($U_{C2}=0$), b) ha viszont vezet, akkor a jobboldali nagyobb kapacitás miatt az ellenkező irányban történik meg ez.



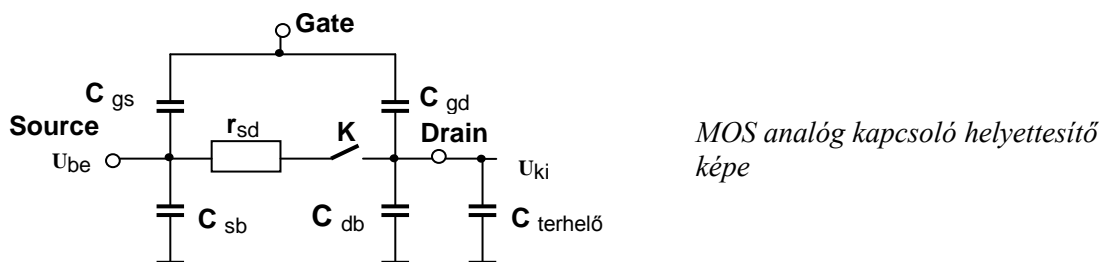
Cache-tag szervezésű memóriák. A ma használatos on-chip cache áramkörök elrendezését az alábbi ábra mutatja be.



A programszámláló alsó szegmense a TAG-RAM memória címzését végzi, a *dekóder* segítségével. Az így megcímezett rekesz tartalma a programszámláló felső szegmensével kerül összehasonlításra a komparátorban, amely ezt egyetlen lépésben végzi el. Egyezés esetén (*Hit*) a CPU az adatot az *adatbank*-ból olvassa ki, amely értelemszerűen sokkal gyorsabb hozzáférést tesz lehetővé, mint a *Main memory*-ből történő olvasás (*Miss*)

6. Analóg áramköri elemek

MOS analóg kapcsoló. Az analóg jelek kapcsolására gyakran alkalmazott eszköz a MOS. Ez a feladat az eszköz számára csakmennyiben tér le szokásos logikai (transzfer gate-es) alkalmazástól, hogy itt különös jelentőséget kap a soros ellenállás és (mivel kisszintű analóg jeleket is hűen kell átvinni) a nagy zavarvédetség. Az analóg kapcsoló üzemben működő MOS tranzisztor helyettesítő képét a 6.3.1. ábra mutatja be.



Az ábrán r_{ds} a drain-source ellenállás, amely lezárt esetben igen nagy érték, viszont nyitásban vezérelt tranzisztornál értéke

$$r_{ds} = \frac{\beta \cdot (U_{GS} - V_T)}{2}$$

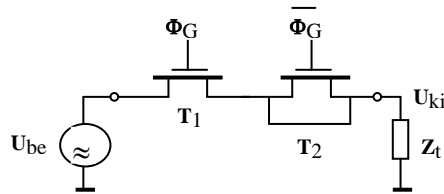
ahol β a fizikai állandóktól és a geometriai méretektől függő ismert állandó, U_{gs} a gate-source feszültség és V_T a küszöbfeszültség (threshold). Látható, hogy elegendően nagy gate-feszültségnél a soros ellenállás elegendően kicsire tehető. Az ellenállással sorban levő K kapcsoló azt szimbolizálja, hogy mindez csak a nyitott tranzisztorra érvényes. A MOS tranzisztoros áramköröknél általában nagy impedanciák jelentkeznek s így ez a soros ellenállás általában nem okoz problémát.

Sokkal nagyobb gondot jelentenek a tranzisztor belső kapacitásai, amelyek az ábrán láthatók s amelyek kapcsolódási pontjait az indexben levő betűk jelölnék. A probléma elsődlegesen (az imént nagy értékűnek ajánlott) gate-feszültséggel van. A tranzisztor nyitáskor a gate-re adott (Volt nagyságrendű) pulzus ugyanis egy kapacitív osztón keresztül becsatolódik az analóg jel útjába és mivel ez utóbbinak a szintje akár néhány száz mV is lehet, még kellően nagy leosztás esetén is lényeges jeltorzítást idéz elő. Meg kell jegyezni azonban, hogy ez csak a gate-impulzus felfutásakor jelentkezik, ezt követően a gate DC-feszültsége nem befolyásolja a kapcsolt jelutatót.

A folyamat követése számítógépes szimuláció nélkül igen bonyolult, hiszen a tranzisztor nyitása közben a feszültségtől függő kapacitások értéke is változik és természetesen r_{ds} értéke is. Vegyük észre, hogy amíg r_{ds} értéke

nagy (vagyis a tranzisztor nem nyitott ki teljesen), addig C_{gd} szerepe igen jelentős, hiszen a kimenet nagyimpedanciájú, csak kapacitív terhelésekkel. Amint viszont r_{ds} értéke közel zérusra csökken le, a kimenetre özvetlenül rákapcsolódik a bemenetet meghajtó (általában kisimpedanciájú) generátor, ami egy sokkal nagyobb feszültség-leosztást eredményez. A gate vezérlése során ezen két határeset között változik a szuperponálódó zavarójel amplitúdója.

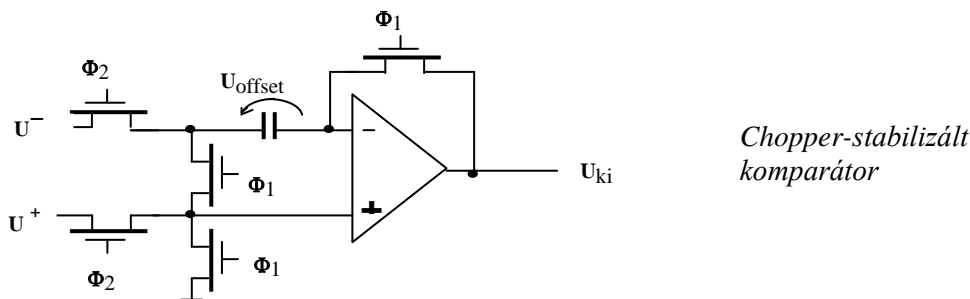
Az áramkörtervezők fő törekvése természetesen az, hogy ezt a nemkívánt effektust a minimálisra csökkentsék. Ennek egyik megoldását mutatja be a 6.3.2. ábra, amely lényegében egy kompenzálásra kialakított hídkapcsolás.



A híd két ága a ponált és negált Φ_G gate-feszültség, a kimenete (ahol a két ág jele kioltja egymást) a kapcsoló kimenete. A jelet ténylegesen a T1 tranzisztor kapcsolja, T2 (amely rövidzáiban van) csak egy hasonló karakterisztikájú kapacitásként működik; amekkora gate-jelet csatol be T1 kapacitása a jel felfutásakor, közel ugyanakkora, de ellenkező előjelű feszültséget csatol be T2 gate-je és a két jel eredője így lényegesen kisebb lesz. A híd tökéletes kiegyenlítése természetesen megoldhatatlan (már csak a technológiai pontatlanságok miatt sem), de jól megválasztott geometriai méretek mellett a zavarójel amplitúdója mintegy a tizedrészére csökkenthető, ami lényeges javulás a zavarvédetségben.

Chopper-stabilizált komparátor. Az átalakítók egy fontos eleme a feszültség-komparátor, amelynek pontossága alapvetően befolyásolja ezeknek az áramköröknek a pontosságát. A legkritikusabb kérdés ezeknél az offset feszültség, amely (bármennyire is kalibráljuk az áramkört eredeti állapotában), a hőmérséklet-függésénél fogva időben változik és ez lerontja a komparálás biztonságát.

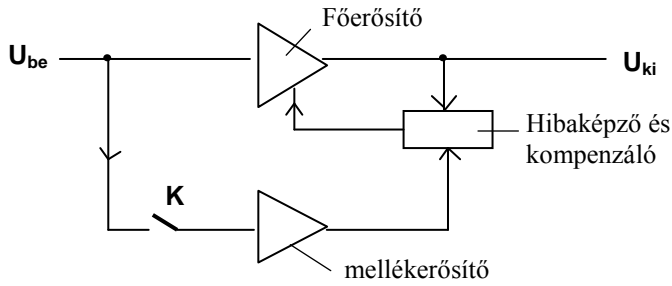
Ugyanakkor azonban van a komparátoroknak egy sajátossága, ami a jól ismert chopper-stabilizálás felhasználását lehetővé teszi, nevezetesen, hogy működésük nem folyamatos: egy-egy összehasonlítás között egy rövid ideig nem dolgoznak s ez lehetőséget ad arra, hogy offset feszültségüket kinullázzuk. Ennek kapcsolástechnikai megoldását mutatja be az ábra.



