Pannon Egyetem
Villamosmérnöki és Információs Tanszék

Digitális Áramkörök

(Villamosmérnök BSc / Mechatronikai mérnök MSc)

4. hét – Több kimenetű logikai függvények grafikus, és számjegyes minimalizálása

Előadó: Dr. Vörösházi Zsolt
voroshazi.zsolt@virt.uni-pannon.hu

Kapcsolódó jegyzet, segédanyag:

- http://www.virt.uni-pannon.hu → Oktatás → Tantárgyak → Digitális Áramkörök (Villamosmérnöki BSc / Mechatronikai mérnöki BSc/MSc).
  - Fóliák, óravázlatok (.ppt)
  - Frissítésük folyamatosan

Több kimenetű logikai függvények minimalizálása

Több-kimenetű logikai függvények minimalizálása

Vizsgált módszerek több kimenetre ($m \geq 2$):

- A.) Számjegyes: Quine-McCluskey
- B.) Grafikus: Karnaugh táblákkal

A.) Számjegyes minimalizálás (Quine-McCluskey)

- **Több kimenet** esetén az eljárást úgy kell kiegészíteni, hogy alkalmas legyen a *több kimenetű prímimplikánsok* előállítására.
- Ehhez minden mintermről, szorzatról vagy végső esetben prímimplikánsról tudni kell, hogy a kimeneti függvények közül melyikben fordult elő (*jelző karakter alkalmazása* szükséges).

Emlékeztető: Szomszédosság 3 feltétel teljesülése

- Bizonyítható, hogy az A.), B.) és C.) (**szükséges, de nem elégséges**) feltételek együttes teljesülése esetén lesz pontosan a két minterm **szomszédos**:
  - A.) indexek különbsége 2^n hatványa, és
  - B.) bináris súlyuk különbsége 1, és
  - C.) a nagyobb bináris súlyú minterm decimális indexe is nagyobb.

Több kimenetű függvények számjegyes minimalizálása

- Kiindulásként az összes megadott kimeneti függvény összes mintermjét - *az egyváltozós Quine-McCluskey módszernél megismert módon* - kell hogy csoportosítsuk I. oszlopban, azaz **bináris súlyok** szerint.
- **Jelzőkaraktert** kell rendelni minden mintermhez:
  - Hozzárendeli a mintermeket az adott kimeneti függvény(ek)hez
  - Bináris számjegy: ’0’ – „nem tartalmazza” / ’1’ – „tartalmazza”
  - Azokon a *helyértéken* ’1’ értékű, ahol a kimeneti függvény tartalmazza az adott mintermet.

1. Példa: Több kimenetű függvény Quine-McCluskey minimalizálása

- Adottak a következő kimeneti függvények (4 bemenet, 2 kimenet):

$$F^4{}_1(A,B,C,D)=\sum_{i=0}^{2^n-1}(0,1,3,5,10,11,14,15)$$

TSH!

$$F^4{}_2(A,B,C,D)=\sum_{i=0}^{2^n-1}(0,1,6,7,8,10,11,12,14,15)$$

1. Példa (folyt.) **I./a lépés**

- **I. oszlop:** Csoportosítás bináris súlyuk szerint:
  - ahol a kimeneti értékük ’1’-s volt.

$$F^4_1(A,B,C,D)=\sum_{i=0}^{2^n-1}(0,1,3,5,10,11,14,15)$$

$$F^4_2(A,B,C,D)=\sum_{i=0}^{2^n-1}(0,1,6,7,8,10,11,12,14,15)$$

Minterm(dec)	Bináris alak	
0	0000	[#0 bináris súly]
1	0001	[#1 bináris súly]
8	1000	
3	0011	[#2 bináris súly]
5	0101	
6	0110	
10	1010	
12	1100	
7	0111	[#3 bináris súly]
11	1011	
14	1110	
15	1111	[#4 bináris súly]

bináris súly szerinti csoportképzések: vonallal elválasztva

1. Példa (folyt.) **I./b lépés**

- **I. oszlop. Jelzőkarakterek ('0', '1') helyértékenként való megadása**
  - '1', ahol a kimeneti függvény tartalmazza az adott mintermet,
  - '0', egyébként.

