

RAZVOJ DIGITALNIH SISTEMOV

Izpit

12. 02. 2019

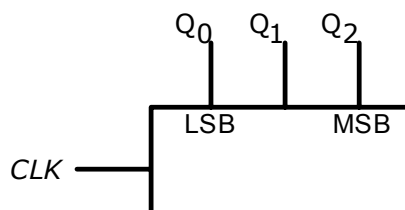
1. Izrazite podano logično funkcijo samo s Shefferjevimi operatorji. Morebitne negacije realizirajte s Shefferjevim operatorjem.

$$f(a, b, c) = \overline{(a + b) \downarrow \bar{c}} \equiv (a \downarrow c)$$

2. Realizirajte funkcijo f v obliki PDNO z redundancami s čim manj izbiralniki 4/1.

$$f(x_1, x_2, x_3, x_4) = V(1, 2, 9, 13, 15) \text{ in } V_x(0, 5, 11, 12)$$

3. Prikažite sintezo sinhronnega 3-bitnega števca navzdol z uporabo T flip-flopov: Zapišite tabelo prehajanja stanj in določite enačbe flip-flopov ter vezje narišite. Imena signalov so razvidna iz spodnje slike.



4. Narišite Mealyev diagram stanj za avtomat končnih stanj, ki ima vhod w in izhod z . Avtomat končnih stanj postavi izhod $z=1$, ko se na vhodu pojavi zaporedje "110" ali "101", sicer je $z=0$. Prekrivanje vzorcev je dovoljeno.

Rešitev 1. naloge

Realizacija s samimi Shefferjevimi (NAND, oziroma \uparrow) operatorji najprej zahteva pretvorbo funkcije v disjunktivno obliko (če se da, normalno).

$$f(a, b, c) = \overline{((a + b) \downarrow \bar{c})} \equiv (a \downarrow c)$$

V ta namen moramo najprej operatorje (EQU, Pierce) v podani funkciji izpisati z disjunkcijami in konjunkcijami. Začnemo na najvišjem nivoju, pri negaciji "čez vse" in EQU funkciji - oboje skupaj predstavlja XOR na najvišjem nivoju, ki jo izpišemo po definiciji XOR $\bar{x} \cdot y + x \cdot \bar{y} = x \oplus y$:

$$\begin{aligned} f(a, b, c) &= ((a + b) \downarrow \bar{c}) \oplus (a \downarrow c) = \\ &= ((a + b) \downarrow \bar{c}) \cdot \overline{(a \downarrow c)} + \overline{((a + b) \downarrow \bar{c})} \cdot (a \downarrow c) \end{aligned}$$

Izpišemo še oba Pierceva (NOR) operatorja, pri čemer izničimo nastali dvojni negacijo drugega in tretjega člena ($x + y = x \downarrow y$):

$$f(a, b, c) = \overline{((a + b) + \bar{c})} \cdot (a + c) + ((a + b) + \bar{c}) \cdot \overline{(a + c)}$$

Uporabimo DeMorganov teorem nad negiranimi členi in nekaj ostalih osnovnih lastnosti Boole-ove logike ($x \cdot x = x$ in $x \cdot \bar{x} = 0$):

$$\begin{aligned} f(a, b, c) &= \overline{(a + b)} \cdot c \cdot (a + c) + ((a + b) + \bar{c}) \cdot \bar{a} \cdot \bar{c} = \\ &= \bar{a} \cdot \bar{b} \cdot c \cdot (a + c) + (a + b + \bar{c}) \cdot \bar{a} \cdot \bar{c} = \\ &= \bar{a} \cdot \bar{b} \cdot c \cdot \bar{a} + \bar{a} \cdot \bar{b} \cdot c \cdot a + \bar{a} \cdot \bar{c} + a \cdot \bar{a} \cdot \bar{c} + b \cdot \bar{a} \cdot \bar{c} = \\ &= \bar{a} \cdot \bar{b} \cdot c + \bar{a} \cdot \bar{c} + a \cdot \bar{a} \cdot \bar{c} + b \cdot \bar{a} \cdot \bar{c} = \bar{a} \cdot \bar{b} \cdot c + \bar{a} \cdot \bar{c} \end{aligned}$$

Dobili smo disjunktivno normalno obliko (DNO):

$$f(a, b, c) = \bar{a} \cdot \bar{b} \cdot c + \bar{a} \cdot \bar{c}$$

Dobljeno DNO pretvorimo v Shefferjevo (NAND) obliko tako, da dvakrat negiramo posamezne člene:

$$\begin{aligned} f(a, b, c) &= \bar{a} \cdot \bar{b} + \bar{a} \cdot \bar{c} = \overline{\overline{\bar{a} \cdot \bar{b}}} + \overline{\overline{\bar{a} \cdot \bar{c}}} \\ &= \overline{\overline{\bar{a} \cdot \bar{b}}} + \overline{\overline{\bar{a} \cdot \bar{c}}} = (\bar{a} \uparrow \bar{b}) \uparrow (\bar{a} \uparrow \bar{c}) \end{aligned}$$

