Megoldás

Digitális technika I. (vimia102)
2. gyakorlat: Boole algebra, logikai függvények, kombinációs hálózatok alapjai

Elméleti anyag:

- Az általános digitális gép: memória + kombinációs hálózat
- A Boole algebra axiómái: a halmaz és a műveletek; a műveletek tulajdonságai, a dualitás elve
- A Boole algebra fontos tulajdonságai: idempotencia, elnyelés, asszociativitás, De Morgan azonosság
- Kétértékű logikai függvények: számuk, a kétváltozós kétértékű függvények és a hozzájuk tartozó kapcsolási szimbólumok Egy bites bináris teljes összeadó tervezése
- Kombinációs hálózatok leírási formái: algebrai alak, igazságtábla, algebrai normál alakok (diszjunktív és konjunktív, minterm, maxterm, rövid alak), Karnaugh tábla
-

Irodalom:

Benesóczky Zoltán: Boole algebra, logikai függvények (2004), elektronikus jegyzet
http://home.mit.bme.hu/%7Ebenes/oktatas/dig-jegyz_052/boole_alg-logf.pdf
Arató Péter: Logikai rendszerek tervezése (jegyzet), 1., 2.1., 2.2. fejezetek

Gyakorló példák:

2.1. Igazolja az alábbi azonosságokat a Boole algebra axiómáihoz és a belőlük levezetett „ismert” tulajdonságaihoz visszanyúlva!

a/ A + /A.B = A + B
b/ A.B + /A.C + B.C = A.B + /A.C
c/ A.B + /A.C = (A + C).(/A + B)

Megoldás:
a1/ A + /A.B = elnyelés =
A + A.B + /A.B = komm és disztr =
A + B.(A + /A) = A + B. 1 = A + B
a2/ A + /A.B = disztr = (A+/A).(A+B) = 1.(A+B) = A+B
b/ B.C = B.C.1 = BC.(A + /A) = A.B.C + /A.B.C, mindkét tagot elnyeli a fenti további két tag
c/ (A + C).(/A + B) = A.B + /A.C + B.C = az előzőhöz hasonlóan B.C-t elnyeli az előző két tag.

Minden ilyen példát sokkal egyszerűbb pl. K táblán ellenőrizni.

2.2. Milyen helyettesítési értékek mellett áll fenn az alábbi egyenlőség (négy példa!)?

a/ A.B.C = A+B+C   b/ A.B.C = /A+/B+/C
c/ /(A.B.C) = A+B+C   d/ (A.B) mod2 C = B

Megoldás:
a/ ABC = 000 vagy 111;
b/ soha;
c/ A két A=B=C esetet kivéve a többi hat;
d/ ABC = 000, 100, 011, 110

Általánosan követhető megoldás, ha felírjuk a két oldal igazságtábláját (vagy K tábláját vagy normál alakját) és mintermenként egyeztetjük.
Persze algebrailag is lehet ügyeskedni, de ez általában sok ötletet igényel.
PL. b. esetén a DeMorgan szabály alapján a baloldal negáltja a jobboldalnak, tehát soha sem egyezik.

2.3. Hozza egyszerűbb alakra az alábbi kifejezéseket, felhasználva a Boole algebra közismert tulajdonságait!

Y1 = /A + A.B./C + (/A + A.B./C).(A + /A./B.C);
Y2 = /((A.B + C.D).(A./C + B./D));
Y3 = B.(A.C + /A./C) + A./C + /A.C;
Y4 = A.C.D + A.B.C + D.(/A + /B) + /A.C.D

