2. hét Kombinációs hálózatok leírási módjai

2.1. A kombinációs hálózat alapfogalmai

Logikai hálózatnak nevezzük azokat a rendszereket, melyeknek bemeneti illetve kimeneti jelei logikai jelek, a kimeneti jeleket a bemeneti jelek függvényében többé-kevésbé bonyolult logikai műveletsorozat eredményeként állítják elő.

Ezeknek a logikai hálózatoknak két típusa van:

1. **Kombinációs hálózatok**
   Kombinációs hálózatoknak nevezzük azokat a logikai hálózatokat, melyeknek kimeneti jelei csak a bemeneti jelek pillanatnyi értékétől függnek. Ezek a hálózatok éppen ezért „Emlékezet" nélküli hálózatok.
2. **Sorrendi hálózatok**
   Sorrendi (szekvenciális) hálózatoknak nevezzük azokat a logikai hálózatokat, melyek kimeneti jelei nemcsak a pillanatnyi bemeneti jelkombinációtól függnek, hanem attól is, hogy korábban milyen bemeneti jelkombinációk voltak. Ezek a hálózatok tehát emlékezettel rendelkeznek.

Ennek az előadásnak a keretein belül tehát a csak a kombinációs, tehát emlékezet nélküli hálózattal fogunk foglalkozni. Ezt a hálózatfajtát többféle módon le lehet írni, hiszen a logikai algebra kapuit /műveleteit is többféle módon tudjuk jelölni.

1. ábra A kombinációs hálózat sémája

A kombinációs hálózatnak több fontos tulajdonsága is van.

- A bemenetek pillanatnyi állapota (a tranziensektől eltekintve egyértelműen meghatározza a kimenetek állapotát, függetlenül attól, hogy korábban milyen bementi állapottal vezéreltük a hálózatot.
- A kombinációs hálózatokban minden bemeneti kombináció egyértelműen és kizárólagosan meghatározza a kimeneti kombinációt.
- A kimeneti kombinációból viszont általában nem tudjuk egyértelműen meghatározni az azt előidéző bemeneti kombinációt, mert nem követelmény, hogy különböző bemeneti kombinációk minden esetben más-más kimeneti kombinációt hozzanak létre.

2.2. A kombinációs hálózat leírási módjai

A kombinációs hálózatot nagyon sokféle módon lehet leírni. Ezek közül a legfontosabb leírási módok két nagy csoportba sorolhatók:

- grafikus
- és nem grafikus megjelenítés.

Didaktikai okokból azonban a leírási formák tárgyalásakor az alábbi sorrendet fogjuk követni:

- szöveges megfogalmazás,
- blokk (grafikus megjelenítés)
- igazságtáblázat (grafikus megjelenítés),
- logikai függvények (algebrai megfogalmazás a Boole logika alapján)
- logikai kapcsolási rajz (grafikus megjelenítés),
- Karnaugh tábla (grafikus megjelenítés)

A leírás módjait egy példán keresztül fogjuk megérteni. Legyen ez a példa egy

szavazat számláló.

2.2.1. Szöveges megfogalmazás

A szöveges megfogalmazás során egyszerűen megfogalmazzuk az adott rendszer logikai szabályait.

Például van egy bírókból álló bizottságunk, akiknek feladata egy közös döntés kialakítása (pl. egy sporteseményen). A bizottságról a következőket tudjuk:

- A bizottság 3 tagból áll,
- Többségi szavazással döntenek.
- A szavazás eredménye IGEN, ha legalább 2 tag IGEN - nel szavaz.

2.2.2. Blokk

A blokkvázlat esetében a rendszerről egy rajzot készítünk, ahol a be és kimenő jeleket tüntetjük fel.

Esetünkben a rendszer bemenője a 3 bíró döntése lesz, míg a kimenő jel az ő döntéseik alapján képződő válasz. Ez a válasz természetesen nem önkényes: a fent meghatározott szabály adja a bírók döntésének függvényében.

A , B, C : bírók személyes döntései (bemenet)

Y a bizottság közös döntése (kimenet)

2. ábra A szavazatszámláló blokkvázlata

2.2.3. Igazságtáblázat

Az igazságtáblázat megszerkesztésénél a következő szabályt vesszük figyelembe:

1. Meghatározzuk a független változókat (esetünkben a rendszer bemenetét: a bírók számát: 3, jelölésük – A, B, C)
   a. *Fontos hogy a változók sorrendje és a kiolvasás sorrendje FELCSERÉLHETŐ De egy feladaton belül nem! Tehát a változó elnevezéseket és sorrendeket konzekvensen kell használni.*
   b. *Fontos figyelni a helyiértékre (MINDIG a JOBB oldal a legkisebb érték)*
2. meghatározzuk a függő változókat (esetünkben a rendszer kimenetét – a bírók közös döntését, jele: Y).
3. Az igazságtáblázatnak annyi oszlopa lesz, ahány változót kaptunk. (Ebben az esetben 4)

C	B	A	Y

3. ábra Az igazságtáblázat készítése 1: független változók

4. A sorok számát az alábbi összefüggéssel tudjuk meghatározni. (n= független változók száma) A számlálást 0-val kezdjük.

$$\mathbf{V = 2^n = 2^3 = 8}$$

i	C	B	A	Y
0				
1				
2				
3				
4				
5				
6				
7				

4. ábra Az igazságtáblázat készítése 2: variációk száma (sorok)

5. A táblázatot kitöltjük minden lehetséges kombinációt figyelembe véve

Minden lehetséges bírói döntés variációi a következők:

a. Minden bíró nemet mond
b. Minden bíró igent mond
c. Egy bíró mond igent
d. Két bíró mond igent

Ha a bíró nemmel szavaz, a változó értéke 0, ha igennel, akkor 1.

i	C	B	A	Y
0	0	0	0	
1	0	0	1	
2	0	1	0	
3	0	1	1	
4	1	0	0	
5	1	0	1	
6	1	1	0	
7	1	1	1	

5. ábra Az igazságtáblázat készítése 3: bemenet meghatározás

6. Meghatározzuk a rendszer kimenetét:

Tudjuk, hogy akkor kapunk igen döntést, ha legalább két bíró igennel szavazott, ezért adott sorban legalább két 1–nek kell lennie ahhoz, hogy a kimeneten is 1 (igaz) értéket kapjunk.

i	C	B	A	Y
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

6. ábra Az igazságtáblázat kimenetének meghatározása

2.2.4. Logikai függvények

A logikai függvények olyan matematikai leképezések, melyek a 0 és 1 számokból álló véges sorozatokhoz rendelik a 0 vagy 1 számot.

Egy logikai függvény tehát olyan n változós függvény, melynek változói a {0,1} halmazból vehetnek fel értéket, a függvényérték pedig szintén a {0,1} halmazból valók. Itt az 1 értékre gyakran mint az igaz, a 0 értékre mint a hamis hivatkoznak (főleg logikai alkalmazásaiban). Formálisan, a $\{0,1\}^n$ Descartes-szorzat segítségével egy f függvény logikai, ha:

$$f: \{0,1\}^n \rightarrow \{0,1\}$$

Az n változós logikai függvények száma 2^{2n}, hiszen az n változó 2^n darab lehetséges értékének mindegyikéhez két értéket rendelhetünk.

2.2.4.1. A logikai függvények meghatározásának menete

A logikai függvényeket a fenti igazságtáblázatból tudjuk képezni. Ehhez a következő lépésekre van szükségünk, miután a szöveges megfogalmazás alapján elkészítjük az értéktáblázatot.

i	C	B	A	Y
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

7. ábra Az igazságtáblázat

2.2.4.1.1. A diszjunkt - alakú függvény felírása.

Az igazságtáblázat tartalmát a következőképpen olvassuk ki.

A Y jelű függő változó értéke 1 (IGAZ), akkor ha

- ha C=0 és B=1 és A=1 (3.sor),vagy
- ha C=1 és B =0 és A=1 (5.sor),vagy
- ha C=1 és B=1 és A=0 (6.sor),vagy
- ha C=l és B=l és A=1 (7.sor).

i	C	B	A	Y
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	**0**	**1**	**1**	**1**
4	1	0	0	0
5	**1**	**0**	**1**	**1**
6	**1**	**1**	**0**	**1**
7	**1**	**1**	**1**	**1**

8. ábra disztjunkt alak leírása

A felírás szabálya a következő:

1) **azokat a sorokat kell figyelembe venni, amelyeknél a függő változó értéke 1;**
2) **az egy sorban levő független változók között ÉS műveletet kell végezni, ahol független változó igaz (egyenes, más kifejezéssel ponált) alakban írandó, ha értéke 1 és tagadott (negált) alakban, ha értéke 0;**
3) **az egyes sorokat leíró ÉS műveletű rész-függvények VAGY művelettel kapcsolódnak egymáshoz.**

Tehát a függvény:

- Soronként

$$Y1 = AB\overline{C}$$
$$Y2 = A\overline{B}C$$
$$Y3 = \overline{A}BC$$
$$Y4 = ABC$$

- A teljes függvény

$$Y = AB\overline{C} + A\overline{B}C + \overline{A}BC + ABC$$

Nézzük most meg az igazságtáblázatból felírt logikai függvény általános jellemzőit.

A függvény rendezett ÉS-VAGY alakú. Az ÉS művelettel összekapcsolt részekben mindegyik változó szerepel egyenes vagy tagadott alakban, vagyis a Veitch diagramnál definiált minterm.

Az egyes mintermek között pedig VAGY műveleteket kell végezni. Az ilyen függvényalakot idegen szóval diszjunktiv kanonikus alaknak (teljes diszjunktiv normál formának) nevezzük.

Ebből a formátumból viszont tudunk egyszerűsíteni.

$$Y = AB\overline{C} + A\overline{B}C + \overline{A}BC + ABC$$

Vigyük az utolsó tagot zárójel mögé

$$Y = \left(AB\overline{C} + ABC\right) + \left(A\overline{B}C + ABC\right) + \left(\overline{A}BC + ABC\right)$$

Mivel tudjuk, hogy

$$\overline{A} + A = \overline{B} + B = \overline{C} + C = 1$$

és

$$A + A + \cdots + A = A$$

tehát

$$ABC + ABC = ABC$$

ezért az egyszerűsített alak:

$$Y = A \cdot B + A \cdot C + B \cdot C$$

VHDL leírás

```
Y <= (A and B) or (A and C) or (B and C)
```

2.2.4.1.2. A konjunkt alakú függvény felírása.

A Y jelű függő változó értéke 0 (HAMIS), akkor ha

- ha C=0 és B=0 és A=0 (0.sor) vagy
- ha C=0 és B=0 és A=1 (1.sor) vagy
- ha C=0 és B=1 és A=0 (2.sor) vagy
- ha C=1 és B=0 és A=0 (4.sor).

i	C	B	A	Y
0	**0**	**0**	**0**	**0**
1	**0**	**0**	**1**	**0**
2	**0**	**1**	**0**	**0**
3	0	1	1	1
4	**1**	**0**	**0**	**0**
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

9. ábra A konjuktív alak felírása

A felírás szabálya ezért a következő:

1) **azokat a sorokat vesszük figyelembe, melyekben a függő változó értéke 0;**
2) **az egy sorban levő független változók között VAGY kapcsolatot írunk elő;**
3) **a független változót egyenes alakban írjuk, ha értéke 0 és tagadott alakban, ham értéke 1;**
4) **az egyes sorokat leíró VAGY függvényeket ÉS művelettel kell összekapcsolni.**

Tehát a függvény, ha az előbbi módszert követjük:

❖ Soronként

$$Y1 = A + B + C$$
$$Y2 = \overline{A} + B + C$$
$$Y3 = A + \overline{B} + C$$
$$Y4 = A + B + \overline{C}$$

❖ A teljes függvény

$$Y = (A + B + C)\left(\overline{A} + B + C\right)\left(A + \overline{B} + C\right)\left(A + B + \overline{C}\right)$$

Ezt szintén tudjuk egyszerűsíteni:

$$Y = (A + B + C)\left(\overline{A} + B + C\right)\left(A + \overline{B} + C\right)\left(A + B + \overline{C}\right)$$

Mivel tudjuk, hogy

$$A \cdot A \cdot \ldots \cdot A = A$$

tehát

$$(A + B + C) \cdot (A + B + C) = A + B + C$$

Ezért az egyszerűbb alak megadható, ha az első taggal felbővítjük:

$$Y = \left(\left(\overline{A} + B + C\right)(A + B + C)\right)\left(\left(A + \overline{B} + C\right)(A + B + C)\right)\left(\left(A + B + \overline{C}\right)(A + B + C)\right)$$

Mert korábban is használtuk az alábbi összefüggést:

$$\overline{A} + A = \overline{B} + B = \overline{C} + C = 1$$

A z egyenlet rövidített alakja a következő:

$$Y = (B + C)(A + C)(A + B)$$

VHDL leírás

```
Y <= (B or C) and (A or C) and (A or B)
```

2.2.4.2. A logikai függvények egyszerűsítése

A logikai függvényeket a használat során fontos lehet egyszerűsíteni. Ezzel ugyanis elérhető, hogy a tervezett áramköri hálózat is egyszerűbb, olcsóbb, hatékonyabb legyen.
Az igazságtáblázat alapján felírt kanonikus alakú függvények a legtöbb esetben egyszerűsíthetőek. Az egyszerűsítés azt jelenti, hogy a logikai algebra megismert tételeinek felhasználásával olyan alakot nyerhetünk, amelyben kevesebb művelet, és vagy kevesebb változó szerepel. Az egyszerűsítésre azért van szükség, mert ez után a feladatot megvalósító logikai hálózat kevesebb áramkört, vagy programozott rendszer (mikrogép) programja kevesebb utasítást tartalmaz Az algebrai módszer mellett kidolgoztak grafikus, illetve matematikai egyszerűsítési eljárásokat is.
Tekintsük át az előző példa lehetséges egyszerűsítését.

2.2.4.2.1. Diszjunktív alak egyszerűsítése

Az előző példában felírt diszjunktív alak a következő volt:

$$Y = AB\overline{C} + A\overline{B}C + \overline{A}BC + ABC$$

Ezt az alakot **disztjunktív teljes normál alaknak** nevezzük, mert minden tagban szerepel minden változó, ill. minden lehetséges variációt sort) tartalmaz.

A diszjunktív teljes normál alakban szereplő szorzatok: **mintermek.**

$$Y = \underbrace{\boxed{AB\overline{C}}}_{m^3_3} + \underbrace{\boxed{A\overline{B}C}}_{m^3_5} + \underbrace{\boxed{\overline{A}BC}}_{m^3_6} + \underbrace{\boxed{ABC}}_{m^3_7}$$

Jelölésük

m^n_i

ahol

n: változók száma

i: a függvény melyik mintermje

A mintermek megfelelő összeadásával bármelyik függvény előállítható. Ezt az alakot lehet rövidíteni úgy is, szumma jel mögé csoportosítjuk az indexeket. (értékeket).

$$Y^3 = \sum(3,5,6,7)$$

Ebből az alakból azonban tudunk egyszerűsíteni:

Vigyük az utolsó tagot zárójel mögé

$$Y = \left(AB\overline{C} + ABC\right) + \left(A\overline{B}C + ABC\right) + \left(\overline{A}BC + ABC\right)$$

Mivel tudjuk, hogy

$$\overline{A} + A = \overline{B} + B = \overline{C} + C = 1$$

és

$$A + A + \cdots + A = A$$

tehát

$$ABC + ABC = ABC$$

ezért az egyszerűsített alak:

$$Y = A \cdot B + A \cdot C + B \cdot C$$

Ezt az alakot **disztjunktív normál, de nem teljes alaknak** nevezzük, mert nem szerepel benne minden tagban minden változó, és nem szerepel minden lehetséges variáció.