Preostale negacije spremenljivk izrazimo z uporabo lastnosti $\bar{x} = x \uparrow x$:

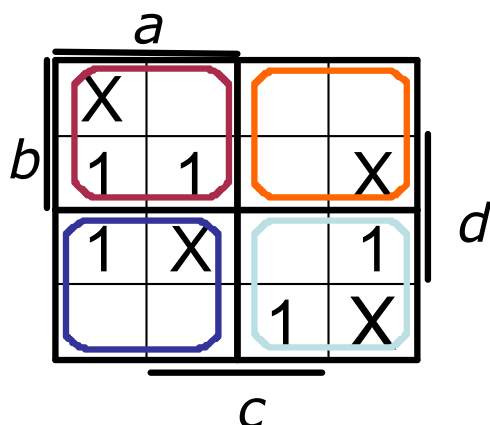
$$\begin{aligned} f(a, b, c) &= (\bar{a} \uparrow \bar{b}) \uparrow (\bar{a} \uparrow \bar{c}) \\ &= ((a \uparrow a) \uparrow (b \uparrow b)) \uparrow ((a \uparrow a) \uparrow (c \uparrow c)) \end{aligned}$$

Rešitev 2. naloge:

Funkcija f je podana v obliki PDNO z redundancami.

$$f = V(1,2,9,13,15) \text{ in } V_x(0,5,11,12)$$

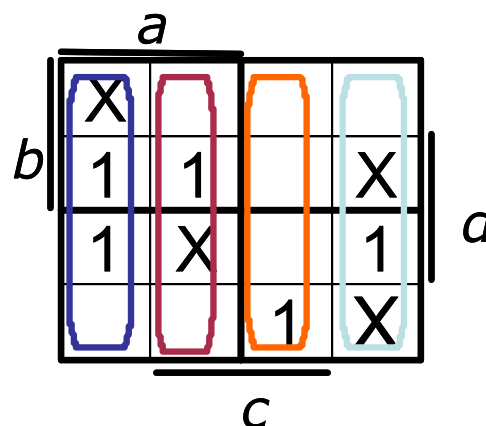
Dobljeno funkcijo vrišemo v Veitchev diagram. Ker iščemo najcenejšo realizacijo z izbiralnikom 4/1, bomo naredili razvoj po vseh kombinacijah naslovnih spremenljivk v Veitchevem diagramu. Če izberemo kot naslovni spremenljivki a in b , potem dobimo:



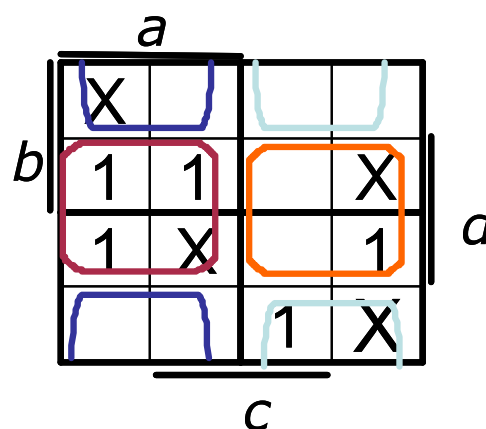
V zgornjem Veitchevem diagramu so označena vsa štiri polja štirih mintermov, če izberemo vhodni spremenljivki a in b . Zgornji levi kvadrat (rdeč) pomeni, da bo to polje izbrano ko bosta $ab="11"$, oranžni kvadrat ko bo $ab="01"$, temno modri ko bo $ab="10"$ in svetlo modri ko bo $ab="00"$. Vsakega od teh kvadratov poskušamo opisati s čimbolj enostavno funkcijo: Vrednost zgornjega levega kvadrata opišemo s spremenljivko d , če postavimo redundanco na '0'. Vrednost spodnjega desnega kvadrata je bolj komplicirana, saj moramo vsako '1' opisati posebej: Za zgornjo '1' v tem kvadratu velja $c \cdot d$, za spodnjo '1' pa $c \cdot d'$. Funkcija bo torej $c \cdot d + c \cdot d'$, kar je enačba funkcije XOR. Najbolj enostavna realizacija je zgornji desni kvadrat, ki je kar '0', če postavimo redundanco na '0'. Zato, da bi pregledali še ostale možnosti, moramo narisati še preostalih pet kombinacij dveh naslovnih vhodov.