A komparátor normális működése a Φ_2 fázisjel alatt történik. Előtte azonban a Φ_1 jel hatására a belső (háromszöggel jelölt, offset-feszültséggel rendelkező) komparátor kimenetét visszacsatoljuk az invertáló bemenetere, és egy soros kapacitás közbeiktatásával, MOS kapcsolók segítségével a bemenetét rövidre zárjuk ill. földeljük. Könnyen belátható módon, ilyenkor a C kapacitás igen jó közelítéssel az offset-feszültség értékére töltődik fel (gondoljunk az offset definíciójára). A Φ_1 fázisjelnek elegendően hosszúnak kell lennie, hogy a töltődés végbemenjen. Ezt követően a Φ_1 fázisjel visszafut, az általa vezérelt tranzisztorok inaktívvá válnak.

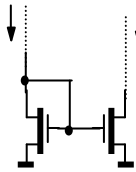
A Φ_2 fázisjel a külső kimeneteket az így kiegészített áramkörre kapcsolva egy olyan komparátorhoz jutunk, amelynek az offset feszültsége nagy pontossággal zérus, mindaddig, amíg a kapacitás tarja a töltését. Mivel MOS áramkörök statikus bemeneti árama gyakorlatilag zérus, ezért ez az idő jóval hosszab, mint egy komparálás ideje. Ezt követően a fenti kalibrálási eljárást meg lehet ismételni s így a működést egymást követő kalibrálásokból és komparálásokból áll.

Chopper-stabilizált folyamatos üzemi műveleti erősítő. Szemben az előzőekben ismertetett komparátor-kalibrálással, a folyamatos működésű áramköröknél (mint pl. a műveleti erősítőknél) ez az eljárás nem ilyen formában nem alkalmazható, mivel nincsenek szünetek, amikor ez véghezvihető lenne. Ennek ellenére sikerült ezt a problémát is megoldani, amint azt a 6.3.4. ábra bemutatja.



Chopper-stabilizált folyamatos működésű műveleti erősítő

Áramtükör. Az ábra a széles feszültségtartományban használható nMOS áramtükört mutatja be, amelynél a tükrözési arány nagymértékben független a kimenet U_{ki} feszültségétől. Összehasonlításként az a) ábrán feltüntettük az egyszerű kéttranszisztoros kapcsolást is, amelynél rövidsatornás eszközök esetében az I_{ki} kimeneti áram az U_{ki} függvénye lesz, vagyis a kapcsolás nem áramgenerátorként működik.



Áramtükör, 2-transzisztoros

7. D/A és A/D átalakítók.

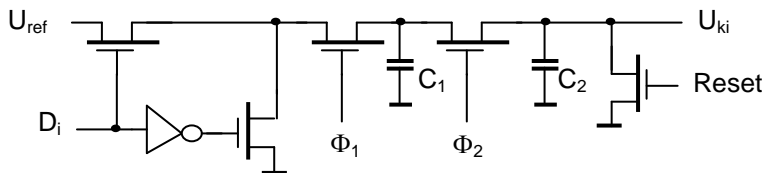
Digitál-Analóg átalakítók. A digitál-analóg átalakítókat három nagy csoportba oszthatjuk a működés módja szerint, nevezetesen

- számláló (integráló) típusú, 2^n lépésben átalakítók,
- bit-soros, n-lépésben átalakítók, valamint
- párhuzamos, vagyis egy lépésben átalakító kapcsolások, ahol n a felbontás.

Töltésfelező, n-lépéses D/A átalakító. A bitsoros átalakítók egy kedvelt megoldása látható a 6.1.1 ábrán, amelynek lényege a nagy pontossággal azonos értékű C_1 és C_2 kapacitásokon végrehajtott töltésfelezés. A működés kezdetén a *Reset* jel kisüti a C_2 kapacitást. Ezután a sorosan beléptetett (elsőként az LSB) D_i adatok rendre a Φ_1 -el vezérelt tranzisztor bemenetére kapcsolják vagy az U_{ref} referenciát, vagy a földet, feltöltve ill. kisütve ezáltal a C_1 kapacitást. A Φ_1 jel visszafutása után a Φ_2 hatására a C_1 és C_2 kapacitások között töltésmegosztás jön létre. Az n-edik lépés után a kimeneten fellépő feszültség értéke

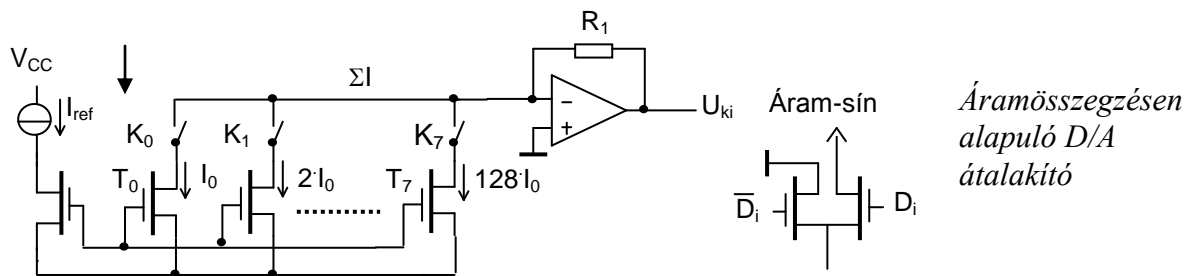
$$U_{ki} = \frac{U_{ref}}{2^n} \sum_{i=0}^{n-1} D_i \cdot 2^i$$

A gyakorlatban 8-10 bitnél nagyobb pontosságú ilyen típusú átalakítót nem alkalmaznak. Nagyobb számú adatsorozat beléptetésekor az az utolsó n-számú adat átalakítása fogadható el, amelyet a $C_1=C_2$ egyenlőség pontossága megenged.



Kapacitív töltésfelező D/A

Egylépéses áramösszegző D/A átalakító. A párhuzamos, tehát egy lépésben történő D/A átalakítás egy ismert eszköze az áramkvantáláson alapuló kapcsolás, amelynek 8-bites változatát az ábra mutatja be.



A kapcsolás binárisan növekvő áramú T0..T7 tranzisztorokból épül fel, amelyek áramának összegzése adja ki az analóg értéket. A tranzisztorok áramát az I_{ref}-el beállított áramtükör szabja meg. A D_i=1 bemeneti adathoz tartozó áram rákapcsolását a ΣI vonalra a K0...K7 kapcsolók végzik, amelyeknek a felépítése a b) ábrán látható. Az áramok összegzését a műveleti erősítő végzi, amelynek nem-invertáló bemenete a földön van, és ennek eredményeképpen a K-kapcsolók a tranzisztorok drain-jét a bemeneti adattól függően a valóságos, valamint a virtuális föld között kapcsolgatják, amiáltal a drain-nek a földhöz mért kapacitásának feszültsége (és így töltése) nem változik. Ez gyorsítja az átalakító működését. A kimeneti feszültség értéke

Az átalakító pontosságát (akárcsak a továbbiakban tárgyalt átalakítóknál is) a binárisan növekvő értékű elemek egymáshoz, valamint a referenciához viszonyított pontossága határozza meg. Ennek biztosítására a következő ún. hármas szabályt kell betartani:

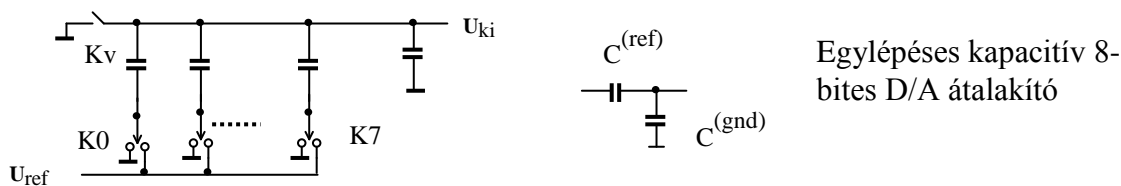
$$U_{ki} = R_1 \sum I$$

1.) A tranzisztorok binárisan növekvő értékű sorozatát nem egyszerűen a gate *w*-szélességének bináris növelésével valósítjuk meg (*w*, 2*w*, 4*w*...128*w*), hanem az I₀ egységnyi áramú tranzisztor multiplikálásával. Ez azt jelenti, hogy a 128I₀ áramú tranzisztor fizikailag 128 db. egységnyi tranzisztor párhuzamos kapcsolásából épül fel. Ezzel a különböző éretű elemek különböző méretváltozásából eredő hibát küszöböljük ki.

