Megoldás

Digitális technika I. (vimia102)
3. gyakorlat: Kombinációs hálózatok minimalizálása, hazárdok, a realizálás kérdései

Elméleti anyag:

- Lényegtelen kombináció (don't care) fogalma
- Kombinációs hálózatok minimalizálása Karnaugh táblán (prímimplikánsok, lényeges prímimplikánsok, lefedési tábla)
- Többkimenetű függvények minimalizálása
- Mit tudunk minimalizálni és mit nem?
- Többszintű kombinációs hálózatok
- Homogén NAND és NOR hálózatok
- Hazárdjelenségek kombinációs hálózatokban: statikus, dinamikus, funkcionális hazárdok
- Hazárd keresés és megszüntetés

Irodalom:

Benesóczky Zoltán: Kombinációs hálózatok egyszerűsítése (elektronikus jegyzet, 2004.) http://home.mit.bme.hu/%7Ebenes/oktatas/dig-jegyz_052/kombh.
Benesóczky Zoltán: Hazárdjelenségek kombinációs hálózatokban (elektronikus jegyzet, 2004.)
http://home.mit.bme.hu/%7Ebenes/oktatas/dig-jegyz_052/hazardok.pdf
Arató Péter: Logikai rendszerek tervezése (jegyzet), 2.3., 2.4. 2.5.

Gyakorló példák:

3.1. Állapítsa meg, hogy az alábbi logikai függvények diszjunktív vagy konjunktív kétszintű megvalósítása tartalmaz kevesebb kapubemenetet!
Rajzolja fel a megoldásokat! (X: a don't care-eket jelöli, a legmagasabb helyiérték az A)

a/ F(A,B,C,D) = SZUMMA (0,1,4,8,11,12,13,14), X:(2,7)
b/ F(A,B,C,D) = PI (0,1,5,7,9) X:(4,12)
c/ F(A,B,C,D) = PI (1,3,5,7,9,13) X:(2,4,11,15)

Megoldás:
a/ A megvalósítandó függvény Karnaugh táblája:

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

A diszjunktív minimális lefedéshez 1 kétváltozós, 3 háromváltozós és 1 négyváltozós hurok kell, ez a kimeneti VAGY kapuval együtt összesen 20 bemeneti láb:

/C./D + /A./B./C + A.B./C + A.B./D + A./B.C.D

CD

AB	00	01	11	10
00	1	1		x
01	1		x	
11	1	1		1
10	1		1	

A konjunktív lefedéshez szintén 1 kétváltozós, 3 háromváltozós és 1 négyváltozós hurok kell, ez a kimeneti ÉS kapuval együtt összesen ismét 20 láb:
(A+/C) . (A+/B+/C) . (/B+/C+/D) . B+/C+D) . (/A+B+C+/D)

b. A megvalósítandó Karnaugh tábla:

CD

AB	00	01	11	10
00			x	
01				0
11			0	0
10	0		x	0

A nullák minimális lefedése (konjunktív lefedés): (/A+/C).(/A+B+D)(/B+/C+D), összesen 11 bemenet, míg az egyesek minimális lefedése (diszjunktiv lefedés): /A./B + B./C +/A.D + /B.D, összesen 12 bemenet.

c/ A megvalósítandó K-tábla:

CD

AB	00	01	11	10
00	x			0
01	x			0
11	0	x		0
10	0		x	0

Ezt a függvényt nézhetem konjunktív szemmel is meg diszjunktívval is,
az eredmény: D.

3.2. Adott az alábbi Karnaugh-tábla!
Adja meg a kétszintű minimális (amikor a hazárd megengedett - l. később) diszjunktív lefedést algebrai alak és kapcsolási rajz formájában.

CDE

AB	000	001	011	010	110	111	101	100
00	0	0	0	0	0	1	1	0
01	1	1	0	0	0	0	1	1
11	1	1	1	0	0	0	0	1
10	0	1	1	0	0	1	1	0

Megoldás:
Az összes primimplikáns:

1. A./B.E
2. A./C.E lényeges
3. B./C./D
4. B./D./E lényeges
5. /A.B./D kell még
6. /A.C./D.E
7. /B.C.E lényeges

A lényeges primimplikánsok nem fedik le a pirossal jelölt mintermeket, ezeket legegyszerűbben a bejelölt /A.B./D fedi le. :

CDE

AB	000	001	011	010	110	111	101	100
00	0	0	0	0	0	1	1	0
01	1	1	0	0	0	0	1	1
11	1	1	1	0	0	0	0	1
10	0	1	1	0	0	1	1	0

3.3. Rajzolja fel a következő függvény Karnaugh-tábláját:

(a.b) mod2 (a.c) mod2 (b.c)

Megoldás:

Ez a függvény 0-t ad, ha legalább két változója 0, 1-et ad, ha legalább két változója 1, vagyis ez a függvény megegyezik a jól ismert egybites teljes összeadó átvitelbit függvényével.

