9. Logikai alapkapcsolások

A digitális készülékek első pillanatban viszonylag bonyolultnak tűnnek, pedig néhány logikai alapkapcsolás többszöri alkalmazásával kialakíthatók. Az alapkapcsolásokból a feladat megoldásához szükséges rendszert tisztán formális módszerekkel alakíthatjuk ki. Ehhez a Boole-algebra nyújt segítséget, amelyet kapcsolásalgebrának is nevezünk. A következő szakaszokban ezért először a kapcsolásalgebra alapjait ismertetjük.

9.1. Logikai alapfüggvények

A hagyományos algebrával ellentétben, a logikai változók csak két diszkrét értéket vehetnek fel, amit általában *L*-nek vagy logikai 0-nak, ill. *H*-nak vagy logikai 1-nek nevezünk. Ezek a logikai értékek a 0 és 1 számoktól egyértelműen megkülönböztethetők, az összefüggésekből mindig nyilvánvaló, hogy melyikről van szó. A logikai változók között három alapvető kapcsolat van, mégpedig a konjunkció, diszjunkció és a negáció. A számalgebrára támaszkodva a következőképpen definiáljuk őket:

szorzás vagy konjunkció: $y = x_1 \wedge x_2 = x_1 \cdot x_2 = x_1 x_2$,
összeadás vagy diszjunkció: $y = x_1 \vee x_2 = x_1 + x_2$,
tagadás vagy negáció: $y = \overline{x}$.

Ezekre a számítási műveletekre a következő tételek érvényesek [9.1]:

Kommutatív szabály:

$$x_1 x_2 = x_2 x_1 , \tag{9.1a}$$

$$x_1 + x_2 = x_2 + x_1 . \tag{9.1b}$$

Asszociatív szabály:

$$x_1 (x_2 x_3) = (x_1 x_2) x_3 , \tag{9.2a}$$

$$x_1 + (x_2 + x_3) = (x_1 + x_2) + x_3 . \tag{9.2b}$$

Disztributív szabály:

$$x_1 (x_2 + x_3) = x_1 x_2 + x_1 x_3 , \tag{9.3a}$$

$$x_1 + x_2 x_3 = (x_1 + x_2)\ (x_1 + x_3) . \tag{9.3b}$$

Abszorpciós szabály:

$$x_1 (x_1 + x_2) = x_1 , \tag{9.4a}$$

$$x_1 + x_1 x_2 = x_1 . \tag{9.4b}$$

Tautológia:

$$xx = x , \tag{9.5a}$$

$$x + x = x . \tag{9.5b}$$

Negáció:

$$x\bar{x} = 0 , \tag{9.6a}$$

$$x + \bar{x} = 1 . \tag{9.6b}$$

Kettős negáció

$$\overline{(\bar{x})} = x . \tag{9.7}$$

De Morgan-szabályok:

$$\overline{x_1 x_2} = \bar{x}_1 + \bar{x}_2 , \tag{9.8a}$$

$$\overline{x_1 + x_2} = \bar{x}_1\, \bar{x}_2 . \tag{9.8b}$$

Műveletek 1-gyel és 0-val:

$$x \cdot 1 = x , \tag{9.9a}$$

$$x + 0 = x , \tag{9.9b}$$

$$x \cdot 0 = 0 , \tag{9.10a}$$

$$x + 1 = 1 . \tag{9.10b}$$

$$\bar{0} = 1 , \tag{9.11a}$$

$$\bar{1} = 0 . \tag{9.11b}$$

E szabályok közül már sokat ismertünk az algebrából, de a (9.3*b*) és (9.4*a* és *b*), (9.5*a* és *b*), (9.10*b*) számokra nem igaz, és számokra a negáció fogalma sem értelmezett. Pl. a $2x$ és x^2 kifejezések a tautológia következtében a kapcsolásalgebrában nem fordulnak elő.

Felcserélve az egyenletpárokat, látható a dualitás fontos szabálya, ugyanis ha valamelyik azonosságban a konjunkciót diszjunkcióval és egyidejűleg 0-t 1-gyel cseréljük fel, akkor azonosságot kapunk.

A (9.9)...(9.11) egyenletek segítségével lehetővé válik a konjunkció és a diszjunkció kiszámítása x_1 és x_2 változó minden lehetséges értékére. A következő táblázatokban (9.1. ábra és 9.2. ábra) a konjunkció és a diszjunkció igazságtáblázatát adjuk meg.

A konjunkció igazságtáblázatából látható, hogy y csak akkor 1, ha x_1 és x_2 is 1.

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

9.1. ábra. Konjunkció igazságtáblázata

$y = x_1 x_2$

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	1

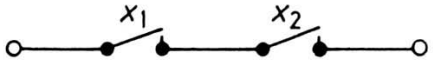
9.2. ábra. Diszjunkció igazságtáblázata

$y = x_1 + x_2$

Ezen az alapon nevezzük a konjunkciót ÉS-függvénynek. A diszjunkció igazságtáblázatában y akkor 1, ha vagy x_1, vagy x_2 értéke 1, ezért a diszjunkciót VAGY-függvénynek nevezzük. Mindkét függvény kibővíthető tetszőleges számú változóra.

A kérdés most az, hogy a logikai műveletek hogyan valósíthatók meg áramkörökkel. Mivel a logikai változóknak csak két diszkrét értéke van, csak olyan áramkörök jöhetnek számításba, amelyek két, jól megkülönböztethető üzemi állapotot vehetnek fel. A logikai változók megvalósításának legegyszerűbb megoldása a 9.3. ábrán látható kapcsoló. Megegyezés alapján a nyitott kapcsolóhoz logikai 0-t, a zárthoz pedig 1-et rendelünk. A K kapcsoló tehát az x változót úgy képviseli, hogy ha $x = 1$, akkor zárva van. Az $\bar{x}$ változó megvalósítására $x = 1$ mellett nyitva van.

9.3. ábra. Logikai változó megvalósítása kapcsolóval

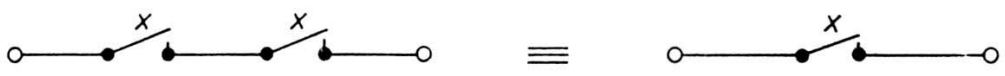


9.4. ábra. ÉS-áramkör

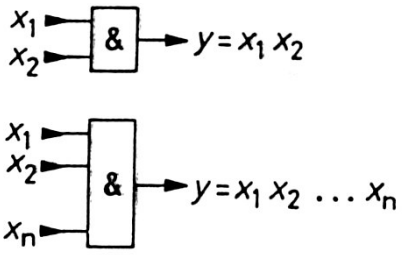
Elsőként határozzuk meg, hogy milyen logikai függvények adódnak, ha az x_1 és x_2 kapcsolót a 9.4. ábrán látható módon sorbakötjük. Az y függő változó értékét azzal jellemezhetjük, hogy az eredő kapcsolóállapot a végpontok között nyitott-e vagy zárt. Mint látható, áram csak akkor folyhat, ha x_1 és x_2 zárva van, ekkor logikai értéke: 1. Következésképpen a kapcsolók soros kapcsolása logikai ÉS-függvényt állít elő. Ennek megfelelően logikai VAGY-függvényt kapunk a kapcsolók párhuzamos kapcsolásával.

A kapcsolókogika segítségével szemléletesen ellenőrizhetjük az előbb megadott tételek érvényességét. Példaként ellenőrizzük a tautológia tételét. A 9.5. ábrán a (9.5*a*) egyenlet két oldalát ábrázoltuk kapcsolók segítségével. Látható, hogy a megadott azonosság teljesül, ha a két soros kapcsolót egyidejűleg zárjuk vagy nyitjuk, vagyis hatásuk kívűlről nézve ugyanolyan, mint egyetlen kapcsolóé.