2.) Az egységnyi tranzisztor minimális mérete legalább 2-3-szorosa a technológia által meghatározott minimális méretnek, hogy a méretszórások ne befolyásolják a áramok értékének pontosságát.

3.) A tranzisztorokat úgy kell elhelyezni a chipen, hogy a közepén elhelyezkedő egységnyi tranzisztor körül egyenletes elosztva foglaljanak helyet a párhuzamosan kapcsolású, nagyobb áramú elemek összetevői. Ezzel elérhető, hogy a chip mentén laterálisan jelentkező inhomogenitások (technológiai egyenetlenségek, hőmérsékleti gradiens) különböző mértékben befolyásolják az egyes összetevőket és ezzel valamiféle kompenzáció jön létre.

Egylépéses kapacitív D/A átalakító. A töltés-kvantáláson alapuló, 8-bites kapacitív átalakítót mutat be az ábra, amelynek lényege a binárisan növekvő értékű kapacitás sor, vagy más néven kapacitás háló. A működés ezen kapacitások által képezett kapacitív osztókon alapszik, amelyet a b) ábra mutat be.



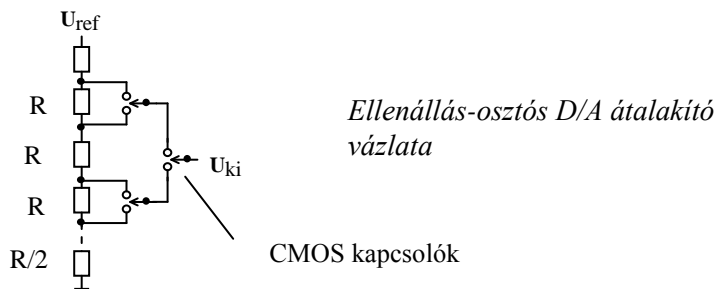
A működés két lépésből áll. Az első lépésben kezdetén a K_v kapcsolóval a kimenetet nullázzuk, és a D_i bemeneti digitális adatokkal vezérelt K₀...K₇ kapcsolókkal pedig minden elemet földre kapcsolunk. Ezzel a kapacitás-hálót nullázzuk. A második lépésben a bemeneti adatoktól függően D_i=1 esetén a kapcsolót az U_{ref} referencia-feszültségre, D_i=0 esetén a földre kapcsoljuk. Ennek eredményeképpen a kapacitások egyrésze a referenciára fog kaocsikódni (ezek összegét C^(ref)-el jelöljük), a maradék rész pedig a földre, ennek jelölése C^(gnd). Az így létrejövő osztó kimenőfeszültsége

$$U_{ki} = U_{ref} \cdot \frac{C^{(ref)}}{C^{(ref)} + C^{(gnd)}} = U_{ref} \cdot \sum_{i=0}^{n-1} \frac{D_i}{2^i}$$

Az áramösszegzéses eljárással szemben a kapacitív átalakító két szempontból is előnyösebb, egyrészt mert statikus áramfelvétele zérus, másrészt a kapacitások értékének pontossága a reprodukálhatóság és a hőmérséklet-függés szempontjából előnyösebb.

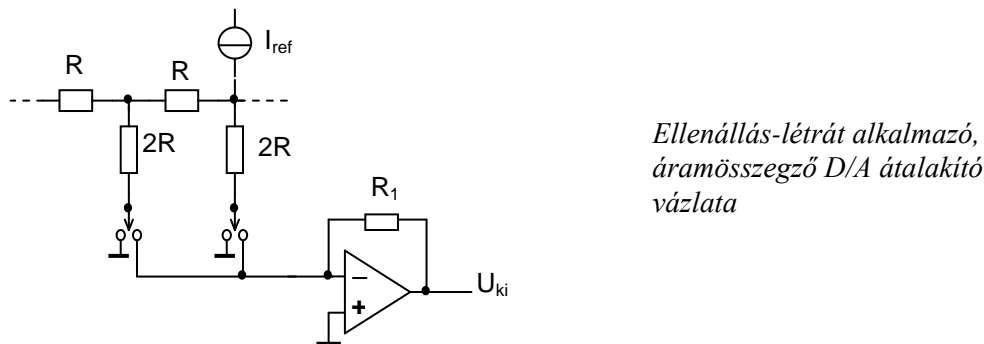
Az előző, az áramösszegzésen alapuló átalakító kapcsán ismertetett hármasszabályt itt is szigorúan be kell tartani. Ez a jelemben annyit jelent, hogy a kapacitás-háló a megnövelt méretekkel tervezett, párhuzamosan kapcsolt egységnyi kapacitások sokaságából áll, amelyek elhelyezkedése a chip felületén egyenletes eloszlású.

Ellenállás-osztós D/A átalakító. A digitális jelek átalakítása analóg jellé olyan módon is megoldható, hogy előre minden analóg értéket előállítunk és egy nagy, a digitális értékekkel vezérelt kapcsolómátrix-szal választjuk ki a hozzátartozó analóg jelet. Egy ilyen megoldást mutat be a 6.1.5 ábra, amely lényegében ellenállásosztó leágazásait kicsatoló bináris fa. Az ellenállások azonos értékűek, tehát relatív pontosságuk szempontjából mikroelektronikailag jól megvalósíthatók, viszont nagy számuk miatt nagyobb felbontások esetén kedvezőtlenül nagy helyfoglalást jelentenek a chipen.



Hasonló módon ilyenkor a kapcsolók nagy száma is előnytelen. Egy további hátrány a (kis helyfoglalású, s ezért kisértékű ellenállásokból adódó) statikusan felvett áram és az ebből adódó nagy disszipáció. Az egyetlen ok, ami miatt mégis helyenkint alkalmazzák az, hogy egy nagyon ötletes integrálási lehetőség kínálkozik a tervező számára, nevezetesen az nMOS kapcsolóknak az osztó felé eső drain-tartománya (azonos adalékoltságú lévén) lehet maga az adalékolt ellenállás. Ez számottevő helymegtakarítást eredményezhet.

Ellenállás-létrát alkalmazó, áramösszegző D/A átalakító. A jólismert R/2R létra bináris feszültségosztását használja ki az ábrán látható átalakító, amelynek nagy előnye az előzőekben vázolt ellenállásláncot alkalmazó megoldással szemben, hogy az ellenállások száma a bitek számában kifejezett felbontás. A két különböző értékű ellenállás azonos nagyságrendbe esik s így mikroelektronikai megvalósításuk a relatív pontosság szempontjából itt is előnyös.



Mint minden ellenállásokkal felépített átalakítóknál, természetesen itt is hátrány a folyamatos áramfelvétel. A kimeneti feszültség értéke

$$U_{ki} = U_{ref} \frac{R_1}{R} \cdot \sum_{i=0}^{n-1} \frac{D_i}{2^i}$$

ahol D_i a kapcsolók állapotát meghatározó digitális adat. A műveleti erősítő invertáló bemenete itt is virtuális föld, ami elengedhetetlen feltétele a létra működésének.

