

RAZVOJ DIGITALNIH SISTEMOV

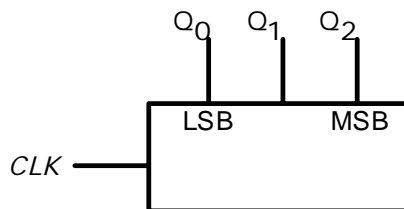
Izpit

07. 02. 2014

1. Zgradite vezje, katerega izhod postane '1', ko se na vhodu pojavi eno izmed praštevil (2, 3, 5, 7, 11, 13). Števila na vhodu so kodirana s 4-bitno Gray-evo kodo. Za realizacijo uporabite samo NAND vrata. Kolikšna je COST funkcija realiziranega vezja?
2. Uporabite PAL3L3 (namišljen čip) za realizacijo naslednjih funkcij:
 - $f_1 = x_1 \oplus x_2$
 - $f_2 = x_1 \cdot x_2 \cdot x_3$
 - $f_3 =$ funkcijo treh spremenljivk, ki vrne '1' pri vsaj dveh enicah na vseh.

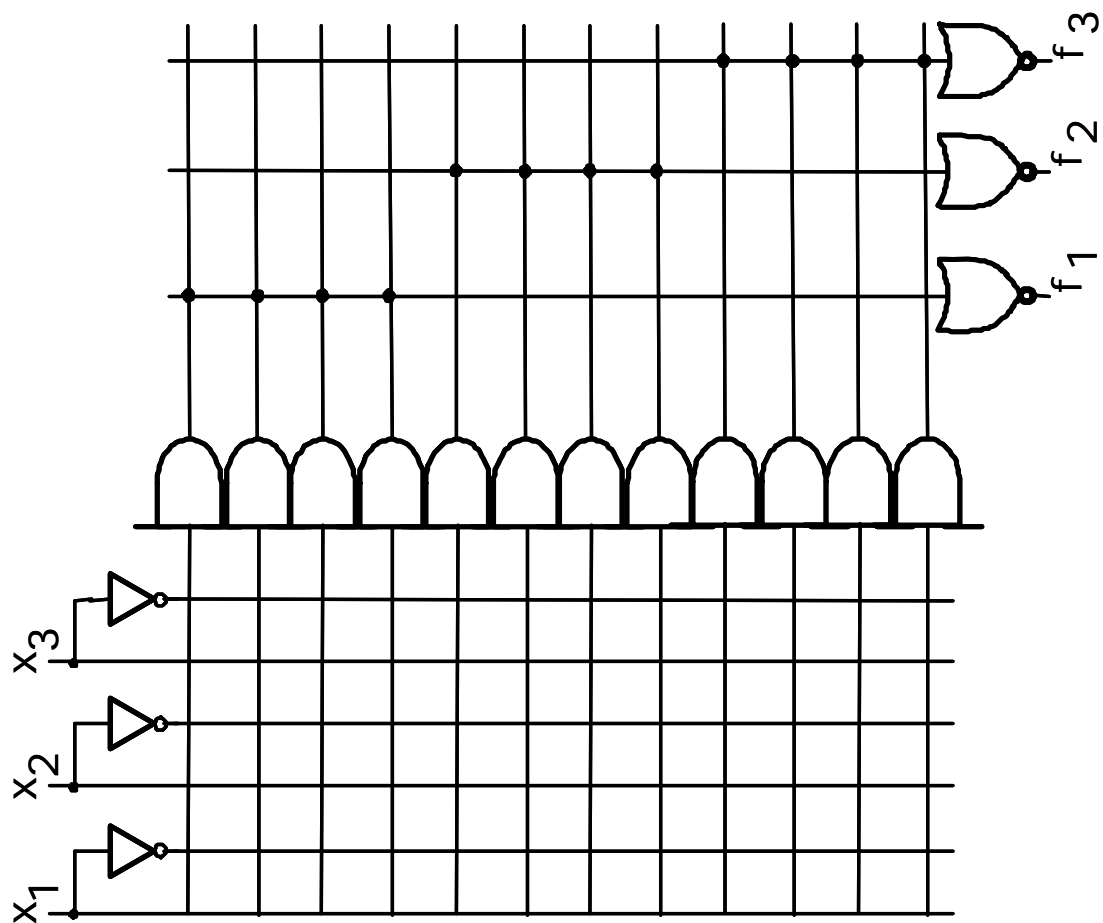
Vezje ima 3 vhode in 3 izhode. Vsaka disjunkcija (OR) ima 4 konjunkcije (AND). Oznaka L pomeni, da je izhod negiran. Programirane povezave označite s piko. Vezje PAL je narisano na hrbtni strani izpita.