Minterm(dec)	F_1	F_2	
0	1	1	√
1	1	1	√
8	0	1	√
3	1	0	√
5	1	0	√
6	0	1	√
10	1	1	√
12	0	1	√
7	0	1	√
11	1	1	√
14	1	1	√
15	1	1	√

jelzőkarakterek

$$F^4{}_1(A,B,C,D)=\sum_{i=0}^{2^n-1}(0,1,3,5,10,11,14,15)$$

$$F^4{}_2(A,B,C,D)=\sum_{i=0}^{2^n-1}(0,1,6,7,8,10,11,12,14,15)$$

1. Példa (folyt.) **II. lépés**

II. Oszlop: Összes létező szomszédos **kételemű** lefedő tömb (hurok) összevonása (I. oszlopból)

Minterm(dec. kül.)	F_1	F_2 (szorzat)
0,1 (1)	**1**	**1**
0,8 (8)	**0**	**1**
1,3 (2)	**1**	**0**
1,5 (4)	**1**	**0**
8,10 (2)	**0**	**1**
8,12 (4)	**0**	**1**
~~**3,7 (4)**~~	~~**0**~~	~~**0**~~
3,11 (8)	**1**	**0**
~~**5,7 (2)**~~	~~**0**~~	~~**0**~~
6,7 (1)	**0**	**1**
6,14 (8)	**0**	**1**
10,11 (1)	**1**	**1**
10,14 (4)	**1**	**1**
12,14 (2)	**0**	**1**
7,15 (8)	**0**	**1**
11,15 (4)	**1**	**1**
14,15 (1)	**1**	**1**

jelzőkarakterek

II. Oszlopot képezve: csak akkor vonható össze két szomszédos minterm az I. oszlop alapján, (vagy két szorzat a további oszlop(ok)ban), ha a jelzőkaraktereikben van legalább egy olyan helyérték, ahol mindkét jelzőkarakter ’1’.

Nem végezhetjük el azokat az összevonásokat sem, ahol az összes jelzőkarakter értéke ’0’.

1. Példa (folyt.) **III. lépés**

III. Oszlop: Összes létező szomszédos **négyelemű** lefedő tömb (hurok) összevonása (II. oszlopból)

Minterm(dec. kül.)	F_1	F_2	(szorzat)
0,1 (1)	1	1	a
0,8 (8)	0	1	b
1,3 (2)	1	0	c
1,5 (4)	1	0	d
8,10 (2)	0	1	√
8,12 (4)	0	1	√
~~3,7 (4)~~	~~0~~	~~0~~	
3,11 (8)	1	0	e
~~5,7 (2)~~	~~0~~	~~0~~	
6,7 (1)	0	1	√
6,14 (8)	0	1	√
10,11 (1)	1	1	√
10,14 (4)	1	1	√
12,14 (2)	0	1	√
7,15 (8)	0	1	√
11,15 (4)	1	1	√
14,15 (1)	1	1	√

prímimplikáns betűzések

Minterm(dec. kül.)	F_1	F_2	(szorzat)
8,10,12,14 (2,4)	0	1	f
6,7,14,15 (1,8)	0	1	g
10,11,14,15 (1,4)	1	1	h

III. oszlop

III → II. csak a jelzőkaraktereikben <u>teljesen</u> megegyező mintermeket (prímimplikáns tagokat) lehet kipipálni!

1. Példa: IV lépés – Prímimplikáns tábla

Kimeneti fgv / szorzat fgv.	Kimeneti fgvek.	F1								F2									
	mintermek / Többkim. Prímimplik.	0	1	3	5	10	11	14	15	0	1	6	7	8	10	11	12	14	15
F1 × F2	0,1 (1) a *	x	x							x	x								
	10,11,14,15 (1,4) h *					x	x	x	x						x	x		x	x
F1	1,3 (2) c		x	x															
	1,5 (4) d *		x		x														
	3,11 (8) e			x			x												
F2	0,8 (8) b									x				x					
	6,7,14,15 (1,8) g *											x	x					x	X
	8,10,12,14 (2,4) f *													x	x		x	x	

Táblázat kitöltésekor egy-egy többkimenetű prímimplikánssal kijelölt sornak abba a sorába kell '*'-ot tenni, amelyhez tartozó mintermeket az illető prímimplikáns tartalmazza → **lényeges prímimplikáns(ok)**
- van olyan minterm, amely oszlopa alatt csak egyetlen '**x**' szerepel

Segédfüggvény felírása (S)

- Ebben a feladatban ránézésre nem volt megállapítható, ezért kell a segédfüggvénnyel felírt alak:

Legegyszerűbb alak a prímimplikánsból „lefedi” a tagot

$$'1' = S = a \cdot (a+d+c) \cdot (c+e) \cdot d \cdot h \cdot h \cdot h \cdot h \cdot (a+b) \cdot a \cdot$$

$$\cdot g \cdot g \cdot (b+f) \cdot (h+f) \cdot h \cdot f \cdot (h+f+g) \cdot (h+g) =$$

Beszorzás elvégzése

$$= a \cdot d \cdot h \cdot f \cdot g \cdot (c+e) = adhfg\underline{c} + adhfg\underline{e}$$

Legegyszerűbb DNF alak

1.) **2.)**

Többkimenetű függvény esetén azonban a legegyszerűbb kétszintű (ÉS-VAGY) elvi logikai rajz felírásához, csak **PRÓBÁLGATÁSSAL** juthatunk el! Mindkét alakot meg kell vizsgálni.