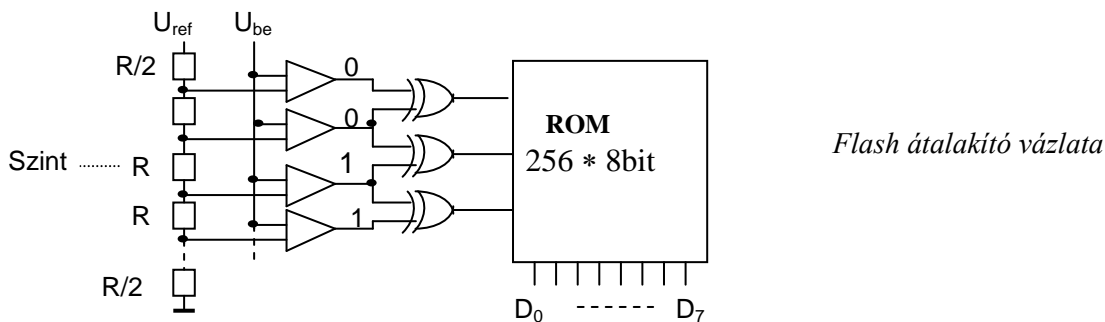
Analóg-Digitál átalakítók. A fokozatos közelítésen (successive approximation) alapuló átalakítók alapja, hogy egy olyan hurkok tartalmaznak, amelyben valamilyen egylépéses D/A átalakító jelet hasonlítjuk össze az

bemeneti analóg jellel. Amennyiben a D/A által szolgáltatott jelet megfelelő stratégiával állítjuk elő, akkor a bemeneti jel viszonylag kevés lépéssel behatárolható. A ma használatos A/D átalakítóknál a közelítés bináris lépésekben történik, amelynek során a) először azt döntjük el, hogy a bemeneti jel kisebb, vagy nagyobb a referencia-feszültség felénél, majd b) ennek ismeretében, hogy az adott fél-tartomány melyik negyedébe esik, c) és így szűkítve a tartományt fokozatosan közelítünk a felbontás által meghatározott legkisebb értékhez. Ebből következően a végrehajtott lépések száma a felbontást meghatározó n -bitszám.

Az egylépéses (flash) átalakítóknál minden diszkrét értéket előállítva ezekkel egy lépésben hasonlítjuk össze a bemeneti analóg jelet.

Az áramkörök egy külön, de igen népszerű családját alkotják a szigma-delta átalakítók, amelyek általában 1-bittal végzik az összehasonlítást, de ennek áraként a szokásosnál jóval nagyobb sebességgel történik a jel mintavételezése.

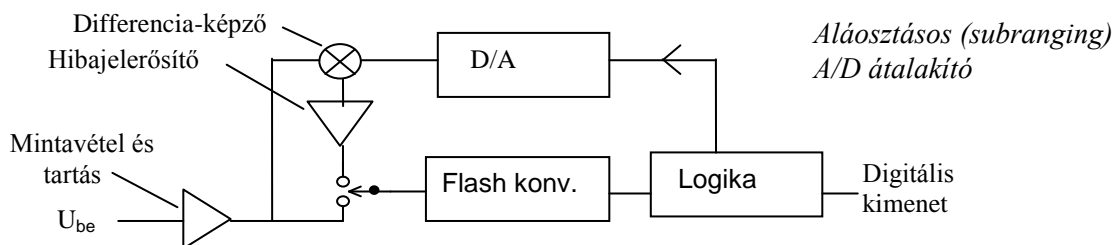
Flash A/D átalakító. Gyors átalakításokra az egylépéses ún. flash áramkörök használatosak, amelyeknek egy jellegzetes elrendezését az ábra mutatja be. Az áramkör egy ellenálás-lánccal az összes diszkrét értéket előállítja, amelyből komparátorok és XOR kapuk segítségével választja ki azon kettő közé eső szintet, ahová a bejövő digitalizálandó feszültség esik. Ezen érték egy ROM memória egy sorát aktíválva a ROM tartalomtól függő, tetszőleges kódban kaphatjuk meg a digitalizált értéket.



Az ábrán megfigyelhetjük, hogy az U_{be} feszültségnél kisebb értékű leágazásoknál a komparátor kimenete "1", míg a nagyobb értékeké "0", következésképp az XOR kapuk között csupán egy lesz, amelynek eltérő a bemenete s így kimenetén "1"-et ad. A komparátorok kimeneti "1" értékeit úgy is tekinthetjük, mint a hőmérő higanyszálát – innen ered az angol "thermometer" elnevezése ennek az átalakítónak.

A flash átalakító legfőbb problémája, hogy nagy bitszámnál a 2^n ellenállás elhelyezése a chipen igen nehézkes s ezért inkább kisfelbontású változatai használatosak.

Aláosztásos (subranging) átalakító. Az egyik legkorszerűbb A/D átalakító típus a subranging áramkör, amelyet az ábra mutat be. Ennek lényege, hogy az átalakítás két lépésben hajtódik végre. Az első lépésben a magasabb helyértékű (MSB) biteket határozzuk meg egy flash konverterrel, majd az így kapott digitális értéket vissza-alakítva, kivonva az eredeti jeltől és a maradékot felerősítve, (a kapcsolót átváltva) ezt újból digitalizáljuk, megkapva ezzel az LSB biteket.



A helyes működés szempontjából igen fontos, hogy az analóg U_{be} jel mintavételezése után, a tartás során a jel értéke ne változzék, továbbá, hogy mind a különbség-képzés, mind a hibajel-erősítés igen pontos legyen. Mivel mind a flash, mind a D/A átalakítás egy lépésben történik, az analóg differencia-képző és erősítő járulékos késleltetésétől eltekintve az átalakítás két lépésben megoldható, a felbontás viszont duplája a felhasznált flash áramkör bitszámának.

Pipeline működésű subranging A/D átalakító. Az előzőekben tárgyalt alóosztásos átalakító sebességét megnövelhetjük, kihasználva, hogy az áramkörben a flash A/D és a visszavezetési ágban levő D/A sohasem dolgozik egyszerre és ennek következtében pipeline működés valósítható meg. Ehhez nem kell más, mint a bemeneti kapcsoló átformalása ill. kiegészítése a 6.2.4. ábrának megfelelően. A működés a következő ütemezés szerint megy végbe:

1) a K2 kapcsoló 1-es állásában a jelfolyam $U_{be}^{(i)}$ i-edik mintavett jelét a flash A/D átalakítóra kapcsoljuk konverziót végzünk az MSB bitekre,

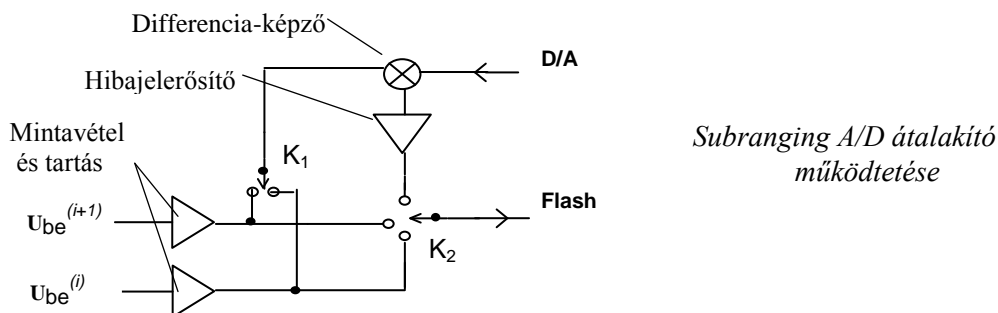
2) a következő ütemben az $U_{be}^{(i)}$ jel digitalizált értékének MSB bitjei visszakerülnek D/A konverzióra, de eközben a K2 kapcsolót 2-es állásba átváltva a következő $U_{be}^{(i+1)}$ mintavett jelet továbbítjuk flash A/D átalakításra,

3) az $U_{be}^{(i)}$ jel MSB bitjeiből visszaalakított analóg jelet a K1 kapcsoló 1-es állásában kivonjuk a bemeneti jelből, felerősítjük, és a K2 kapcsoló 3-as állásában ezt a maradékot újból visszavezetjük a flash konverterre az LSB bitek meghatározására, összeállítva a teljes digitalizált kimeneti jelet, miközben az $U_{be}^{(i+1)}$ jel digitalizált MSB értékein D/A átalakítást végzünk,

4) a K2 kapcsoló 3-as, valamint a K1 kapcsoló 2-es állásában most az $U_{be}^{(i+1)}$ jel kivonását, erősítését, az LSB bitekre történő flash A/D átalakítását, valamint a teljes digitalizált jelnek a kimenetre való továbbítását végezzük el,

5) egy új ciklus eső lépéseként a flash átalakítóra vezetjük a következő, (i+2)-ik mintavett jelet.

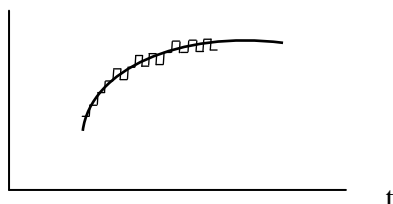
Mint látható, már ebben az egyszerű elrendezésben is a két jel átalakítására 6 helyett csak 5 ütemet használtunk fel. Egyes elemek multiplikálásával a sebesség tovább növelhető és ezzel lehetővé válik közel 1GHz-es jelek digitalizálása is.



