

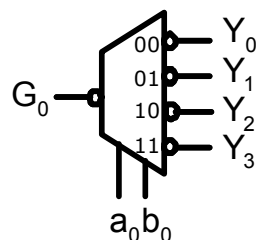
# RAZVOJ DIGITALNIH SISTEMOV

Izpit

28. 08. 2019

- Realizirajte funkcijo  $f^b = V(1, 2, 4, 7)$  z redundantnimi mintermi pri  $V_x(0, 3, 6)$  z enim TTL dekodirjem 74139. Dekoder 74139 ima vhod za omogočenje elementa ( $G$ ) in izhode  $Y_0, Y_1, Y_2, Y_3$  v negativni logiki. Njegovo delovanje povzema spodnja tabela:

$G_0$	$a_0$	$b_0$	$Y_0$	$Y_1$	$Y_2$	$Y_3$
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

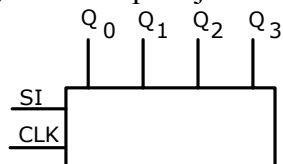


- Uporabite ROM vezje za realizacijo funkcij:

$$g_1 = x_1 + x_2 \cdot \overline{x_3} \quad g_2 = \overline{x_1} \cdot x_3 + x_1 \cdot x_2$$

ROM vezje ima 3 naslovne spremenljivke in 2 bitno vsebino. Narišite shemo ROM vezja in v shemi označite programirane povezave oz. 'varovalke' s piko (●).

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



- Minimizirajte podani avtomat končnih stanj z uporabo metode z razdelki ter zapišite tabelo prehajanja stanj nastalega minimalnega avtomata.

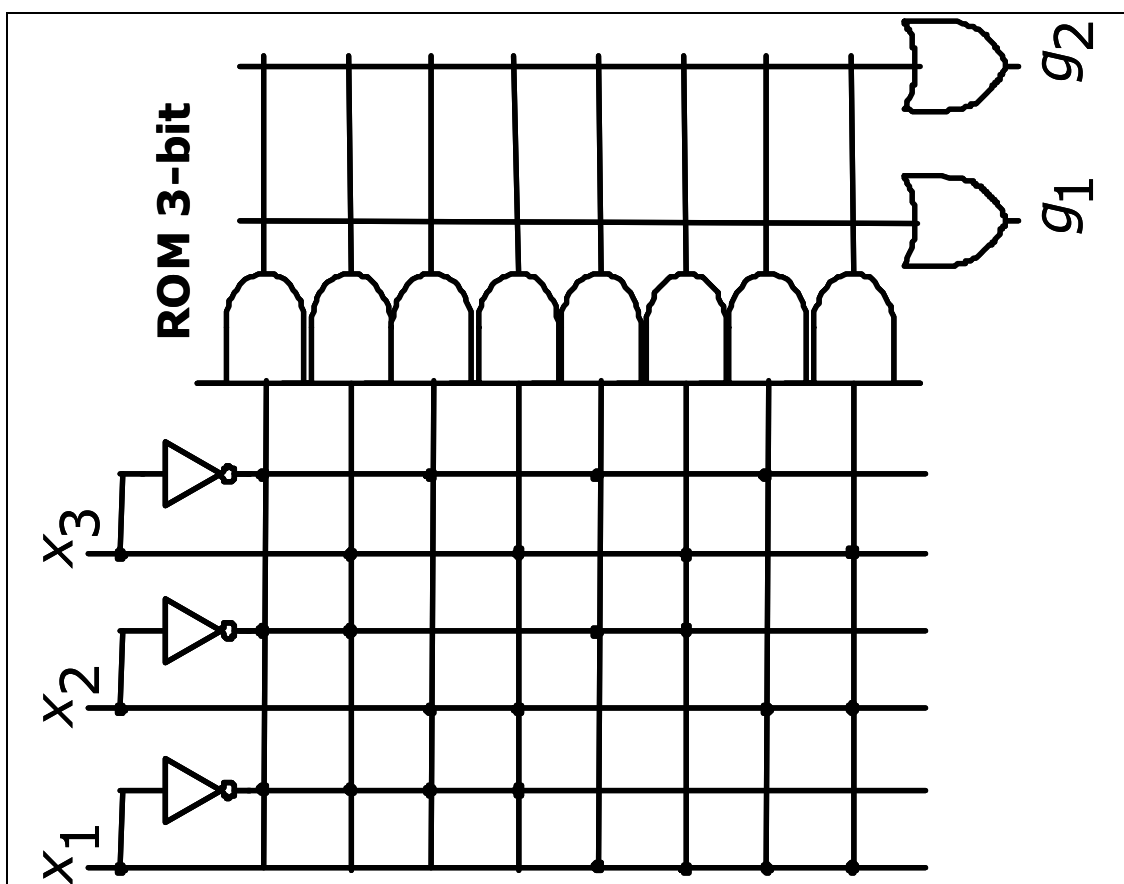
<i>Trenutno stanje</i>	<i>Naslednje stanje</i>		<i>Izhod</i>
	$w=0$	$w=1$	
A	B	C	1
B	D	F	1
C	F	E	0
D	B	G	1
E	F	C	0
F	E	D	0
G	F	G	0

**OBRNITE LIST**

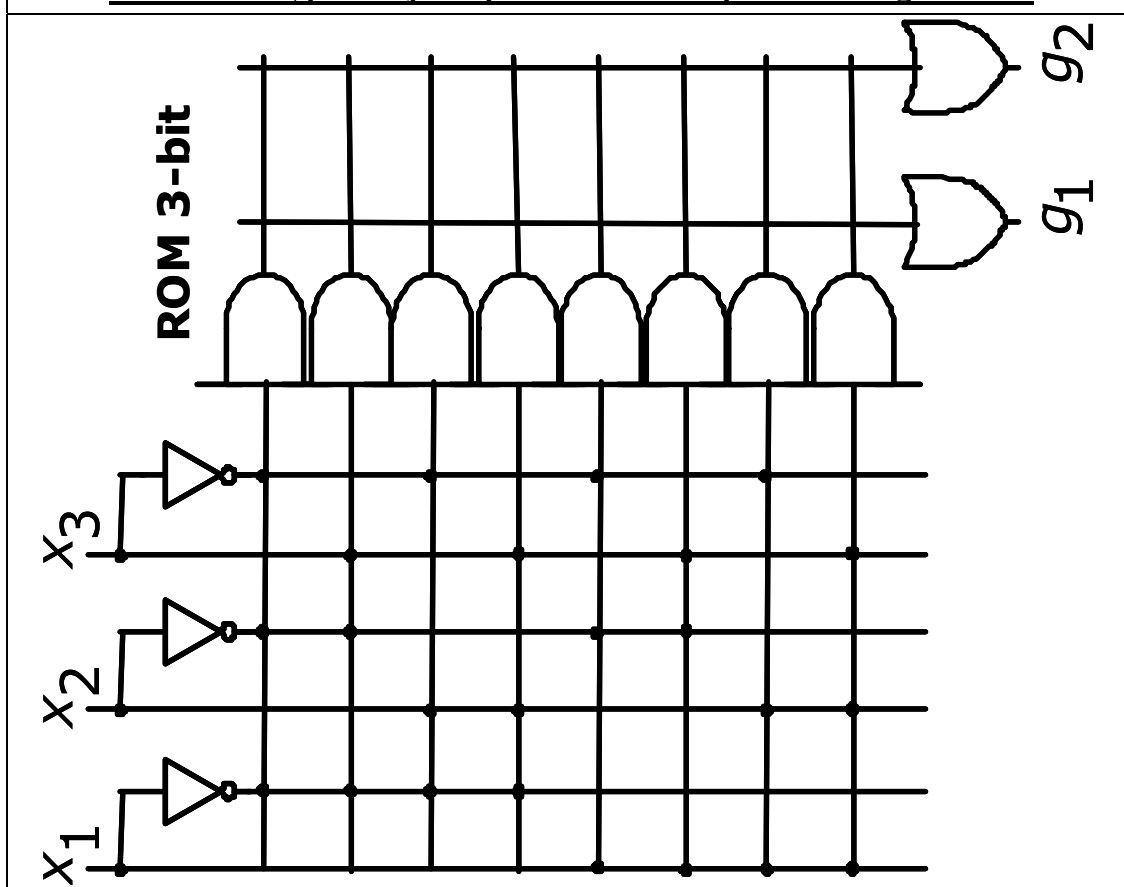
Čas pisanja je 60 minut. Vsaka naloga je vredna 10 točk.

Na list z rešitvami se podpišite in napišite še vpisno številko ter kateri predmet pišete (VSŠ, UNI).

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



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



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

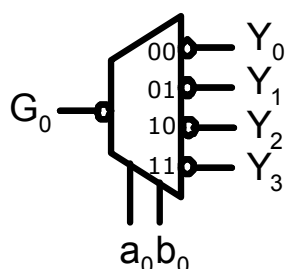
Na list z rešitvami se podpišite in napišite še vpisno številko ter kateri predmet pišete.

Rezultati bodo objavljeni na spletni strani e-študent.

Rešitev 1. naloge:

Delovanje dekoderja 74139<sup>1</sup> povzema spodnja tabela:

$G_0$	$a_0$	$b_0$	$Y_0$	$Y_1$	$Y_2$	$Y_3$
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0



Funkcijo  $f$  narišemo v Veitchev diagram, da si jo lažje predstavljamo:

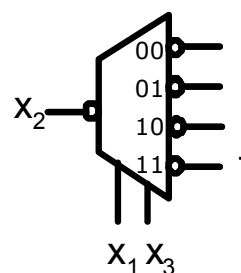
	$x_1$			
$x_2$	X	1	X	1
	1	0	1	X
	$x_3$			

Čim imamo na voljo dekodirnik z ENABLE vhomom v negativni logiki preverimo ali obstaja spremenljivka v osnovni ali negirani obliki, pri kateri so vsa polja enaka '1' vključno z redundancami X.

V zgornjem Veitchevem diagramu je to spremenljivka  $x_2$ : Namreč, če je  $x_2=1$ , potem lahko vse redundance izberemo tako, da bo  $f=1$  za vse vrednosti  $x_2=1$ . Ko določimo spremenljivko za omogočenje elementa ( $G$ ), opazujemo samo preostali del Veitchevega diagrama. Spodnje 4 vrednosti diagrama narišemo v novem Veitchevem diagramu 2 spremenljivk.

	$x_1$	
$x_3$	0	1
	1	X

Dekoder ima aktivno nizke izhode, zato iz nastalega diagrama realiziramo *negacijo* funkcije  $f$  zato v Veitchevem diagramu združujemo ničle, kar nastopa recimo v primeru ko sta  $x_1=1$  in  $x_3=1$ . Obstajata tudi rešitev, ko je  $x_1$  negiran na vhodu za omogočanje.



<sup>1</sup><http://www.alldatasheet.com/view.jsp?Searchword=74HC139>

Rešitev 2. naloge:

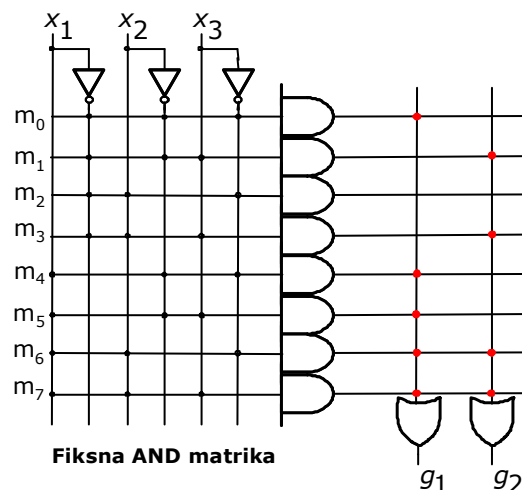
Če se funkcije ne nahajajo v popolni disjunktivni normalni obliki (PDNO), jih prevedemo v to obliko z uporabo pravil Boole-ove algebre. Funkcijo lahko tudi izpišemo v Veitch-ev diagram in izpišemo številke mintermov, kjer je funkcija enaka '1'.