A VHDL leírás pedig

Y <= (A and B) or (A and C) or (B and C)

2.2.4.2.2. Konjunktív alak egyszerűsítése

Az előző példában felírt konjunktív alak a következő volt:

$$Y = (A + B + C)(\overline{A} + B + C)(A + \overline{B} + C)(A + B + \overline{C})$$

Ezt az alakot **konjunktív teljes normál alaknak** nevezzük, mert minden tagban szerepel minden változó, ill. minden lehetséges variációt, sort) tartalmaz.

A konjunktív teljes normál alakban szereplő szorzatok: **maxtermek.**

Jelölésük

$$M^n{}_i$$

ahol

n: változók száma

i: a függvény melyik maxtermje

$$Y = \underbrace{\boxed{(A + B + C)}}_{M^3{}_0}\underbrace{\boxed{(\overline{A} + B + C)}}_{M^3{}_1}\underbrace{\boxed{(A + \overline{B} + C)}}_{M^3{}_2}\underbrace{\boxed{(A + B + \overline{C})}}_{M^3{}_4}$$

A maxtermek megfelelő összeszorzásával bármelyik függvény előállítható. Ezt az alakot lehet rövidíteni úgy is, hogy $\prod$ jel mögé csoportosítjuk az indexeket (értékeket).

$$Y^3 = \prod(0,1,2,4)$$

Ezt a teljes alakot szintén tudjuk egyszerűsíteni:

$$Y = (A + B + C)(\overline{A} + B + C)(A + \overline{B} + C)(A + B + \overline{C})$$

Mivel tudjuk, hogy

$$A \cdot A \cdot \ldots \cdot A = A$$

tehát

$$(A + B + C) \cdot (A + B + C) = A + B + C$$

Ezért az egyszerűbb alak megadható, ha az első taggal felbővítjük:

$$Y = \left((\overline{A} + B + C)(A + B + C)\right)\left((A + \overline{B} + C)(A + B + C)\right)\left((A + B + \overline{C})(A + B + C)\right)$$

Mert korábban is használtuk az alábbi összefüggést:

$$\overline{A} + A = \overline{B} + B = \overline{C} + C = 1$$

Az egyenlet rövidített alakja a következő:

$$Y = (B + C)(A + C)(A + B)$$

A VHDL leírás pedig

Y <= (B or C) and (A or C) and (A or B)

2.2.5. Logikai kapcsolási rajz

A logikai kapcsolási rajzzal lehet a legszemléletesebben ábrázolni a feladatot. Ekkor kiválaszthatjuk, hogy melyik felírást választjuk a két lehetőségből. Az alábbi ábrán válasszuk a diszjunkt alakot.

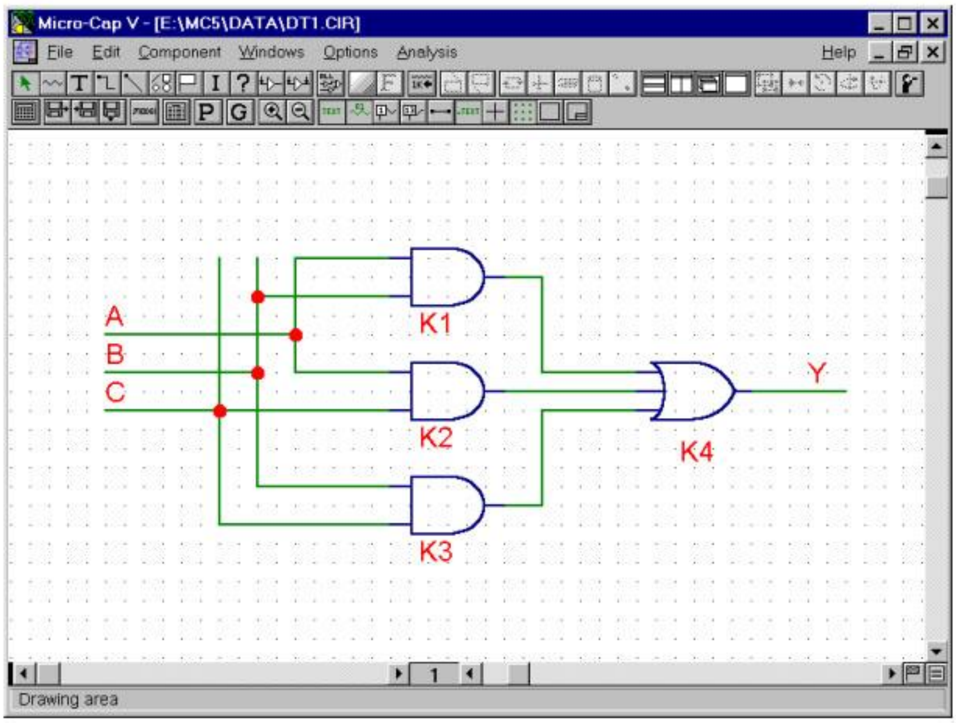
Ekkor a nem teljes diszjunkt alakot figyelembe véve, ahol az egyenlet

$$Y = A \cdot B + A \cdot C + B \cdot C$$

VHDL leírás pedig

Y <= (A and B) or (A and C) or (B and C)

ekkor a bemeneti jeleket A, B, C) először három ÉS kapuval kell összekötni, és az egyenlet alapján az ÉS kapuk eredményét kell egy 3 bemenetű VAGY kapura vezetni.



10. ábra A diszunktív alak kapcsolási rajza

2.2.6. Karnaugh[1] tábla

2.2.6.1. A Karnaugh tábla definíciója

A logikai függvények grafikus ábrázolásainak közös jellemzője, hogy az elemi - egy műveletes - logikai függvényekhez (minterm, maxterm) sík-, vagy térbeli geometriai alakzatot rendelünk. A lehetséges egyszerűsítések közül csak a Karnaugh (ejtsd: karnó) – táblázatos eljárással foglalkozunk.

A Karnaugh táblázat formailag a Veitch diagramból származtatott. A különbségek a változók megadásának (a peremezésnek) módjában, valamint abban van, hogy egy elemi négyszög mintermet, vagy maxtermet is jelképezhet. Tulajdonképpen az igazságtáblázat célszerűen átalakított változatáról beszélhetünk, hiszen a rendszer igazságtáblázatának pontosan annyi sora van, ahány cellája a (mintermje) a Karnaugh – táblának. Azaz n db független változó esetén 2^n. Fontos dolog, hogy a Gray és a Johnson kódoláshoz hasonlóan a kódolást úgy kell elvégezni, hogy az egymás melletti oszlopok, ill. sorok mindig csak egy változóban térjenek el egymástól.

Egy n változós függvény tehát 2^n db elemi négyzetből álló táblázatban szemléltethető.

A módszer előnye:

- gyorsabb,
- biztosabban jó eredményt adó,
- kevesebb munkát igénylő módszer

Hátránya pedig, hogy:

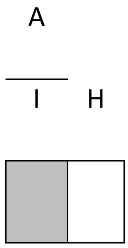
- legfeljebb 4 (esetleg 5) változóig használható.

[1] Maurice Karnaugh (1924 -) amerikai fizikus. Nevéhez fűződik a Boole algebra alapján létrehozott tábla, mely segíti a villamos hálózatok , áramkörök felírását, egyszerűsítését, melyet a Bell Laboratóriumban dolgozott ki. Magasan képzett fizikus, a Yale egyetemen végzett, és az IEEE tagja.

2.2.6.2. A Karnaugh – tábla fajtái

2.2.6.2.1. Egy változós Karnaugh – tábla

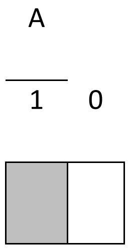
A Karnaugh – tábla tulajdonképpen egy módosított Veitch - diagram.



11. ábra Egy változós Veitch – diagram

A Karnaugh – tábla esetén a halmazt egy négyszögben ábrázoljuk. A változókat a négyszög szélére írt jelöléssel tüntetjük fel. Az értékeket a négyszög szélére irt kódolással adjuk meg.

Minden változó 1 értékéhez a teljes terület egyik felét, míg a 0 értékéhez pedig a másik felét rendeljük. A változó IGAZ (1) értékéhez vonalat húzunk, mert ezzel jelöljük meg ezt a térfelet.

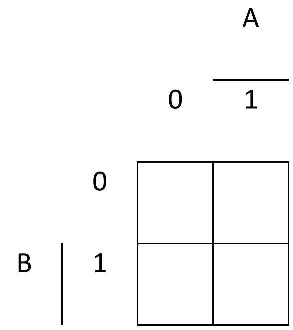


12. ábra Egy változós Karnaugh – tábla

Több változó esetén is így fogunk eljárni.

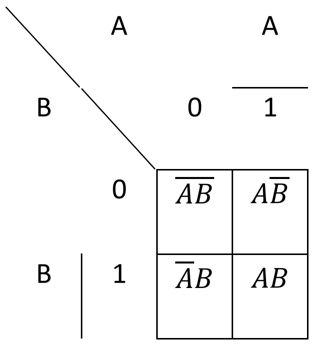
2.2.6.2.2. Két változós Karnaugh – tábla

Két változó esetén már figyelni kell arra, hogy a felezést úgy forgatjuk, hogy a változókhoz rendelt területeket jól meg lehessen különböztetni. Ezért a változók kódolását úgy kell végezni, hogy az egymás melletti oszlopok, ill. sorok mindig csak egy változóban térjenek el egymástól.



13. ábra Két változós Karnaugh – tábla

A táblában a változók értékei a korábbiak miatt értelem szerűen a következőképpen alakulnak.



14. ábra A két változós Karnaugh - tábla tartalma

A Karnaugh tábla kapcsán azt is fontos észrevenni, hogy a táblázat milyen kapcsolatban van az igazságtáblával és a kettes számrendszerrel / bináris számkódolással.

Két változó esetén az igazságtáblának 4 sora lesz, és a következő módon fog kinézni:

i	B	A	Y
0	0	0	
1	0	1	
2	1	0	
3	1	1	

15. ábra A két változós igazságtábla

Vegyük észre, hogy az egyes sorokban A és B változó értékei természetesen beírhatók a Karnaugh – táblába.

		A	A
B		0	1
	0	00	10
B	1	01	11

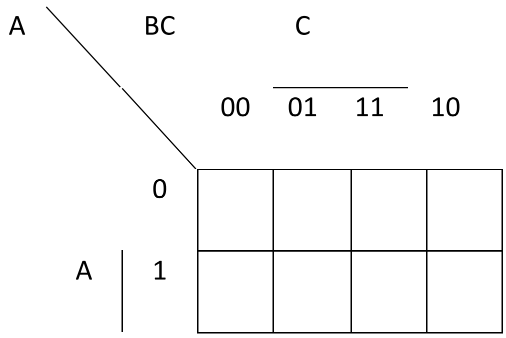
Az igazság tábla értékei azonban az egyes sorokban kettes számrendszerben adja az egyes sorok „sorszámának” értékét. Ezért mindenegyes cellához egy saját érték rendelhető.

		A	A
B		0	1
	0	0	2
B	1	1	3

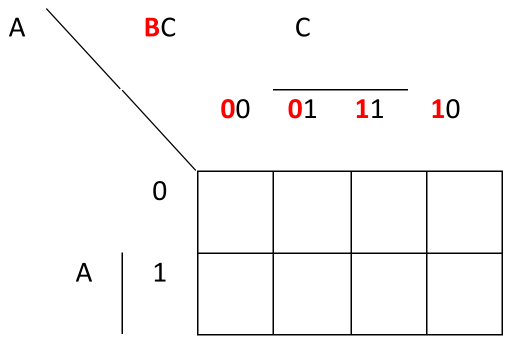
2.2.6.2.3. Három változós Karnaugh – tábla

A három változó esetén az előbbi meggondolások érvényben maradnak, ezért ezeket ugyan úgy figyelembe kell venni: meg kell különböztetni a változókhoz rendelt területeket, és az egymás melletti sorok és oszlopok csak egy változóban térhetnek el egymástól.

Viszont a három változó miatt táblázat oszlopai a BC változó-pár lehetséges értékkombinációihoz vannak rendelve. Az oszlop-peremezést úgy kell végezni, hogy a szomszédos oszlopok csak egyetlen változó-értékben különbözzenek. A harmadik változó A értéke szerint két sora van a táblázatnak. Az egyikben A=0,a másikban pedig A=1. Az egyes elemi négyszögekhez tehát a változók különböző értékvariáció tartoznak. A peremezés megváltoztatható, de csak úgy, hogy a szomszédos sorok, oszlopok egy változóban különbözhetnek. (A táblázat szélső oszlopai, illetve sorai mindig szomszédosak).



16. ábra Három változós Karnaugh – tábla



17. ábra Három változós (B elemei kiemelve) Karnaugh – tábla

Az eddigiekben csak az ábrázolás formai részével foglalkoztunk. Nézzük most meg a logikai tartalmat is egy kicsit bővebben. A két hozzárendelés szerint beszélünk Kp ill. Ks diagramról. A Kp jelölés az ÉS-VAGY (vagyis a mintermek logikai összege – tehát a diszjunkt alakot leíró), míg a Ks a VAGY-ÉS (maxtermek logikai szorzata – konjunktív alakot leíró) műveletes összerendelést jelenti. A leírt kikötések betartásával - az előző fejezetben megismert – mindkét logikai függvényalak (diszjunktív, ill. konjunktív) ábrázolható, és egyszerűsíthető Karnaugh – diagram segítségével

Az előbbi tartalommegadásnál az ÉS – VAGY irányt vettük figyelembe, ennek alapján készült a táblázat kitöltése, tehát a diszjunktív alakot vettük figyelembe. Három változóra a diszjunktív alakot felírva az alábbi eredményt kapjuk.

A	BC		C		
		00	01	11	10
	0	$\overline{ABC}$	$\overline{AB}C$	$\overline{A}BC$	$\overline{A}B\overline{C}$
A	1	$A\overline{BC}$	$A\overline{B}C$	ABC	$AB\overline{C}$

18. ábra Három változós Karnaugh – tábla tartalma

Természetesen három változóra is felírhatjuk az igazságtáblát, és az így kapott értékeket itt is behelyettesíthetjük a Karnaugh – tábla cellájába.

i	A	B	C	Y
0	0	0	0	
1	0	0	1	
2	0	1	0	
3	0	1	1	
4	1	0	0	
5	1	0	1	
6	1	1	0	
7	1	1	1	

19. ábra A három változós igazságtábla

A \ BC			C	C	
		00	01	11	10
	0	000	001	011	010
A	1	100	101	111	110

20. ábra A három változós Karnaugh - tábla bináris értékekkel

A \ BC			C	C	
		00	01	11	10
	0	0	1	3	2
A	1	4	5	7	6

21. ábra A három változós Karnaugh - tábla cellaértékei

2.2.6.2.4. Négy változós Karnaugh – tábla

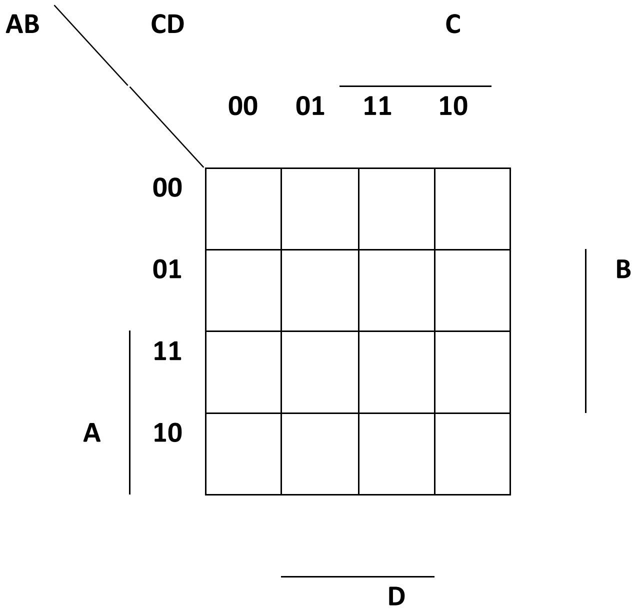
Természetesen a táblázat megrajzolható négy változó esetére is. Ilyenkor azonban fel kell bővíteni a cellákat, hiszen az igazságtáblában 4 változó szerepel., ezért 16 sorunk lesz.