Subranging A/D átalakító működése

Sigma-delta ($\Sigma\Delta$) átalakító. Az 1-bites átalakító igen gyakori mintavétellel rendre összehasonlítja a visszaalakított értéket a mért értékkel és a különbségtől függően egy Δ -nyi mennyiséget hozzáad ill. kivon az aktuális értékből, és ennek megfelelően logikai „1” ill. „0” értéket ad a bitfolyamhoz. Elegendően gyakori mintavétel esetén (vagyis elegendően kis Δ mennyiségnél) a lépcső jól közelíti a függvényt. Aluláteresztő szűrővel a lépcsőzöttség kisimítható.

amplitúdó



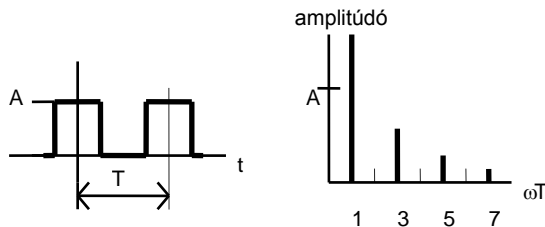
A függvény lépcsős közelítése

8. Távközlési hálózatok áramkörei és jellemzőik

Sávszélesség. A megvalósítható átviteli sebesség (amely a hálózat egyik leglényegesebb jellemzője) a hálózat sávszélességétől függ. Mellőzve a digitalizált jelek spektrum-analízisének részletes tárgyalását, a sávszélesség hatását a 2.1 ábrán kívánjuk szemléltetni. Az *a*) ábra egy periodikus, T ismétlődési frekvenciával rendelkező, A -amplitúdójú impulzus-sorozat időfüggvényét mutatja be, amely az 1,0,1,0... logikai jelnek felel meg. A *b*) ábrán ezen jel frekvenciaspektruma látható, amely a Fourier-sorfejtés alapján az

$$f(t) = \frac{4A}{\pi} \left[\cos(\omega_0 T) \cdot \frac{1}{3} \cos(3\omega_0 T) \cdot \frac{1}{5} \cos(5\omega_0 T) \right]$$

közelítő (csak az első három komponens tartalmazó) összefüggésből adódik. Ha az átvivő hálózat sávszélessége kisebb, mint $3\omega_0$ (vagyis csak az alapharmonikust viszi át), akkor a négyszögjel helyett az alapharmonikusnak megfelelő koszinusz-függvényt kapunk (ezt alkalmazzák a szinkronizáció során, lásd később a *preamble* jelet az Ethernet csomagban). Ahogy szélesítjük a sávot a magasabb felharmonikusok bevonásával, úgy "szögletesedik" a jel és közelít egyre inkább a négyszögjelhez. A megbízható átvitel céljából ezért a hálózat eredő (*overall*) sávszélességét az alapharmonikus frekvenciájánál nagyobbra szokás választani.



A logikai 1,0,1,0 impulzus-sorozat a) időfüggvénye és b) frekvencia-spektruma

Adatátviteli sebesség. Digitális hálózatokat az adatátviteli sebességgel, vagyis az időegység alatt átvitt bitek számával jellemezhetjük, amit bit/s-ben adnak meg. Szokás még az átvitelt az ún. *jelzési sebesség*-gel megadni, vagy közismert néven *baud rate*-el. $1 \text{ baud} = \log_2 N \text{ bit/s}$, ahol N a kódolásban használt jelszintek száma. Ha az átvitelt kétállapotú jelekkel valósítjuk meg, akkor a baud-rate és a bit/s azonos számértéket adnak. Ha azonban a jelet négy szint felhasználásával vesszük át, akkor az adott baud rate-hez kétszeres bit/s-ban megadott adatátviteli sebesség tartozik, hiszen egy jel két bit átvitelét jelenti.

Jitter. A jelek késleltetésének változása érthető módon torzítást okoz az információ-átvitelben. A rövididejű (100ms-nál rövidebb idő alatt mérhető) változásokat *jitter*-nek nevezzük. A jitter oka elsődlegesen az elektronika, ahol az egyes fokozatoknál a vezérlő logikai jelek véges-idejű felfutása miatt (az adott küszöbnél történő) átkapcsolás ideje változó.

Zaj. Az átviteli csatorna zaja (ebbe beleérve az eszközök elektronikus zaján túlmenően a külső zavarokat is) elsődlegesen a jel amplitúdóját módosítja és így lerontja a jel érzékelésének megbízhatóságát. A zaj mértékét az *S/N jel/zaj-viszony* (signal-to-noise ratio) fejezi ki. Egy digitális átviteli rendszer megbízhatóságának mérőszáma *bithiba-arány* (Bit Error Rate, BER), amely az időegység (általában egy másodperc) alatt mért

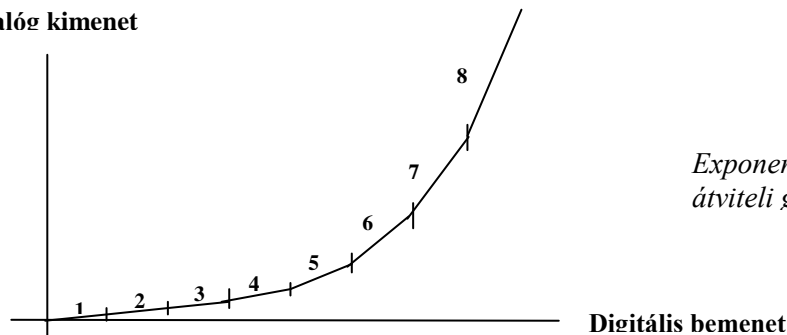
$$BER = \frac{\text{Hibásan vett bitek száma}}{\text{Összes adott bitek száma}}$$

Távközlési hálózatoknál alkalmazott áramkörök. A digitális átvitel alapvetően két csoportba osztható, nevezetesen a szinkron és az aszinkron átvitelre. *Szinkron* (*Synchronous Transfer Mode, STM*) átvitelnél az átvitt információs egység (pl. karakterek) azonos ütemben követik egymást, közöttük tehát idő-megszakítás nincs. Ezt úgy is mondhatnánk, hogy az átvitel előre lefoglalt időszelvényekben történik. A szinkron működést a közös órajel biztosítja, amit egyes esetekben külön csatornán (vezetéken) juttatnak el a vételi oldalra (mint pl. többvezetékes, ún. bit-párhuzamos rendszereknél). Szinkron átvitelt szoktak használni (a folyamatos üzem végett) a digitális 64 kb/s-os telefon, a telefax, valamint a nem tömörített TV -jelek átvitelére.

Aszinkron átvitelnél az információs egységek (karakterek, jel-csomagok) között szünetek lehetnek; az adatok kezdetét START, végét STOP bitek ill. karakterek jelzik. Ezek az ún. Start-Stop átviteli módok. Az adatok vétele természetesen a jel ciklusidejének megfelelően kell történnék, ezért ezt a vételi oldalon a ciklusidőt pontosan elő kell állítani. Ennek egyik módja a vevőoldalon egy nagy pontosságú oszcillátor (ezt alkalmazzák pl. az RS232-rendszerrel), vagy a ciklusidő visszaállítása magából az adatból (ha az megfelelő kódolással elegendő számú jel-átmenetet tartalmaz).

Dinamika kompresszió. A telefonhang dinamikájának lineáris átvitelére az alkalmazott 8bit kevés és ezért átvitel előtt egy logaritmikus görbe szerinti kompressziót használnak. A vevőoldalon a visszaállítás az ábra szerinti exponenciális görbével történik, amely 8 húr tartalmaz. Az átvitt 8 bitből 3 ezen húr valamelyikét jelöli ki, a fennmaradó 5 bit pedig egy-egy húr belül lineárisan 32 diszkrét értéket ad meg.