Megjegyzés: a mod2-vel történő megvalósítás statikus hazárdos (lásd a 3.6. példát), míg a függvény szokásos kétszintű diszjunktív vagy konjunktív megvalósítása hazárdmentes.

ab\c	0	1
00		
01		1

11	1	1
10		1

3.4. Adja meg a kétszintű minimális diszjunktív lefedést (hazárd lehetséges), majd a hazárdmentes megoldást is!

	CDE							
AB	000	001	011	010	110	111	101	100
00		1	1		1	1	1	
01		1	1	1	1	1	1	
11		1	1	1	1			
10				1				

Megoldás:
Az összes primimplikáns:
1. A./C.D./E lényeges
2. /A.B.D
3. /A.C.D lényeges
4. /A.E lényeges
5. B./C.D
6. B./C.E lényeges
7. B.D./E lényeges
7. /A.B.D

A lényeges primimplikánsok lefedik az összes mintermet.

Ebben a megvalósításban statikus hazárd van, ezt a B./C.D hurokkal lehet kiküszöbölni.

3.5. Rajzolja fel az alábbi függvényeket közvetlenül megvalósító kombinációs hálózatot és vizsgálja meg statikus vagy dinamikus hazárd szempontjából! Ha van hazárd, akkor sorolja fel, hogy milyen átmenetnél milyen hazárdot talált és módosítsa úgy a kapcsolást, hogy hazárdmentes legyen! (Két feladat)

a/ F(A,B,C,D) = (/C + B.C).(A.D + /A.C)
b/ F(A,B) = A.X + B.X ; ahol X = /(A.B)

Megoldás:
a/ ÉS – VAGY – ÉS típusú háromszintű hálózat, ezért célszerű először az alacsonyabb szintű hazárdokat megkeresni.

F(A,B,C,D) = (/C + BC)(AD + /AC) = f1 * f2

Statikus hazárdok f1=/C + BC esetén

	CD			
AB	00	01	11	10
00	1	1		

Statikus hazárdok f2=AD + /AC esetén

	CD			
AB	00	01	11	10
00			1	1

01	1	1	1	1
11	1	1	1	1
10	1	1		

01			1	1
11		1	1	
10		1	1	

A	B	C	D
x	1	0↔1	x

A	**B**	**C**	**D**
0↔1	x	1	1

Statikus hazárd léphet fel F-ben, ha:

- az egyik alacsonyabb szintű függvény értéke konstans 1, miközben a másik függvény kimenetén statikus hazárd van (bal Karnough-tábla)
- az egyik alacsonyabb szintű függvény kimenetének értéke megváltozik (pl. 0→1) és a másik függvény kimenetének megváltozása pedig ennek az ellentéte (pl. 1→0) (jobb Karnaugh-tábla)

CD

AB	00	01	11	10
00	0	0	0	0
01	0	0	1	1
11	0	1	1	0
10	0	1	0	0

CD

AB	00	01	11	10
00	0	0	0	0
01	0	0	1	1
11	0	1	1	0
10	0	1	0	0

Dinamikus hazárd léphet fel F-ben, ha az egyik alacsonyabb szintű függvény kimenetének értéke megváltozik és a másik függvény kimenetén pedig statikus hazárd van.

CD

AB	00	01	11	10
00	0	0	0	0
01	0	0	1	1
11	0	1	1	0
10	0	1	0	0

A fentiek alapján a megoldás:

F(A,B,C,D)	**A**	**B**	**C**	**D**
Statikus hazárdok	1	1	0↔1	1
	0↔1	1	1	1
	0	0	0↔1	x
Dinamikus hazárdok	0	1	0↔1	x

b/ Első ránézésre ez egy inhomogén hálózat (a NAND kapu miatt), ezért kínálkozik az útérzékenyítési megoldás.
(Második ránézésre X = /(A.B) = /A + /B és így máris a szabályos VAGY – ÉS VAGY hálózatot kapjuk, ami ismét vizsgálható az előző példa módszerével is.