Tehát 4 változó esetén az igazságtábla:

i	A	B	C	D	Y
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	
3	0	0	1	1	
4	0	1	0	0	
5	0	1	0	1	
6	0	1	1	0	
7	0	1	1	1	
8	1	0	0	0	
9	1	0	0	1	
10	1	0	1	0	
11	1	0	1	1	
12	1	1	0	0	
13	1	1	0	1	
14	1	1	1	0	
15	1	1	1	1	

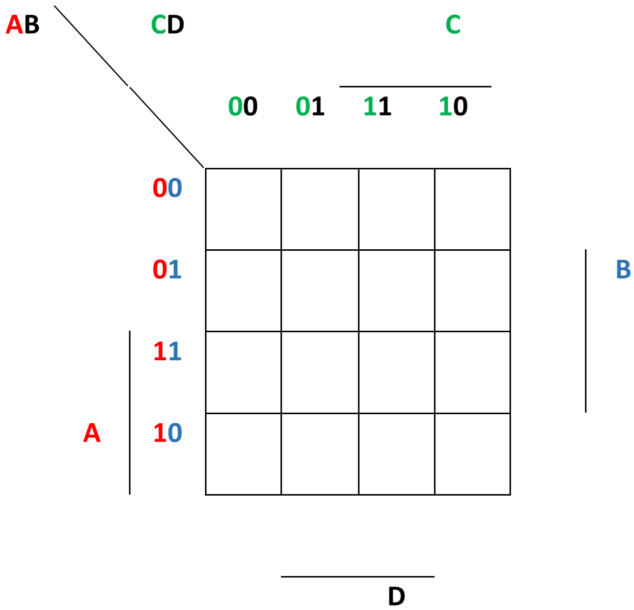
22. ábra A négy változós Igazságtábla

Az ábrázolásához azonban a Karnaugh – táblában új sorokat kell természetesen felvennünk.



23. ábra A négy változós Karnaugh – tábla

Fontos lekövetni, hogy melyik változó mely cellában Igaz vagy hamis értékű. Ezért a következő ábrán színekkel jelöltük meg az összetartozó értékeket.



24. ábra A négy változós Karnaugh – tábla

AB \ CD	00	01	11	10
00	$\overline{A}\,\overline{B}\,\overline{C}\,\overline{D}$	$\overline{A}\,\overline{B}\,\overline{C}D$	$\overline{A}\,\overline{B}CD$	$\overline{A}\,\overline{B}C\overline{D}$
01	$\overline{A}B\overline{C}\,\overline{D}$	$\overline{A}B\overline{C}D$	$\overline{A}BCD$	$\overline{A}BC\overline{D}$
11	$AB\overline{C}\,\overline{D}$	$AB\overline{C}D$	$ABCD$	$ABC\overline{D}$
10	$A\overline{B}\,\overline{C}\,\overline{D}$	$A\overline{B}\,\overline{C}D$	$A\overline{B}CD$	$A\overline{B}C\overline{D}$

C (columns 11, 10); B (rows 01, 11); A (rows 11, 10); D (columns 01, 11)

25. ábra A négy változós Karnaugh – tábla értékei (változók)

AB \ CD	00	01	11	10
00	0000	0001	0011	0010
01	0100	0101	0111	0110
11	1100	1101	1111	1110
10	1000	1001	1011	1010

C (columns 11, 10); B (rows 01, 11); A (rows 11, 10); D (columns 01, 11)

26. ábra A négy változós Karnaugh – tábla értékei (binaris)

	AB \ CD	00	01	11 (C)	10 (C)	
	00	0	1	3	2	
	01	4	5	7	6	B
A	11	12	13	15	14	B
A	10	8	9	11	10	
			D	D		

27. ábra A négy változós Karnaugh – tábla értékei (decimális)

2.2.6.3. A Karnaugh – tábla kitöltése

Mivel megállapítottuk, hogy az előbbi példában (szavazatszámláló) a táblázat a diszjunktív logikai függvény értékeit tartalmazza, ezért a példaként használt szavazatszámláló estét is le tudjuk írni a segítségével.

Mivel tudjuk, hogy a feladat teljes diszjunktív alakja a következő:

$$Y = AB\overline{C} + A\overline{B}C + \overline{A}BC + ABC$$

Ebben a logikai függvényben meg tudjuk határozni a mintermeket.

$$Y = \underbrace{\boxed{AB\overline{C}}}_{m^3_3} + \underbrace{\boxed{A\overline{B}C}}_{m^3_5} + \underbrace{\boxed{\overline{A}BC}}_{m^3_6} + \underbrace{\boxed{ABC}}_{m^3_7}$$

Ha megfigyeljük a három változós Karnaugh – táblát, akkor azt is észrevehetjük, hogy ezek a mintermek megtalálhatók a táblázatban.

Itt láthatjuk azt az összefüggést, hogy a táblázat cellái mintermeket jelölnek (A konjunktív alak felírása esetén maxtermeket jelölnének). Azokba a mintermekbe, melyek megjelennek az egyenletben egy 1-es értéket írunk.

Tehát mivel az egyenlet:

$$Y = AB\overline{C} + A\overline{B}C + \overline{A}BC + ABC$$

Ezért a kitöltött Karnaugh tábla:

A \ BC	00	01 (C)	11 (C)	10
0	$\overline{ABC}$	$\overline{AB}C$	$\overline{A}BC$	$\overline{A}B\overline{C}$
1 (A)	$A\overline{BC}$	$A\overline{B}C$	ABC	$AB\overline{C}$

28. ábra Az eAz egyenlet Karnaugh táblás felírása

A \ BC	00	01 (C)	11 (C)	10
0	**0**	**0**	**1**	**0**
1 (A)	**0**	**1**	**1**	**1**

29. ábra A kitöltött Karnaugh – tábla

Sok esetben a könnyebb használat érdekében a 0 vértékű változókat nem jelöljük. Ha pedig figyelembe vesszük a mezők értékeit, akkor a teljes Diszjunkt függvény minterm alakját is megkaphatjuk. Ez az egyik oka a táblázat használatának: a mintermmel megadott függvény nagyon könnyen kiolvasható a táblázatból.

A \ BC	00	01	11	10
0	0	1	3	2
A 1	4	5	7	6

30. ábra A három változós Karnaugh - tábla cellaértékeiből kiolvasható a mintermek felírása

$$Y^3 = \sum(3,5,6,7)$$

2.2.6.4. A Karnaugh – tábla egyszerűsítése

Természetesen ugyanúgy, ahogyan az algebrai alak esetében, itt is élhetünk az egyszerűsítés lehetőségével. Tulajdonképpen a táblát pont azért alakították ki, hogy az egyszerűsítés gyorsan és könnyen elvégezhető legyen.

Az egyszerűsítés elve az algebrai egyszerűsítéseknél is használt közös tényező kiemelése.

Másrészt kihasználjuk azt is, hogy a táblában egymás melletti (alatti) cellában olyan mintermek, vannak amelyek csak 1 változóban térnek el.

Az egyszerűsítés során az azonos részt kiemelhetjük, a megmaradó változó és negáltja pedig kiesik, azaz kihasználjuk a korábban felhasznált azonosság igazságtartalmát:

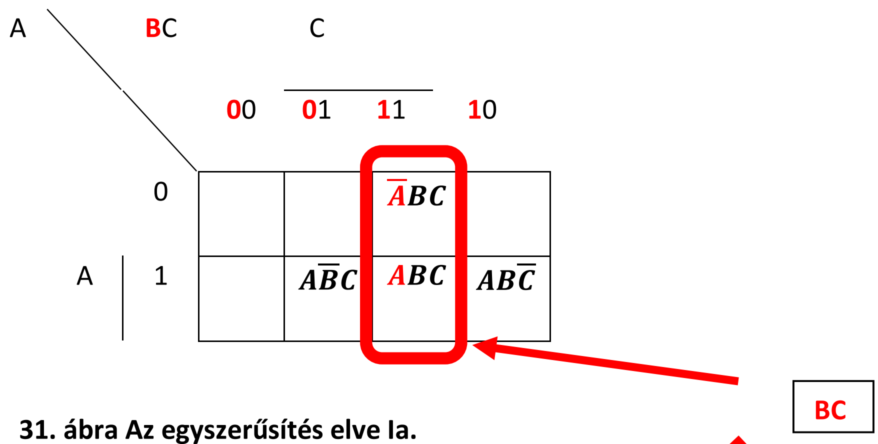
$$\overline{A} + A = \overline{B} + B = \overline{C} + C = 1$$

Az egyszerűsítés során éppen ezért az ilyen kieső párokat fogjuk megkeresni. Ezek pedig nem lesznek mások, mint a Karnaugh táblában egymás mellé került 1-es értékek (hiszen ezek fognak olyan tagot tartalmazni, melyek kiejthetők.

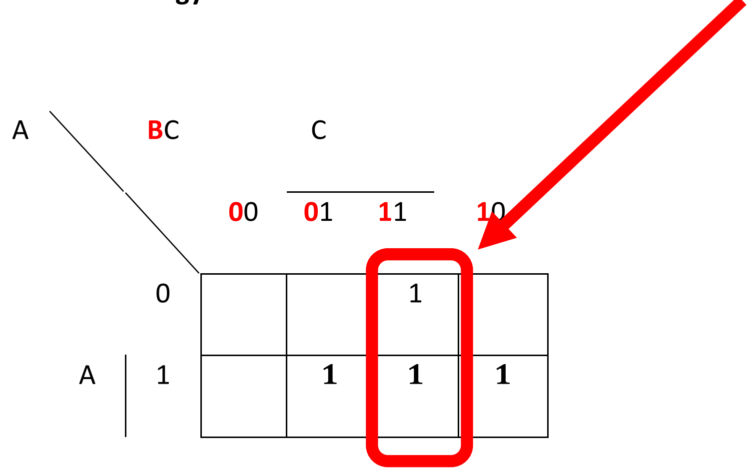
1) **Az egymás mellett lévő összevonandó 1-eseket egy hurokkal vesszük körül, és ezután már csak ennek a huroknak az eredményt tüntetjük fel.** Ennek a módszernek nagy előnye, hogy mivel a szomszédos mintermek, vagyis az összevonási lehetőségek azonnal észrevehetők.

A gyakorlatban ez a következő módon zajlik:

Három hurkot képezünk, és a tagonként ejtjük ki A változót és negáltját, majd B változót és Negáltját, végül C-t és negáltját.



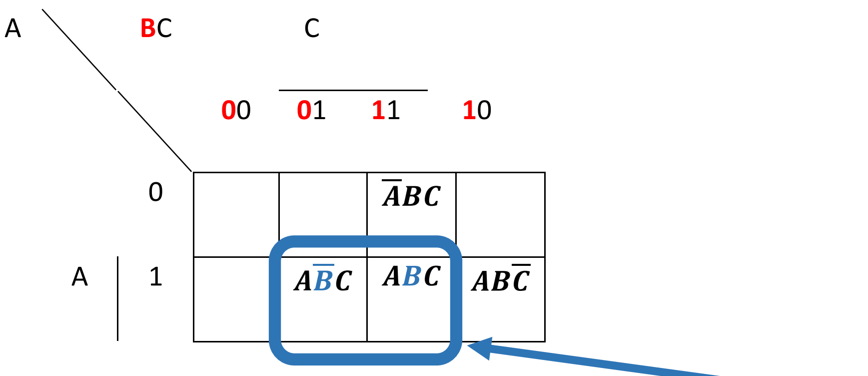
31. ábra Az egyszerűsítés elve Ia.



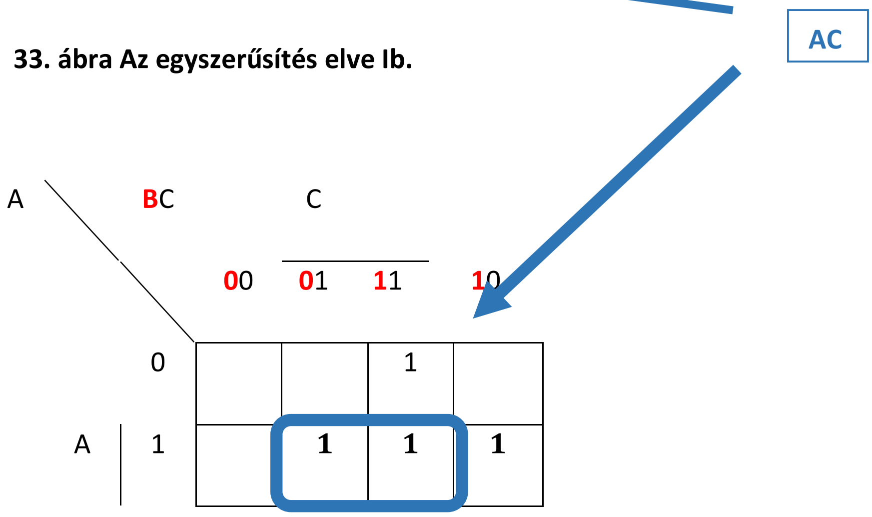
32. ábra Az egyszerűsítés elve IIa.

A példában a hurokkal körülvett mintermekben B = 1 és C = 1 állandó

Másképpen a hurokban A 1-essel és 0-ával is szerepel (nem állandó), ezért kiesik. Az összevonás eredménye BC.



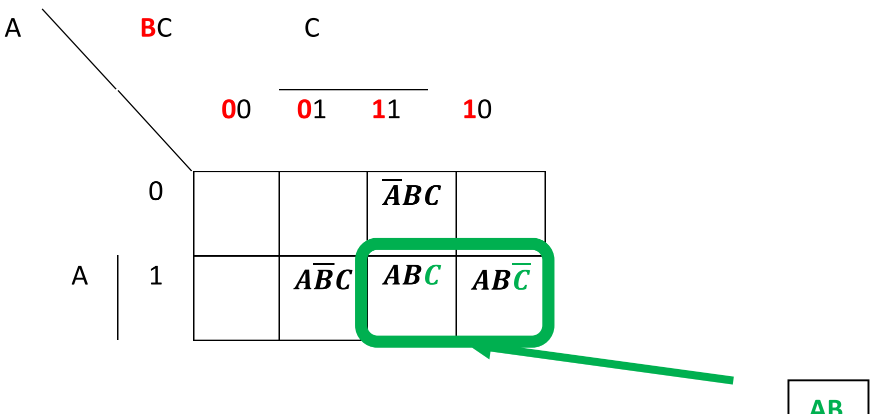
33. ábra Az egyszerűsítés elve Ib.