3. Prikažite sintezo sinhronega 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.

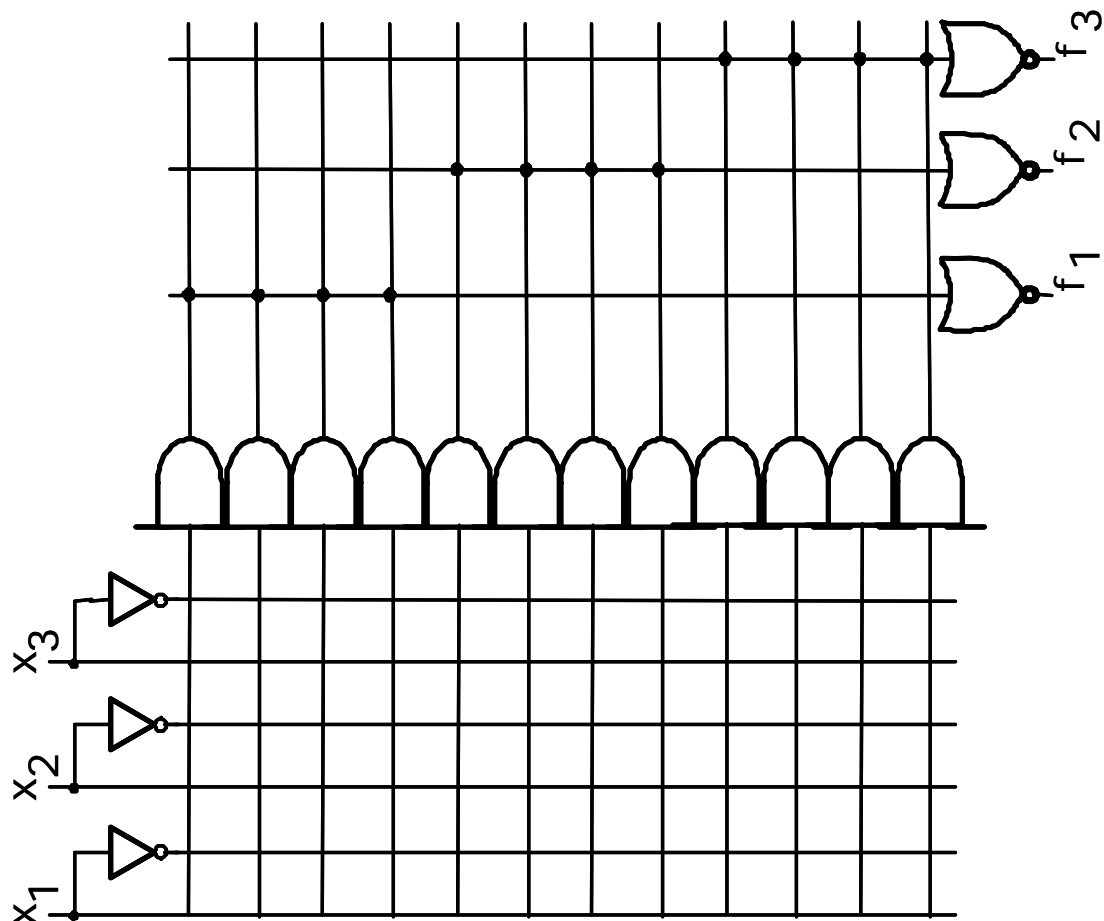


4. Narišite diagram stanj za avtomat končnih stanj, ki ima vhod w in izhod z. Avtomat končnih stanj postavi izhod $z=1$, ko se na vhodu pojavi zaporedje "110" ali "101", sicer je $z=0$. Prekrivanje vzorcev je dovoljeno. Delovanje avtomata končnih stanj povzema spodnje časovno zaporedje vhoda in izhoda. Tip avtomata je razviden iz podanega časovnega zaporedja.