1. 1.) Tekintsük az ‘adhfgc’ szorzatot

- **1)** Most az ‘***adhfgc’-***prímimplikáns készlettel kell az optimális lefedést magvalósítani (prímimplikáns tábla alapján leolvasva):

Kimenetenként felírt segédfüggvények:

$$s_1 = a \cdot (a + c + d) \cdot c \cdot d \cdot h \cdot h \cdot h \cdot h = a \cdot c \cdot d \cdot h \Rightarrow acdh$$

$$s_2 = (a + b) \cdot a \cdot g \cdot g \cdot f \cdot (f + h) \cdot h \cdot (b + f) \cdot (h + g + f) \cdot (h + g) =$$

$$= a \cdot g \cdot f \cdot h \Rightarrow afgh$$

$$\Downarrow$$

Kimeneti függvények:

$$F_1 = a + c + d + h = ?$$

$$F_2 = a + f + g + h = ?$$

1. 1.) Prímimplikánsok VAGY kapcsolatából képzett F1,F2 kétkimenetű függvények megadása

ABCD

- a **0,1 :** 0000, 0001 ⇒ $\overline{A}\,\overline{B}\,\overline{C}$
- c **1,3:** 0001, 0011 ⇒ $\overline{A}\,\overline{B}D$
- d **1,5:** 0001, 0101 ⇒ $\overline{A}\,\overline{C}D$
- h **10,11,14,15:** 1010, 1011, 1110, 1111 ⇒ AC
- f **8,10,12,14:** 1000, 1010, 1100, 1110 ⇒ $A\overline{D}$
- g **6,7,14,15:** 0110, 0111, 1110, 1111 ⇒ BC

MEGJ: A mintermen belüli, adott helyértéken lévő ‘0’-’1’ kombinációk kiesnek!

Próbálgatás után az F1,F2 többkimenetű függvények realizálnak itt egy lehetséges elvi kapcsolási rajzot! (Arató könyv: 2.46. ábra – 85. oldal)

Kapott kimeneti függvények:

$$F_1 = a + c + d + h = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}D + \overline{A}\,\overline{C}D + AC$$

$$F_2 = a + f + g + h = \overline{A}\,\overline{B}\,\overline{C} + A\overline{D} + BC + AC$$

1. 2.) Tekintsük az ‘adhfg<u>e</u>’ szorzatot

- **2.)** Most az ‘***adhfge’-***prímimplikáns készlettel kell az optimális lefedést magvalósítani (prímimplikáns tábla alapján leolvasva):

Kimenetenként felírt segédfüggvények:

$$s_1 = a \cdot (a+d) \cdot e \cdot d \cdot h \cdot (h+e) \cdot h \cdot h = a \cdot e \cdot d \cdot h \Rightarrow adeh$$

$$s_2 = (a+b) \cdot a \cdot g \cdot g \cdot f \cdot (f+h) \cdot h \cdot (b+f) \cdot (h+g+f) \cdot (h+g) =$$

$$= a \cdot g \cdot f \cdot h \Rightarrow afgh$$

$$\Downarrow$$

Kimeneti függvények:

$$F_1 = a+d+e+h = ?$$

$$F_2 = a+f+g+h = ? \longleftarrow$$ Ugyanaz, mint 1)-ben!!!

1. 2.) Prímimplikánsok VAGY kapcsolatából képzett F1,F2 kétkimenetű függvények megadása

	ABCD	
a 0,1 :	0000 0001	$\Rightarrow \overline{A}\,\overline{B}\,\overline{C}$
e 3,11:	0011 1011	$\Rightarrow \overline{B}CD$
d 1,5:	0001 0101	$\Rightarrow \overline{A}\,\overline{C}D$
h 10,11,14,15:	1010 1011 1110 1111	$\Rightarrow AC$
f 8,10,12,14:	1000 1010 1100 1110	$\Rightarrow A\overline{D}$
g 6,7,14,15:	0110 0111 1110 1111	$\Rightarrow BC$

MEGJ: A mintermen belüli, adott helyértéken lévő ‘0’-’1’ kombinációk kiesnek!

Próbálgatás után az F1,F2 többkimenetű függvények realizálnak itt egy lehetséges elvi kapcsolási rajzot!