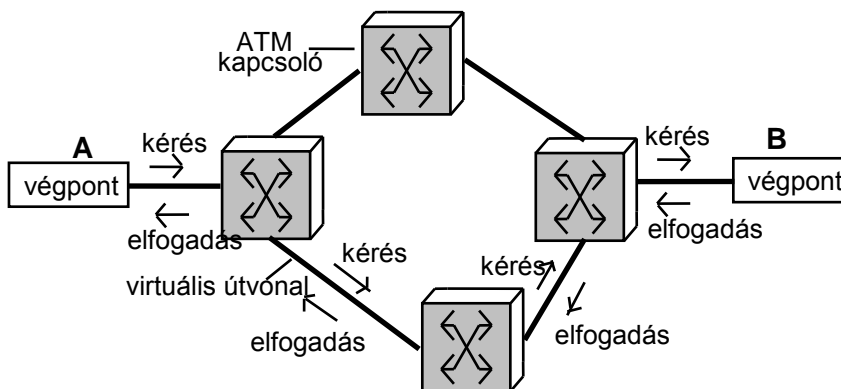
Analóg kimenet



Exponenciális karakterisztikájú
átviteli görbe

ATM hálózatok. Az ATM hálózatok egy olyan hierarchikusan felépített, összeköttetés-alapú rendszerek, amelyekben a csomagok lényegében állandó késleltetéssel vihetők át.

A hierarchikus felépítés azt jelenti, hogy az összeköttetés két szinten határozható meg, nevezetesen a látszólagos útvonal (*virtual path*) és a látszólagos csatorna (*virtual channel*) megadásával; az előbbi gyakorta virtuális áramkörnek is nevezik. A virtuális csatornákat egy-egy virtuális útvonalba nyalábolják össze és ezeket az ún. ATM-kapcsolókkal összefogva alakul ki a hálózat, amint azt az ábra mutatja vázlatosan. Látható, hogy az ATM látszólagos kapcsolatok és látszólagos csatlakozások összefűzött sorozata.

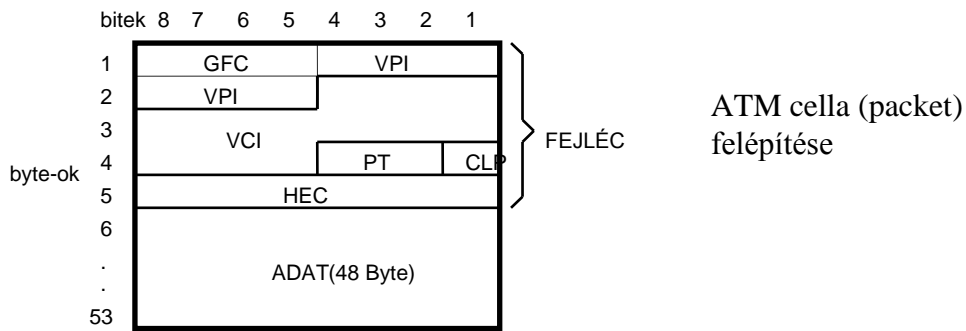


ATM hálózat vázlat

A hálózat a két aktuális végpont (az adó és a vevő) között látható több átviteli útvonalat tud megvalósítani. Ezek közül egy adott átvitelre egyet épít fel, mégpedig az adat-átviteli csatornákon kívül, egy külön útvonalon. Ennek során az A-végpont egy kérést (request) juttat el a B-végponthoz, aki ezt elfogadva megtörténik az útvonal kijelölése. Ez lényegében azt jelenti, hogy az útvonal mentén levő kapcsolókban egy ún. kapcsoló-tábla épül fel, amely megszabja az adat-csomag útvonalát. Ezt követően az A-végpont elküldi az üzenet első csomagját, amelynek fejlécében szerepel az *VPI* címke (Virtual Path Identifier, Látszólagos útvonal azonosító), valamint a *VCI* címke (Virtual Channel Identifier, Látszólagos csatorna azonosító), ahogy azt a 2. fejezet 2.7 ábráján bemutattuk. Minden egyes kapcsoló, amelynek a bemenetére kerül az adatsomag, leolvassa ezeket a címkeket, ennek alapján a megfelelő kimenetre kapcsolja a csomagot, miközben a címkeket az új útvonalnak ill. az abban szereplő új csatornának megfelelően átírja. A címke tehát csak az egy szakaszra vonatkozó adatokat hordozza.

Az adatoknak csomagokban történő átvitele lehetőséget ad arra, hogy egy vonalon több különböző, csomagokra bontott információt vigyünk át. Ez az ATM rendszerű telefon-hálózatok alapja, ahol (szemben a korábbi időosztásos rendszerekkel, amelynél minden csatorna egy fix idő-résben működik) az információ csomagokra (*cellákra*) bontva az éppen rendelkezésre álló csatornán (ill. vonal-szakaszokon) jut el a célállomáshoz, függetlenül az időzítéstől. Az időbeliség figyelembevétele (cellák összefűzése, ütemezése) a célállomás feladata, ha erre egyáltalán szükség van.

Az ATM cellák (packet-ek) felépítése szigorúan kötött, amint azt az ábra mutatja: az 53 byte-os cella egy 5 byte-os (vagy másképpen *octet-es*) fejlécszt tartalmaz, ezt követi a 48 byte-os adat.



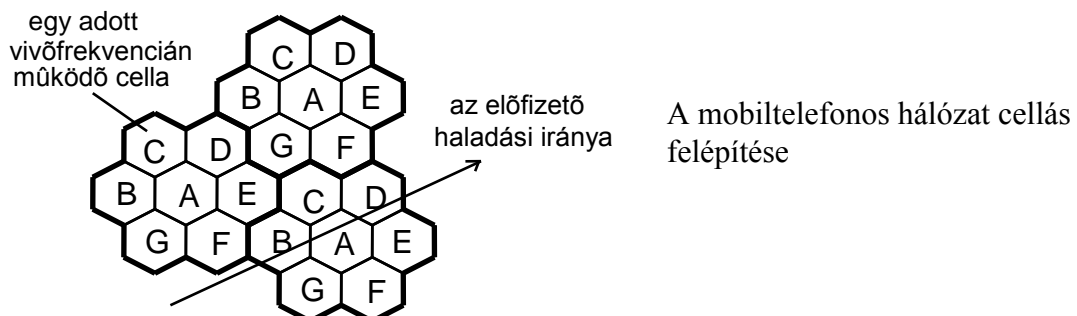
A cella egyes elemei a következők: GFC (általános folyam vezérlő), VPI (virtuális útvonal azonosító), VCI (virtuális csatorna azonosító), PT (adat típusa), CKP (cella-vesztés prioritás), HEC (fejléc hiba-ellenőrzés).

Számítógép hálózatokban alkalmazott áramkörök. A csomagkapcsolásos átvitelre más szabványok is születtek. Az *Ethernet*-hálózatoknál a csomag (cella) hossza jóval nagyobb, ami a nagyobb adat-tömbök átvitelénél előnyösebb. Az *Ethernet* szabvány szerint felépülő hálózatokon küldött adatsomagok szerkezetét az ábra mutatja be.

Preamble	SFD	Célállomás címe	Forrás címe	Hossz	Adat	FCS
62 bit	2 bit	6 bájt	6 bájt	2 bájt	46-1500 bájt	4 bájt

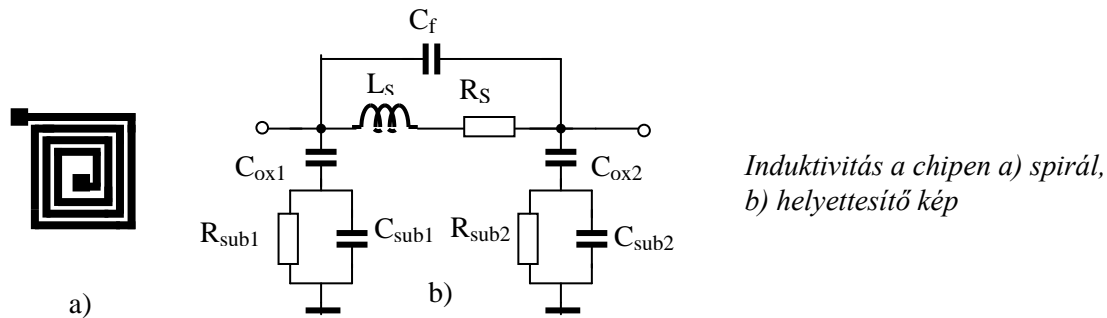
Mint látható, a keret elején egy viszonylag hosszú preamble helyezkedik el, amely egy Manchester-kódolású változó (jelsorozat) és amely a bitidő (órajel) előállítására, vagyis lényegében az (1,0) átmenetek szinkronizálására szolgál. Ha az átvitel során a preamble-ből elvesznének bitek, úgy ezt az állomás a továbbküldés előtt visszaállítja. Ezt követi a *keret indítás* (Start of Frame Delimiter, SFD) mező, amely két logikai „1”, és a tényleges adat kezdetét jelzi. Ezt a küldő-, majd a célállomás címe követi. Ezután az adat hosszát megadó két bájt következik, majd maga az adat, amelynek minimálisan 46 bájt hosszúságúnak kell lennie (ha ennél rövidebb az információ, akkor azt ki kell egészíteni); Az adatot a *Keretellenőrző Sorozat* (Frame Check Sequence, FCS) *hibajelző kódok* zárják. Az *Ethernet*-hálózatoknál a csomag (cella) hossza, mint látható, jóval nagyobb, mint például az *ATM*-csomagok esetében, ami a nagyobb adat-tömbök átvitelénél előnyösebb.