CLK	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}	t_{12}
w	0	1	1	0	1	1	0	1	1	1	0	0	0
z	—	0	0	1	1	0	1	1	0	0	1	0	0



Če se zmotite, prečrtajte napačno shemo in uporabite drugo shemo!



Čas pisanja je 60 minut. Vsaka naloga je vredna 10 točk.

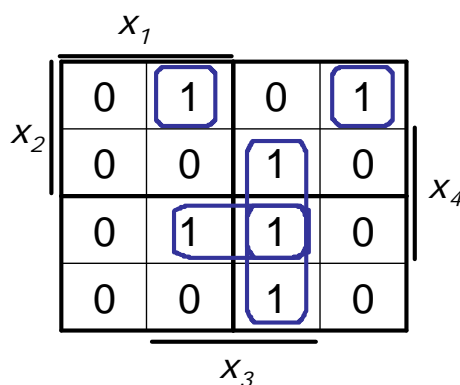
Na list z rešitvami se podpišite in napišite še vpisno številko ter kateri predmet pišete.

Rezultati bodo objavljeni na: <https://estudent.fri.uni-lj.si>

Rešitev 1. naloge:

S 4-bitno Grayevo kodo lahko kodiramo 16 števil (0 - 15). Najprej zapišemo pravilnostno tabelo v kateri desetiška števila kodiramo v Grayevi kodi (npr. $7_{10}=0100_{\text{Gray}}$). Izhod mora biti '1' vsakič, ko se na vhodu pojavi praštevilo. Izpisano funkcijo nato izpišemo v Veitch-ev diagram, od koder bomo lahko izvajali postopek minimizacije v PDNO, iz te oblike pa bomo prešli na PSNO (NAND operatorji).

št.	x_1	x_2	x_3	x_4	f
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	1	1
3	0	0	1	0	1
4	0	1	1	0	0
5	0	1	1	1	1
6	0	1	0	1	0
7	0	1	0	0	1
8	1	1	0	0	0
9	1	1	0	1	0
10	1	1	1	1	0
11	1	1	1	0	1
12	1	0	1	0	0
13	1	0	1	1	1
14	1	0	0	1	0
15	1	0	0	0	0



$$f_{MDNO} = \overline{x_1} \cdot \overline{x_2} \cdot x_3 + \overline{x_2} \cdot x_3 \cdot x_4 + \overline{x_1} \cdot x_3 \cdot x_4 + x_1 \cdot x_2 \cdot x_3 \cdot \overline{x_4} + \overline{x_1} \cdot x_2 \cdot \overline{x_3} \cdot \overline{x_4}$$

Da bi iz PDNO prešli na realizacijo s samimi NAND vrati, funkcijo v PDNO dvakrat negiramo in uporabimo de Morgan-ov teorem:

$$\overline{\overline{f_{MDNO}}} = \overline{\overline{x_1 \cdot \overline{x_2} \cdot x_3 + \overline{x_2} \cdot x_3 \cdot x_4 + \overline{x_1} \cdot x_3 \cdot x_4 + x_1 \cdot x_2 \cdot x_3 \cdot \overline{x_4} + \overline{x_1} \cdot x_2 \cdot \overline{x_3} \cdot \overline{x_4}}}$$

$$f_{PSNO} = \overline{(\overline{x_1 \cdot \overline{x_2} \cdot x_3}) \cdot (\overline{\overline{x_2} \cdot x_3 \cdot x_4}) \cdot (\overline{\overline{x_1} \cdot x_3 \cdot x_4}) \cdot (\overline{x_1 \cdot x_2 \cdot x_3 \cdot \overline{x_4}}) \cdot (\overline{\overline{x_1} \cdot x_2 \cdot \overline{x_3} \cdot \overline{x_4}})}$$

Za nastalo PSNO moramo določiti COST vezja (brez inverterjev):

OBLIKA	VRAT	VHODOV	COST
PSNO	6	22	28

Čas pisanja je 60 minut. Vsaka naloga je vredna 10 točk.