A lényeg az, hogy két dinamikus hazárdot találunk:

- A=1, B változik

- B=1, A változik

3.6.
a/ Van-e statikus vagy dinamikus hazárd az alábbi logikai függvényt közvetlenül megvalósító hálózatban?
b/ Tervezzen hazárdmentes kétszintű, csupán NAND kapukat tartalmazó ekvivalens hálózatot!

Z = (A.B) mod2 (B.C) mod2 (C.A)

Megoldás:
A mod2-es kimenet miatt az útérzékenyítés ajánlatos.
Ha B=C=1 és az A változik, akkor a három közül két ÉS kapu kimenete is változik, ami a kimeneten statikus hazárdot eredményezhet. Hasonló az eset a másik két változónál is, hiszen a kapcsolás szimmetrikus rájuk.
Ha B és C közül csak egyik 1-es és A változik, akkor csak egy ÉS kapu kimenetén lesz változás, így nem lesz a kimeneten hazárd.
Ha pedig B=C=0, akkor semelyik ÉS kapu sem terjeszti a változást, így hazárd sem lesz.

A fenti függvény hazárdmentes megvalósítása pl. a kétszintű minimális diszkjunktív vagy konjunktív az egybites teljes összeadó megoldás:

Z = a.b + a.c + b.c vagy

Z = (a+b).(a+c).(b+c)

3.7.
Furfangos hallgató úgy kódolja binárisan a decimális számjegyeket (0-9), hogy héttel megszorozza őket és ezt az eredményt írja le hat biten binárisan.

a. Mennyi a (minimális) Hamming távolsága ennek a kódkészletnek?

A kódszókészlet Hamming távolsága d=2, mert

- **d=2-re van példa, pl. az 1 és a 2 kódjai (000111 és 001110) között 2 a Hamming távolság,**
- **d=1 nem lehet, mert csak egy biten eltérő két kódszó különbsége kettő kerek hatványa, ami nem osztható héttel.**

b. Tervezze meg a kódolót, azaz adja meg a logikai függvényeit annak a kombinációs hálózatnak, amely elvégzi a héttel való szorzást:

DCBA=0000-ból fedcba=000000-t csinál,
DCBA=0001-ből fedcba=000111-t csinál,
DCBA=0010-ból fedcba=001110-t csinál,
………………………………………………
DCBA=1000-ból fedcba=111000-t csinál,
DCBA=1001-ból fedcba=111111-t csinál.

A megoldást itt nem részletezzük, csupán a kétszintű diszjunktív lefedés eredményeit adjuk meg:
f = D + C.B + C.A = D + C.(B + A)
e = D + B.A + C./A./B
d = D + C./A + B./A = D + /A.(C + B)

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

c. Most tervezze meg a dekódert, azaz adja meg annak a kombinációs hálózatnak a logikai függvényeit, amely ezt a furcsa kódot visszaalakítja NBCD kóddá, azaz:

fedcba=000000 helyett DCBA=0000-t ad,
fedcba=000111 helyett DCBA=0001-et ad,
fedcba=001110 helyett DCBA=0010-át ad,

fedcba=111000 helyett DCBA=1000-át és
fedcba=111111 helyett DCBA=1001-et ad ki a kimeneten.

A logikai függvényeket a lehető legegyszerűbb alakban adja meg!

A hatváltozós Karnaugh táblán berajzoltuk az egyes értelmes kódszavakat, az üresen hagyott helyek don't care-k.

				cba				
fed	**000**	**001**	**011**	**010**	**110**	**111**	**101**	**100**
000	**0**					**1**		
001					**2**			
011								**4**
010							**3**	
110		**7**						
111	**8**					**9**		
101				**6**				
100			**5**					

Fenti K táblából megkaphatók a visszakódolt NBCD bitek K táblái:

D:

				cba				
fed	**000**	**001**	**011**	**010**	**110**	**111**	**101**	**100**
000	**0**					**0**		
001					**0**			
011								**0**
010							**0**	
110		**0**						
111	**1**					**1**		
101				**0**				
100			**0**					

Amiből nyilvánvaló a legegyszerűbb megvalósítás:
D = f.e.d

C:

cba

fed	000	001	011	010	110	111	101	100
000	0					0		
001					0			
011								1
010							0	
110		1						
111	0					0		
101					1			
100				1				