Mobil telefon hálózatok. A cellás rádiótelefon hálózatok működésének alapja az, hogy az ultrarövid-hullámú sávban a jel intenzitása a távolsággal olyan mértékben csökken, hogy adott távolságon túl a kérdéses vivőfrekvencia újból felhasználható összeköttetések létrehozására. A rendelkezésre álló frekvenciasávot vivő-frekvenciákra bontva, kialakítható egy olyan cella-rendszer (topológiájában hasonlóan a méheknél levő hatszögű sejt-rendszerhez), amelynél az egymáshoz közel levő cellákhoz *egymástól eltérő* frekvenciák tartoznak, de bizonyos távolságokon (cella-számon) túl ezek a frekvenciák ismétlődnek; így korlátozott számú frekvenciával egy tetszés szerinti nagyságú terület fedhető le. A valóságban a cellák alakját, elhelyezkedését, egymástól való távolságát az antennák kialakítása és a terjedési viszonyok (sík terep vagy magas házak) szabják meg. Minél magasabb frekvenciasávban üzemel egy rádiótelefon, annál kisebb méretű cellák kellene. A cellák lehetnek 10 km-nél nagyobbak, de vannak 100 m-nél kisebbek is (pl. egy épületen belül). Természetesen alapvető követelmény az, hogy az azonos frekvencián működő cellák jelei egymással ne interferáljanak.



Minden cellában egy bázisállomás helyezkedik el, amely a mobil készülékekkel a kapcsolatot a kijelölt ultrarövid frekvencián tartja. A mozgó előfizetőt a rendszer oly módon követi, hogy egy másik cellába átlépve automatikusan annak egy csatornájára tevődik át a kapcsolat. A bázisállomás adóteljesítménye néhányszor 10W (max. 200W), a mobil készülékeké néhány Watt. A bázisállomások a központtal általában mikrohullámú összeköttetéssel tartják a kapcsolatot. A világon többféle frekvenciasávban működő, egymástól némileg eltérő előírások szerint működő mobil telefonrendszere terjedtek el. Európában ma a GSM rendszer a legelterjedtebb, amely a 900 MHz ill. 1,8 GHz frekvenciasávokban üzemel.

A 900 MHz-es frekvencia már olyan magas, hogy a chipen megjelenő tranzisztor-kapacitásokhoz (ill. az ott egyszerűen megvalósítható monolit kapacitásokhoz) tartozó rezonáns induktivitások értéke már néhányszor 10nH értékűre csökken le, ami a chipen létrehozott fém spirállal megvalósítható. Ennek sematikus rajzát az ábra mutatja be.



Induktivitás a chipen a) spirál, b) helyettesítő kép

Az így kialakított induktivitás jósági tényezője érthető módon igen alacsony ($Q \approx 5-8$), hiszen a tekercs terét alulról közvetlenül egy szilícium-réteg zárja le. Az induktivitás helyettesítő képét a 11.3/b ábra mutatja be, ahol, mint látható, a szilícium felé sorosan helyezkedik el a spirál alatt levő oxidréteg kapacitása, valamint szubsztrát hatását leíró RC-tag. Az induktivitással sorban levő ellenállás a veszteséget, az áthidaló kondenzátor a menetek közti kapacitást veszi figyelembe. A valóságban egy sokmenetes spirálnál több ilyen π -taggal jellemezhető az induktivitás, sőt, elvben minden menethez egy ilyen tag rendelhető. Látható, hogy az így monolit technikával megvalósított induktivitás távol van az ideális reaktanciától, részben a veszteségek, részben pedig a saját rezonanciák miatt. Ezek a tényezők rendkívül megnehezítik tervezést, és a veszteségek erősen korlátozzák a velük létrehozott LC-szűrők szelektivitását.

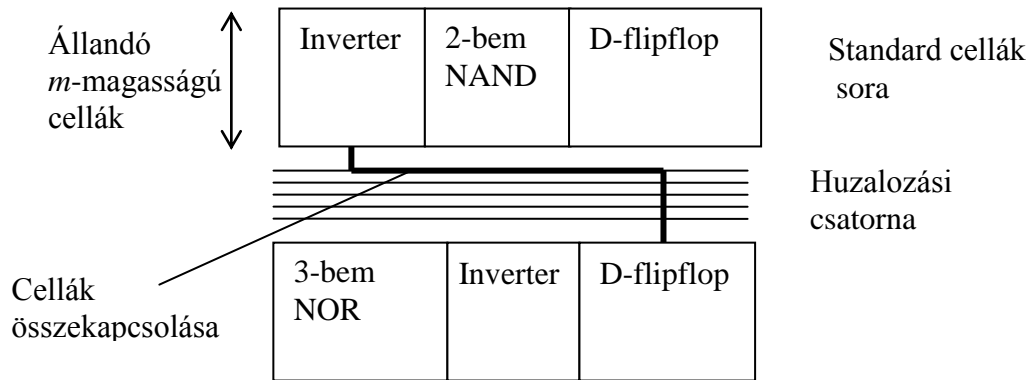
A Bluetooth vezeték nélküli hálózat. A Bluetooth módszer alapját a szórt frekvencia spektrum a gyors nyugtázás és a frekvencia ugrálás (hopping) jelenti, amelynek segítségével még zajos környezetben is képes működni. Amint egy Bluetooth egység vett vagy elküldött egy csomagot, átugrik egy másik frekvenciára (azonos algoritmus alapján ugyanoda ugrik a többi egység is, amelyek közös közlési viszonyban vesznek részt). A frekvencia ugrások 1 MHz-es tartományú, 79 különböző frekvenciaérték között történnek, amely mentes az interferenciától.

9. Áramkörök tervezése.

Megoldási lehetőségek: 1) Nyák+SSI+MSI, 2) Dedikált LSI/VLSI full-custom szilikonra tervezéssel, 3) μP , μC , DSP, 4) FPGA, 5) SoC

Full-custom tervezés: teljesen egzedi elrendezés és egyedi cellák alkalmazása. CAD (Computer-Aided Design) támogatás, de manuális beavatkozás is kell az optimális elrendezéshez. A tervező kész cellákat is felhasználhat, de tervezhet újakat is – de ezeket szimulálni kell SPICE-el és lehetőleg verifikálni chipen is. Jó helykihasználás érhető el, de nagy a hibalehetőség (javító ciklusok...!), hosszabb idő.

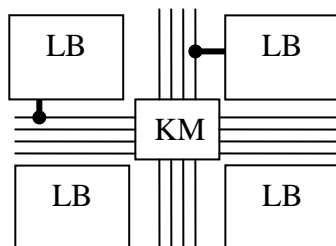
Cellás tervezés: adott magasságú cellák egy sorban, amelyeket összekötési csatornákkal kapcsolnak össze. A CAD rendszer automatikusan elhelyezi és összeköti a cellákat. Hely-kihasználása rosszabb, mint a full custom tervezésé, de gyors és megbízható.



Mikroprocesszoros realizálás. Előnye, hogy könnyen módosítható, de a végrehajtési idő sokkal hosszabb, mint kapukkal és tárolókkal megvalósítva.