Na list z rešitvami se podpišite in napišite še vpisno številko ter kateri predmet pišete.

Rezultati bodo objavljeni na: <https://estudent.fri.uni-lj.si>

Rešitev 2. naloge:

Za funkcije zapišemo najprej pravilnostno tabelo, nato narišemo Veitch–eve diagrame.

x_1	x_2	x_3	f_1	f_2	f_3
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	1	0	0
0	1	1	1	0	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	0	0	1
1	1	1	0	1	1

Vezje PAL ima negirane izhode, zato bomo pri realizaciji funkcij z Veitch–evimi diagrami realizirali \overline{f} in ne f .

$\overline{f_1}$:

	x_1		
x_2	0	0	1
	1	1	0
		x_3	

Prvo funkcijo zapišemo enostavno, saj je negacija XOR funkcije dveh spremenljivk kar funkcija ekvivalence:

$$\overline{f_1} = x_1 \cdot x_2 + \overline{x_1} \cdot \overline{x_2}$$

Podobno lahko naredimo za drugo funkcijo, kjer za negacijo konjunkcije treh spremenljivk uporabimo De Morgan–ovo enakost.

$\overline{f_2}$:

	x_1		
x_2	0	1	0
	0	0	0
		x_3	

$$f_2 = \overline{x_1 \cdot x_2 \cdot x_3}$$

$$\overline{f_2} = \overline{x_1} + \overline{x_2} + \overline{x_3}$$

Zadnjo funkcijo minimiziramo z uporabo Veitch–evega diagrama, tako da zbiramo ničle.

$\overline{f_3}$:

	x_1		
x_2	1	1	1
	0	1	0
		x_3	

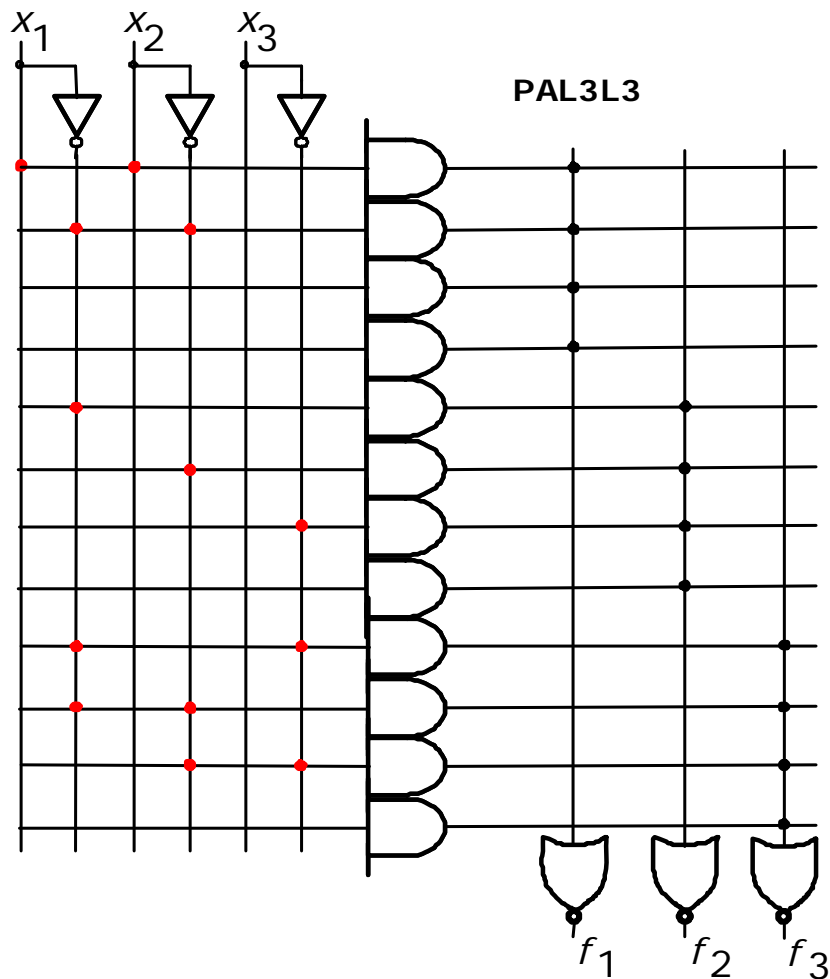
$$\overline{f_3} = \overline{x_1} \cdot \overline{x_3} + \overline{x_1} \cdot \overline{x_2} + \overline{x_2} \cdot \overline{x_3}$$