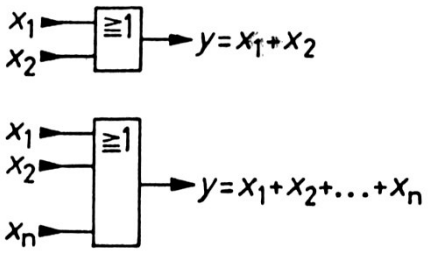
A logikai változók értékét elektromos feszültségek is képviselhetik. Ilyeneket már a 8.1. szakaszban megismertünk. Ott két szintet különböztetünk meg, mégpedig a H és L szinteket, ezekhez most az 1 és a 0 logikai értékeket rendeljük hozzá.



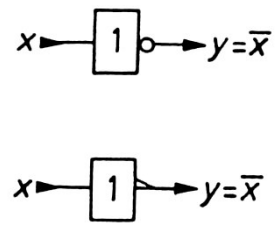
9.5. ábra. A tautológia szemléltetése $xx = x$



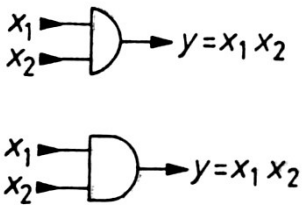
9.6. ábra. ÉS-kapu rajzjele



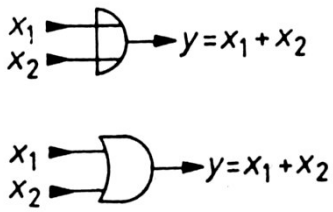
9.7. ábra. VAGY-kapu rajzjele



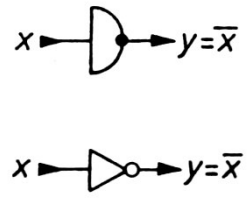
9.8. ábra. NEM-kapu rajzjele



9.9. ábra. ÉS-kapu régi rajzjele



9.10. ábra. VAGY-kapu régi rajzjele



9.11. ábra. NEM-kapu régi rajzjele

A pozitív logika a $H = 1$ és az $L = 0$ megfeleltetést jelenti. Ezzel ellentétes hozzárendelés is lehetséges, ezt negatív logikának nevezzük.

A logikai alapfüggvények erre alkalmas elektronikus áramkörökkel valósíthatók meg. Az ilyen áramköröknek egy vagy több bemenetük és egy kimenetük van. Általános gyűjtőnevük kapu. A bemeneti feszültségszintek és a logikai függvény határozza meg a kimeneti szintet. Mivel nagyon sok elektronikus lehetőség adódik ugyanazon logikai függvény megvalósítására, ezért az egyszerűség kedvéért áramköri jelöléseket vezetünk be, amelyek a belső felépítéstől függetlenül csupán a logikai függvényt jelölik. Ezeket a jelöléseket a 9.6...9.8. ábrán foglaltuk össze. Magyarázó leírás például a [9.2] és [3.4] irodalmakban található.

Mivel a digitális technikában a feszültség, mint fizikai mennyiség érdektelen, és csak az általa képviselt logikai állapot lényeges, ezért a be- és kimeneteket nem U_1, U_2... stb-vel, hanem közvetlenül a logikai változóval jelöljük. (A korábban használt jelöléseket a 9.9...9.11. ábrákon foglaltuk össze.)

9.2. Logikai függvények előállítása

A digitális technikában a feladatokat többnyire függvénytáblázat formájában adjuk meg, amelyet igazságtáblázatnak is nevezünk. Az első feladat az, hogy találjunk olyan logikai függvényeket, amelyek a függvénytáblázatnak eleget tesznek. Ezután az áramkört a logikai alapkapcsolások megfelelő kombinációival megvalósítjuk. A logikai függvények felírására sokszor a diszjunktív normálalakot használjuk. Felírásához a következőképpen járunk el:

1. A logikai függvény diszjunktív normálalakjának meghatározására először minden olyan sort megkeresünk a függvénytáblázatban, amelyben az y kimeneti változó logikai 1 értékű.

2. Minden ilyen sorban képezzük a bemeneti változók konjunkcióját, tehát x_i-t írunk, ha a függvénytáblázatban az illető változónál 1-es áll, egyébként pedig $\bar{x}_i$-at. Ilyen módon éppen annyi konjunkciót kapunk, mint ahány $y = 1$ eset van.

3. A keresett függvényt végül úgy kapjuk meg, hogy az összes konjunkció diszjunkcióját képezzük. Az eljárást a 9.12. ábrán látható igazságtáblázat segítségével mutatjuk be. A 3., 5. és 7. sorban $y = 1$. Először képezzük e sorok bemeneti változóihoz tartozó konjunkciókat.

Sorok	x_1	x_2	x_3	y
1	0	0	0	0
2	0	0	1	0
3	0	1	0	1
4	0	1	1	0
5	1	0	0	1
6	1	0	1	0
7	1	1	0	1
8	1	1	1	0

9.12. ábra. Példa egy igazságtáblázatra

$$3.\ \text{sor:}\ K_3 = \bar{x}_1 x_2 \bar{x}_3\ ,$$

$$5.\ \text{sor:}\ K_5 = x_1 \bar{x}_2 \bar{x}_3\ ,$$

$$7.\ \text{sor:}\ K_7 = x_1 x_2 \bar{x}_3\ .$$

A keresett függvény a konjunkciók diszjunkciójaként adódik:

$$y = K_3 + K_5 + K_7\ ,$$

$$y = \bar{x}_1 x_2 \bar{x}_3 + x_1 \bar{x}_2 \bar{x}_3 + x_1 x_2 \bar{x}_3\ .$$

Ez a keresett logikai függvény diszjunktív normálalakja. Az egyszerűsítéshez alkalmazzuk először a (9.3*a*) egyenletet, akkor a következőt kapjuk:

$$y = [\bar{x}_1 x_2 + x_1 (\bar{x}_2 + x_2)]\bar{x}_3\ .$$

A (9.6*b*) és (9.9*a*) egyenletek további egyszerűsítést tesznek lehetővé:

$$y = (\bar{x}_1 x_2 + x_1)\,\bar{x}_3\ .$$

A (9.3*b*) egyenlet felhasználásával:

$$y = (x_1 + x_2)\,(x_1 + \bar{x}_1)\,\bar{x}_3\ .$$

Végül a (9.6*b*) és (9.9*a*) összefüggés újabb alkalmazásával megkapjuk a végeredményt:

$$y = (x_1 + x_2)\,\bar{x}_3\ .$$

Azokban az esetekben, amikor a sorok többségéhez $y = 1$ kimeneti változó érték tartozik, előnyösebb a konjunktív normálalak. Ekkor azokat a sorokat keressük

meg a függvénytáblázatban, amelyeknél $y = 0$. Minden ilyen sorra képezzük minden bemeneti változó diszjunkcióját úgy, hogy x_i-t írunk, ha a függvénytáblázatban az illető változónál 0 áll, egyébként pedig $\bar{x}_i$-at. Ilyen módon éppen annyi diszjunkciót képezünk, mint ahány sorban $y = 0$. Végül a keresett függvényt úgy kapjuk meg, hogy az összes diszjunkció konjunkcióját képezzük. Alkalmazzuk ezt a módszert az előbbi függvénytáblázat esetére. Először képezzük a sorok diszjunkcióját. Az 1., 2., 4., 6., és 8. sorban $y = 0$.

1. sor: $D_1 = x_1 + x_2 + x_3$,

2. sor: $D_2 = x_1 + x_2 + \bar{x}_3$,

4. sor: $D_4 = x_1 + \bar{x}_2 + \bar{x}_3$,

6. sor: $D_6 = \bar{x}_1 + \bar{x}_2 + \bar{x}_3$,

8. sor: $D_8 = \bar{x}_1 + \bar{x}_2 + \bar{x}_3$.