Kapott kimeneti függvények:

$$F_1 = a + d + e + h = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,\overline{C}D + \overline{B}CD + AC$$

$$F_2 = a + f + g + h = \overline{A}\,\overline{B}\,\overline{C} + A\overline{D} + BC + AC$$

1. Példa: eredmény

- Összehasonlítva a prímimplikánsokkal kifejezett F1,F2 kimeneti függvényeket, az látható, hogy
  - F2 függvények realizálásában nincs eltérés a két módszer (1) és (2) között. Továbbá, mindként F1 függvény ugyananyi prímimplikáns taggal írható fel.
  - Viszont F1 függvényben az (2) módszer esetén a 3. prímimplikáns egyik változóját nem kell negálni!
- Következtetés: két módszer nem különbözik ÉS kapuk szintjén, viszont a VAGY kapuk szintjét tekintve **a (2) „módszer optimálisabb”:**
  - Egy vezetéket takarítunk meg (ÉS szint előtt nem invertálunk).

1.) módszer

$$F_1 = a + c + d + h = \bar{A}\bar{B}\bar{C} + \boxed{\bar{A}\bar{B}D} + \bar{A}\bar{C}D + AC$$

$$F_2 = a + f + g + h = \bar{A}\bar{B}\bar{C} + A\bar{D} + BC + AC$$

2.) módszer

$$F_1 = a + d + e + h = \bar{A}\bar{B}\bar{C} + \bar{A}\bar{C}D + \boxed{\bar{B}CD} + AC$$

$$F_2 = a + f + g + h = \bar{A}\bar{B}\bar{C} + A\bar{D} + BC + AC$$

Több-kimenetű függvényekre a Quine-McCluskey módszer alkalmazása: NTSH hálózatok esetén

- Ugyanúgy kell eljárni, mint az egyváltozós számjegyes minimalizálás NTSH esetén, azaz:
- Közömbös (x) dont'care mintermek megadásakor
  - az összevonásoknál I.-II.-III. stb. oszlopok felírásánál a **dont'care értékeket fix '1' nek** tekintjük, továbbá
  - a közömbös mintermeket nem kell figyelembe venni a prímimplikáns tábla felírásakor (hiszen azok lefedéséről nem kell gondoskodnunk)
  - Végül, legtöbb esetben az **S** segédfüggvény felírása ad jó megoldást eredményül

2. Példa: Több kimenetű függvény számjegyes minimalizálása NTSH hálózat esetén

- Adottak a következő kimeneti függvények (4 bemenet, 3 kimenet):

NTSH!

$$F^4{}_1(A,B,C,D) = \sum_{i=0}^{2^n-1} \left[(0,5,7,8,10) + (2,4,13,15)\right]$$

$$F^4{}_2(A,B,C,D) = \sum_{i=0}^{2^n-1} \left[(3,7,8,10,11) + (0,15)\right]$$

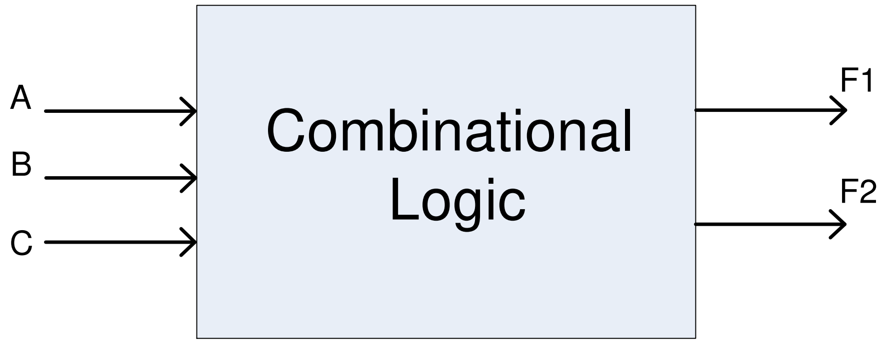
$$F^4{}_3(A,B,C,D) = \sum_{i=0}^{2^n-1} \left[(0,3,6,14,15) + (7,8)\right]$$

HF: valósítsa meg a legegyszerűbb kétszintű ÉS-VAGY elvi logikai kapcsolási rajzot!

B.) Grafikus minimalizálás

- Több kimenetű KH. egyszerűsítése: olyan, mintha kimenetenként egy-egy logikai függvénnyel írnánk le
- Külön-külön egyszerűsítve eljut(hat)unk a legegyszerűbb diszjunktív, vagy konjunktív logikai alakhoz,
  - azonban, mivel ugyanazon független (bemeneti) változókon értelmezettek → további **egyszerűsítésre** adódhat lehetőség

1. Példa:



- Legyen adott a következő 2-kimenetű, 3-bemenetű függvény:

$$F_1(A,B,C) = \overline{A}B\overline{C} + \overline{A}BC + ABC = m_2 + m_3 + m_7$$

$$F_2(A,B,C) = \overline{A}BC + A\overline{B}C + ABC = m_3 + m_5 + m_7$$

- Alkalmazzuk a grafikus minimalizáláshoz a Karnaugh táblát külön-külön:

F1:

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

$$F_1(A,B,C) = \overline{A}B + BC$$

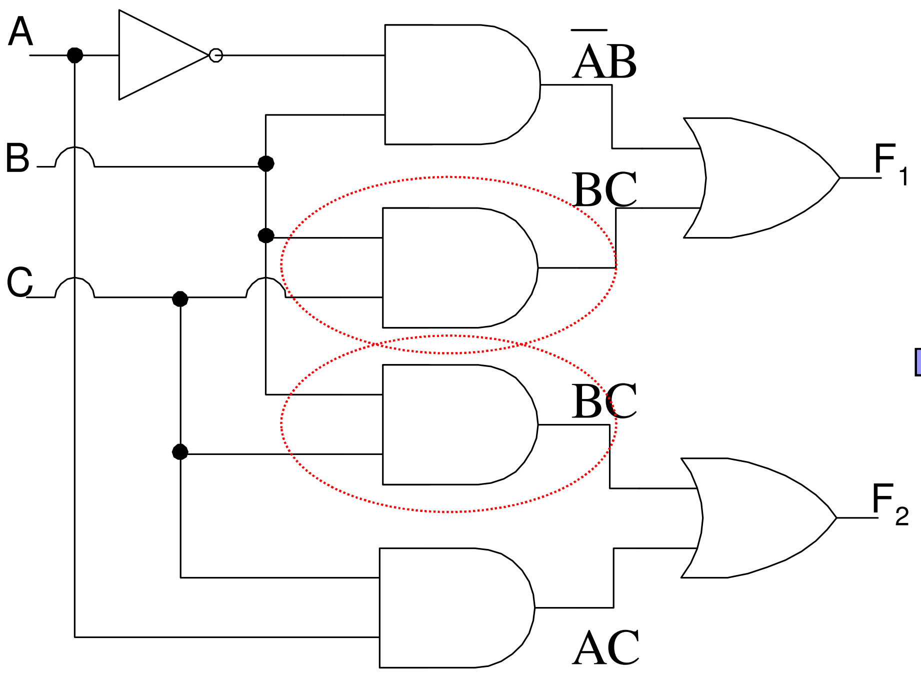
F2:

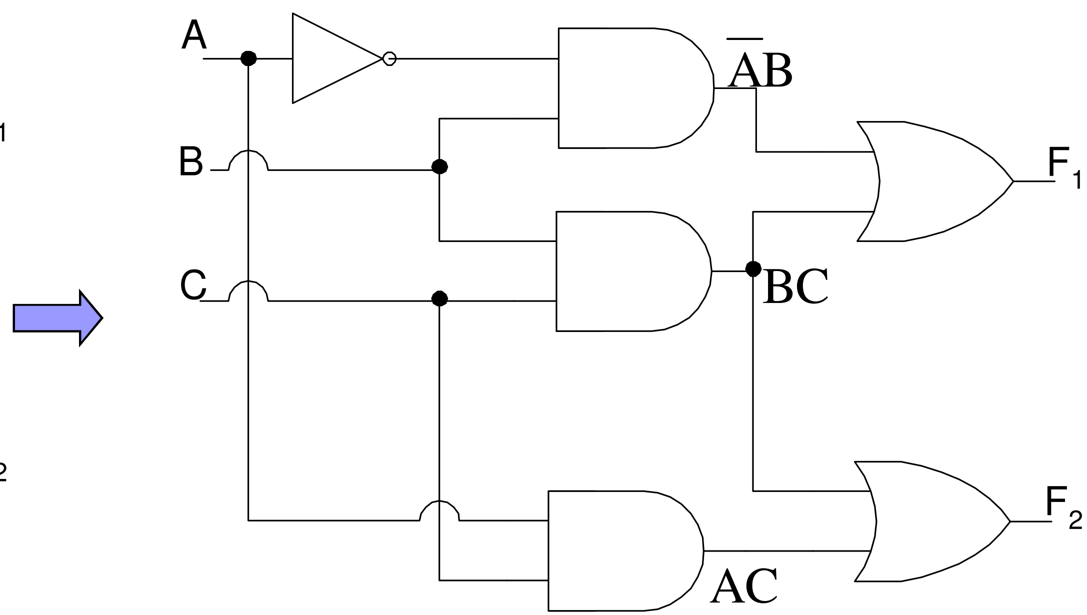
			C	C	
				B	B
A \ BC		00	01	11	10
	0	0_{0}	0_{1}	1_{3}	0_{2}
A	1	0_{4}	1_{5}	1_{7}	0_{6}