FPGA (Field Programmable Gate Array) áramkörök. Összefoglaló néven ASIC, Application Specific Integrated Circuit. Az áramköri feladatok megvalósításakor alapvető fontosságú, hogy mindezt a fejlesztő saját maga a laborjában („field”), a technológiai laboratóriumok (félvezetőgyárak) bevonása nélkül, önállóan és lehetőleg gyorsan végezhesse el. Ezt szolgálják az egyre nagyobb szerepet kapó felhasználó orientált ASIC áramkörök. Az idők folyamán ezeknek sokféle típusa alakult ki. A legelső áramkörök az ún. *device array*-k voltak, amelyek tranzisztorokat, valamint analóg áramköri feladat megoldására ellenállásokat, diódákat, kapacitásikat, zener-diódát, stb. is tartalmaztak; ezeket az elemeket felhasználó tervének megfelelően ugyan még egy technológiai laborban kellett utólagos fémmezéssel összekapcsolni, de az előállítási folyamat már lényegesen leegyszerűsödött az előre elkészített lemezekkel (ez az ún. *master-slice* technika). A következő lépés a *gate-array* áramkörök megjelenése volt, ahol már előre elkészített kapukat és tárolókat tartalmazott a chip, ám ezeket még mindig technológiai eszközökkel kellett összefémézni.

A lényeges áttörést a technológiai lépések teljes kiiktatására a programozható eszközök kifejlesztése jelentette és ezzel valósult meg a ma széles körben alkalmazott FPGA (Field Programmable Gate Array) áramkörök. Felépítésüket az ábra mutatja: kész logikai blokkokat (LB) kész vezeték-szakaszok kapcsolják össze, a kereszteződéseknél pedig kapcsolómátrix (KM) van.



Az FPGA-knek jelenleg három alapvető megvalósítási formája használatos, nevezetesen 1) a SRAM-tárolással megoldott, 2) az EPROM/EEPROM bázisú, valamint 3) az antifuse megoldású áramkörök.

Mindhárom megoldásnál az áramkör logikai cellákból épül fel. Amelyek invertet, kapukat valamint egy, vagy két flipflop-ot tartalmaz; ezek összekötésével alakítják ki a kívánt kapcsolást, oly módon, hogy ezek a cellák CMOS kapcsolókkal vezeték-szakaszokra kapcsolhatók és ily módon az egyes cellák között a kontaktus létrehozható. A kapcsolók bakapcsolt állapotát a korábban említett három eszköz valamelyike vezérli. Ilyen módon a cellákból tetszés szerinti kapcsolás alakítható ki, tehát az áramkör elektronikusan „konfigurálható”. Az összeköttetés értelemszerűen a szomszédos cellák között valósítható meg a legegyszerűbben, a távolabbi cellák összekötésére ún. *long-range* vonalakat használnak.

A konfigurálás jellege eltérő a három típusnál. A flipflop esetében az adott feladatra beprogramozott konfiguráció elillan a tápfeszültség eltűnésekor (volatile), ami egy intelligens programozó eszköz jelenlétét követeli

meg. Az illékonyosság egyes esetekben (pl. titkosságnál) lehet előnyös is. Az EPROM/EEPROM bázisú vezérlésnél az áramkör *non-volatile*, ami általában könnyebbséget jelent az áramkör-tervező ill. gyártó számára.

A harmadik, ún. *antifuse* megoldás egy speciális, könnyen átüthető szigetelőréteget használ fel, amelynek eredeti szigetelési ellenállása $>1\text{Mohm}$, de a ráadott kb. 18V értékű feszültség hatására átüt és $<300\text{ohm}$ ellenállást mutat, ami lényegében rövidzárnak tekinthető. A lényeges újonság ebben az eszközben, hogy az alacsony átütési feszültséget elviseli az áramkör többi része, mint pl. a dekóder, amely a kijelölt átütendő pontokhoz ezt a feszültséget odajuttatja.

Mint látható, a flipflop-os programozás lehetővé teszi az áramkör működés közbeni átkonfigurálását is, ami bizonyos esetekben igen hasznos lehet; erre a másik két megoldásnál nincsen mód. Az EPROM/EEPROM bázisú megoldásnál az átprogramozás hosszabb (általában több ms) ideig tart és így működés közben az átkonfigurálás nem előnyös. Az eszköz viszont törölhető, vagyis tervezési hiba, változó áramköri feladatok stb. esetén újra felhasználható. Végül az antifuse csak egyszer programozható, hiszen az átütött szigetelő nem állítható helyre. Ez az egyszer használhatóság megnehezíti az alkalmazhatóságot, de ugyanakkor igen előnyös, ha igen megbízható (robosztus) programra van szükség, mint pl. a sugárzásokkal terhelt űrkutatási feladatoknál.

Az FPGA áramkörök tervezését korszerű fejlesztő-rendszerek segítik elő, amelyek az áramkör valamilyen formájú leírásából (VHDL hardver leíró nyelv, ORCAD kapcsolási rajz, stb.) automatikusan implementálják a feladatot az adott chipre (ill. az adott családra), feltéve, hogy az feladat méreténél fogva implementálható az adott chipre.

System-on-Chip (SoC) áramkörök. A felhasználó által programozható áramkörök legfejlettebb formája a SoC, amely a cellákból felépülő gate-array mellett egy intelligens eszközt, egy mikrokontrollert és egy memória-blokkot is tartalmaz. A memóriát a felhasználó particionálhatja oly módon, hogy egyrészt a mikrokontroller használja mint program ill. adat-memóriát, másik részét a gate array-hez rendelheti, amely ezt az ún. *memória-éhes* feladatokhoz használhatja (mivel az egyes cellákban levő 1-2 tároló kevés ezekhez a feladatokhoz). A SoC chip alkalmazását igen korszerű fejlesztő-rendszerek segítik, ám ami jelenleg nem kellően megoldott, az az összetett feladatok szétosztása a mikrokontroller és a gate array között.

Az áramkör egy fejlett típusa: mátrixba rendezett cellákból áll, 4×4 cella alkot egy szub-mátrixot, amelyek mindegyikéhez egy duál-port memória ($32 \times 4\text{bit}$ "free RAM) kapcsolódik. A cellákat bonyolult sínrendszer köti össze egymással ill. a freeRAM-mal, valamint a beégyazott 8-bites μ kontrollerrel:

- ◆ Egy összekötési hálózat segítségével minden cella egy-egy adatvonallal tud kapcsolódni a 8 legközelebbi szomszédja közül kettőhöz, amelyeket elhelyezkedésük szerint az égtájnak megfelelően nevezünk el, így North-West (NW), North-East (NE), stb.,

- ◆ A cellák között egy $3 \times 5 = 15$ vonallal rendelkező sínrendszer helyezkedik el. A három 5-vonalas busz közül ez egyik az ún. *lokális* busz, a másik kettő nagyobb távolságok áthidalására szolgáló ún. *express* busz. A lokális busz horizontális és vertikális szegmensei egyaránt 4-4 cellával köthetők össze; mindkét végükön ún. *repeaterek* (horizontális és vertikális *repeaterek*) helyezkednek el, amelyeknél vagy azonos irányban folytatódik a busz egy-egy vonala, vagy egy rá merőlegesen futó lokális sínre kapcsolódva eltér; végül lehetősége van (amennyiben egy távolabb fekvő cellához kell csatlakoznia) egy *express-busz*hoz kapcsolódni. Az *express-busz* nyolc cellát fog át, de ezeket programozhatóan ki is kerülheti.

A 900 MHz-es sávban működő mobiltelefon-rendszer az előfizetőtől a bázisállomás felé, azaz a *felfelé* történő adatátvitelle (uplink) 124 db frekvencia-csatornát tartalmaz a 890,2-914,8 MHz sávban, $(914,8-890,2)/123 = 200\text{ kHz}$ -es távolságokban. Hasonló felosztású az előfizető felé, azaz a *lefelé* történő átvitel (downlink) is. A rendszer időosztásos nyalábolást alkalmaz oly módon, hogy mindegyik frekvencia-csatorna 8 időrésre van osztva, amelyek mindegyikében időben eltolva 8 adatcsatorna működik. Az így adódó $8 \times 124 = 992$ adatcsatornának azonban csak egy része használható, mert különben frekvencia-konfliktus lépne fel a szomszédos cellákkal.

A 8 időrésben átvitt adatok egy ún. *keretet* alkotnak, amelyekből ezután bonyolult szerkezetű ún. *többszereket* (multiframe) képeznek. Erre azért van szükség, mert a tényleges adatok (vagyis az *információ*) mellett sok más, vezérlésre, keretjelzésre, adattípus-megkülönböztetésre, összehangolásra, stb. szolgáló jeleket is át kell vinni.