A keresett logikai függvény konjunktív normálalakja a $D_1 \ldots D_8$ diszjunkció konjunkciójaként adódik:

$$y = (x_1 + x_2 + x_3)(x_1 + x_2 + \bar{x}_3)(x_1 + \bar{x}_2 + \bar{x}_3)(\bar{x}_1 + x_2 + \bar{x}_3)(\bar{x}_1 + \bar{x}_2 + \bar{x}_3).$$

Látható, hogy az y kifejezés lényegesen bonyolultabb, mint a diszjunktív normálalak. Ez azért van így, mert a függvénytáblázatban y öt esetben 0, és csak három esetben 1. Ha a felírt összefüggést a (9.1)...(9.11) tételek alkalmazásával leegyszerűsítjük, akkor a (9.12)-vel egyező végeredményt kapjuk:

$$y = (x_1 + x_2)\bar{x}_3 .$$

A logikai függvények egyszerűsítése során – felhasználva az előzőekben megadott tételeket – könnyen megeshet, hogy nem veszünk észre egy lehetséges egyszerűsítést. Ennek oka főként az, hogy az egyik disztributív szabály (a (9.3*b*) egyenlet) gyakorlat híján nehezen ismerhető fel. Ilyen esetben előnyös, ha az egyszerűsítésre váró egyenlet negáltját képezzük. Bonyolultabb egyenletekre is minden nehézség nélkül alkalmazhatjuk a De Morgan-szabályt, ugyanis csupán a (+) művelet helyett (·) műveletet végzünk és viszont, ezenkívül a változókat és konstansokat negáljuk. Az így átalakított egyenlet többnyire könnyen tovább egyszerűsíthető. Úgy jutunk a végeredményhez, hogy az egész egyenletet még egyszer negáljuk.

9.2.1. Karnaugh-diagram

A Karnaugh-diagram egy másik fontos segédeszköz, amellyel a lehető legegyszerűbb alakú logikai függvények állíthatók elő. Ez a diagram nem más, mint az eredeti függvénytáblázat más elrendezésben. A bemeneti változók értékeit ebben az esetben nem egyszerűen egymás alá írjuk, hanem a sakktáblaszerűen felosztott tábla vízszintes és függőleges szélein helyezzük el. Páros számú bemeneti változó esetén a változók felét az egyik oldalra, a másik felét pedig a háló másik szélére

írjuk. Páratlan számú változó esetén az egyik oldalra egyel több változó kerül, mint a másikra. A bemeneti függvényértékek különböző kombinációinak elrendezését úgy kell végrehajtani, hogy mindig csak egy változó értéke változzon, ha az egyik kockából a szomszéd kockába lépünk át. A táblázatba beírjuk még a táblázat szélein álló bemeneti változók értékeihez tartozó y kimeneti változóértékeket is. A 9.13. ábra az ÉS-függvény igazságtáblázatát két változóra, a 9.14. ábra pedig a hozzá tartozó igazságtáblázatot tartalmazza.

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

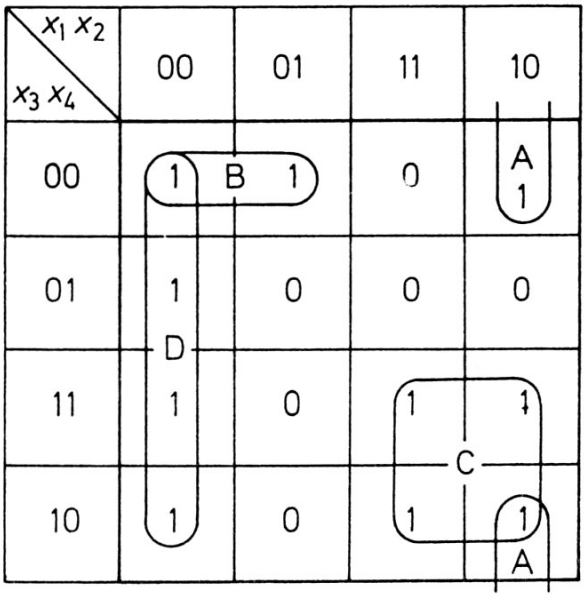
9.13. ábra. ÉS-függvény igazságtáblázata

x_2 \ x_1	0	1
0	0	0
1	0	1

9.14. ábra. ÉS-függvény Karnaugh-diagramja

A Karnaugh-diagram tulajdonképpen az igazságtáblázat leegyszerűsített formájú ábrázolása, amelyből a hozzá tartozó logikai függvény diszjunktív normálalakja a már leírt módon megkapható. A módszer előnye az, hogy a lehetséges egyszerűsítések könnyen felismerhetők. Ezt a gyakorlatban a következő táblázat és a 9.15. ábra példájával szemléltetjük:

x_1	x_2	x_3	x_4	y
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1



9.15. ábra. Igazságtáblázat a hozzá tartozó Karnaugh-diagrammal

Állítsuk elő a diszjunktív normálalakot. Ehhez először a korábban leírtak szerint minden olyan mezőnek a bemeneti változókból álló konjunkcióját képezzük, ahol 1 áll. A bal felső mezőre a következő konjunkció adódik:

$$K_1 = \bar{x}_1 \bar{x}_2 \bar{x}_3 \bar{x}_4 .$$

A tőle jobbra levő mező konjunkciója:

$$K_2 = \bar{x}_1 x_2 \bar{x}_3 \bar{x}_4 .$$

Ha végül az összes konjunkció diszjunkcióját képezzük, akkor többek között a következő kifejezés szerepel az eredményben:

$$K_1 + K_2 = \bar{x}_1 \bar{x}_2 \bar{x}_3 \bar{x}_4 + \bar{x}_1 x_2 \bar{x}_3 \bar{x}_4 .$$

A kifejezés egyszerűsíthető:

$$K_1 + K_2 = \bar{x}_1 \bar{x}_3 \bar{x}_4 (\bar{x}_2 + x_2) = \bar{x}_1 \bar{x}_3 \bar{x}_4 .$$

Ebből látható a Karnaugh-diagram általános egyszerűsítési szabálya. Ha a Karnaugh-diagram szomszédos mezőiben 1-esek állnak, akkor a diszjunktív normálalak előállítása során csak azoknak a bemeneti változóknak a konjunkciója szerepel, amelyek a szomszédos mezők minden csoportjában közösek. Példánkban eszerint, a 9.15. ábra alapján, a B két elemes csoportra a

$$K_B = \bar{x}_1 \bar{x}_3 \bar{x}_4$$

egyszerűsített eredményt közvetlenül kapjuk, amely megegyezik az előzőekben megadott függvénnyel. Azok a mezők is belevonhatók egy ilyen csoportba, amelyek a bal és jobb oldali szélen egy sorba vagy a felső és alsó szélen egy oszlopba esnek. Például a 9.15. ábrán így kaptuk a negyedik oszlop alsó és felső mezejében található egyenesekből az A kettős csoportot. Erre a következő eredményt írhatjuk fel:

$$K_A = x_1 \bar{x}_2 \bar{x}_4 .$$

Ugyanilyen egyszerűsítési szabály érvényes egy csupa 1-esből álló sorra vagy oszlopra, pl. a 9.15. ábrán látható D csoportra. Erre a

$$K_D = \bar{x}_1 \bar{x}_2$$

összefüggést kapjuk. A harmadik fontos eset, amire az egyszerűsítő szabály alkalmazható, a 9.15. ábrán látható C négyes csoport, amelyre:

$$K_C = x_1 x_3 .$$

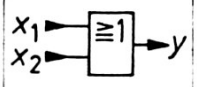
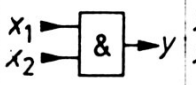
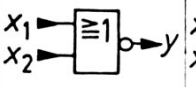
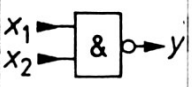
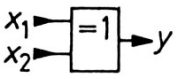
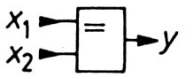
A diszjunktív normálalakra a következő leegyszerűsített végeredményt kapjuk:

$$y = K_A + K_B + K_C + K_D ,$$

$$y = x_1 \bar{x}_2 \bar{x}_4 + \bar{x}_1 \bar{x}_3 \bar{x}_4 + x_1 x_3 + \bar{x}_1 \bar{x}_2 .$$