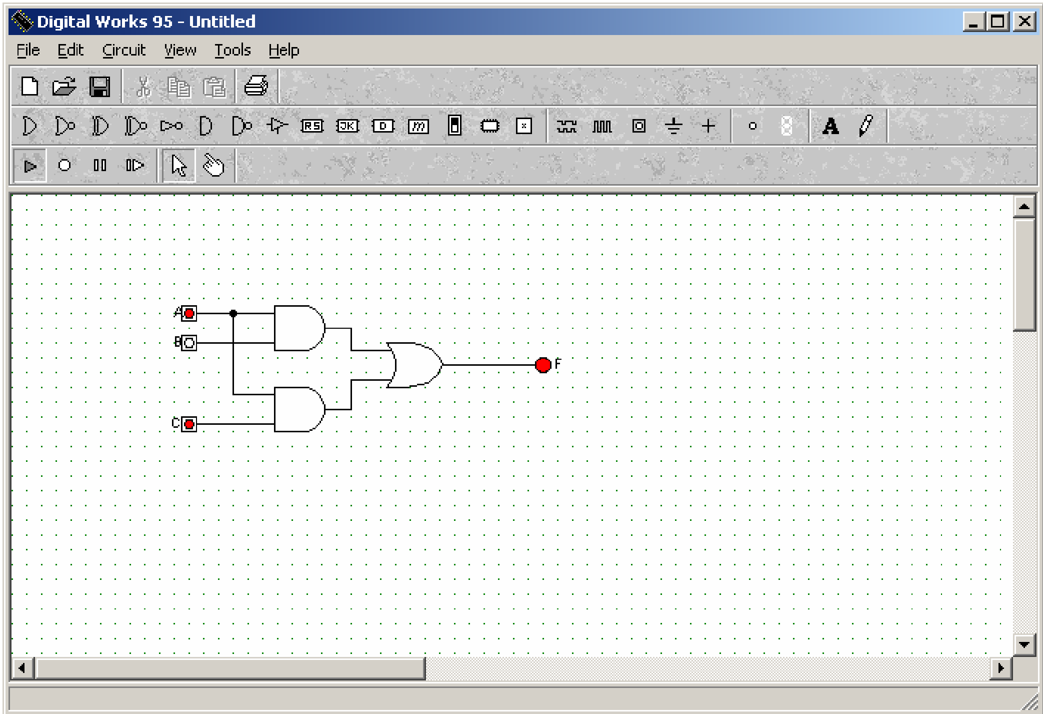
Megoldás:
Y1 = /A + A.B./C, mert ehhez még hozzáadjuk ennek valahányszorosát (elnyelési tétel: a = a.b). Ez a kifejezés tovább egyszerűsíthető a 2.1. a/ példa eredménye alapján:
Y1 =
/A + B./C
Y2 = /(A.B./C + A.B./D + 0 + 0) = DeMorgan = /(A.B./C) . /(A.B./D) = DeMorgan
= (/A + /B + C).(/A + /B + D) = beszorozva és sok elnyelési tétel =
/A + /B + C.D
Y3 = Bővítünk a második tag B-szeresével (elnyelési tétel) =
= B.(A.C + /A./C) + A./C + /A.C + B.(A./C + /A.C) = B-t kiemeljük és B.1-et kapunk = B + /A.C + A. /C (mindezt K táblán nagyon egyszerű)
Y4 = egy kiemelés és egy szorzás elvégzése = C.D + A.B.C + /A.D + /B.D =
= CD-t hosszabban felírva = C.D.(/A + /B + A.B), elvégezve ezt a szorzást, azt kajuk, hohy mindhárom tagot elnyeli a fenti kifejezés utolsó három tagja, CD tehát nem kell! = A.B.C + /A.D + /B.D

2.4. Tervezzen „tiltott NBCD kódot” felismerő hálózatot!
A hálózat feladata, hogy a kimenetén jelezze, ha a bemenetére érkező ABCD négybites kód >= 1010-nál.

Megoldás:
Az 1010…1111 kódokat felismerő hálózat Karnaugh táblája:

AB \ CD	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	1	1	1	1
10	0	0	1	1

Ennek a függvénynek „ránézésre jó megvalósítása: F = A.B + A.C = A.(B+C)



2.5. Tervezzen kétbemenetű programozható kaput!
A hálózatnak két adatbemenete (a, b) és két funkcióbemenete (f, g) van. A kapu a funkciókódtól függően a következő módon viselkedjen:

fg = 00: a AND b
fg = 01: a NAND b
fg = 10: a
fg = 11: b

Törekedjen arra, hogy a felhasznált "szumma kapubemenetszám" lehetőleg kevés legyen! Mennyivel sikerült megoldania?