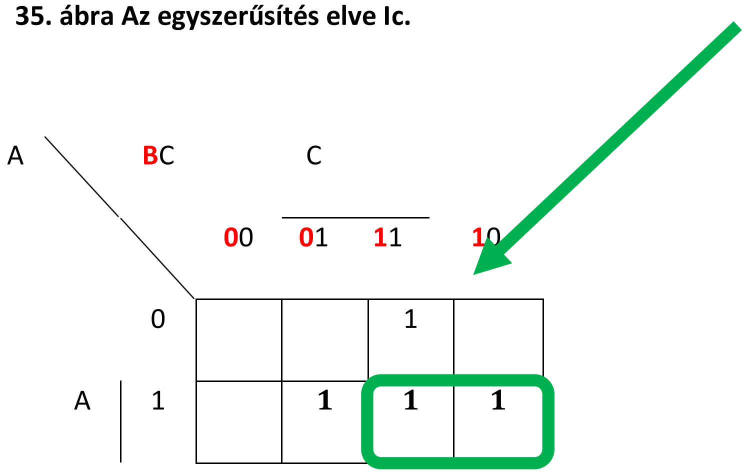
34. ábra Az egyszerűsítés elve IIb.

A példában a hurokkal körülvett mintermekben A = 1 és C = 1 állandó. Másképpen a hurokban B 1-essel és 0-ával is szerepel (nem állandó), ezért kiesik.

Az összevonás eredménye AC.



35. ábra Az egyszerűsítés elve Ic.

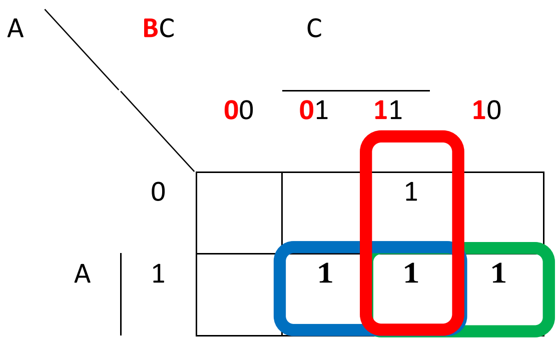


36. ábra Az egyszerűsítés elve IIc.

A példában a hurokkal körülvett mintermekben A = 1 és B = 1 állandó.

Másképpen a hurokban C 1-essel és 0-ával is szerepel (nem állandó), ezért kiesik. Az összevonás eredménye AB.

Összesítve az alábbi eredményre jutunk:



37. ábra Az egyszerűsítés elve abc összesítve.

Ezért az egyszerűsítés eredménye a következő:

$$Y = A \cdot B + A \cdot C + B \cdot C$$

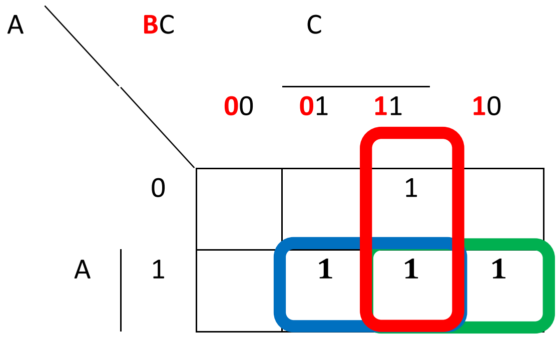
Ez pedig nem más, mint a korábban algebrai úton is meghatározott nem teljes diszjunktív alak. Természetesen a konjuktív alakkal ugyanígy tudunk eljárni.

Természetesen ezt nagyobb táblázatban is megtehetjük.

Az egyszerűsítésnek vannak további szabályai is:

2) **Egy logikai függvényben egy tagot tetszés szerint ismételhetünk az egyszerűsítés érdekében, így a Karnaugh-tábla bármely 1-esét is akárhány hurokba bevonhatjuk.**

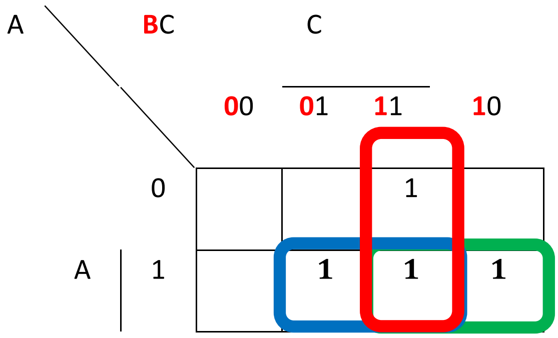
Ez jól látható az előbbi példában, ahol a középső alsó eggyes három huroknak is tagja.



38. ábra A középső alsó egyes 3 hurok tagja

3) **Minden 1-est legalább egyszer be kell vonni legalább egy hurokba.**

Az előző példában minden egyes tagja egy huroknak.

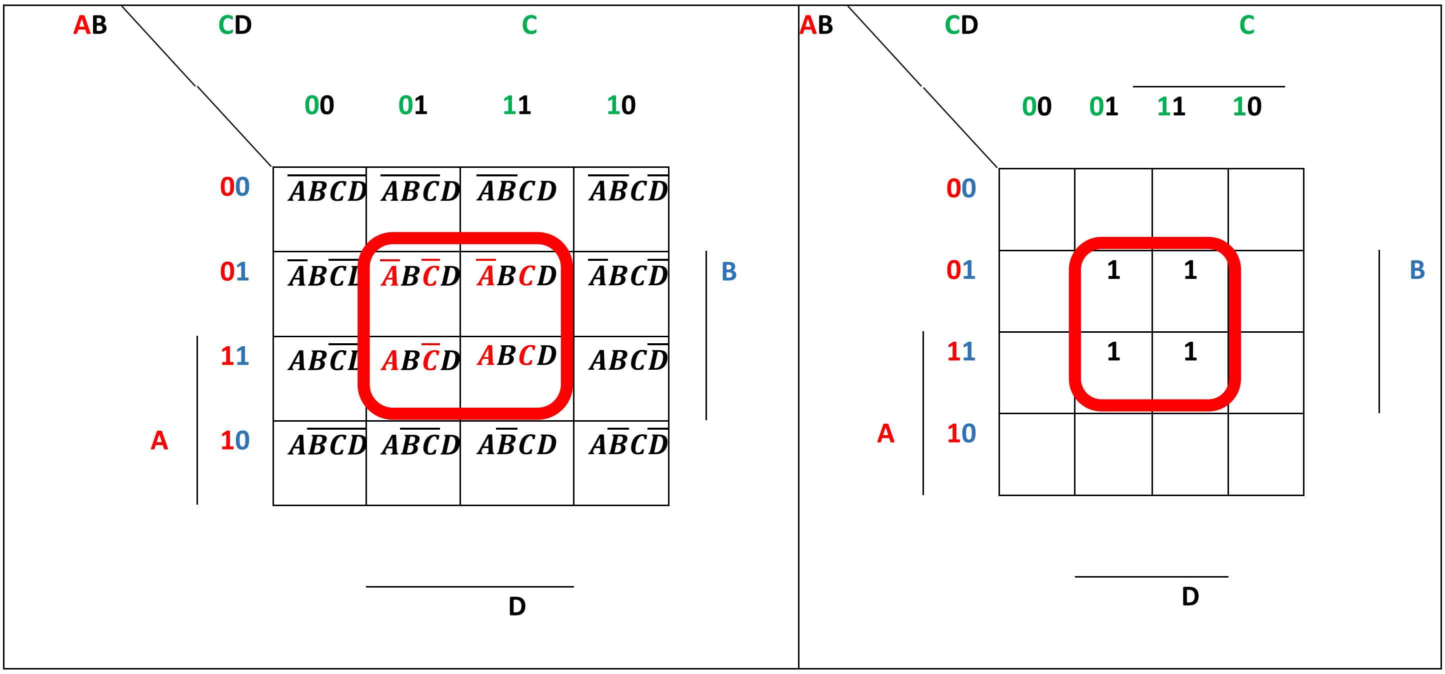


39. ábra Minden egyes legalább egy hurok tagja

4) **Ha nem tudjuk összevonni semmivel, egyetlen cella alkotja a hurkot (nem lehet egyszerűsíteni).**

5) Nemcsak 2, hanem 2 bármely egész számú hatványa darabszámú szomszédos minterm összevonható.

Négy változós esetben ez nagyon szépen levezethető.



Tehát az eredmény:

$$Y = BD$$

6) A tábla a széleken összefügg.

AB \ CD	00	01	11	10
00	$\overline{ABCD}$	$\overline{ABC}D$	$\overline{AB}CD$	$\overline{AB}C\overline{D}$
01	$\overline{A}B\overline{CD}$	$\overline{A}B\overline{C}D$	$\overline{A}BCD$	$\overline{A}BC\overline{D}$
11	$AB\overline{CD}$	$AB\overline{C}D$	$ABCD$	$ABC\overline{D}$
10	$A\overline{BCD}$	$A\overline{BC}D$	$A\overline{B}CD$	$A\overline{B}C\overline{D}$

A, B, C, D

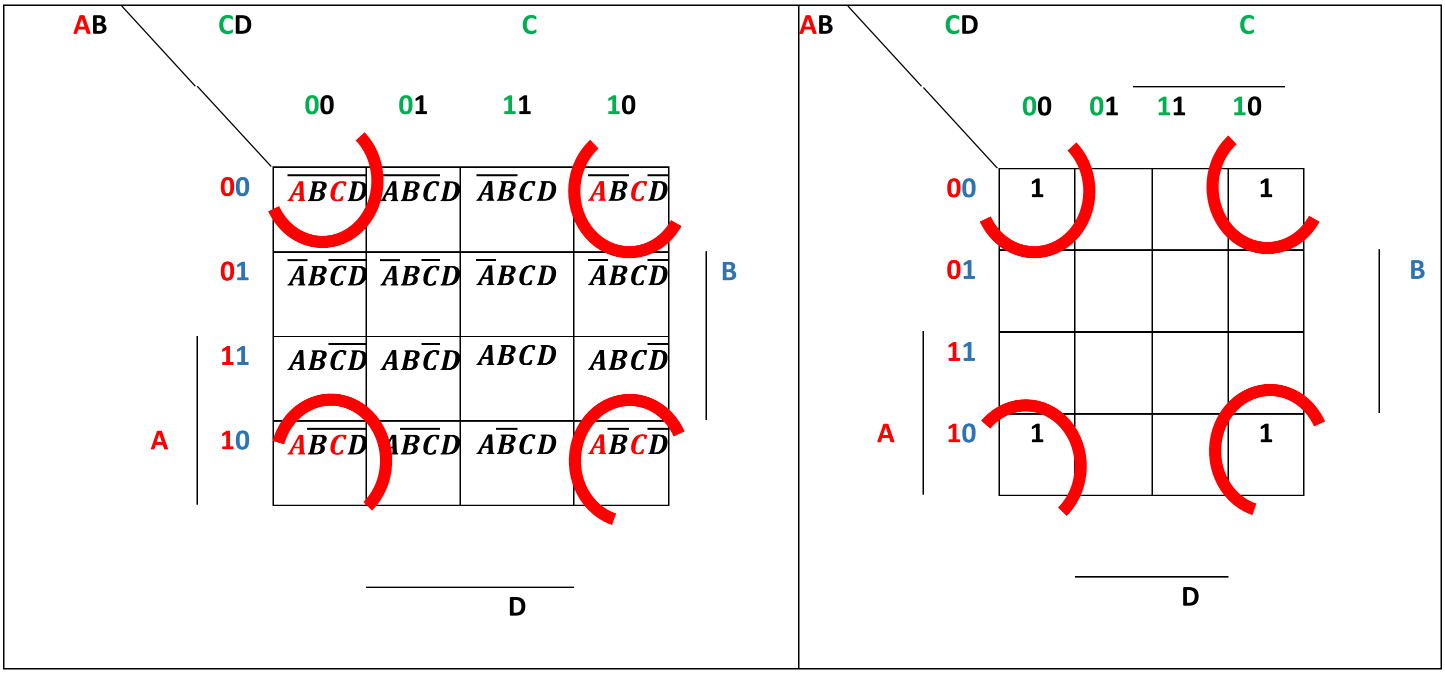
AB \ CD	00	01	11	10
00	1			1
01				
11				
10	1			1

A, B, C, D

Tehát az eredmény:

$$Y = \overline{BD}$$

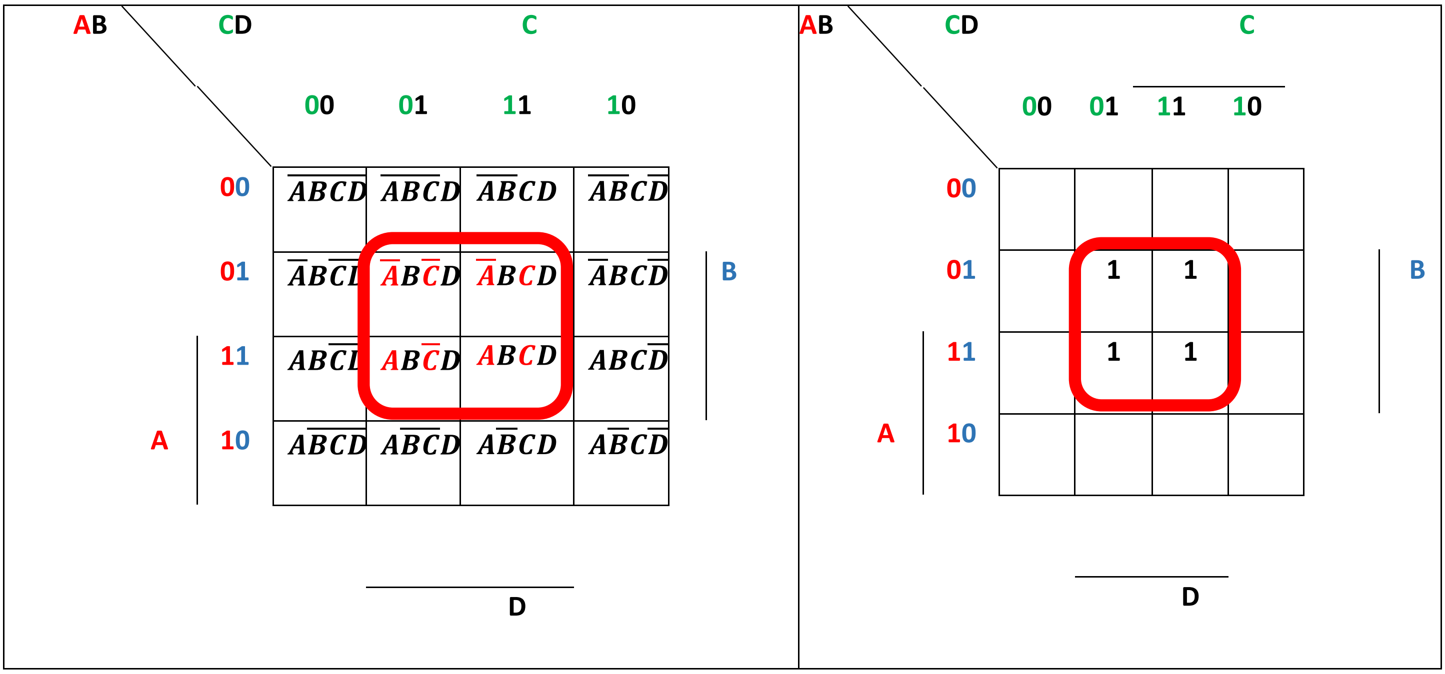
7) **Egymás mellettinek ill. alattinak számítanak a sorok, ill. oszlopok két végén levő 1-esek is.**