9.3. Származtatott alapfüggvények

A korábbi fejezetekben bebizonyítottuk, hogy tetszőleges logikai függvény a VAGY-, ÉS-, NEM-alapfüggvények megfelelő kombinációjával előállítható. A származtatott alapfüggvények egyes tagjai a kapcsolástechnikában olyan gyak-

Bemeneti változók		$y=x_1+x_2=$ $=x_1$ VAGY x_2	$y=x_1x_2=$ $=x_1$ ÉS x_2	$y=\overline{x_1+x_2}=$ $=x_1$ NV x_2	$y=\overline{x_1x_2}=$ $=x_1$ NÉ x_2	$y=x_1+x_2=$ $=x_1$ KIZV $x_2=$ $=x_1$ ANTIV x_2	$y=\overline{x_1+x_2}=$ $=x_1$ KIZNV $x_2=$ $=x_1$ EKVIV x_2
x_1	x_2						
0	0	0	0	1	1	0	1
0	1	1	0	0	1	1	0
1	0	1	0	0	1	1	0
1	1	1	1	0	0	0	1
							

9.16. ábra. ÉS- ill. VAGY-függvényekből származtatott alapfüggvények

ran fordulnak elő, hogy megkülönböztetésül külön nevet is kaptak. Igazságtáblázatukat két bemeneti változó esetén és áramköri jelölésüket a 9.16. ábrán összefoglaltuk. A NEM – VAGY – (NV) és NEM – ÉS – (NÉ)-függvényeket a VAGY- ill., ÉS-függvények negációjával kapjuk. (NEM – VAGY-not or; NEM – ÉS-not and):

A táblázat alapján:

$$x_1 \text{NV} x_2 = \overline{x_1 + x_2} = \bar{x}_1 \bar{x}_2 , \tag{9.12}$$

$$x_1 \text{NÉ} x_2 = \overline{x_1 x_2} = \bar{x}_1 + \bar{x}_2 . \tag{9.13}$$

Az ekvivalenciafüggvény esetén y akkor 1, ha minden bemeneti változó egyenlő értékű. A diszjunktív normálalak képzésekor az igazságtáblázatból az

$$y_3 = x_1 \text{EKVIV} x_2 = \bar{x}_1 \bar{x}_2 + x_1 x_2$$

összefüggést kapjuk. Az antivalenciafüggvény az ekvivalenciafüggvény negáltja, amelynél y csak akkor 1, ha a bemeneti változók különbözők. A diszjunktív normálalak:

$$y_4 = x_1 \text{ANTIV} x_2 = \bar{x}_1 \bar{x}_2 + x_1 \bar{x}_2 .$$

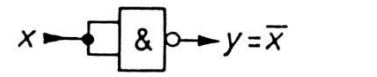
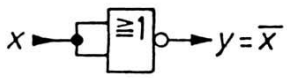
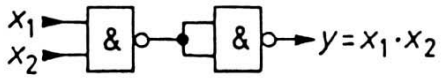
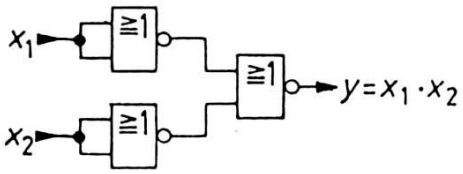
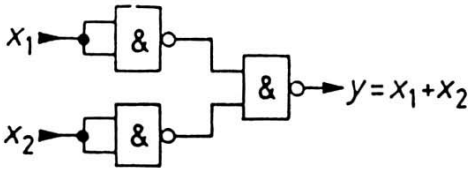
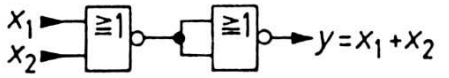
Az antivalenciafüggvénynek még egy másik jelentése is adódik az igazságtáblázatból: az hogy a VAGY-függvénnyel azonos, kivéve, ha minden bemeneti változó értéke: 1. Ezért kizáró VAGY-függvénynek is nevezzük.

Ennek megfelelően az ekvivalenciafüggvény is kizáró NEM – VAGY-függvényként jelölhető.

Ha integrált áramkörökkel építünk fel egy áramkört, gyakran elkerülhetetlen, hogy kizárólag NEM – ÉS- vagy NEM – VAGY-kaput használjunk. Ehhez a logikai függvényt úgy kell átalakítani, hogy csak a használható alapfüggvények szerepeljenek benne. Ezt a legegyszerűbb módon úgy tehetjük meg, hogy először felírjuk az összefüggéseket az alapfüggvényekkel. Az ÉS-függvényre:

$$x_1 x_2 = \overline{\overline{x_1 x_2}} = \overline{x_1 \text{NÉ} x_2} ,$$

$$x_1 x_2 = \bar{\bar{x}}_1 \bar{\bar{x}}_2 = \overline{\bar{x}_1 + \bar{x}_2} = \bar{x}_1 \text{NV} \bar{x}_2 .$$

Logikai függvény	Kapufajta	
	NEM – ÉS	NEM – VAGY
NEM		
ÉS		
VAGY		

9.17. ábra. Alapfüggvények megvalósítása NEM – VAGY- és NEM – ÉS-kapukkal

A VAGY-függvényre:

$$x_1 + x_2 = \overline{\overline{x}_1 + \overline{x}_2} = \overline{\overline{x_1 x_2}} = \overline{x}_1 \text{NÉ} \overline{x}_2 ,$$

$$x_1 + x_2 = \overline{\overline{x_1 + x_2}} = \overline{x_1 \text{NV} x_2} .$$

Ebből kapjuk a 9.17. ábrán feltüntetett megoldási lehetőségeket.

9.4. Logikai alapáramkörök kapcsolástechnikai megvalósítása

Az előző szakaszokban már logikai áramköröket használtunk anélkül, hogy belső felépítésüket ismernénk. Ez a tárgyalási mód azért indokolt, mert napjainkban a digitális áramkörtechnikában majdnem kizárólag integrált áramköröket használnak, amelyeknek tápfeszültség-csatlakozási pontokon kívül csak bemeneti és kimeneti kivezetései vannak.

Egy-egy alapáramkör megvalósítására egész sor áramkörtechnikai megoldás létezik, amelyek teljesítményfelvételben, tápfeszültségigényben, *H* és *L* szintben, sebességben és kimeneti terhelhetőségben térnek el egymástól. Az áramkörtípusok helyes megválasztása érdekében, ismerjük meg nagy vonalakban belső felépítésüket is. A következő pontokban összefoglaljuk a legfontosabb áramkörcsaládokat.

Az integrált áramkörök összekapcsolása során gyakran több bemenetet kell egy kimenetre kapcsolni. Mivel csak azonos áramkörcsaládhoz tartozó integrált áramkörök jöhetnek számításba, ezért megadható, hogy egy kimenet maximálisan hány bemenettel terhelhető. Ez a szám a kimeneti terhelhetőség (fan out). Előfordulnak olyan bemenetek is, amelyek a kimenetet a szokásosnál jobban terhelik. Az

ilyen áramkörök bemeneti terhelése (fan in) nagyobb, mint 1 egységterhelés. A digitális áramkörök összekapcsolása során valamely kimenetet terhelő bemeneti egységterhelések összege nem lehet nagyobb, mint amit az illető áramkör maximális kimeneti terhelhetősége megenged. A 10-es kimeneti terhelhetőség tehát azt jelenti, hogy a kimenet 10 egységterhelést visel el. Ha a kimeneti terhelhetőség nem megfelelő, akkor az egyszerű kapu helyett egy teljesítményfokozatot (buffert) kell közbeiktatni.