Megoldás:
A Karnaugh a tábla:

fg \ ab	00	01	11	10
00			1	
01	1	1		1
11		1	1	
10			1	1

Egy lehetséges kapcsolás algebrai egyenletként:

Y = /f..g./a+f.g.b+f./g.a+bg.a.b+/f.g./b

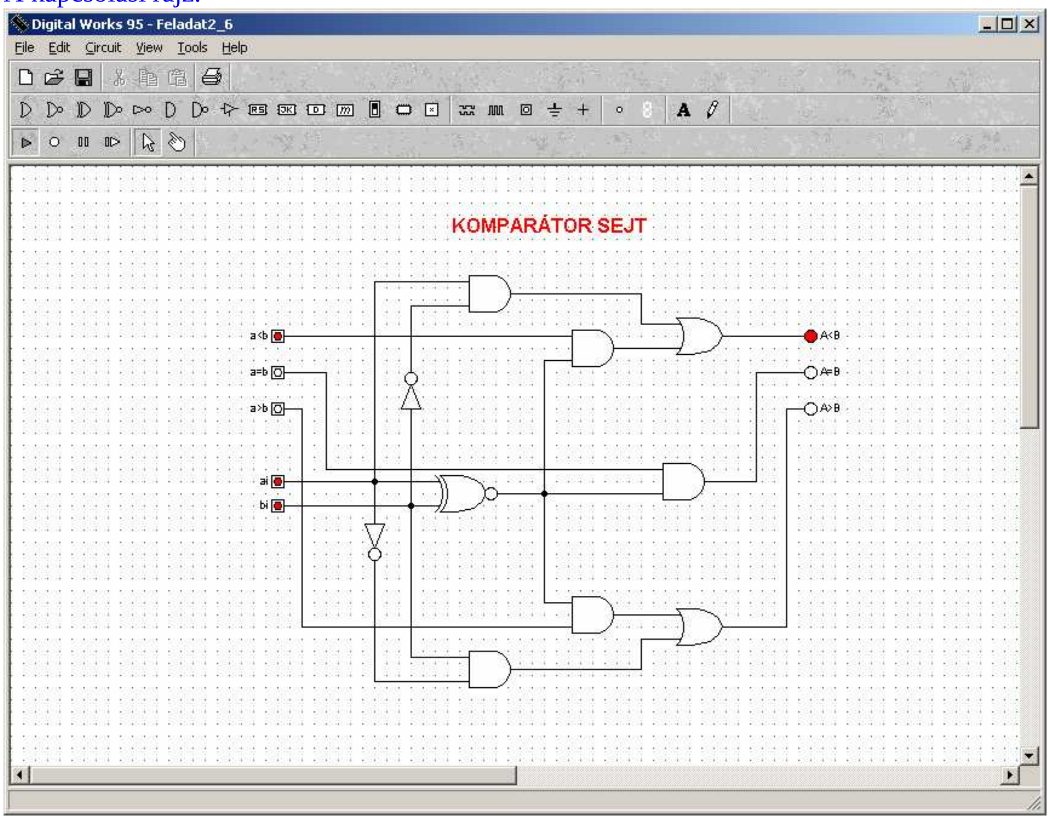
2.6. Tervezzen komparátor sejtet!
Az áramkörnek két adatbemenete (ai, bi), három „kaszkádosító" bemenete (a<b, a=b, a>b) és három „kaszkádosító" kimenete (A<B, A=B, A>B) van.

Megoldás:
Kaszkádosítsunk a kisebb helyértékek felől a nagyobbak irányába!
„Logikai” alapon a függvények egy lehetséges felírása:

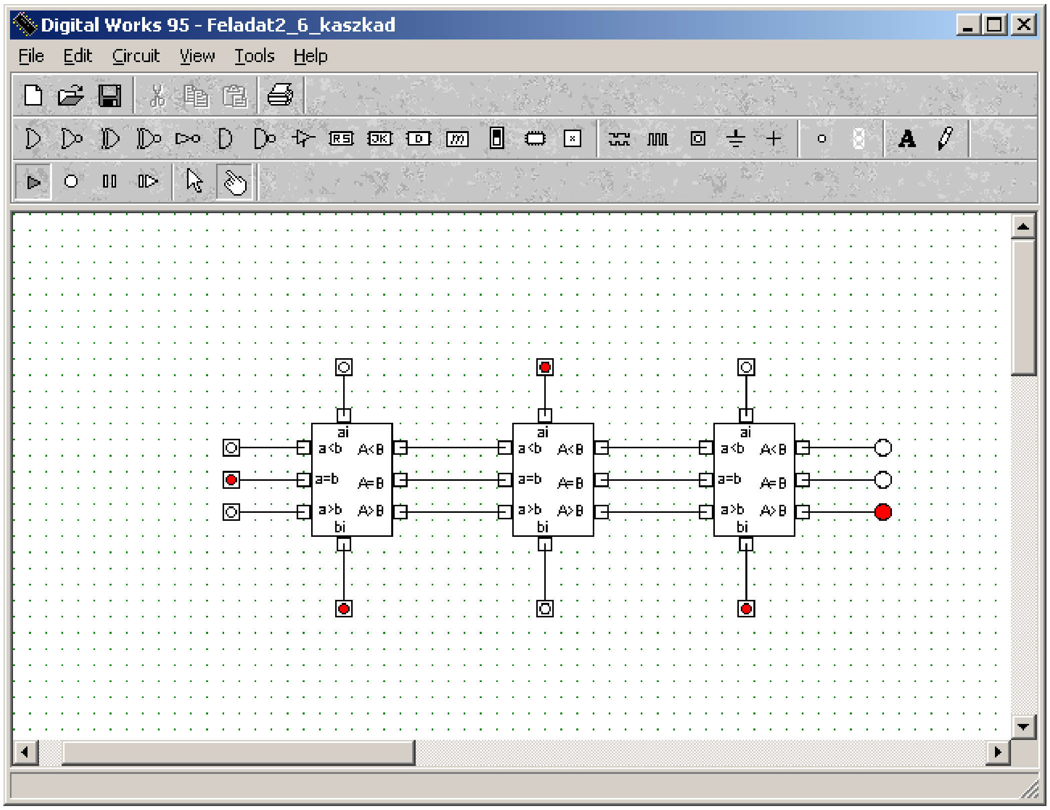
1. (A<B) = (ai ekv bi). (a<b) + /ai.bi
2. (A=B) = (ai ekv bi).(a = b)
3. (A>B) = (ai ekv bi).(a>b) + ai./bi

ad.1. A<B, ha egyforma biteknél az alacsonyabb helyértékeknél a<b VAGY ai=0 és bi=1
ad.2. AQ=B, ha egyformák a bitek és az alacsonyabb helyértékeken is egyenlőség van
ad.3. mint 1.

A kapcsolási rajz:



A komparátor sejtből makrót csinálva (Feladat2_6macro), eggyel magasabb szinten több bites komparátort is készíthetünk, az alábbi ábra három biteset mutat.



Nehéz példák az érdeklődőknek:

2.n1. Egy gyilkosság nyomozása során a következő információk gyűltek össze:
A gyilkosról kizárható, hogy

1. kék szemű ÉS sportos;
2. fekete hajú ÉS alacsony ÉS NEM kékszemű;
3. NEM fekete hajú ÉS NEM alacsony ÉS sportos;
4. NEM fekete hajú ÉS sportos és NEM kékszemű ÉS alacsony;
5. NEM fekete hajú ÉS NEM sportos ÉS NEM visel tornacipőt;
6. tornacipőt visel ÉS NEM sportos;
7. fekete hajú ÉS NEM alacsony.

Végül is a rendőrség előzetes letartóztatásba helyezett három gyanúsítottat, akik közül

- az egyik feketehajú, kékszemű, alacsony és kövér állásnélküli nyomdász,
- a másik feketehajú, kékszemű, magas, sportos tenisztréner,
- a harmadik feketehajú, szürkeszemű, alacsony beteges kinézetű varrónő volt.

Mi a véleménye, szükség volt-e a három letartóztatásra?

2.n2. A szorgalmas hallgató pénzfeldobással „véletlen” logikai függvényeket állít elő:

a. 50-50%-os valószínűséggel az ***A*** vagy a ***B*** logikai változót írja le
b. az egész eddigi függvényt zárójelbe teszi
c. véletlengenerál (50-50%!) logikai összeadás (**+**) vagy szorzás (**.**) műveletet,
d. véletlengenerál ***A*** vagy ***B*** logikai változót
e. ugrás a b. pontra jó sokáig

Mit tud mondani az előállított logikai függvényről?
Pl. egy lehetséges függvény:

F = ((((((***A***)+***B***).***A***).***B***)+***B***).**A**) ….