Tehát az eredmény:

$$Y = \overline{BD}$$

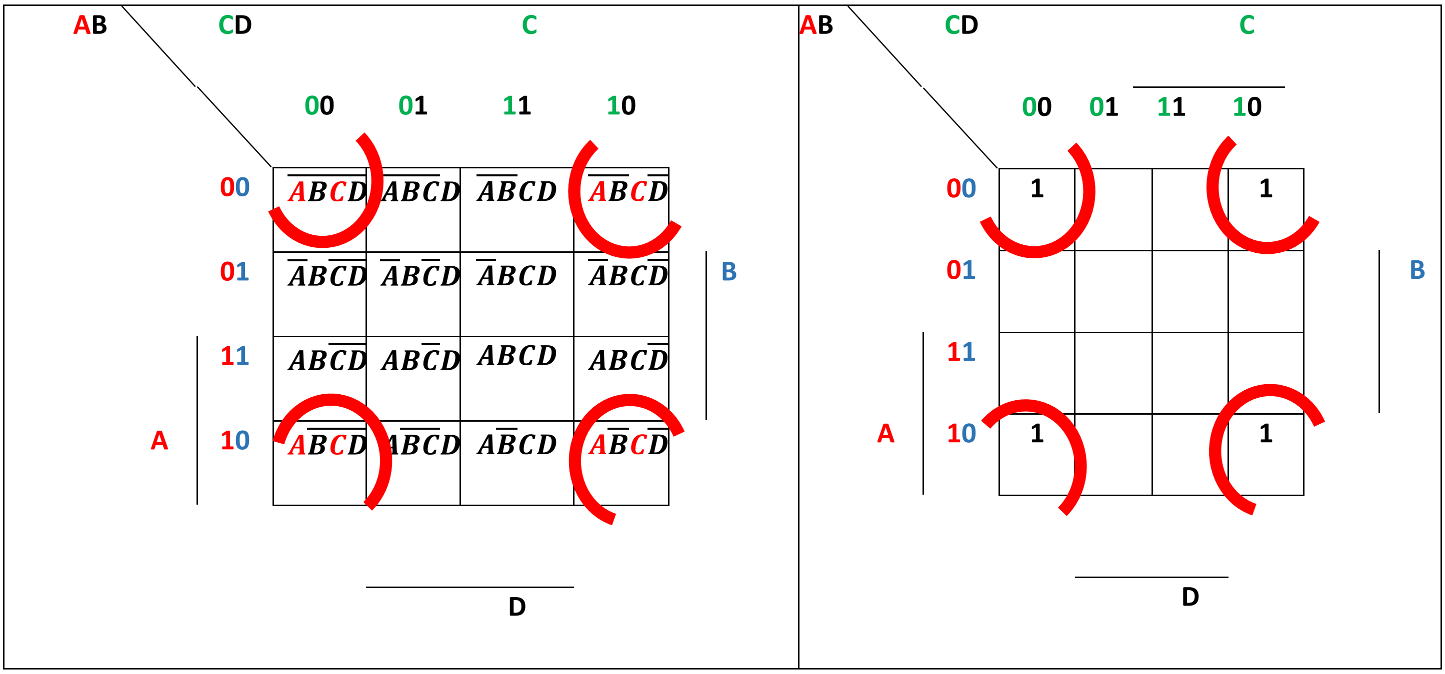
8) 4 db négyzet alakban elhelyezkedő 1-es összevonható



Tehát az eredmény:

$$Y = BD$$

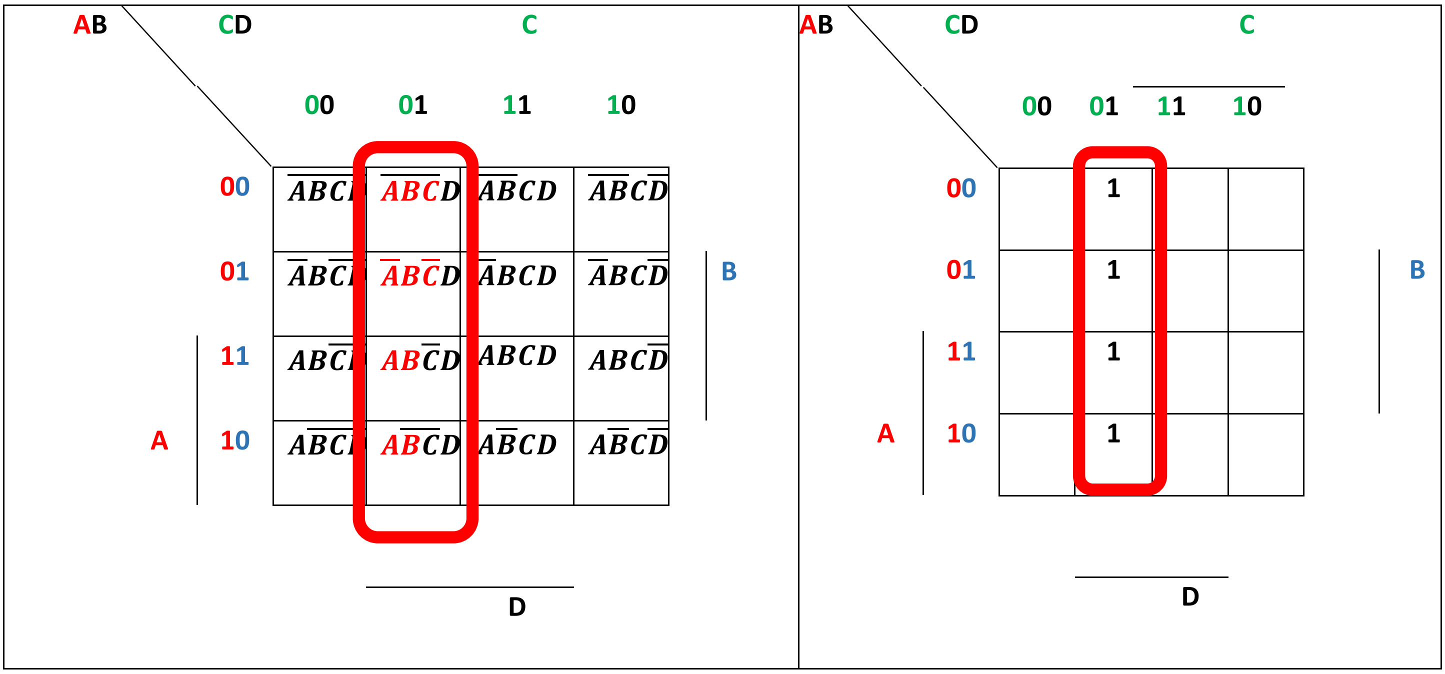
9) A négy sarokban lévő 1-es is négyzet alaknak számít



Tehát az eredmény:

$$Y = \overline{BD}$$

10) Teljes sorok valamint teljes oszlopok összevonhatók



Tehát az eredmény:

$$Y = \overline{C}D$$

11) Két szomszédos teljes sor vagy oszlop összevonható

AB \ CD	00	01	11	10	
00	$\overline{ABCD}$	$\overline{ABC}D$	$\overline{AB}CD$	$\overline{AB}C\overline{D}$	
01	$\overline{A}B\overline{CD}$	$\overline{A}B\overline{C}D$	$\overline{A}BCD$	$\overline{A}BC\overline{D}$	B
A 11	$AB\overline{CD}$	$AB\overline{C}D$	$ABCD$	$ABC\overline{D}$	B
A 10	$A\overline{BCD}$	$A\overline{BC}D$	$A\overline{B}CD$	$A\overline{B}C\overline{D}$	
		D	D		

C: 11, 10

AB \ CD	00	01	11	10	
00					
01					B
A 11	1	1	1	1	B
A 10	1	1	1	1	
		D	D		

C: 11, 10

Tehát az eredmény:

$$Y = A$$

12) Minél nagyobb hurkokat képzünk, annál több változó esik ki, annál egyszerűbb természetesen a végeredmény.

2.3. További példa: Digitális komparátor

Vegyünk egy további példát, ahol megtanulhatjuk a kombinációs hálózatok felírását, természetesen a korábban megismert felírási lehetőségekkel.

2.3.1. Szöveges megfogalmazás

Legyen egy digitális komparatorunk, melynek az a feladata, hogy két binárisan felírt számot hasonlítson össze. A két szám legyen eltárolva két biten.

2.3.2. Blokk

Ebben az esetben a rendszer bemenete az a négy bit ahol a két számot eltároljuk, kimenete pedig az a három változó, melyek igaz értékeivel képes eldönteni a két szám viszonyát. Erre azért van szükség, mert a három kimenet állapota csak igaz vagy hamis lehet, egyéb értéket nem vehet fel. Így az igaz állapot „helyzete” mondja meg, hogy mi a két szám viszonya.

Ez azt jelenti, hogy

- $Y_0 = 1$ ha $A > B$ egyébként 0
- $Y_1 = 1$ ha $A = B$ egyébként 0
- $Y_2 = 1$ ha $A < B$ egyébként 0

40. ábra A komparátor.

A rendszerben legyen (Azért, hogy az előbb tanult felírási módot tudjuk követni $A_0 = D$, $A_1=C$, $B_0= B$ és $B_1= A$.).

2.3.3. Igazságtáblázat

Ezután felrajzoljuk a rendszer igazságtáblázatát, amely 4 független változót és 3 függő változót tartalmaz, ezért 7 oszlopa lesz, viszont 4 független változójából következik, hogy 16 sorral fog rendelkezni. Felírjuk A, B C, D lehetséges kombinációit, és kitöltjük a táblázatot.

i	A	B	C	D	Y_0	Y_1	Y_2
0	0	0	0	0	0	1	0
1	0	0	0	1	0	0	1
2	0	0	1	0	0	0	1
3	0	0	1	1	0	0	1
4	0	1	0	0	1	0	0
5	0	1	0	1	0	1	0
6	0	1	1	0	0	0	1
7	0	1	1	1	0	0	1
8	1	0	0	0	1	0	0
9	1	0	0	1	1	0	0
10	1	0	1	0	0	1	0
11	1	0	1	1	0	0	1
12	1	1	0	0	1	0	0
13	1	1	0	1	1	0	0
14	1	1	1	0	1	0	0
15	1	1	1	1	0	1	0

41. ábra A komparátor igazságtáblázata

2.3.4. Logikai függvények

A logikai függvény felírásához csak egy egyenletrendszert veszünk most figyelembe, hogy egyszerűbben fel tudjuk írni a függvényt.

Tekintsük most csak azt az esetet, amikor

– $Y_0 = 1$ ha A > B egyébként 0

– Ekkor az igazságtáblában az alábbi sorok lesznek fontosak.

	i	A	B	C	D	Y_0	Y_1	Y_2
	0	0	0	0	0	0	1	0
	1	0	0	0	1	0	0	1
	2	0	0	1	0	0	0	1
	3	0	0	1	1	0	0	1
Y1	**4**	**0**	**1**	**0**	**0**	**1**	0	0
	5	0	1	0	1	0	1	0
	6	0	1	1	0	0	0	1
	7	0	1	1	1	0	0	1
Y2	**8**	**1**	**0**	**0**	**0**	**1**	0	0
Y3	**9**	**1**	**0**	**0**	**1**	**1**	0	0
	10	1	0	1	0	0	1	0
	11	1	0	1	1	0	0	1
Y4	**12**	**1**	**1**	**0**	**0**	**1**	0	0
Y5	**13**	**1**	**1**	**0**	**1**	**1**	0	0
Y6	**14**	**1**	**1**	**1**	**0**	**1**	0	0
	15	1	1	1	1	0	1	0

Az ebből felírható részegyenletek (soronként meghatározott mintermek):

$$Y1 = \overline{A}B\overline{CD}$$

$$Y2 = A\overline{BCD}$$

$$Y3 = A\overline{BC}D$$

$$Y4 = AB\overline{CD}$$

$$Y5 = AB\overline{C}D$$

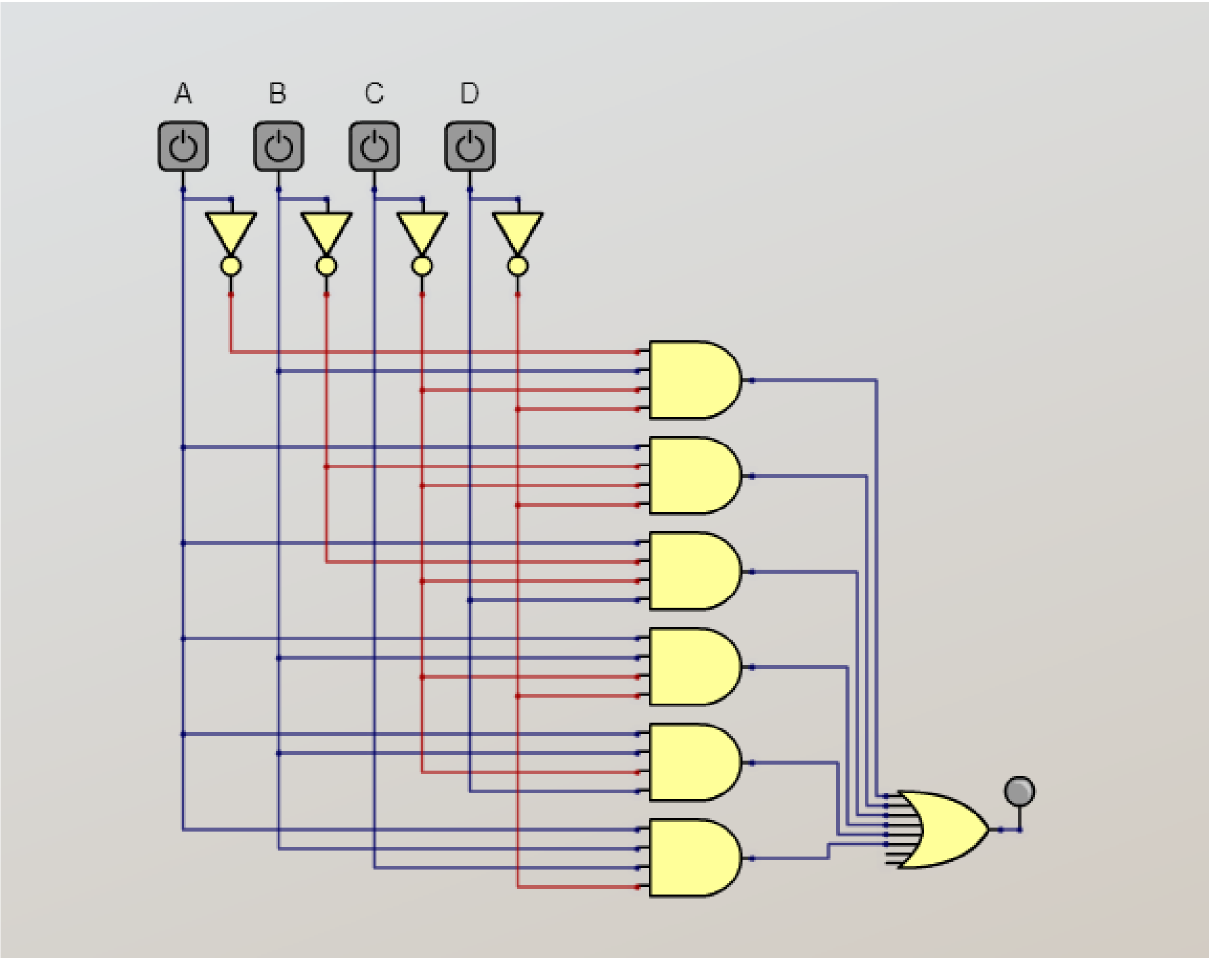
$$Y6 = ABC\overline{D}$$

Tehát az egyenlet:

$$Y = \overline{A}B\overline{CD} + A\overline{BCD} + A\overline{BC}D + AB\overline{CD} + AB\overline{C}D + ABC\overline{D}$$

2.3.5. Kapcsolási rajz

Ebből az egyenletből felépíthetjük a hálózatot is. Ehhez 6 darab És és egy VAGY kapura van szükségünk. Ezen kívül természetesen minden bemenetnél képeznünk kell a negáltat is.