Egy kapu minden bemeneti állapotához meghatározott kimeneti állapot tartozik. A 8. fejezetben leírtak szerint ezeket az állapotokat *H* és *L* jellel különböztetjük meg attól függően, hogy a feszültség U_{max}-nál nagyobb, vagy U_{min}-nél kisebb. Valamely kapu működésének szinttáblázata a 9.18. ábrán látható.

U_1	U_2	U_{ki}
L	L	H
L	H	H
H	L	H
H	H	L

9.18. ábra. Példa a szinttáblázatra

x_1	x_2	y
0	0	1
0	1	1
1	0	1
1	1	0

9.19. ábra. Pozitív logikájú igazságtáblázat: NEM – ÉS-függvény

x_1	x_2	y
1	1	0
1	0	0
0	1	0
0	0	1

9.20. ábra. Negatív logikájú igazságtáblázat: NEM – VAGY-függvény

A kapu által megvalósított logikai függvény azonban nem egyértelmű, ha a szintállapot és a logikai állapot közötti kapcsolat nincs tisztázva. A hozzárendelés önkényes is lehet, de egy készüléken belül egységesnek és értelemszerűnek kell lennie. A

$$H \triangleq 1 \quad \text{és} \quad L \triangleq 0$$

összerendelést pozitív logikának nevezzük. Példánkban a 9.19. ábrán látható igazságtáblázatot kapjuk, amely a NEM – VAGY-függvénnyel azonosítható. A

$$H \triangleq 0 \quad \text{és} \quad L \triangleq 1$$

összerendelést negatív logikának nevezzük. Negatív logika esetén példánkra a 9.20. ábrán látható igazságtáblázatot kapjuk, ami nem más, mint a NEM – VAGY-függvény.

Egy és ugyanazon áramkör a logika megválasztása szerint egyszer NEM – VAGY-, másszor pedig NEM – ÉS-kapcsolásként működik. Ha a megvalósítandó függvény adott, és eldöntöttük, hogy melyik áramkörcsaláddal valósítjuk meg, akkor azt a logikát alkalmazzuk, amelyikkel egyszerűbb a kapcsolás. Ha negatív logikára térünk át, a függvényeket a következőképpen kell megcserélnünk:

NV <=> NÉ,
VAGY <=> ÉS,
NEM <=> NEM.

erősítőt használunk komparátorként. Ilyen áramkörök a kereskedelemben vonali vevő (line receiver) néven kaphatók. Az áramköri elrendezést a 9.51. ábra szemlélteti.

9.6. Kombinációs logikai hálózatok

Kombinációs hálózatnak nevezünk minden állapottárolót nem tartalmazó digitális áramköri elrendezést. A kimeneti változók értékeit egyértelműen meghatározzák a bemeneti változók értékei. Ez a különbség a szekvenciális hálózatokkal szemben, amelyeknél a kimeneti állapotot az előző állapotok is befolyásolják.

A kimeneti változók hozzárendelését a bemeneti változókhoz az igazságtáblázat vagy Boole-függvények segítségével írhatjuk le. Áramköri realizálásuk, pl. ROM-mal (csak olvasható tár, *R*ead *O*nly *M*emory) valósulhat meg, amelyek az igazságtáblázatot tárolják. Ezeknél a bemeneti változók mint címek szerepelnek. A másik lehetőség kapuk alkalmazása vagy programozott kapuelrendezés (PAL) használata, amelyeket a Boole-függvények szerint alakítunk ki. A továbbiakban néhány egyszerű kapcsolást ismertetünk. Azokkal az áramkörökkel, amelyeket digitális számítási feladatok elvégzésre alakítanak ki, a 19. fejezetben foglalkozunk.

A kombinációs hálózatokat főleg a számokkal végzett aritmetikai műveletek elvégzésére és számok átkódolására használjuk. Ahhoz, hogy a számokat logikai változók segítségével ábrázolhassuk, a kétértékű (bináris) információk sorozatát kell felírni. A bináris szám egysége a bit. A bit tehát egy bináris helyértékhez rendelt szám. Egy speciális bináris számábrázolás a bináris kód, amelynél növekvő helyértékekhez kettő növekvő hatványait rendeljük. Az 1-es számmal a logikai 1-et jelöljük, a 0-val pedig a logikai 0-t. A logikai változókat, amelyekkel az egyes helyértékeket jellemezzük, kisbetűvel jelöljük. Az n helyértékű szám ábrázolása bináris kódban a következő:

$$x_n = x_{n-1} \cdot 2^{n-1} + x_{n-2} \cdot 2^{n-2} + \ldots + x_1 2^1 + x_0 2^0 .$$

Természetesen mindig egyértelműen meg kell különböztetni, hogy adott műveletet számokkal, vagy logikai változókkal kell-e elvégezni. A különbséget példával tesszük érthetőbbé. Számítsuk ki az $1+1$ kifejezést. Ha feltételezzük, hogy a $(+)$ műveleti jel decimális számrendszerben összeadási utasítást jelöl, akkor az

$$1+1 = 2$$

eredményt kapjuk.

Ezzel szemben a kettes számrendszerben az eredmény:

$$1+1 = 10 \text{ (olvasd: egy, nulla)} .$$

Ha feltételezzük, hogy a $(+)$ műveleti jel a logikai változók diszjunkcióját jelenti, akkor ez adódik:

$$1+1 = 1$$

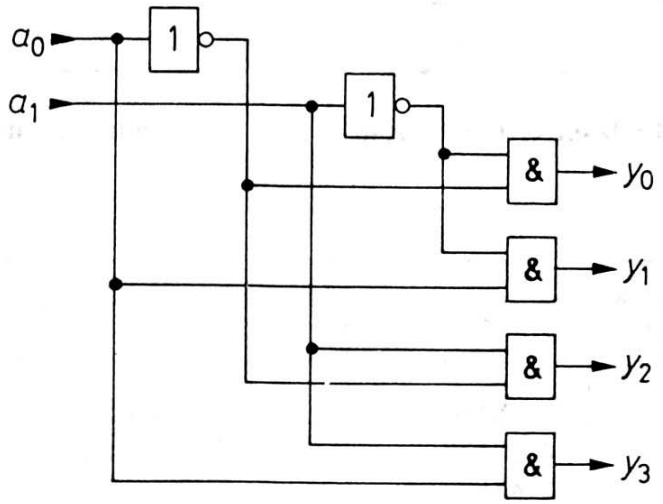
9.6.1. *n*-ből 1 dekódoló

Az *n*-ből 1 dekódoló olyan áramkör, amelynek *n* kimenete és ld *n* bemenete van. Az y_J kimenetet 0-tól $(n-1)$-ig számozzuk. Egy meghatározott kimenet akkor lesz logikai 1 értékű, ha a bemenetre adott *A* bináris szám értéke egyenlő az illető kimenet *J* sorszámával, egyébként logikai 0 értékű. A 4-ből 1 kód igazságtáblázatát a 9.52. ábra szemlélteti. Az a_0 és a_1 bemeneti változók az *A* szám bináris kódját ábrázolják. Ebből közvetlenül kiolvashatjuk a dekódoló függvény diszjunktív normál alakját. Az áramköri megoldást a 9.53. ábra szemlélteti.