Itt több jó megoldás is adódik!
Diszjunktív megvalósítással:
C = f./e + f.e./d +/f.e.d = /c.a + /c.b + c./b./a = /c.(a+b) + c./b./a
Mod2 kapu használatával:
C = f mod2 (e.d) = c mod2 (b+a)

B:

cba

fed	000	001	011	010	110	111	101	100
000	0					0		
001					1			
011								0
010							1	
110		1						
111	0					0		
101					1			
100				0				

A minimális megvalósítás:

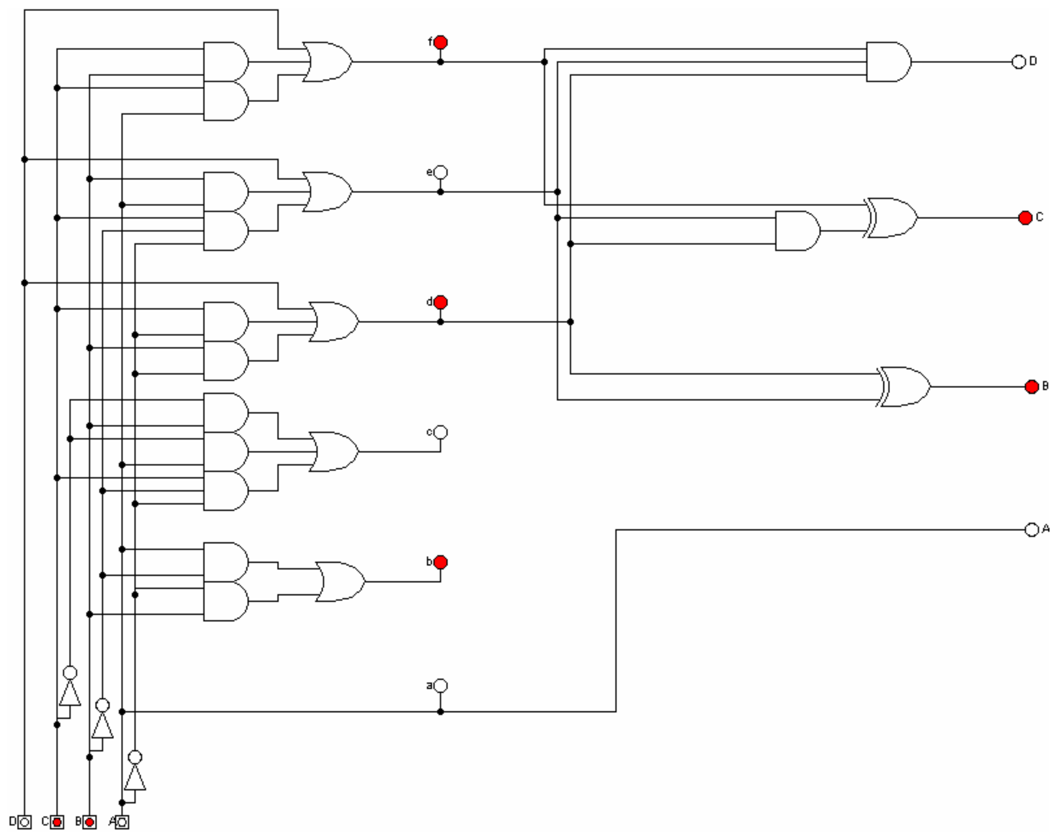
B = e mod2 d

A:

				cba				
fed	**000**	**001**	**011**	**010**	**110**	**111**	**101**	**100**
000	**0**					**1**		
001					**0**			
011								**0**
010							**1**	
110		**1**						
111	**0**					**1**		
101				**0**				
100			**1**					

A = a
Az átkódoló logikai függvényeket "jó szemmel" közvetlenül a kódolási táblából is ki lehet olvasni!

A kódoló – dekódoló együttes kapcsolási rajzát mutatja a Feladat3_7.dwm ábrája:



Nehéz példák az érdeklődőknek:

3n.1. A 3.2. feladatban tegye lényegtelen kombinációvá (don’t care) születés és névnapja (hónap és nap) mintermjeit.
Pl. ha 11.30 és 12.05 a kétnevezetes nap, akkor az ABCDE = 01011; 11110; 01100; 00101 mintermek lesznek lényegtelenek . Adja meg a kétszintű minimális lefedést!

3n.2. Valaki valamikor valahol azt állította, hogy egy don’t care-eket **nem tartalmazó** függvény kétszintű hazárdmentes megvalósításához az **összes** prímimplikánsra szükség van. Igaza van?