$$F_2(A,B,C) = AC + BC$$

‘BC’ közös prímimplikánsok

1. példa: Elvi logikai rajz





Kimeneti minimalizálással kapott elvi logikai rajz

$$F_1(A,B,C) = \overline{A}B + BC$$

$$F_2(A,B,C) = AC + BC$$

A közös ‘BC’ prímimplikáns egyszeri megvalósításával kapott elvi logikai rajz

1. Példa (folyt)

- Mivel a 'BC' közös prímimplikáns mindkét (F1, F2) logikai függvényben szerepel, felesleges mindkét elvi logikai kapcsolásban kétszer megvalósítani.
- Ezért a minimalizálási eljárást módosítani kell!
  - Cél: közös prímimplikánsok meghatározása
  - Előző példa alapján képzett szorzatfüggvény

$F_1 \cdot F_2$

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

$$F_1 \cdot F_2 = BC$$

DEF: Közös/nem-közös prímimplikánsok meghatározása

- Alapgondolat: a két (több) függvény **közös prímimplikánsai** (amelyek közösen lefedik a mintermeket) biztosan *prímimplikánsai* lesznek a két (több) függvény **szorzatának**.
  - Két függvénynél: $F_1 \times F_2=1$, ha $F_1=1$ és $F_2=1$
- Közös Karnaugh táblában: azokon a helyeken lesz '1', ahol a mindkét (mindegyik) Karnaugh táblában is '1' volt.
- A szorzatfüggvényben természetesen lehetnek **nem-közös prímimplikánsok** is (lásd köv: **Példa 2**), melyeket nem lehet figyelmen kívül hagyni a szorzatfüggvény optimális lefedéséhez.

2. Példa: Adottak a következő Karnaugh táblák

F_1

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

e

F_2

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

f

$F_1 \cdot F_2$

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

a

b

c

d

összes

prímimplikáns

2. Példa (folyt.) alapján:

Megállapítható, hogy:

- ***'a'*** prímimplikáns közös (F_1, és F_2-ben is pontosan így szerepel)
- ***'b'*** prímimplikáns nem közös (mivel F_1-ben nem prímimplikánsként, hanem az **'e'** jelű prímimplikánsnak csak egy **részeként**, azaz tovább egyszerűsíthető szorzatként szerepel!)
- ***'c'*** prímimplikáns nem közös (mivel F_1-ben nem prímimplikánsként, hanem az ***'e'*** jelű prímimplikánsnak csak egy **részeként**, F_2-ben pedig az ***'f'*** prímimplikáns tartalmazásaként szerepel, azaz tovább egyszerűsíthető szorzatként szerepel!)
- ***'d'*** prímimplikáns nem közös (mivel F_2-ben nem prímimplikánsként, hanem az ***'f'*** jelű prímimplikánsnak csak egy **részeként** szerepel, azaz tovább egyszerűsíthető szorzatként szerepel!)

Több kimenetű függvény optimális lefedésének meghatározása

Figyelembe kell venni tehát:

- **Kimeneti függvények** prímimplikánsait, melyek lehetnek:
  - Közösek
  - Nem közösek
- Összes lehetséges **szorzatfüggvény** prímimplikánsait is.

3. Példa

- <u>Adottak</u>: n=4 bemenetű, 3 kimenetű K.H. Realizáljuk a legegyszerűbb kétszintű ÉS-VAGY (DNF) alakú elvi logikai rajzot! Egyszerűsítésként a *grafikus minimalizálást (Karnaugh táblákat)* alkalmazzuk!

$$F^4{}_1(A,B,C,D)=\sum_{i=0}^{2^n-1}(0,5,7,8,13,15)$$

$$F^4{}_2(A,B,C,D)=\sum_{i=0}^{2^n-1}(0,5,8,10,14)$$

$$F^4{}_3(A,B,C,D)=\sum_{i=0}^{2^n-1}(0,1,3,5,8,14)$$

3. Példa (folyt.) – I. lépés

F_1

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

F_2

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

F_3

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

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

$$F_1 = BD + \overline{B}\,\overline{C}\,\overline{D}$$

$$F_2 = AC\overline{D} + \overline{B}\,\overline{C}\,\overline{D} + \overline{A}B\overline{C}D + A\overline{B}\,\overline{D}$$

$$F_3 = \overline{B}\,\overline{C}\,\overline{D} + \overline{A}\,\overline{C}D + \overline{A}\,\overline{B}D + \overline{A}\,\overline{B}\,\overline{C} + ABC\overline{D}$$

Szorzat függvények előállítása összes lehetséges módon!

