

# RAZVOJ DIGITALNIH SISTEMOV

2. kolokvij  
24. 1. 2012

1. Prikažite postopek celoštevilskega deljenja nepredznačenih števil v dvojiškem sistemu na primeru:

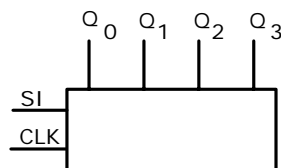
$$47_{10}/9_{10}$$

2. Uporabite PAL3L3 (namišljen čip) za realizacijo naslednjih funkcij:

- $f_1 = x_1 \oplus x_2$
- $f_2$  = konjunkcijo treh spremenljivk
- $f_3$  = funkcijo treh spremenljivk, ki vrne '1' pri vsaj dveh enicah na vseh 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.

3. Sestavite 4-bitni pomikalni register s T-celicami in izbiralniki 2/1. Register ima zaporedni vhod *SI* (ang. serial input), in vzporedni izhod ( $Q_0, Q_1, Q_2, Q_3$ ).



4. Narišite tabelo prehajanja stanj Moore-ovega avtomata končnih stanj, ki deluje kot krmilje za kavni avtomat. Kava stane 15 centov, plačujemo pa lahko s kovancema za 5 in 10 centov. Krmilje ima:

- vhod *5cent*, ki postane '1', ko uporabnik vrže v avtomat kovanec za 5 centov,
- vhod *10cent* ki postane '1', ko uporabnik vrže v avtomat kovanec za 10 centov,
- izhod *p*, ki postane '1', ko uporabnik vrže v avtomat skupno 15 centov.

Avtomat ne vrača drobiža in se ob detekciji plačila 15 centov ne vrača nazaj v začetno stanje, ampak ostane v končnem stanju. Vnos dveh kovancev naenkrat ni mogoč.

Rešitev 2. naloge:

Pretvorimo število v dvojiški zapis  $47_{10}=2F_{16}=0010\ 1111_2$

Podobno  $9_{10}=9_{16}=1001_2$

				0	0	1	0	1	kvocient=5 <sub>10</sub>				
1	0	0	1	0	0	1	0	1	1	1	1	1	START
				1	0	0	1						2 < 9
				0	0	1	0	1					
					1	0	0	1					5 < 9
					0	1	0	1	1				
						1	0	0	1				11>9→ odštejemo
						0	0	1	0	1			5 < 9
							1	0	0	1			
							0	1	0	1	1		11>9→ odštejemo
								1	0	0	1		
								0	0	1	0		ostanek=2 <sub>10</sub>

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
	$x_3$		

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

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

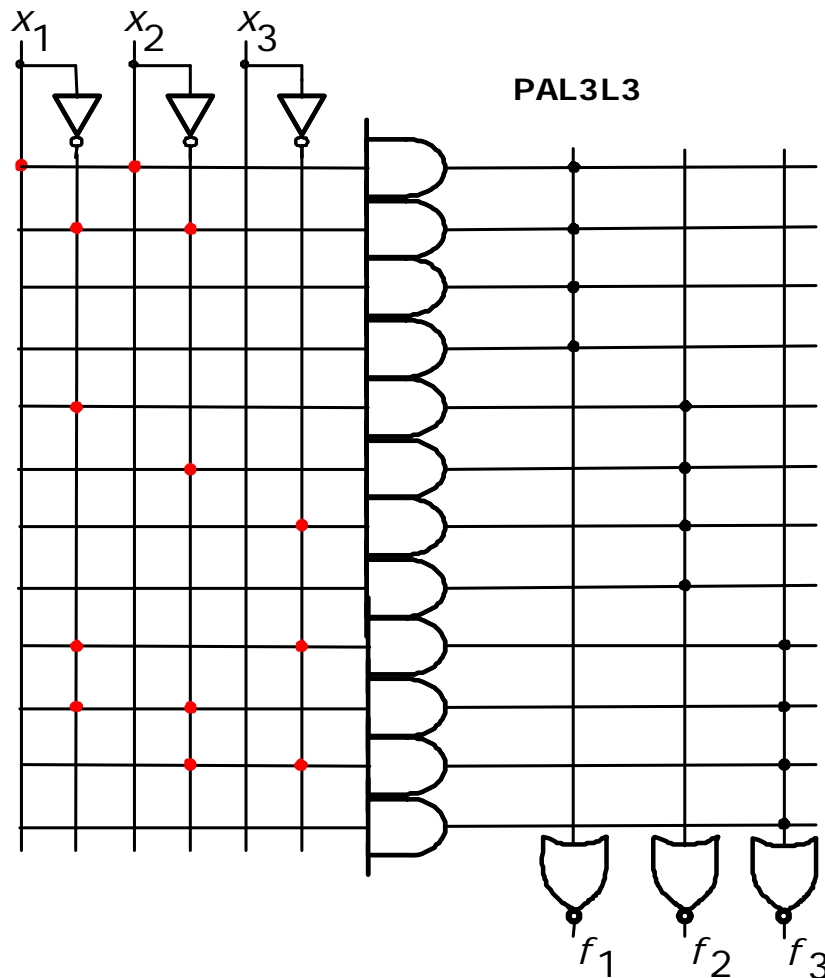
Zadnjo funkcijo minimiziramo z uporabo Veitch–evega diagrama, tako da zbiramo ničle.