A	a_1	a_0	y_3	y_2	y_1	y_0
0	0	0	0	0	0	1
1	0	1	0	0	1	0
2	1	0	0	1	0	0
3	1	1	1	0	0	0

9.52. ábra. 4-ből 1 dekódoló igazságtáblázata

$y_0 = \bar{a}_0\bar{a}_1$, $y_1 = a_0\bar{a}_1$, $y_2 = \bar{a}_0 a_1$, $y_3 = a_0 a_1$



9.53. ábra. 4-ből 1 dekódoló kapcsolás

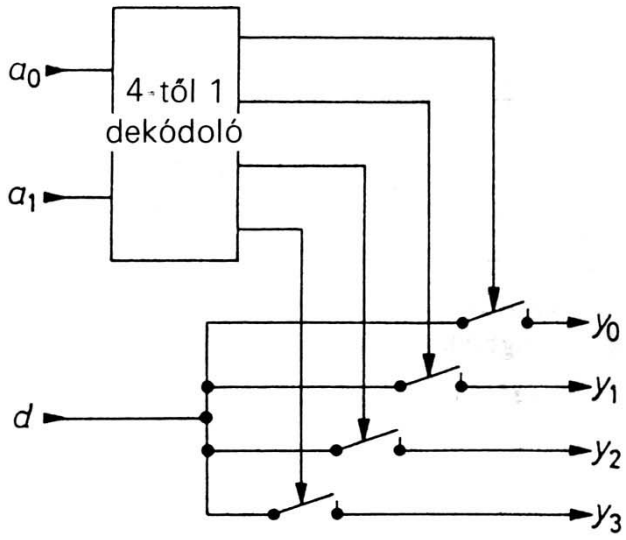
Monolit integrált áramkörös realizálásnál az ÉS kapuk helyett gyakran NEM – VAGY-kapukat használunk. A kimeneti változók emiatt negatív logikájúak.

IC típusok	TTL	CMOS
10 kimenetű	74 LS42	4028

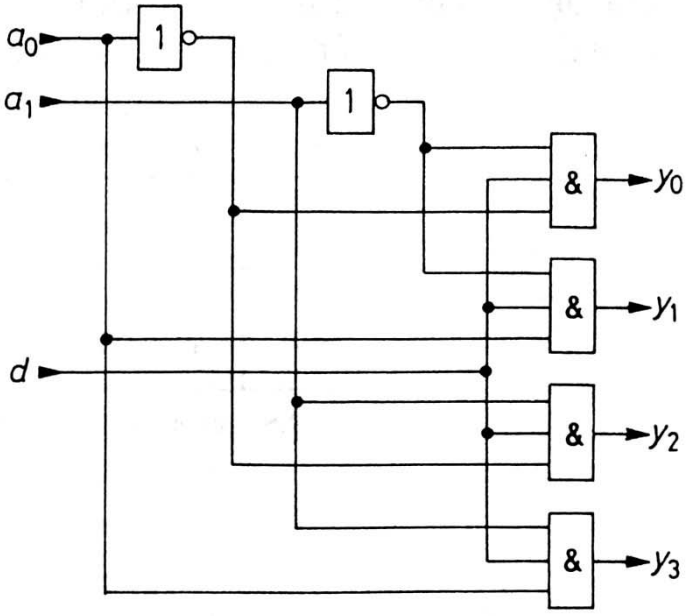
További integráltáramkör-típusokat a következő, demultiplexerekről szóló pontban találhatunk.

9.6.2. Demultiplexerek

Demultiplexerrel a d bemeneti információ különböző kimenetekre juttatható. Ez a kapcsolás az n-ből 1 kód kibővített változata. A megcímzett kimenet nem logikai 1-re változik, hanem a d bemeneti változó értékét veszi fel. Elvi működését a 9.54. ábra szemlélteti, a 9.55. ábra pedig a kapukkal megvalósított kapcsolástechnikai megoldást mutatja.



9.54. ábra. 4-ből 1 demultiplexer elvi működése



9.55. ábra. Demultiplexer kapcsolása

$y_0 = \bar{a}_0\bar{a}_1 d,\ y_1 = a_0\bar{a}_1 d,\ y_2 = \bar{a}_0 a_1 d,\ y_3 = a_0 a_1 d$

Kimenetek száma	TTL	ECL	CMOS
16	74 LS154		4514
8	74 LS138	10162	74 HC138
8	74 ALS538*		40 H138
2×4	74 LS139	10172	74 HC139
2×4	74 ALS539*		4555

* Kimeneti polaritás átkapcsolható.

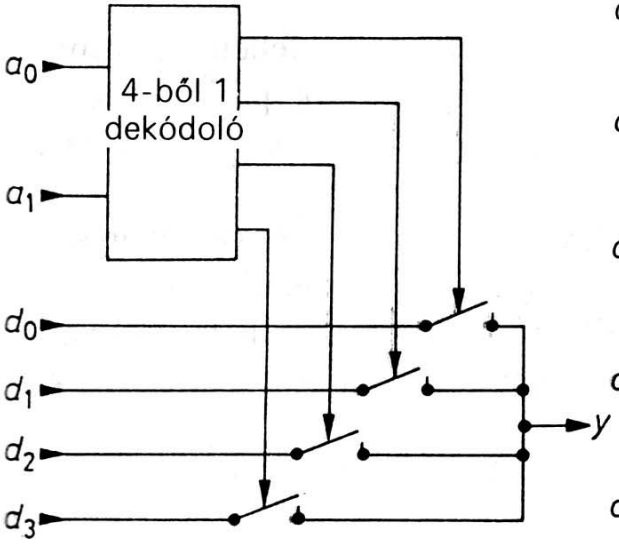
9.56. ábra. Integrált demultiplexer áramkörök

Amennyiben d = állandó = 1, akkor a demultiplexer n-ből 1 dekódolóként működik. A leggyakrabban használt demultiplexereket a 9.56. ábrán foglaltuk össze.

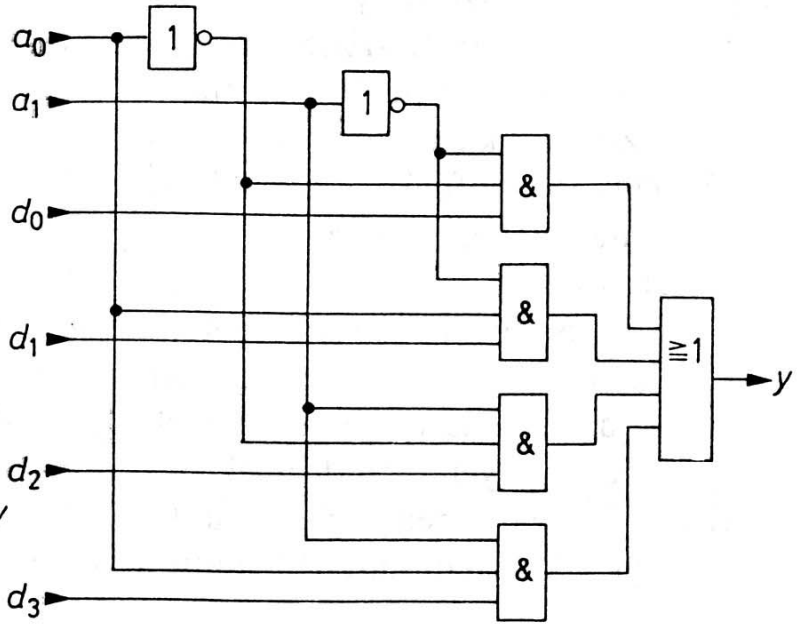
9.6.3. Multiplexerek

A demultiplexer fordítottját multiplexernek nevezzük. A 9.54. ábrán látható elvi kapcsolásból kiindulva úgy valósítható meg, hogy a kimeneteket és a bemeneteket felcseréljük. Így jutunk a 9.57. ábrán feltüntetett elvi kapcsoláshoz. Működése az ábra alapján nagyon egyszerűen megérthető. Az n-ből 1 dekódoló az n bemenet közül mindig azt választja ki, amelynek indexe megegyezik a dekódoló bemenetre adott számmal, és ezt a bemenetet összeköti a kimenettel. Az ennek megfelelő áramköri megoldást a 9.58. ábrán látjuk.