Čas pisanja je 60 minut. Vsaka naloga je vredna 10 točk. Na list z rešitvami se podpišite in napišite še vpisno številko. Rezultati bodo objavljeni na domači strani predmeta.

Če izberemo kot naslovni spremenljivki a in c , dobimo levi Veitchev diagram, če a in d , pa desnega. Podobno kot v prejšnjem primeru poiščemo realizacije ustreznih kvadratov in iščemo najcenejšo realizacijo: Izogibamo se veliko različnim funkcijam in iščemo drugače kvadratov, ki vsebujejo same '1' ali same '0'. Pri razvoju po a in c imamo pri $ac="01"$ najneugodnejšo funkcijo, saj vsebuje eno samo '1'.

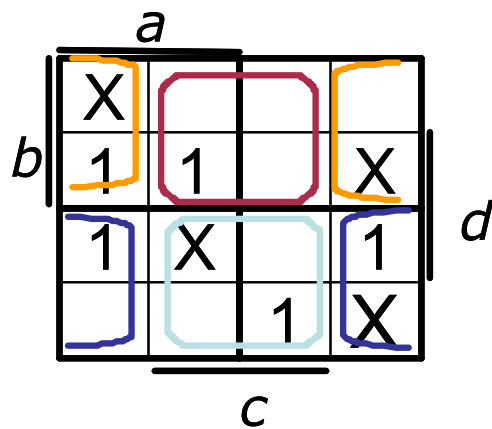


Pri razvoju po a in d nikjer ne nastopa ena sama '1' ali tri '1' ali diagonalna (XOR) dveh '1'.

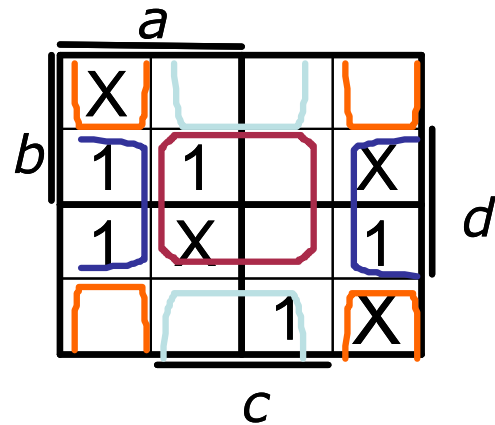
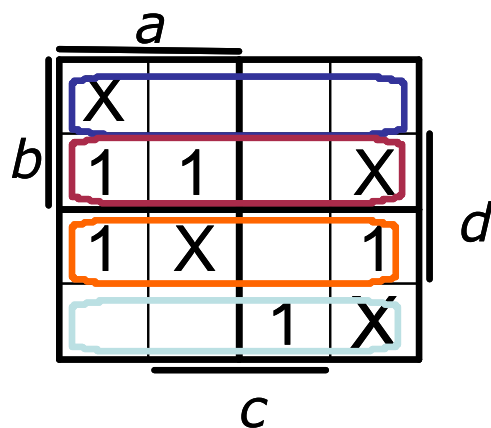


Nato izberemo naslovni spremenljivki b in c , (levi Veitchev diagram) in b in d (desni diagram). Pri razvoju po b in c

imamo pri $bc="11"$ najneugodnejšo funkcijo (rdeč), saj vsebuje eno samo '1'.

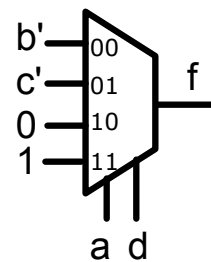


Pri razvoju po b in d dobimo eno (od dveh) možno realizacijo s funkcijskimi ostanki: $F_{00}=a'$; $F_{01}=c'$; $F_{10}=0$ in $F_{11}=a$.



Zadnja kombinacija naslovnih vhodov je cd . Pri razvoju po c in d imamo pri $cd="10"$ najneugodnejšo funkcijo (svetlo modro), saj vsebuje eno samo '1'. Najbolj ugodna kombinacija za realizacijo je torej razvoj po spremenljivkah a in d .

Končna realizacija funkcije je lahko:



Druga možna rešitev je kombinacija naslovnih spremenljivk b in d .