$$\begin{aligned} g_1(x_1, x_2, x_3) &= x_1 + \overline{x_2} \cdot \overline{x_3} = x_1 \cdot (\overline{x_2} \cdot \overline{x_3} + x_2 \cdot \overline{x_3} + \overline{x_2} \cdot x_3 + x_2 \cdot x_3) + (\overline{x_1} + x_1) \cdot \overline{x_2} \cdot \overline{x_3} \\ g_1(x_1, x_2, x_3) &= x_1 \cdot \overline{x_2} \cdot \overline{x_3} + x_1 \cdot x_2 \cdot \overline{x_3} + x_1 \cdot \overline{x_2} \cdot x_3 + x_1 \cdot x_2 \cdot x_3 + \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} + \overline{x_1} \cdot x_2 \cdot \overline{x_3} \\ g_1(x_1, x_2, x_3) &= x_1 \cdot \overline{x_2} \cdot \overline{x_3} + x_1 \cdot x_2 \cdot \overline{x_3} + x_1 \cdot \overline{x_2} \cdot x_3 + x_1 \cdot x_2 \cdot x_3 + \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \\ g_1(x_1, x_2, x_3) &= V(4, 6, 5, 7, 0) \end{aligned}$$

Podobno storimo še za preostale funkcije:

$$\begin{aligned} g_2 &= \overline{x_1} \cdot x_3 + x_1 \cdot x_2 = \overline{x_1} \cdot (\overline{x_2} + x_2) \cdot x_3 + x_1 \cdot x_2 \cdot (\overline{x_3} + x_3) \\ g_2(x_1, x_2, x_3) &= \overline{x_1} \cdot \overline{x_2} \cdot x_3 + \overline{x_1} \cdot x_2 \cdot x_3 + x_1 \cdot x_2 \cdot \overline{x_3} + x_1 \cdot x_2 \cdot x_3 \\ g_2(x_1, x_2, x_3) &= V(1, 3, 6, 7) \end{aligned}$$

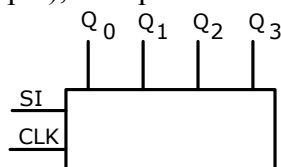
PDNO je najprimernejša oblika za realizacijo z ROM, ker je matrika AND fiksna. Programirane vrednosti AND matrike predstavljajo vse minterme funkcije treh spremenljivk ( $x_1 \ x_2 \ x_3$ ) od  $m_0$  do  $m_7$ . Številka minterma določa naslov lokacije ROM pomnilnika.



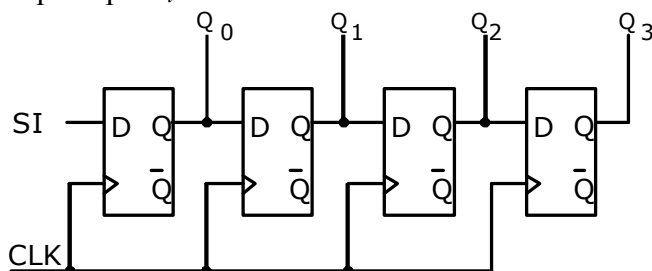
Narišemo celotno vezje ROM strukture in vstavimo pike (●) v OR matriki tam, kjer želimo programirati določeno spremenljivko v členu PDNO.

### Rešitev 3. naloge:

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



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

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

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

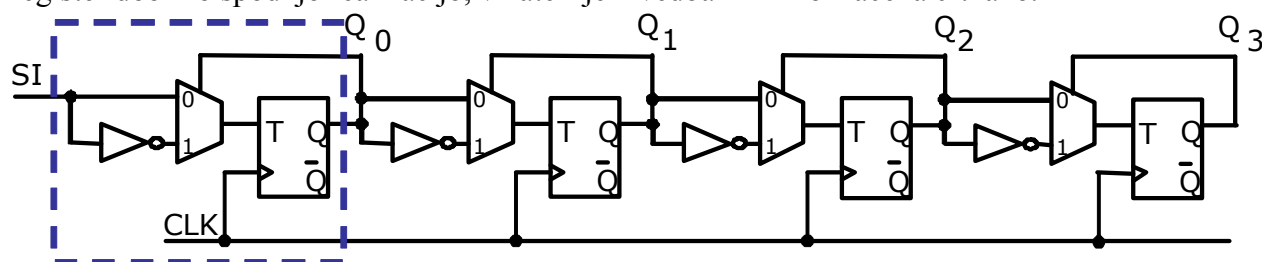
Iz tabele sledi, da je  $T$  vhod XOR operacija  $Q(t)$  in vhoda D-FF, ki ga realiziramo.

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

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

$x$	$f$
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.



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 spletni strani e-študent.