A CMOS-technikában a multiplexerek mind kapukkal, mind analóg kapcsolókkal (transmission gate) kivitelezhetők. Analóg kapcsolók alkalmazásakor a jelátvitel kétirányú lehet. Ezért ebben az esetben a multiplexer elvileg megegyezik a demultiplexerrel, ami a 9.54. és 9.57. ábra összehasonlításából is látható. A kapcsolást ebben az esetben analóg multiplexernek (demultiplexernek) is nevezzük.

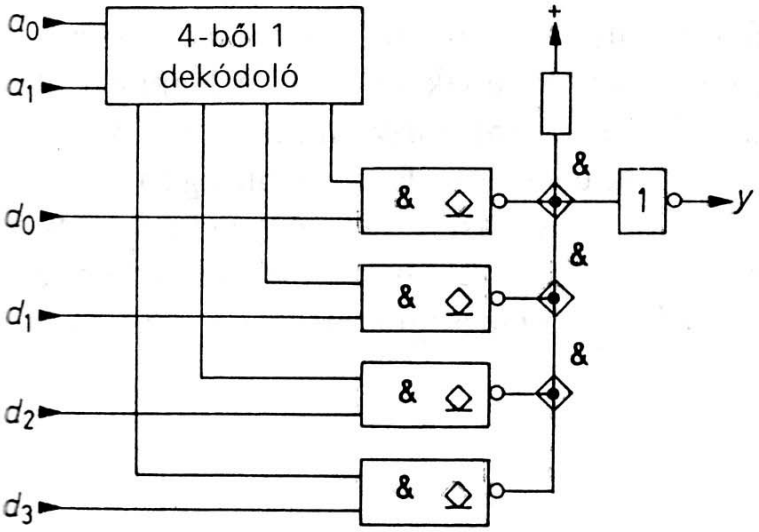


9.57. ábra. Multiplexer elvi működése

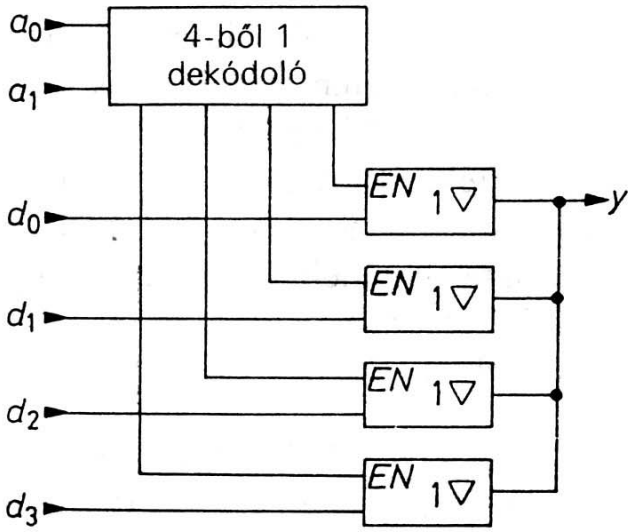


9.58. ábra. Multiplexer kapcsolása

$$y = \bar{a}_0\bar{a}_1 d + a_0\bar{a}_1 d_1 + \bar{a}_0 a_1 d_2 + a_0 a_1 d_3$$



9.59. ábra. Nyitott kollektoros kapus multiplexer



9.60. ábra. Háromállapotú kimenetű kapus multiplexer

Bemenetek száma	TTL	ECL	CMOS digitális	CMOS analóg
16	74 LS150		4515	4067
2×8	74 LS451*			4097
8	74 LS151	10164	4512	4051
4×4	74 LS453*			
2×4	74 LS153	10174	4539	4052
8×2	74 LS604			
4×2	74 LS157	10159	4519	4066

* MMI típus.

9.61. ábra. Integrált multiplexer áramkörök

A multiplexerek VAGY-függvényeit huzalozott VAGY-logikával is kivitelezhetjük. Ezt a megoldást nyitott kollektorú kimenetekre a 9.59. ábra mutatja. Mivel itt pozitív logikában ÉS-függvény adódik, ezért – hasonlóan a 9.30. ábrához – a jelet invertálva kell továbbvezetni.

Ha szeretnénk elkerülni a nyitott kollektorú kimenetek miatt fellépő hátrányt, ami a nagyobb felfutási időben jelentkezik, akkor kapcsoljunk párhuzamosan tristate-kimeneteket, amelyekből mindig csak egy aktív. Ezt a megoldást a 9.60. ábrán láthatjuk.

A 9.59. és 9.60. ábrán feltüntetett lehetőségeket VAGY-függvények megvalósítására az integrált multiplexerekben nem használjuk. Jelentőségük mégis nagy akkor, ha a multiplexer bemenetére dolgozó jelforrások geometriailag távol vannak egymástól. Ilyen elrendezések buszrendszerekkel kapcsolatban alakulnak ki, ezekkel a 21. fejezetben foglalkozunk majd.

A leggyakrabban használt multiplexereket a 9.61. ábrán foglaltuk össze.

9.6.4. Prioritásdekódoló

Az *n*-ből 1 kód bináris kóddá való átalakításához prioritásdekódolót használunk. Kimenetén az a bináris szám jelenik meg, amelyik megfelel a legnagyobb logikai 1 értékű bemenet sorszámának. Az ennél kisebb sorszámú bemeneti változók értéke érdektelen. A dekódoló nevét is innen kapta. E tulajdonsága miatt a kapcsolás nemcsak *n*-ből 1 kód átalakítására alkalmas, hanem összegkód átalakítására is, amelynél nemcsak egyetlen helyen van 1-es, hanem az összes ennél kisebb sorszámú helyen is. A prioritás dekóderoló igazságtáblázatát a 9.62. ábra szemlélteti.

Integráltáramkör-típusok:

10-ből 1 kód: SN 74147 (TTL);

8-ból 1 kód bővíthető kivitelben: SN 74148 (TTL); MC 10165 (ECL); MC14532 (CMOS).

J	x_9	x_8	x_7	x_6	x_5	x_4	x_3	x_2	x_1	y_3	y_2	y_1	y_0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	1	0	0	0	1
2	0	0	0	0	0	0	0	1	×	0	0	1	0
3	0	0	0	0	0	0	1	×	×	0	0	1	1
4	0	0	0	0	0	1	×	×	×	0	1	0	0
5	0	0	0	0	1	×	×	×	×	0	1	0	1
6	0	0	0	1	×	×	×	×	×	0	1	1	0
7	0	0	1	×	×	×	×	×	×	0	1	1	1
8	0	1	×	×	×	×	×	×	×	1	0	0	0
9	1	×	×	×	×	×	×	×	×	1	0	0	1

9.62. ábra. Prioritásdekódoló igazságtáblázata. × jelentése: tetszőleges

9.7. Függvény-jelölésrendszer

A digitális áramkörök szimbólumainak új jelölésrendszere nemcsak arra korlátozódik, hogy az eddigi kör alakú jelölések négyszögletesre változtak, hanem jelentős lépés történt azáltal, hogy az új szimbólumrendszer keretein belül új, ún. függvény-jelölésrendszert vezettek be, amivel bonyolult kapcsolásokat is áttekinthetően jellemezhetünk [9.3, 9.4, 9.5].

Az új jelölésrendszer alapgondolata, hogy a kapuk áramköri jelöléseinek pontosan meghatározott leírási szabályaival azt is megadjuk, hogy az egyes változók mint befolyásolnak más változókat. Vezérlő és vezérelt csatlakozókat különböztetünk meg. Arra is van lehetőség, hogy egy vezérelt csatlakozó más tekintetben vezérlő csatlakozóként működjön.