Pri realizaciji PAL vezja upoštevamo poenostavljeno strukturo, pri kateri ne vezemo vsake povezave na konjunkcije, saj so AND vrata na narisani strukturi 6–vhodna. Vezje PAL3L3 je AND–NOR arhitekture in vsebuje 4 konjunkcije na en NOR člen. Pri PAL vezju je programabilen samo AND del vezja.

Čas pisanja je 60 minut. Vsaka naloga je vredna 10 točk.

Na list z rešitvami se podpišite in napišite še vpisno številko ter kateri predmet pišete.

Rezultati bodo objavljeni na: <https://estudent.fri.uni-lj.si>



Predstavljeno PAL3L3 vezje je sicer izmišljeno, vendar demonstrira strukturo in uporabo večjih (realnih) PAL vezij kot so npr. PAL14L4, GAL16V8 in GAL22V10. GAL vezja so nadgradnja osnovne PAL strukture. Slednji so izključno kombinacijski, GAL vezja pa imajo v OLMC (ang. Output Logic MacroCell) strukturi še D-FF, s katerim lahko realiziramo tudi sekvenčna vezja.

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 T_0 se vidi:

$$T_0 = 1$$

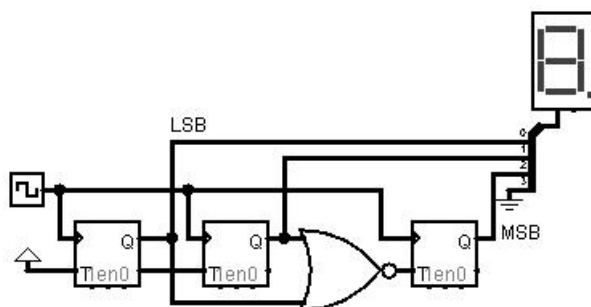
Z opazovanjem stolpcev trenutnega stanja določimo T_1 :

$$T_1 = \overline{Q_0}$$

Podobno lahko določimo T_2 :

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

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



Čas pisanja je 60 minut. Vsaka naloga je vredna 10 točk.

Na list z rešitvami se podpišite in napišite še vpisno številko ter kateri predmet pišete.

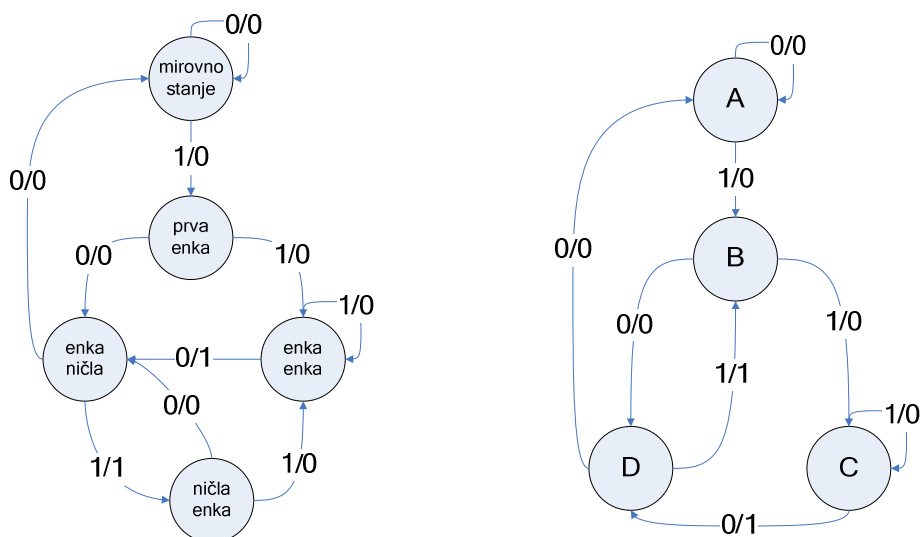
Rezultati bodo objavljeni na: <https://estudent.fri.uni-lj.si>

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.