42. ábraA komparátor kapcsolási rajza

2.3.6. Karnaugh - tábla

A Karnaugh tábla felírásához szükség van az előbbi egyenletre.

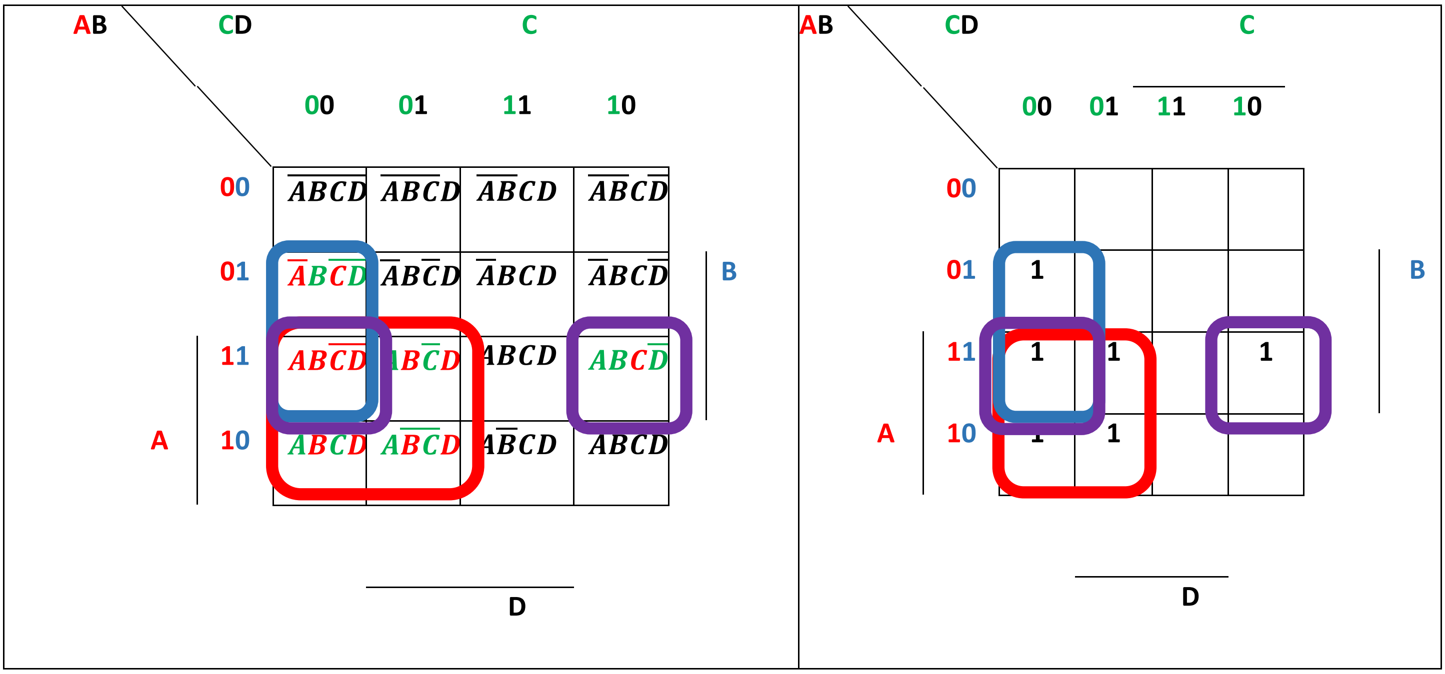
$$Y = \overline{A}B\overline{CD} + A\overline{BCD} + A\overline{BC}D + AB\overline{CD} + AB\overline{C}D + ABC\overline{D}$$

Ennek elemeit keressük meg a táblában, és írunk a helyére egyes értéket.

AB \ CD		00	01	11	10	
				C	C	
	00	$\overline{ABCD}$	$\overline{ABC}D$	$\overline{AB}CD$	$\overline{AB}C\overline{D}$	
	01	$\overline{A}B\overline{CD}$	$\overline{A}B\overline{C}D$	$\overline{A}BCD$	$\overline{A}BC\overline{D}$	B
A	11	$AB\overline{CD}$	$AB\overline{C}D$	$ABCD$	$ABC\overline{D}$	B
A	10	$A\overline{BCD}$	$A\overline{BC}D$	$A\overline{B}CD$	$A\overline{B}C\overline{D}$	
			D	D		

AB \ CD		00	01	11	10	
				C	C	
	00					
	01	1				B
A	11	1	1		1	B
A	10	1	1			
			D	D		

Ezután a tábla egyszerűsíthető: (Pirossal a kieső tagok):

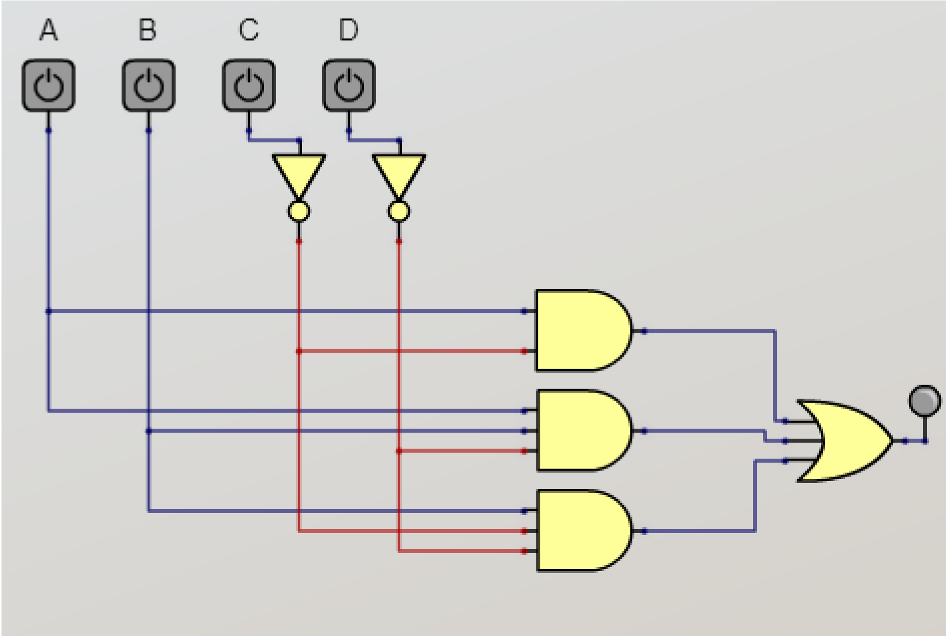


Az egyszerűsítés után kapható egyenlet:

$$Y = \overline{A}B\overline{CD} + A\overline{BCD} + A\overline{BC}D + AB\overline{CD} + AB\overline{C}D + ABC\overline{D}$$

$$Y = A\overline{C} + AB\overline{D} + B\overline{CD}$$

Ezzel az egyszerűsítéssel tényleg egy könnyebben kezelhető rendszert kaphatunk: így már csak 3 ÉS kapu és egy VAGY Kapu szükséges a rendszer felépítéséhez. A négy negált helyett pedig csak kettőt használunk.



43. ábra Az egyszerűsített kapcsolási rajz

+1) Közömbös függvényértékek kezelése (Dont't care)

Közömbös függvényértékek esetén tudjuk, hogy

- ❖ Biztos, hogy az adott változókombináció nem következik be
  pl. BCD kód
- ❖ Nem lényeges, hogy az adott változókombinációnál hogy viselkedik a rendszer
  Pl. olyan memória cím, ahol nincs fizikailag semmilyen eszköz
- ❖ Kikötjük, hogy nem szabad az adott változókombinációt a bemenetre kötni (pl. Reserved)
- ❖ X-szel jelöljük.

Az egyszerűsítés során pedig:

- ❖ Cél minél több 1-est, minél nagyobb hurokba bevonni.
- ❖ Az X-eket nem kötelező de be lehet vonni hurokba, ezért ha kell X-ek felhasználásával egyszerűsítünk, hogy minél egyszerűbb alakot kapjunk.

$$Y = A + C + BD + \overline{BD}$$

2.4. Több kimenetű rendszerek

Több kimenetű rendszerek esetén a feladat egyszerű: annyi Karnaugh táblát kell felrajzolni, ahány kimenetünk van.

Az alábbiakban erre is tekintsünk egy példát.

2.4.1. Karnaugh tábla

Két (több) kimenet esetén a rendszert több táblába tudjuk felírni.

Kettő vagy több kimenettel rendelkező kombinációs hálózatok minimalizálásánál két külön táblába írjuk fel a rendszert. A kimeneti függvényeket külön-külön egyszerűsítjük, független de közös bemenetekkel rendelkező alhálózatokra bontva a rendszert. A függvényeket külön-külön írjuk fel, de az esetleges közös elemeket csak egyszer valósítjuk meg.

A rendszerünkről tudjuk, hogy a következő módon írható fel:

$$P = \sum^{4}(13,15)$$

$$Q = \sum^{4}(4,5,10,11,12.13,15)$$

Ezért két táblát tudunk felírni: két kimenetünk van.

P- re felírva:

$$P = \sum^{4}(13,15)$$

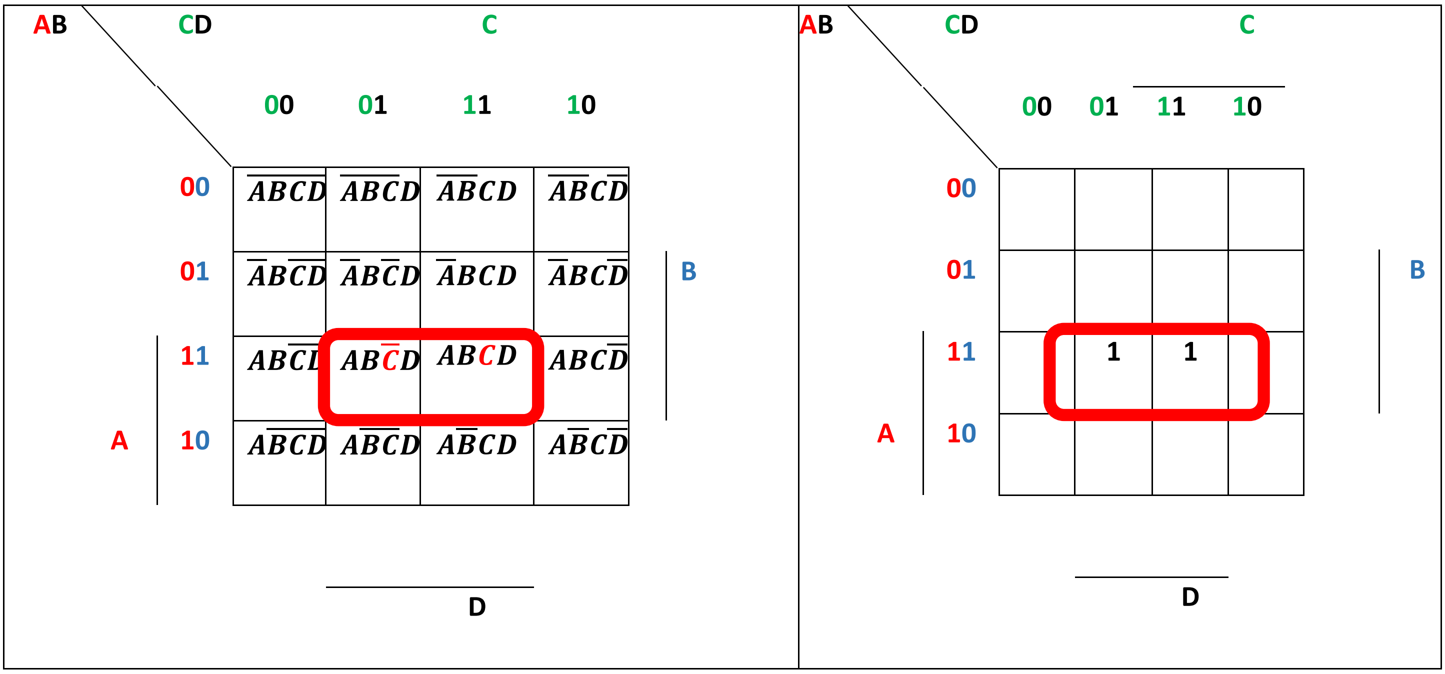
AB \ CD — C, D, A, B

AB \ CD	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

AB \ CD	00	01	11	10
00	$\overline{A}\,\overline{B}\,\overline{C}\,\overline{D}$	$\overline{A}\,\overline{B}\,\overline{C}D$	$\overline{A}\,\overline{B}CD$	$\overline{A}\,\overline{B}C\overline{D}$
01	$\overline{A}B\overline{C}\,\overline{D}$	$\overline{A}B\overline{C}D$	$\overline{A}BCD$	$\overline{A}BC\overline{D}$
11	$AB\overline{C}\,\overline{D}$	$AB\overline{C}D$	$ABCD$	$ABC\overline{D}$
10	$A\overline{B}\,\overline{C}\,\overline{D}$	$A\overline{B}\,\overline{C}D$	$A\overline{B}CD$	$A\overline{B}C\overline{D}$

AB \ CD	00	01	11	10
00				
01				
11		1	1	
10				

Ez természetesen egyszerűsíthető:

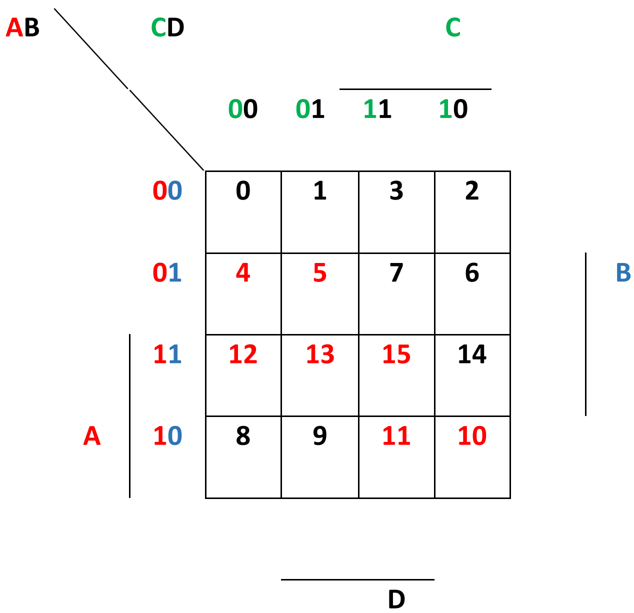


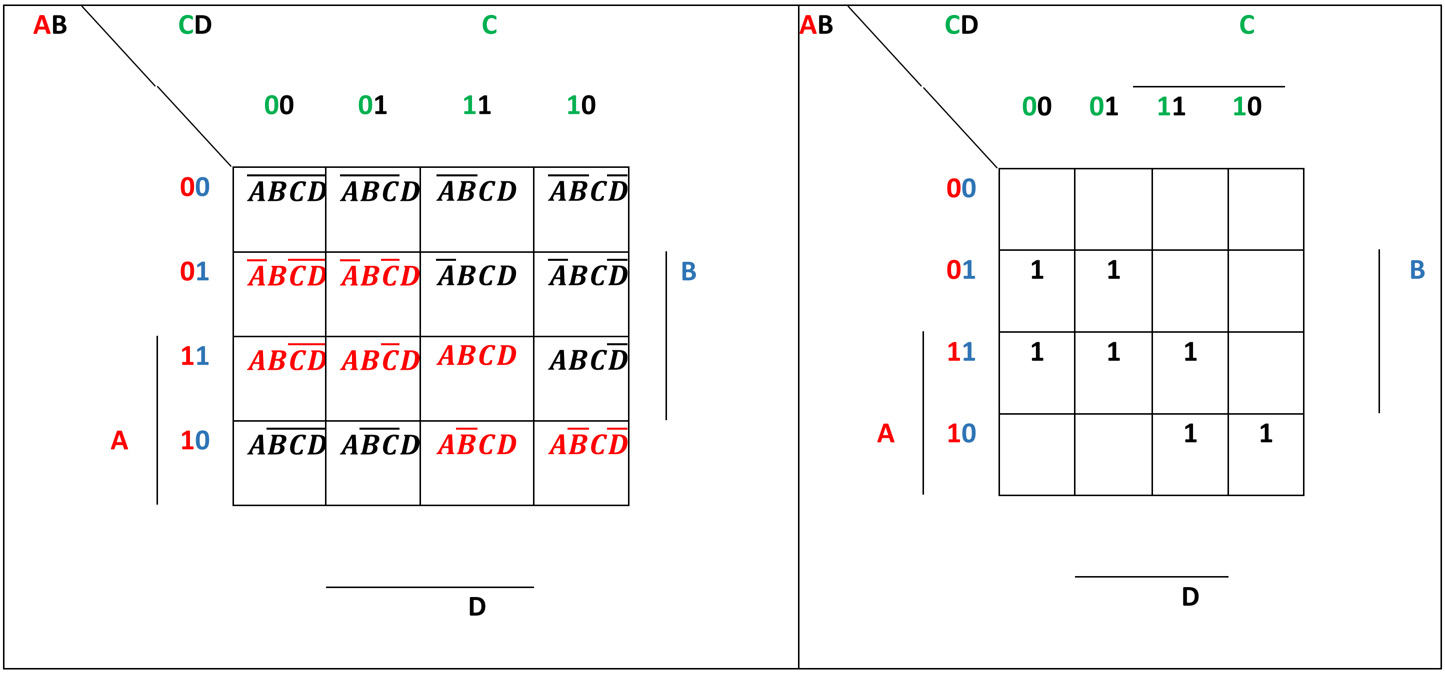
Ezért

$$P = ABD$$

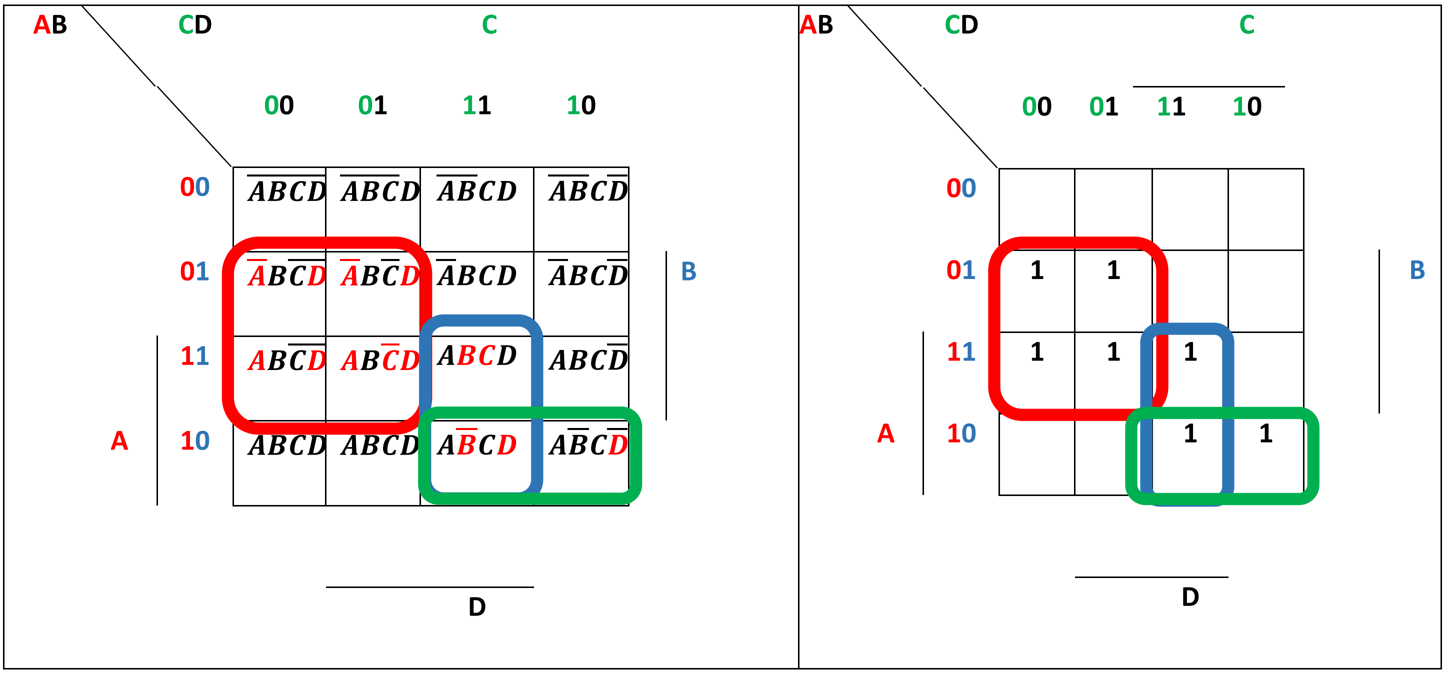
Q-ra felírva

$$Q = \sum^{4} (4,5,10,11,12.13,15)$$





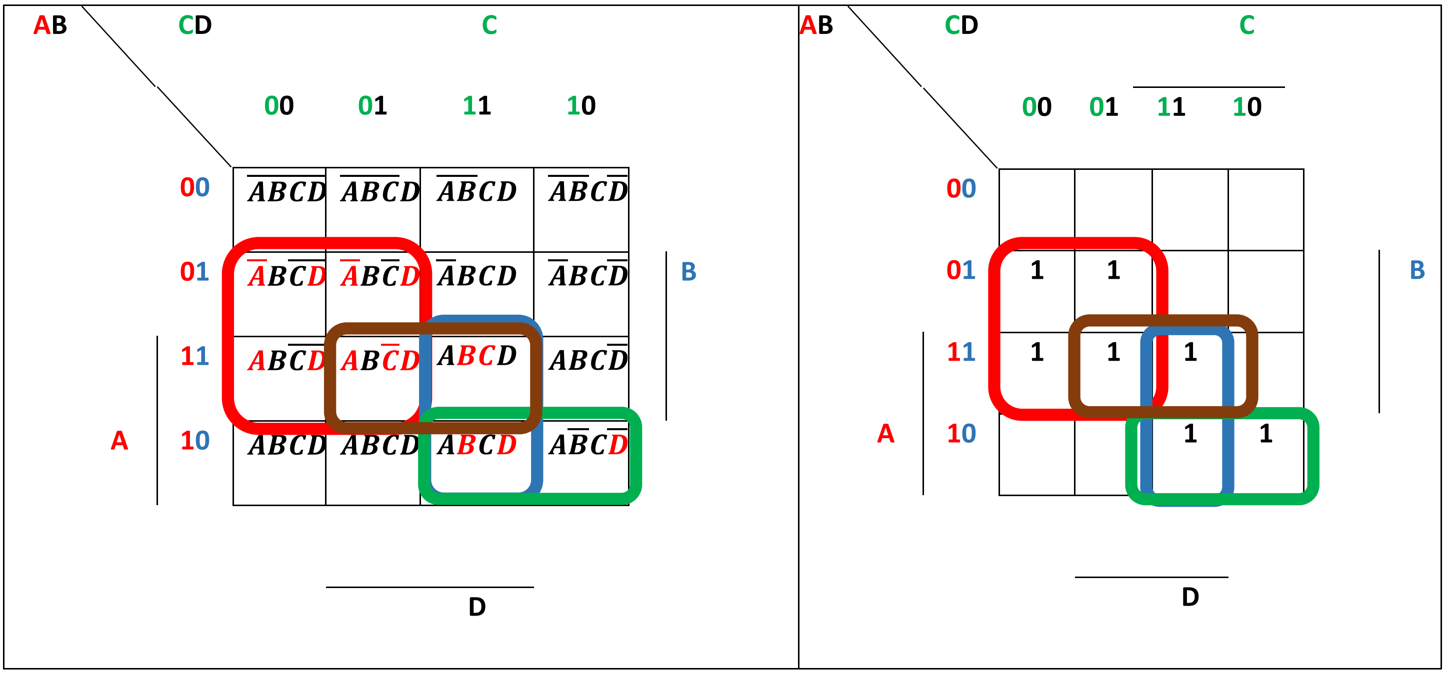
Természetesen ez is egyszerűsíthető:



Ezért az egyszerűsítés után az egyenlet a következő módon írható fel:

$$Q = B\overline{C} + ACD + A\overline{B}C$$

A két rendszert egy táblába érdemes foglalni, hogy az azonos részeket csak egyszer valósítsuk meg.



$$Q = B\overline{C} + ABD + A\overline{B}C$$

2.4.2. Igazságtáblázat

A két kimenet mintermjei alapján az igazságtáblázat is kitölthető:

$$P = \sum^{4}(13,15)$$

$$Q = \sum^{4}(4,5,10,11,12.13,15)$$

i	A	B	C	D	P	Q
0	0	0	0	0	0	0
1	0	0	0	1	0	0
2	0	0	1	0	0	0
3	0	0	1	1	0	0
4	**0**	**1**	**0**	**0**	**0**	**1**
5	**0**	**1**	**0**	**1**	**0**	**1**
6	0	1	1	0	0	0
7	0	1	1	1	0	0
8	1	0	0	0	0	0
9	1	0	0	1	0	0
10	**1**	**0**	**1**	**0**	**0**	**1**
11	**1**	**0**	**1**	**1**	**0**	**1**
12	**1**	**1**	**0**	**0**	**0**	**1**
13	1	1	0	1	1	1
14	1	1	1	0	1	0
15	1	1	1	1	1	1

2.4.3. Logikai függvények (algebrai megfogalmazás a Boole logika alapján)

A két rendszer felírása:

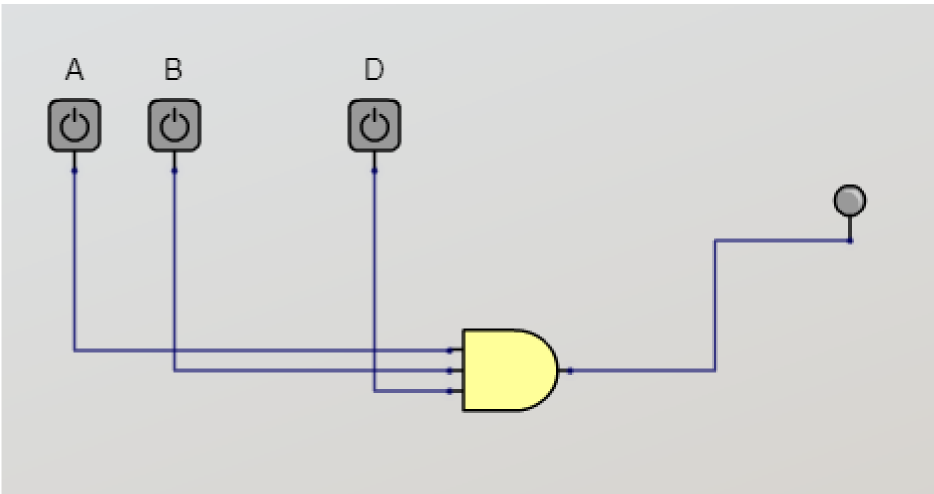
$$P = ABD$$

$$Q = B\overline{C} + ACD + A\overline{B}C$$

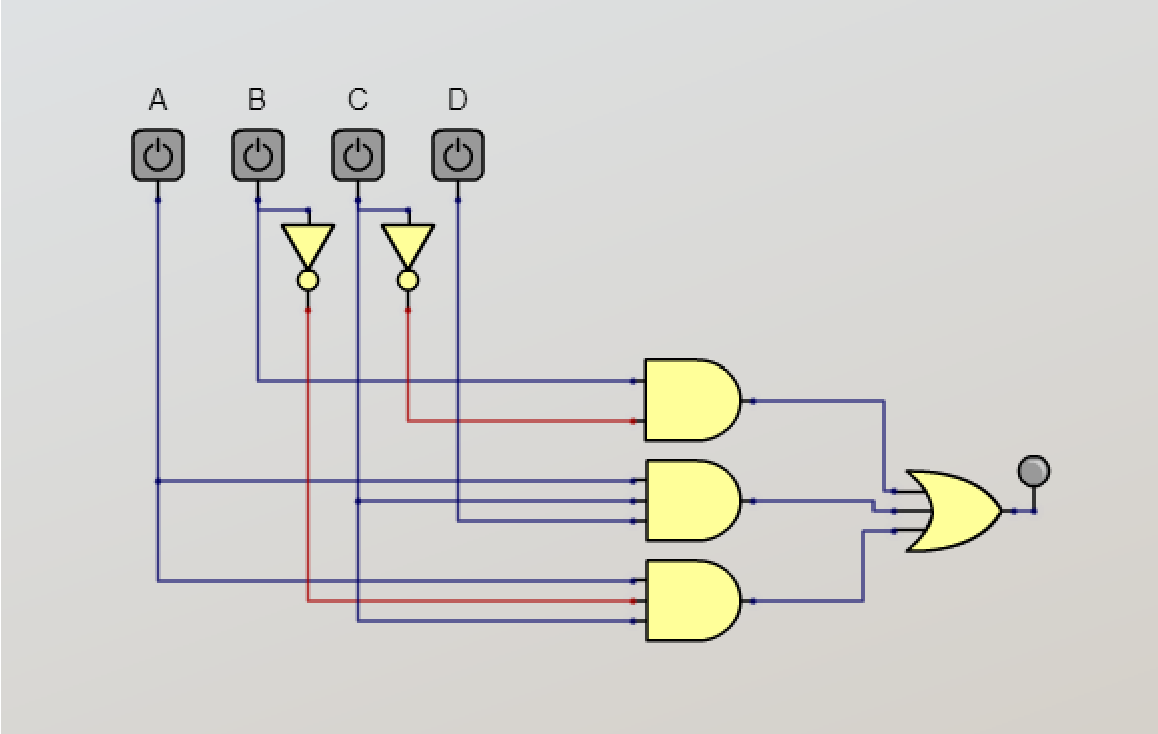
A teljes rendszerre pedig:

$$Q = B\overline{C} + ABD + A\overline{B}C$$

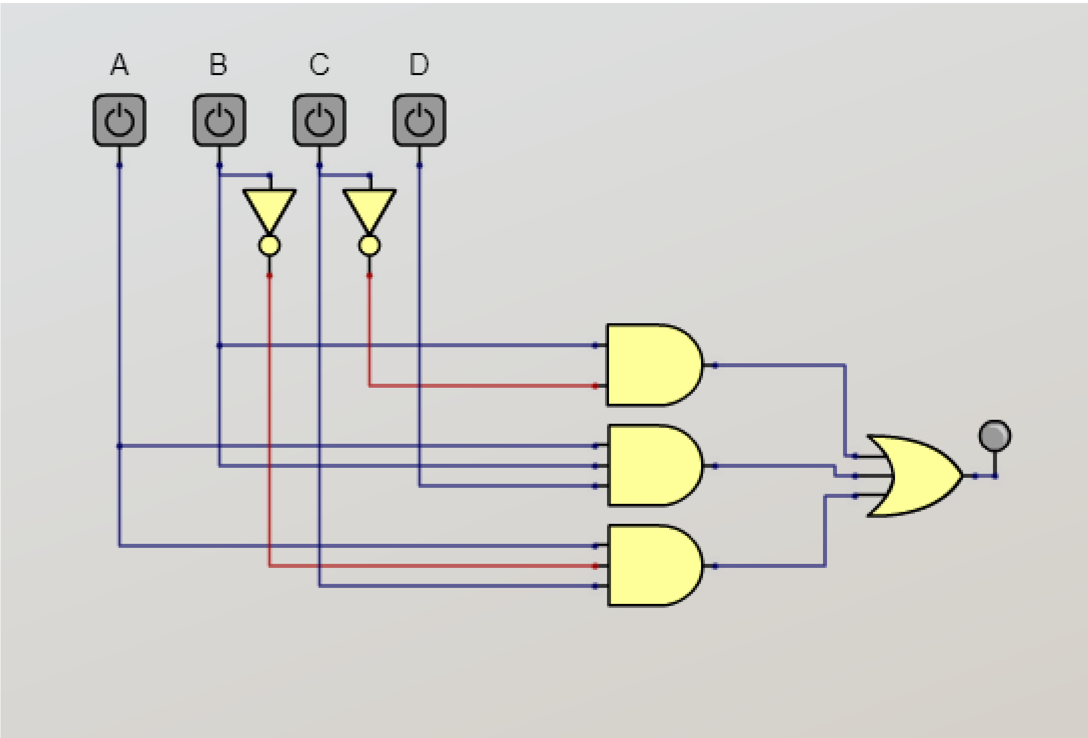
Ha felrajzoljuk a kapcsolásokat jól látható az egyszerűsítés elve is:



44. ábra P=ABD



45. ábra $Q = B\overline{C} + ACD + A\overline{B}C$



$Q = B\overline{C} + ABD + A\overline{B}C$

2.5. Ellenőrző kérdések

1) Definiálja a logikai hálózat fogalmát!
2) Definiálja a kombinációs hálózat fogalmát!
3) Definiálja a sorrendi hálózat fogalmát!
4) Milyen fontos tulajdonságai vannak a kombinációs hálózatoknak?
5) Milyen módon tudunk felírni egy kombinációs hálózatot?
6) Írja le a szavazatszámláló példa szöveges megfogalmazását!
7) Mit tud a blokkos felírási módról? Válaszát példával illusztrálja!
8) Írja fel a szavazatszámláló működését leíró igazságtáblázatot!
9) Hogyan lehet kiolvasni az igazságtáblázatból a logikai függvény diszjunkt alakját?
10) Hogyan tudjuk felírni a logikai függvény konjunkt alakját az igazságtáblázat ismeretében?
11) Definiálja a minterm fogalmát!
12) Definiálja a maxterm fogalmát!
13) Mit tud a Karnaugh – tábláról?
14) Milyen típusait ismeri a Karnaugh – táblának?
15) Hogyan határozza meg egy adott rendszerben a szükséges Karnaugh – tábla méretét?
16) Mi a kapcsolat a Karnaugh - tábla és az igazságtáblázat között?
17) Mi a kapcsolat a Karnaugh - tábla és a mintermek ill. a maxtermek között?
18) Milyen módszereket ismer egy hálózat egyszerűsítésére?
19) Írja fel a Karnaugh – tábla egyszerűsítésének szabályait! Az állításokat rajzzal illusztrálja!
20) Mit nevezünk digitális komparátornak?
21) Hogyan írhatunk fel egy hálózatot, ha tudjuk róla, hogy két kimenete van? Válaszát példával illusztrálja!

2.6. Feladatok

1) Írja fel a következő hálózatot Karnaugh – táblával, igazságtáblázattal, logikai függvénnyel és a hozzá tartozó kapcsolási rajzzal! Ahol lehet, egyszerűsítsen!

$$F = \sum^{3}(1,5,7)$$

2) Irodalom Írja fel a következő hálózatot Karnaugh – táblával, igazságtáblázattal, logikai függvénnyel és a hozzá tartozó kapcsolási rajzzal! Ahol lehet, egyszerűsítsen!

$$F = \sum^{3}(3,5,7)$$

3) Írja fel a következő hálózatot Karnaugh – táblával, igazságtáblázattal, logikai függvénnyel és a hozzá tartozó kapcsolási rajzzal! Ahol lehet, egyszerűsítsen!

$$F = \sum^{3}(3,5,6,7)$$

4) Írja fel a következő hálózatot Karnaugh – táblával, igazságtáblázattal, logikai függvénnyel és a hozzá tartozó kapcsolási rajzzal! Ahol lehet, egyszerűsítsen!

$$F = \sum^{3}(2,3,5,6,7)$$

5) Írja fel a következő hálózatot Karnaugh – táblával, igazságtáblázattal, logikai függvénnyel és a hozzá tartozó kapcsolási rajzzal! Ahol lehet, egyszerűsítsen!

$$F = \sum^{3}(0,2,3,7)$$

6) Írja fel a következő hálózatot blokkal, Karnaugh – táblával, igazságtáblázattal, logikai függvénnyel és a hozzá tartozó kapcsolási rajzzal! Ahol lehet, egyszerűsítsen!

$$F = \sum^{4}(5,7,12,14)$$

7) Írja fel a következő hálózatot blokkal, Karnaugh – táblával, igazságtáblázattal, logikai függvénnyel és a hozzá tartozó kapcsolási rajzzal! Ahol lehet, egyszerűsítsen!

$$F = \sum^{4}(5,7,12,14,9)$$

8) Írja fel a következő hálózatot blokkal, Karnaugh – táblával, igazságtáblázattal, logikai függvénnyel és a hozzá tartozó kapcsolási rajzzal! Ahol lehet, egyszerűsítsen!

$$F = \sum^{4}(0,2,5,7,8,9,10,12,14)$$

9) Írja fel a következő hálózatot blokkal, Karnaugh – táblával, igazságtáblázattal, logikai függvénnyel és a hozzá tartozó kapcsolási rajzzal! Ahol lehet, egyszerűsítsen!

$$F = \sum^{4}(2,4,5,8,9,10,14)$$

10) Írja fel a következő hálózatot blokkal, Karnaugh – táblával, igazságtáblázattal, logikai függvénnyel és a hozzá tartozó kapcsolási rajzzal! Ahol lehet, egyszerűsítsen!

$$F = \sum^{4}(0,2,4,5,8,10)$$

2.7. Irodalom

Kóré László: Digitális elektronika I. (BMF 1121)

Zsom Gyula: Digitális technika I. (Műszaki Könyvkiadó, Budapest, 2000, KVK 49-273/I, ISBN 963 6 1786 6)

Zsom Gyula: Digitális technika II. (Műszaki Könyvkiadó, Budapest, 2000, KVK 49-273/II, ISBN 963 16 1787 4)

Arató Péter: Logikai rendszerek tervezése (Tankönyvkiadó, Budapest, 1990, Műegyetemi Kiadó 2004, 55013)

Zalotay Péter: Digitális technika (http://www.kobakbt.hu/jegyzet/DigitHW.pdf)

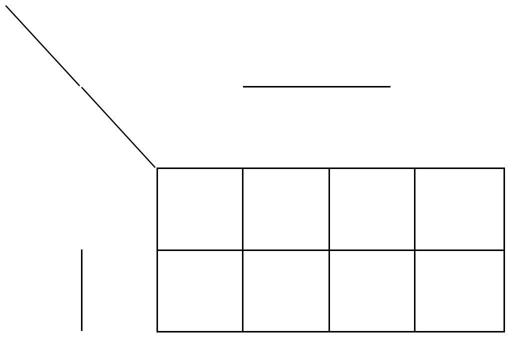
Rőmer Mária: Digitális rendszerek áramkörei (Műszaki Könyvkiadó, Budapest, 1989, KVK 49-223)

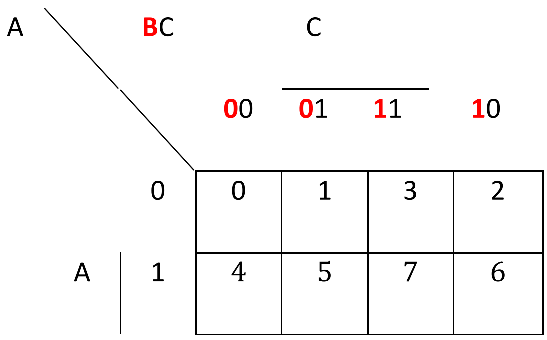
Rőmer Mária: Digitális technika példatár (KKMF 1105, Budapest 1999)

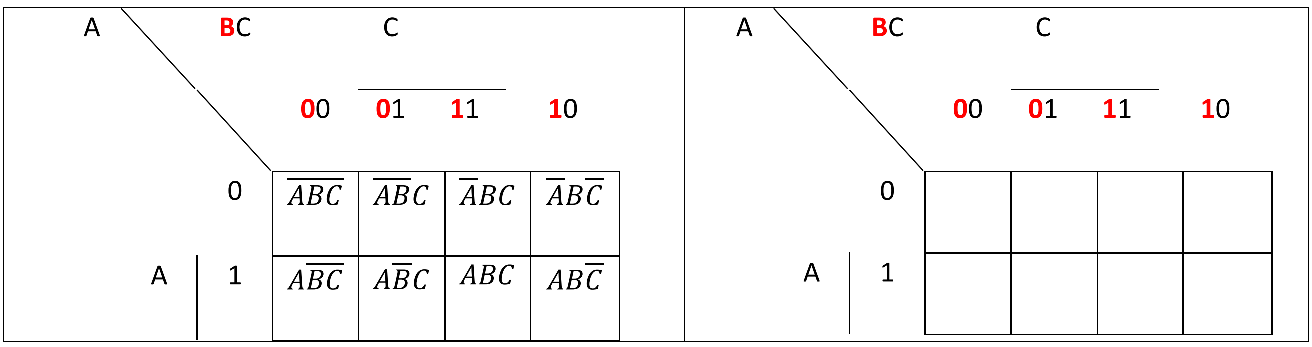
Matijevics István: Digitális Technika Interaktív példatár (ISBN 978-963-279-528-7 Szegedi Tudományegyetem)

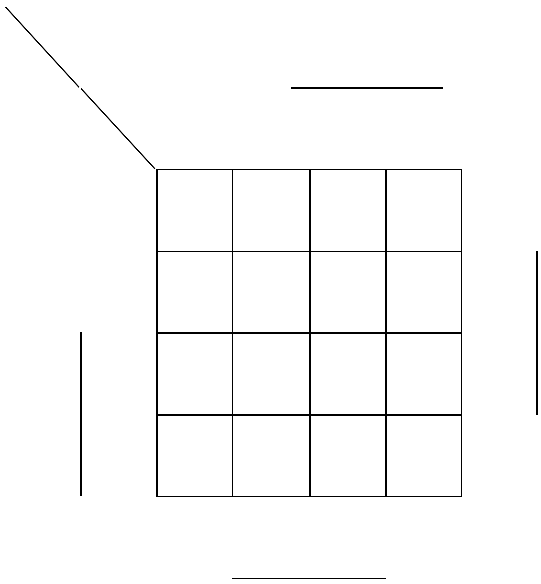
http://www.inf.u–szeged.hu/projectdirs/digipeldatar/digitalis_peldatar.html

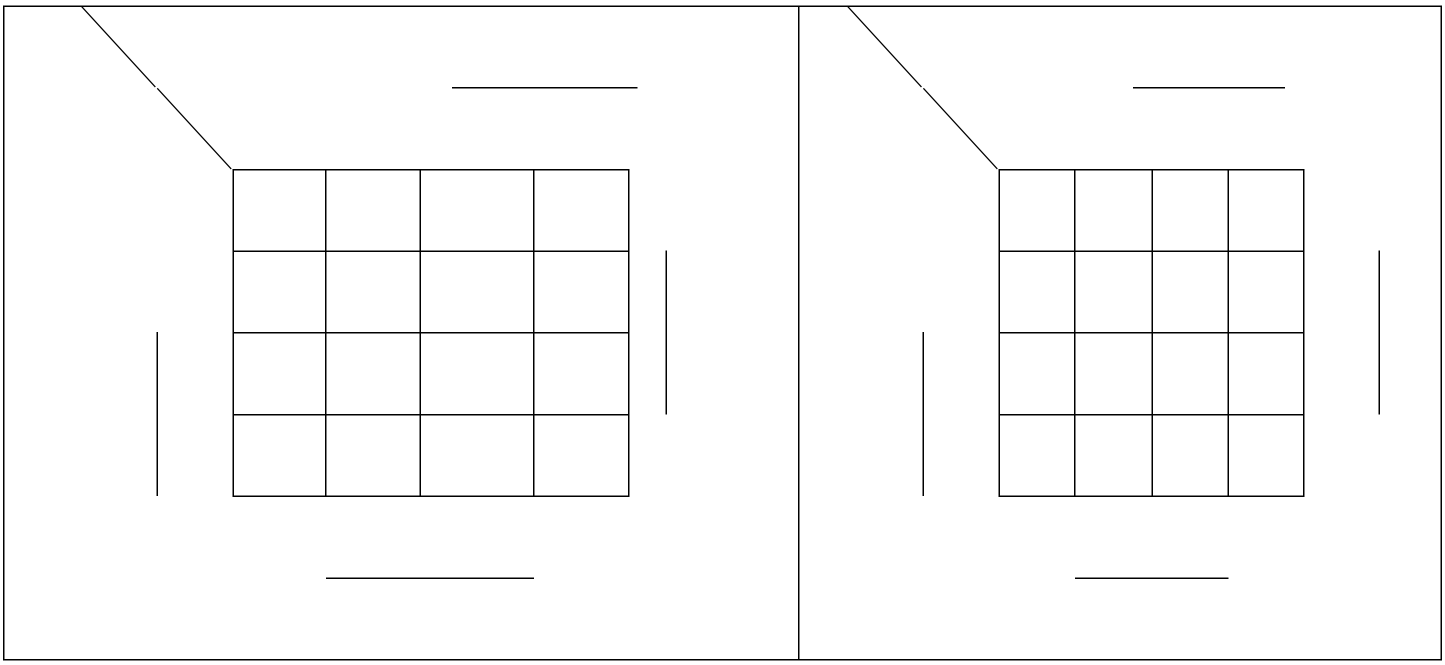
2.8. Feladatmegoldást segítő











AB \ CD	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

C (11, 10); B (01, 11); A (11, 10); D (01, 11)

AB \ CD	00	01	11	10
00	$\overline{ABCD}$	$\overline{ABC}D$	$\overline{AB}CD$	$\overline{AB}C\overline{D}$
01	$\overline{A}B\overline{CD}$	$\overline{A}B\overline{C}D$	$\overline{A}BCD$	$\overline{A}BC\overline{D}$
11	$AB\overline{CD}$	$AB\overline{C}D$	$ABCD$	$ABC\overline{D}$
10	$A\overline{BCD}$	$A\overline{BC}D$	$A\overline{B}CD$	$A\overline{B}C\overline{D}$

C (11, 10); B (01, 11); A (11, 10); D (01, 11)

AB \ CD	00	01	11	10
00				
01				
11				
10				

C (11, 10); B (01, 11); A (11, 10); D (01, 11)