Rešitev 4. naloge:

V prvi iteraciji zberemo skupaj vsa stanja v enem razdelku:  $P_1 = (ABCDEFGG)$

Trenutno stanje	Naslednje stanje		Izhod z
	w=0	w=1	
A	B	C	1
B	D	F	1
C	F	E	0
D	B	G	1
E	F	C	0
F	E	D	0
G	F	G	0

Naslednja iteracija loči stanja, ki imajo različne izhode:  $P_2 = (ABD)(CEFG)$

- Pregledamo vsa naslednja stanja pri vhodu 0 in 1 v vsakem bloku:
    - Blok (ABD):
      - Naslednja stanja pri w=0 (BDB)
      - Naslednja stanja pri w=1 (CFG)
    - Blok (CEFG):
      - Naslednja stanja pri w=0 (FFEF)
      - Naslednja stanja pri w=1 (ECDG)
- Vsa stanja niso v enem bloku. Problem je pri stanju F, ki ima naslednje stanje D. Zato bo stanje F NEEKVIVALENTNO ostalim CEG.
- Novo stanje F zato postavimo v svojo skupino.

Naslednja iteracija loči stanje F od ostalih  $P_3 = (ABD)(CEG)(F)$

- Blok (ABD):
  - Naslednja stanja pri w=0 (BDB)  
So vsa v istem bloku
  - Naslednja stanja pri w=1 (CFG) Niso v istem bloku, ker je F v drugem bloku kot C in G. Zato bo stanje B v novem bloku.
- Blok (CEG):
  - Naslednja stanja pri w=0 (FFF)
  - Naslednja stanja pri w=1 (ECG) C, E in G imamo lahko še vedno za ekvivalentna

Trenutno stanje	Naslednje stanje		Izhod z
	w=0	w=1	
A	B	C	1
B	D	F	1
C	F	E	0
D	B	G	1
E	F	C	0

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 spletni strani e-študent.

<i>F</i>	<i>E</i>	<i>D</i>	<i>0</i>
<i>G</i>	<i>F</i>	<i>G</i>	<i>0</i>

Naslednja iteracija loči stanje B od ostalih  $P_4=(AD)(B)(CEG)(F)$

- Blok (*AD*)
  - Naslednja stanja pri  $w=0$  (*BB*)
  - Naslednja stanja pri  $w=1$  (*CG*)
  - So vsa v istem bloku.
- Blok (*CEG*)
  - Naslednja stanja pri  $w=0$  (*FFF*)
  - Naslednja stanja pri  $w=1$  (*ECG*) So vsa v istem bloku.

Trenutno stanje	Naslednje stanje		Izhod z
	w=0	w=1	
<i>A</i>	<i>B</i>	<i>C</i>	<i>1</i>
<i>B</i>	<i>D</i>	<i>F</i>	<i>1</i>
<i>C</i>	<i>F</i>	<i>E</i>	<i>0</i>
<i>D</i>	<i>B</i>	<i>G</i>	<i>1</i>
<i>E</i>	<i>F</i>	<i>C</i>	<i>0</i>
<i>F</i>	<i>E</i>	<i>D</i>	<i>0</i>
<i>G</i>	<i>F</i>	<i>G</i>	<i>0</i>

$P_5=(AD)(B)(CEG)(F)$

Iteraciji  $P_5$  in  $P_4$  sta enaki, zato se postopek minimizacije zaključi. Stanji *A* in *D* sta ekvivalentni. Stanja *C*, *E* in *G* so ekvivalentna.

- Tabelo stanj zapišemo na novo
- Izbrišemo vrstice za *D*, *E* in *G*
- Zamenjamo stanja:  $D \rightarrow A$  in vse  $E \rightarrow C$  ter  $G \rightarrow C$

Rezultat je nova tabela stanj minimiziranega avtomata:

Trenutno stanje	Naslednje stanje		Izhod z
	w=0	w=1	
<i>A</i>	<i>B</i>	<i>C</i>	<i>1</i>
<i>B</i>	<i>A</i>	<i>F</i>	<i>1</i>
<i>C</i>	<i>F</i>	<i>C</i>	<i>0</i>
<i>F</i>	<i>C</i>	<i>A</i>	<i>0</i>

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 spletni strani e-študent.