3. Példa (folyt.) – II. lépés

Összes lehetséges szorzatfüggvény Karnaugh táblái (összes lehetséges prímimplikáns képzése)

$F_1 \cdot F_2$

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

D: columns 01, 11

$F_1 \cdot F_3$

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

D: columns 01, 11

$F_2 \cdot F_3$

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

D: columns 01, 11

$$F_1 \cdot F_2 = \overline{B}\,\overline{C}\,\overline{D} + \overline{A}B\overline{C}D$$

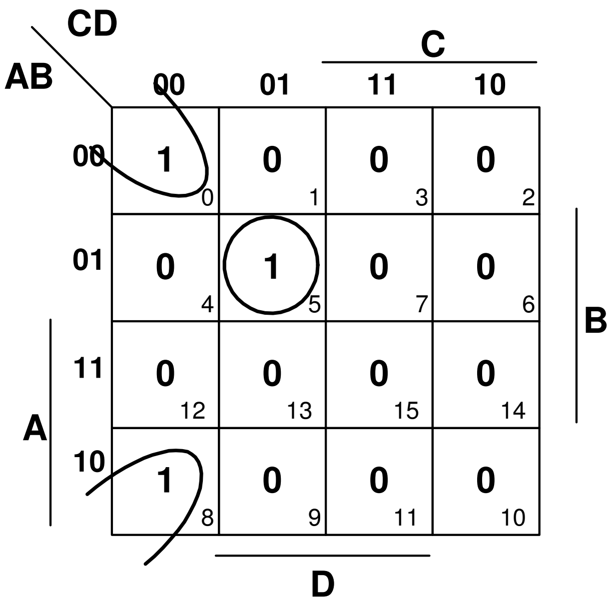
$$F_1 \cdot F_3 = \overline{B}\,\overline{C}\,\overline{D} + \overline{A}B\overline{C}D$$

$$F_2 \cdot F_3 = \overline{B}\,\overline{C}\,\overline{D} + \overline{A}B\overline{C}D + ABC\overline{D}$$

Végül $F_1 \cdot F_2 \cdot F_3$ szorzat előállítása!

3. Példa (folyt.) – III. lépés

$F_1 \cdot F_2 \cdot F_3$



$$F_1 \cdot F_2 \cdot F_3 = \overline{B}\,\overline{C}\,\overline{D} + \overline{A}B\overline{C}D$$

Szorzatfüggvény prímimplikánsai

Optimális lefedéshez a szorzatfüggvények prímimplikánsait is figyelembe kell venni, de nehéz lehet áttekinteni a Karnaugh tábláikat.

→ Prímimplikáns táblát használunk leolvasásukhoz!

3. Példa: IV lépés – Prímimplikáns tábla

Kimeneti fgv / szorzat fgv.	Kimeneti fgvek.	F1						F2					F3					
											!							
	mintermek / Többkim. Prímimplik.	$\bar A\bar B\bar C\bar D$	$\bar AB\bar CD$	$\bar ABCD$	$A\bar B\bar C\bar D$	$AB\bar CD$	$ABCD$	$\bar A\bar B\bar C\bar D$	$\bar AB\bar CD$	$A\bar B\bar C\bar D$	$A\bar BC\bar D$	$ABC\bar D$	$\bar A\bar B\bar C\bar D$	$\bar A\bar B\bar CD$	$\bar A\bar BCD$	$\bar AB\bar CD$	$A\bar B\bar C\bar D$	$ABC\bar D$
F1	BD * a		x	x		x	x											
F2	$AC\bar D$ b										x	x						
	$A\bar B\bar D$ c									x	x							
F3	$\bar A\bar B\bar C$ d												x	x				
	$\bar A\bar CD$ e													x		x		
	$\bar A\bar BD$ * f													x	x			
F2×F3	$ABC\bar D$ * g											x						x
F1×F2×F3	$\bar B\bar C\bar D$ * h	x			x			x		x			x				x	
	$\bar AB\bar CD$ * i		x						x							x		

Táblázat kitöltésekor egy-egy többkimenetű prímimplikánssal kijelölt sornak abba a sorába kell '*'-ot tenni, amelyhez tartozó mintermeket az illető prímimplikáns tartalmazza → **lényeges prímimplikáns(ok)**
- van olyan minterm, amely oszlopa alatt csak egyetlen '**x**' szerepel, majd x kiemelhető

3. Példa (folyt.) – V. lépés

- Feladat: a többkimenetű prímimplikánsok készletéből azokat kiválasztani, amelyek a legegyszerűbb elvi logikai rajzot eredményezi.
  - Ehhez **'S'** segédfüggvény felírása.