Rešitev 3. naloge:

Postopek sinteze zahteva, da zapišemo tabelo prehajanja stanj števca:

Trenutno stanje			Naslednje stanje			Enačbe FF		
Q ₂	Q ₁	Q ₀	Q ₂	Q ₁	Q ₀	T ₂	T ₁	T ₀
0	0	0	1	1	1	1	1	1
0	0	1	0	0	0	0	0	1
0	1	0	0	0	1	0	1	1
0	1	1	0	1	0	0	0	1
1	0	0	0	1	1	1	1	1
1	0	1	1	0	0	0	0	1
1	1	0	1	0	1	0	1	1
1	1	1	1	1	0	0	0	1

Normalna analiza bi zahtevala, da narišemo Veitch-eve diagrame za tri spremenljivke za vsak vhod T-FF, vendar ker so T-FF po svoji naravi primerni za realizacijo števec, so praviloma njihove vhodne enačbe zelo enostavne. Iz tabele prehajanja stanj števca določimo enačbe T-FF:

Iz stolpca T₀ se vidi:

$$T_0 = 1$$

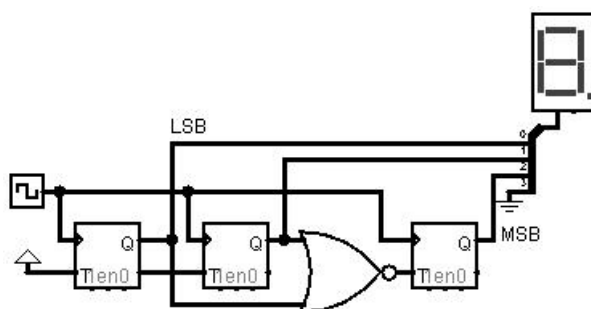
Z opazovanjem stolpcev trenutnega stanja določimo T₁:

$$T_1 = \overline{Q_0}$$

Podobno lahko določimo T₂:

$$T_2 = \overline{Q_0} \cdot \overline{Q_1} = \overline{Q_0 + Q_1}$$

Opis delovanja in vezje števca je v predlogah vaj na domači strani predmeta v imeniku Logisim\counter\ counter_7_0_using_T_FF.circ:

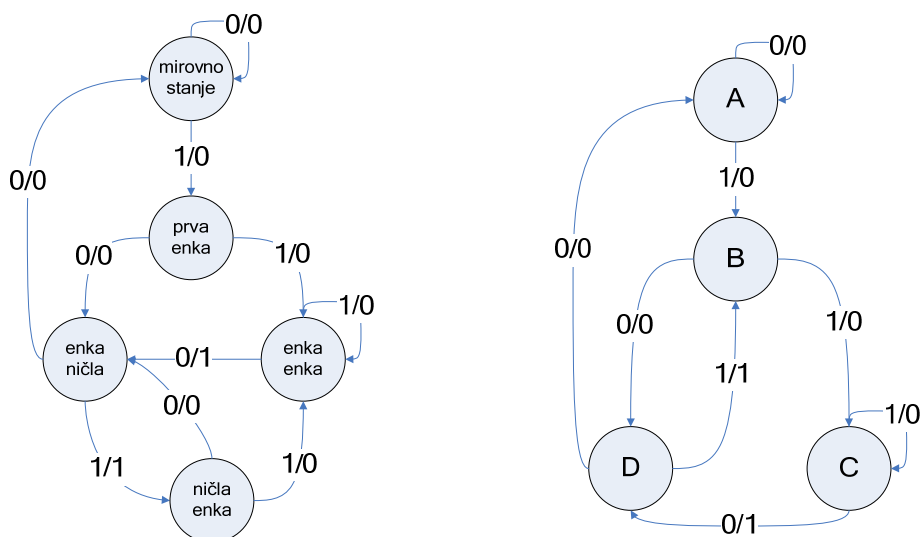


Rešitev 4. naloge:

Avtomat je Mealy–eve izvedbe, ker izhod postane '1' takoj, ko se pojavi zadnja vrednost zaznavanega zaporedja.

Zapišemo začetno stanje A, v katerem ostajamo toliko časa, dokler se ne začne ena od sekvenc, ki ju zaznavamo. Obe sekvenci se začneta z '1', zato v stanje B preidemo, ko je na vhodu prva '1'. V stanju B ne moremo ostati, saj se na vhodu lahko pojavi '0' ali '1' – v obeh primerih gre za del zaznavanega zaporedja "10X" ali "11X". Iz stanja B preidemo v stanje C, če se vmes pojavi '1', tako da v tem stanju pomeni detekcijo sekvence "11X", v stanje D pa preidemo če se pojavi na vhodu '0', kar pomeni detekcijo sekvence "10X".

Prekrivanje zaporedij: Če se v stanju C pojavi '1' na vhodu, potem gre za sekvenco "111" na vhodu – kar še vedno pomeni, da ostajamo v stanju C, saj je prekrivanje vzorcev dovoljeno. Drugače se diagram obnaša, ko smo v stanju D in pride na vhod še ena '0' – takrat smo imeli na vhodu sekvenco "100", tako da se moramo vrniti v stanje A, saj se nobena od zaznavanih sekvenc ne začneja z '0'.



Stanju "ničla enka" se lahko izognemo, saj je to stanje ekvivalentno stanju "prva enka" in dobimo izvedbo avtomata, prikazano na desni strani.