$\overline{f_3}$ :

	$x_1$		
$x_2$	1	1	1
	0	1	0
	$x_3$		

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

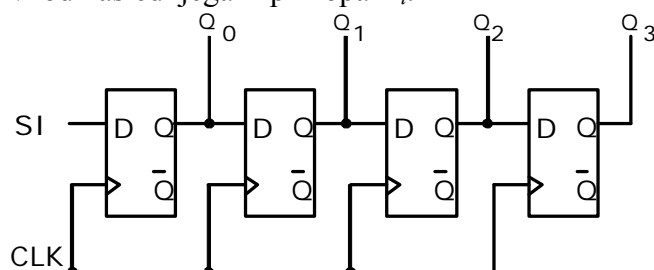
Pri realizaciji PAL vezja upoštevamo poenostavljeno strukturo, pri kateri ne vežemo vsake povezave na konjunkcije, saj so AND vrata na narisani strukturi 6-vhodna. Vezje PAL3L3 je AND-NOR arhitekture in vsebuje 4 konjunkcije na en NOR člen. Pri PAL vezju je programabilen samo AND del vezja.



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 kombinatorni, GAL vezja pa imajo v OLMC (ang. Output Logic MacroCell) strukturi še D-FF, s katerim lahko realiziramo tudi sekvenčna vezja.

Rešitev 3. naloge:

Zaporedno-vzporedni (SIPO) pomikalni register, realiziran s pomočjo D-FF, je veriga kaskadno vezanih D-FF, v kateri je izhod prejšnjega flip-flopa  $Q_{i-1}$  vezan na vhod naslednjega flip-flopa  $D_i$ .



Če želimo pomikalni register sestaviti iz T-FF in 2/1 izbiralnikov, moramo pravzaprav realizirati celico D-FF s pomočjo T-FF in 2/1 izbiralnikov. V ta namen zapišemo tabelo D-FF, pri kateri dodamo izhodni stolpec T vhoda.

$D$	$Q(t)$	$Q(t+1)$	$T$
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

Iz tabele sledi, da je  $T$  vhod XOR operacija  $Q(t)$  in vhoda D-FF, ki ga realiziramo.

$$Q(t+1) = Q(t) \oplus D$$

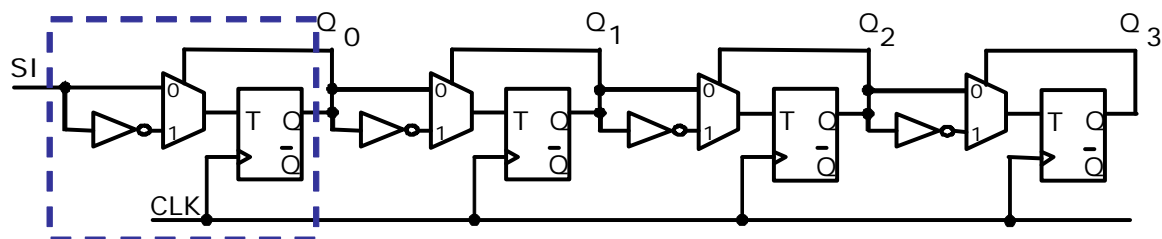
XOR vrata moramo realizirati s pomočjo 2/1 izbiralnikov, zato zapišemo enačbo XOR funkcije:

$$f = x \oplus y = \bar{x} \cdot y + x \cdot \bar{y}$$

Funkcijo  $f$  realiziramo z izbiralnikom tako, da naredimo razvoj po spremenljivki  $x$  in dobimo:

$x$	$f$
0	$y$
1	$y'$

Če nastali D-FF iz T-FF in 2/1 izbiralnika sestavimo skupaj v 4-bitni pomikalni register dobimo spodnjo realizacijo, v kateri je izvedba D-FF označena črtkano.



Opis delovanja in vezje pomikalnega registra je v predlogah avditornih vaj na domači strani predmeta: Logisim\shift\_reg\shift\_reg\_4bit\_using\_tff\_mux.circ

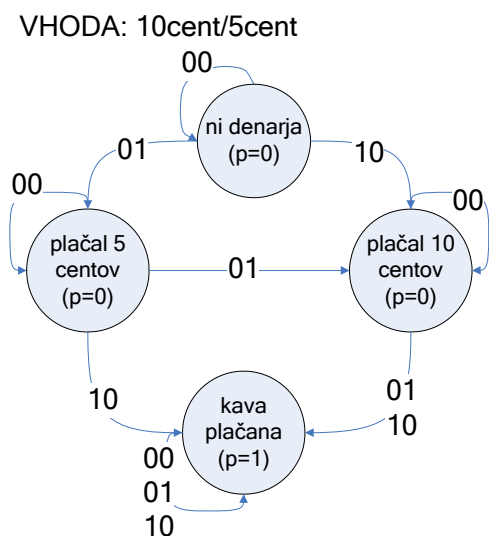
Čas pisanja je 60 minut. Vsaka naloga je vredna 10 točk.

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:

Moore—ova realizacija avtomata končnih stanj. Opis diagrama stanj:



Na začetku se nahajamo v stanju "ni denarja", v katerem je izhod  $p=0$ . Vhoda v avtomat sta dva: 10cent in 5cent, kar na diagramu kodiramo kot 10cent/5cent.

Mehanizem za vnos kovancev preprečuje hkraten vnos dveh kovancev, torej je kombinacija (10cent/5cent=11) nemogoča, zato bo avtomat od tu lahko prešel v poljubno stanje (X). Če uporabnik ni vrgel denarja v avtomat (10cent/5cent=00), potem ostaja v stanju "ni denarja". Če uporabnik vrže v avtomat 5 centov (10cent/5cent=01), potem preide v stanje "plačal 5 centov". Če uporabnik vrže v avtomat 10 centov (10cent/5cent=10), potem preide v stanje "plačal 10 centov". Ne glede na to koliko je

vrgel bo izhod v teh dveh stanjih enak  $p=0$ , ker še ni plačal celotne cene kave. Če smo v stanju "plačal 5 centov" in uporabnik vrže v avtomat 10 centov (10cent/5cent=10), potem preide v stanje "kava plačana", kjer postavimo izhod ( $p=1$ ). Stanje "kava plačana" je končno in tam tudi ostanemo za vse možne kombinacije. Če smo v stanju "plačal 10 centov" in uporabnik vrže v avtomat 5 ali 10 centov (10cent/5cent=10 oz. 01), potem podobno preidemo v stanje "kava plačana", kjer postavimo izhod ( $p=1$ ).

Naredimo tabelo prehajanja stanj:

trenutno stanje	naslednje stanje (5cent/10cent)				izhod p
	00	01	10	11	
ni denarja	ni denarja	plačal 5 centov	plačal 10 centov	X	0
plačal 5 centov	plačal 5 centov	plačal 10 centov	kava plačana	X	0
plačal 10 centov	plačal 10 centov	kava plačana	kava plačana	X	0
kava plačana	kava plačana	kava plačana	kava plačana	X	1

Celotno sintezo avtomata si lahko ogledate v zbirki vaj na domači strani predmeta.