Jelölés	Jelentés
G	ÉS
V	VAGY
N	kizáró VAGY (vezérelhető negáció)
Z	változatlan átvitel
C	órajel
S	set
R	reset
EN	enable
M	üzemmód
L	terhelés (LOAD)
T	váltókapcsoló
A	cím
CT	tartalom (pl. számláló állapota)

9.63. ábra. Függvény-jelölésrendszer szimbólumai (jelölései)

Az új jelölésrendszerben a függőségeknek tizenegy különböző változatát különböztetjük meg. Ezeket a 9.63. ábra szerinti betűkkel jelöljük. Ezeket a betűket a követelményeknek megfelelően a szimbólumon belül a vezérlő csatlakozó mellé írjuk. A betű után az azonosító számot írjuk. Ezt a számot minden olyan csatlakozó mellé odaírjuk, amelynek logikai szerepe ugyanaz, mint a betűjelzéssel is ellátotté.

A 9.64. ábra példaként egy meghajtó kapu kibővítését mutatja ÉS-kapuvá az új jelzésrendszer segítségével. Ennek megfelelően a 9.65. és 9.66. ábra VAGY-, ill. kizáró VAGY-kapuvá bővítését mutatja.

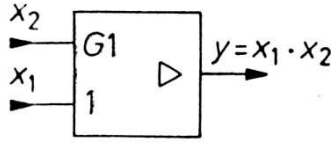
Egy csatlakozó egyszerre több más csatlakozót is vezérelhet. Ebben az esetben az azonosítási számokat, amint a 9.67. ábrán is látjuk, vesszővel választjuk el egymástól. A műveleteket egymás után balról jobbra haladva végzi el az áramkör.

A 9.68. ábra arra mutat példát, hogy egy vezérlő bemenet több más csatlakozóval is kapcsolatba kerül. Az azonosító számon lévő felülvonás, negációs jel azt jelenti, hogy az illető változó a vezérlő változó invertáltjától függ.

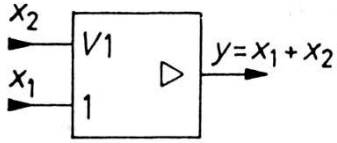
Amint a 9.69. ábrán is látható, több csatlakozó egyetlen vezérlő változóvá is egyesíthető. Azonosítási számként olyan bináris szám szerepel, amely a kapocsjelen belül látható értékű. A lehetséges tartomány a kapocsjel mögött található. A 0/3 azt jelenti, hogy a szám a 0...3 tartományban bármi lehet. Például az x_0 bemenet csak akkor működik, ha az a_0 és a_1 vezérlőbemeneteken bináris 0 van.

Az eddig bemutatott példákon láttuk, hogy a vezérlő bemeneteket csak az azonosítási számok jelölhetik. Vannak olyan esetek, amikor a csatlakozókat egyéb okokból a működésbeli szerepükre utaló betűjelzéssel látjuk el, pl. az adatoknál D. Ilyen esetben az azonosító számot vesszővel választjuk el a betűtől, pl.: 1, *D*.

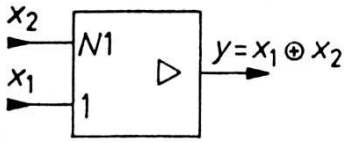
A 9.70. ábra a különböző működési módok (*M*; mode) használatára mutat példát, valamint a tartalom (*con*tent, CT) módosítását és egyben vezérlő hatását



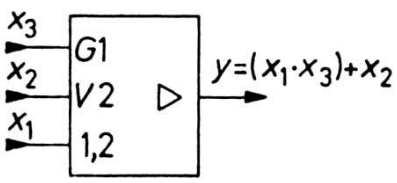
9.64. ábra. ÉS-függvény



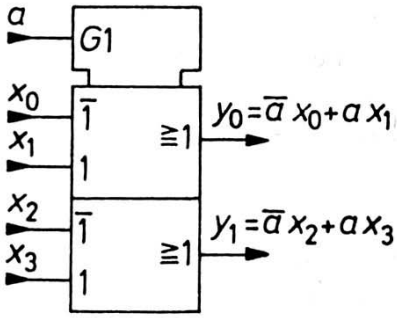
9.65. ábra. VAGY-függvény



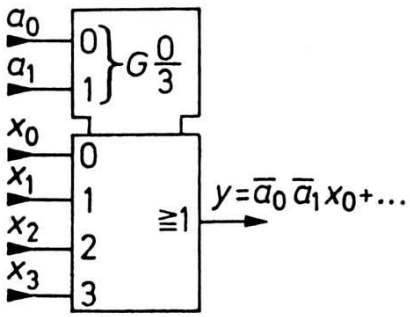
9.66. ábra. Kizáró VAGY-függvény



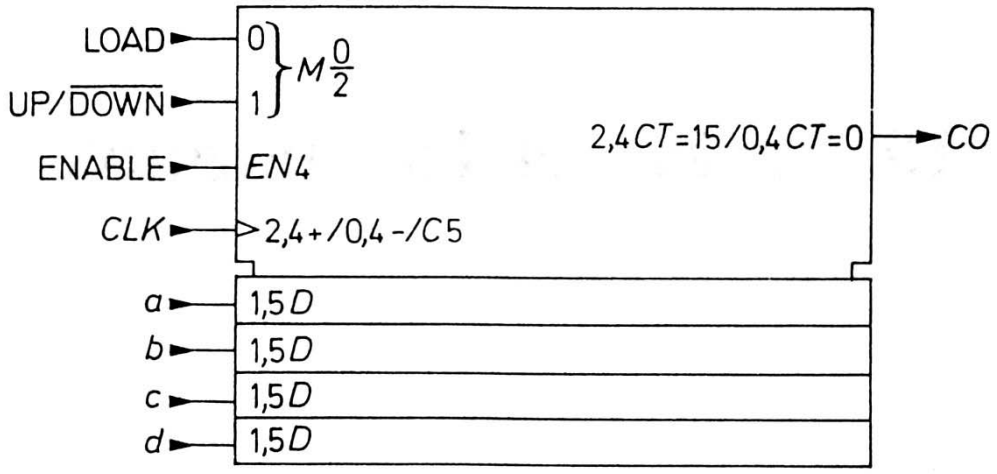
9.67. ábra. Bemenet többszörös vezérlése



9.68. ábra. Több bemenet együttes vezérlése a duál kettőből egy multiplexer példáján szemléltetve



9.69. ábra. Több vezérlő változójú vezérlő blokk, a négyből egy multiplexer példáján szemléltetve



9.70. ábra. Több működésmód leírása, a példa egy szinkron beíróbemenetű előre-hátraszámláló

szemlélteti. Egy párhuzamos beíróbemenetű előre-, hátraszámláló látható. Működési módtól függően a CLK órajel más-más hatású.

Az órajelbemenet 2,4+ jelzése azt jelenti, hogy a számláló állása növekszik, ha a működési mód 2 (LOAD = 0, UP = 1) és ENABLE = 1. Ennek megfelelően 0 működési módban visszafelé számol. Ez utóbbi működési feltétel: 0,4−. Az egyes csatlakozók működési módjait / (per) jellel választjuk el és egymás mellé írjuk.

A harmadik működési mód a párhuzamos adatbeírás. Az 1,5 D jelölés azt jelenti, hogy 1 működési módban az órajellel szinkron párhuzamos beírás folyik. Ezzel szemben az 1 D jel aszinkron működést jelentene.

A *CO* átvitel kimenetet a számláló tartalma vezérli. A kimeneten 1 lesz, ha előreszámlálásnál a tartalom 15(2,4 CT = 15), vagy akkor, ha hátraszámlálásnál a tartalom 0(0,4 CT = 0).