$$S = h\cdot(a+i)\cdot a\cdot h\cdot a\cdot a\cdot h\cdot i\cdot(c+h)\cdot(b+c)\cdot(b+g)\cdot(d+h)\cdot(d+e+f)\cdot f\cdot(e+i)\cdot h\cdot g =$$

$$= \underbrace{afghi}\cdot(a+i)\cdot(c+h)\cdot\underline{(b+c)\cdot}(b+g)\cdot(d+h)\cdot(d+e+f)\cdot(e+i) = \underset{1)}{\underline{afghic}} + \underset{2)}{\underline{afghib}}$$

Kiemelhető, mivel minden keletkező segédfüggvénybeli szorzatban szerepel

Többkimenetű függvény esetén azonban a legegyszerűbb kétszintű (ÉS-VAGY) elvi logikai rajz felírásához, csak **<u>PRÓBÁLGATÁSSAL</u>** juthatunk el!

Mindkét alakot meg kell vizsgálni!

PRÓBÁLGATÁS

- A próbálgatás során csak a *legkevesebb tényezőt tartalmazó szorzatokat* vesszük figyelembe.
- A kiválasztott szorzatban az összes több kimenetű prímimplikáns közül azok szerepelnek, amelyeket az elvi logikai rajzban ÉS kapukkal valósítunk meg.
- Ezeknek az ÉS kapuknak a kimeneteit úgy kell VAGY kapuk bemeneteire vezetni, hogy a VAGY szinten a kapubemenetek száma az adott kiválasztással *minimális* legyen.

3. 1) Tekintsük az ‘afghic’ szorzatot

- **1)** Most az ‘***afghic’-***prímimplikáns készlettel kell az optimális lefedést magvalósítani (prímimplikáns tábla alapján leolvasva):

Kimenetenként felírt segédfüggvények:

$$s_1 = h \cdot (a+i) \cdot a \cdot h \cdot a \cdot a = h \cdot (a+i) \cdot a = ha$$

$$s_2 = h \cdot i \cdot (c+h) \cdot c \cdot g = hicg$$

$$s_3 = h \cdot f \cdot f \cdot i \cdot h \cdot g = hfig$$

$$\Downarrow$$

Kimeneti függvények:

$$F_1 = h + a = \overline{B}\,\overline{C}\,\overline{D} + BD$$

+ tag

$$F_2 = h + i + c + g = \overline{B}\,\overline{C}\,\overline{D} + \overline{A}B\overline{C}D + A\overline{B}\,\overline{D} + ABC\overline{D}$$

$$F_3 = h + f + i + g = \overline{B}\,\overline{C}\,\overline{D} + \overline{A}\,\overline{B}D + \overline{A}B\overline{C}D + ABC\overline{D}$$

Próbálgatás után az F1,F2,F3 többkimenetű függvények realizálnak itt egy lehetséges elvi kapcsolási rajzot! (Arató könyv: 2.43. ábra – 78. oldal)

3. 2) Tekintsük az ‘afghib’ szorzatot

- **2)** Most pedig az ‘***afghib’-***prímimplikáns készlettel kell az optimális lefedést magvalósítani (prímimplikáns tábla alapján leolvasva):

Kimenetenként felírt segédfüggvények:

$$s_1 = h \cdot (a+i) \cdot a \cdot h \cdot a \cdot a = h \cdot (a+i) \cdot a = ha$$

$$s_2 = h \cdot i \cdot h \cdot b \cdot (b+g) = hib$$

$$s_3 = h \cdot f \cdot f \cdot i \cdot h \cdot g = hfig$$

Kimeneti függvények:

$$F_1 = h + a = \overline{B}\,\overline{C}\,\overline{D} + BD$$

$$F_2 = h + i + b = \overline{B}\,\overline{C}\,\overline{D} + \overline{A}B\overline{C}D + AC\overline{D}$$

$$F_3 = h + f + i + g = \overline{B}\,\overline{C}\,\overline{D} + \overline{A}\,\overline{B}D + \overline{A}B\overline{C}D + ABC\overline{D}$$

Próbálgatás után az F1,F2,F3 többkimenetű függvények realizálnak itt egy lehetséges elvi kapcsolási rajzot! (Arató könyv: 2.44. ábra – 79. oldal)

3. Példa: eredmény

- Összehasonlítva a prímimplikánsokkal kifejezett F1,F2,F3 kimeneti függvényeket, az látható, hogy
  - F1, ill. F3 függvények realizálásában nincs eltérés a két módszer (1) és (2) között.
  - Viszont F2 függvényben az (2) módszer esetén egy taggal csökken a prímimplikánsok száma
- Következtetés: két módszer nem különbözik ÉS kapuk szintjén, viszont a VAGY kapuk szintjét tekintve **a (2) módszer optimálisabb:**
  - Egy kapubemenetet takarítunk meg (VAGY szinten).