

Hermidæmi02 Greining og Hönnun stýrikerfa TÖV201G

Donn Eunice Bartido deb5@hi.is

Janúar 2023

Verkefnið

Verkefnið gengur út á að hanna flétturas sem kóðar frá BCD kóðun yfir í Excess-3 kóðun með notkun NAND rökrásahliða

Hönnunarforsendur

- flétturas á að umbreyta 4 bita merki sem er á BCD kóða yfir í Excess-3 kóða.
- Aðeins gildin 0,1,2....9 fullgild (þ.e við höfum aðeins áhuga á tölunum 0-9 í BCD).
- Rásin á að sýna Excess-3 stöðuna með 4x útmerkjum (t.d LED einingar í Cedar logic)
- Útmerkin eiga að vera í réttri röð m.v Excess-3 kóðann.
- MSB lengst til vinstri og LSB lengst til hægri
- Rásin notast aðeins við NAND hlið.



Aðferðin mín

1. Skilgreina hvað BCD og Excess-3 kóði er.
2. Setja upp Sanntöflu og K-map til að finna Boolean Jöfnu
3. Hönnun Flétturása
4. Prófun á rása.

1) BCD yfir í Excess-3 kóði;

Decimal Digit	BCD code	Excess-3 code
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110

Decimal Digit	BCD code	Excess-3 code
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

2.a) Sannleikstaflan og K-map á Excess-3 code

Sannleikstafla : **A B C D**

	A	B	C	D
1	0	0	0	0
2	0	0	0	1
3	0	0	1	0
4	0	0	1	1
5	0	1	0	0
6	0	1	0	1
7	0	1	1	
8	0	1	1	1
9	1	0	0	0
10	1	0	0	1
11	1	0	1	0
12	1	0	1	1
13	1	1	0	0
14	1	1	0	1
15	1	1	1	0
16	1	1	1	1

Sannleikstafla: **w x y z**

W	X	Y	Z
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
x	x	x	x
x	x	x	x
x	x	x	x
x	x	x	x
x	x	x	x
x	x	x	x

2.b) K-map til að finna Boolean Jöfnu

Til að finna út hvernig við hönnum circuit þá nota ég k-map aðferðina fyrir Excess-3 kóða sem úttak af öllum bitinum af BCD sem inntak. (ss þar sem BCD er 1 þá set ég það upp í k-map)

K-map w

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

K-map x

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

K-map y

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

K-map z

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

2.c) Boolean Jafna

- $w = A + BC + BD$
 - Groups:
 - $A(8,9,10,11,12,13,14,15)$
 - $BC(6,7,12,15)$
 - $BD(5,7,12,15)$
- $x = B'C + B'D + BC'D'$
 - Groups:

- $\mathbf{B}'\mathbf{C}(2,3,10,11)$
 - $\mathbf{B}'\mathbf{D}(1,3,9,11),$
- $\mathbf{y} = \mathbf{CD} + \mathbf{C}'\mathbf{D}'$
 - Groups:
 - $\mathbf{CD}(3,7,11,15)$
 - $\mathbf{C}'\mathbf{D}'(0,4,8,12)$
- $\mathbf{z} = \mathbf{D}'$
 - Groups;
 - $\mathbf{D}'(0,2,4,6,8,10,12,14)$

Hönnun flétturása

ath notast aðeins við NAND Rökrásarhliða

NAND Gate implementation for simplified function

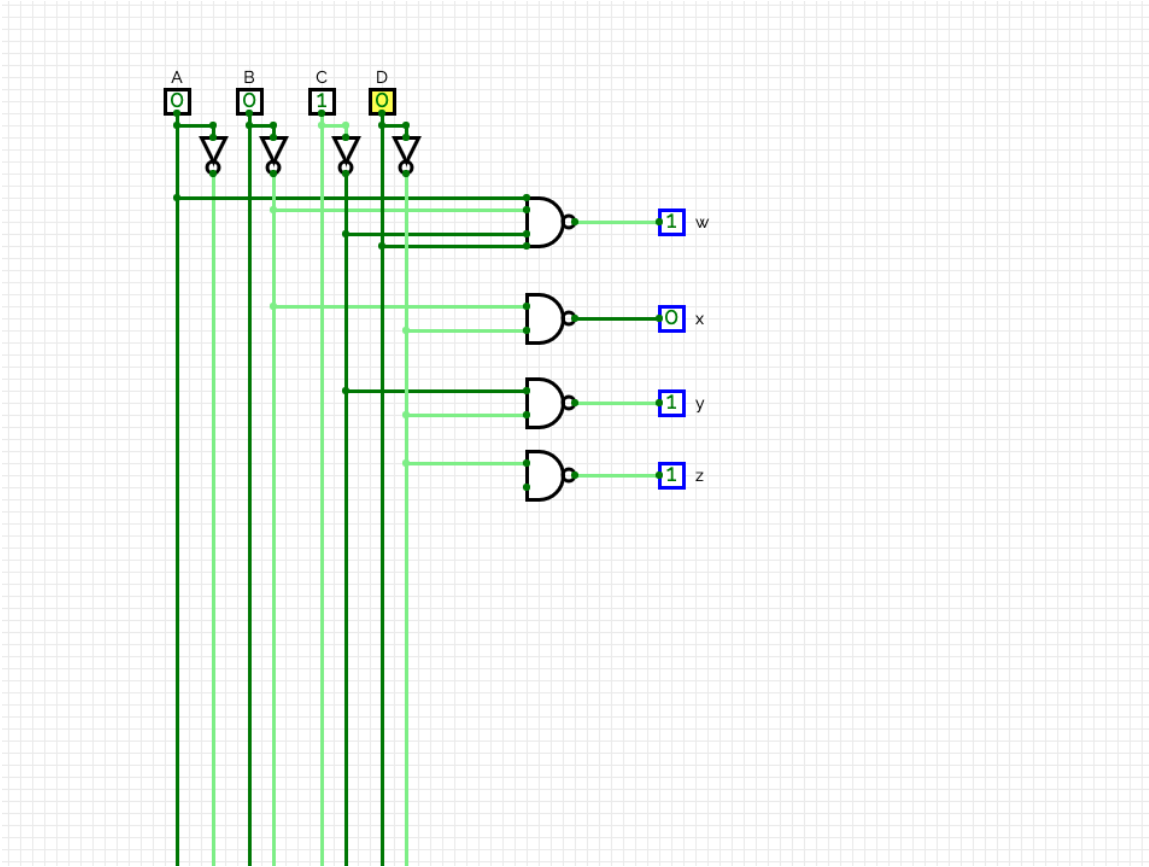
- $w' = (A'BC'D')' = A'' + B' + C' +$
- $x' = (B'C + B'D + BC'D')' = (B'C)'(B'D)'(BC'D')' = (B + C')(B + D')(B'C + BC' + BD) = (B + C')(B + D')(B'C + BD')$
- $y' = (CD + C'D')' = (C'D)'(C'D')'(CD)' = (C' + D')(C + D)(C'D' + CD)$
- $z' = D$

Nú getum við notað **DeMorgan** regluna til að umbreyta Boolean jöfnuna í NAND form:

- $w = (A'N)(BN)(CN)(DN)$
- $x = (B'N)(C'N)(BCN)(B'CN)(BD'N)$
- $y = (C'N)(D'N)(CDN)(C'DN)$
- $z = D$

þar sem N táknar negation, sem er í raun NAND-gátt. Nand formið er alltaf útfært með Nand-gátum og er oft notað til að spara á fjölda gáttanna.

Hönnun Flétturása



In []: