

Dobio sam 1.87 i neznam šta mi je tačno, a šta nije, ali možda nekom pomogne.

1.

Zadan je sljedeći VHDL kod:

```
entity Element is port(  
    clock, Q, R, S, T: in std_logic;  
    Qout: out std_logic);  
end Element;  
  
architecture beh of Element is  
    signal Qint: std_logic;  
begin  
  
    process(...)  
        variable sel: std_logic_vector(1 downto 0);  
    begin  
        if Q='1' then  
            Qint<= '0';  
        elsif R='0' then  
            Qint<= '1';  
        elsif falling_edge(clock) then  
            sel:=S&T;  
            case sel is  
                when "00"=> Qint<= '0';  
                when "01"=> Qint<= '1';  
                when "10"=> Qint<= Qint;  
                when "11"=> Qint<= Qint;  
                when others=> null;  
            end case;  
        end if;  
        Qout<= Qint;  
    end process;  
  
end beh;
```

Označite sve signale koji čine minimalnu listu osjetljivosti.

- ☒ Qint
 - ☒ R
 - ☒ Q
 - ☐ T
 - ☐ S
 - ☒ clock
-

2.

Zadan je sljedeći VHDL kod:

```
entity Element is port(  
    cp, G, H: in std_logic;  
    Qout: out std_logic);  
end Element;  
  
architecture beh of Element is  
    signal Qint: std_logic;  
begin  
  
    process(cp, G)  
    begin  
        if rising_edge(cp) then  
            if H= '0' then  
                Qint<= not Qint;  
            end if;  
        end if;  
        if G='0' then  
            Qint<= '1';  
        end if;  
    end process;  
  
    Qout<= Qint;  
  
end beh;
```

Označite sve ulaze bistabila koji djeluju sinkrono. ☐ G ☒ H

3.

Zadan je sljedeći VHDL kod:

```
entity Element is port(  
    clock, V, W, X, Y: in std_logic;  
    Qout: out std_logic);  
end Element;  
  
architecture beh of Element is  
    signal Qint: std_logic;  
begin  
  
    process(clock, V, W)  
        variable sel: std_logic_vector(1 downto 0);  
    begin  
        if V='1' then  
            Qint<= '0';  
        elsif W='0' then  
            Qint<= '1';  
        elsif falling_edge(clock) then  
            sel:=X&Y;  
            case sel is  
                when "00"=> Qint<= '0';  
                when "01"=> Qint<= Qint;  
                when "10"=> Qint<= '0';  
                when "11"=> Qint<= '0';  
                when others=> null;  
            end case;  
        end if;  
    end process;  
  
    Qout<= Qint;  
  
end beh;
```

Označite asinkroni ulaz najvišeg prioriteta.

- ☐ W
 - ☐ X
 - ☐ Y
 - ☒ V
-

4.

Zadan je sljedeći VHDL kod:

```
entity Element is port(  
    clock, F, G, H: in std_logic;  
    Qout: out std_logic);  
end Element;  
  
architecture beh of Element is  
    signal Qint: std_logic;  
begin  
  
    process(clock, F, G)  
    begin  
        if F='0' then  
            Qint<= '1';  
        elsif G='0' then  
            Qint<= '1';  
        elsif rising_edge(clock) then  
            if H= '0' then  
                Qint<= not Qint;  
            end if;  
        end if;  
    end process;  
  
    Qout<= Qint;  
  
end beh;
```

Na što djeluje ulaz F?

- ☐ na padajući brid
 - ☒ na logičku razinu 0
 - ☐ na rastući brid
 - ☐ na logičku razinu 1
-

5.

Stroj s konačnim brojem stanja zadan je tablicom u nastavku. Stroj ima jedan jednobitni ulaz, te jedan 3-bitni izlaz.

Trenutno stanje	Pobuda U	Sljedeće stanje	Izlaz
S2	0	S4	2
S2	1	S3	3
S5	0	S3	4
S5	1	S3	5
S0	0	S5	0
S0	1	S5	7
S1	0	S6	2
S1	1	S5	0
S3	0	S6	0
S3	1	S1	2
S4	0	S2	0
S4	1	S6	7
S7	0	S2	2
S7	1	S7	4
S6	0	S2	4
S6	1	S0	5

Za realizaciju tog stroja na raspolaganju su 3 JK bistabila. Pri tome se svako stanje kodira prema tablici prikazanoj u nastavku.

Stanje	Kod stanja		
	Q2	Q1	Q0
S0	0	1	0
S1	0	1	1
S2	0	0	0
S3	1	0	0
S4	1	0	1
S5	0	0	1
S6	1	1	1
S7	1	1	0

Projektirajte taj sklop uporabom zadanih bistabila. Nije dozvoljeno obavljati minimizaciju broja stanja stroja (naime, iako se ovo uobičajeno čini prilikom projektiranja, za potrebe strojnog ocjenjivanja u ovom se zadatku ne smije). U polja za unos rješenja za svaki ulaz bistabila, te za svaki izlaz stroja unesite minimizirani algebarski zapis funkcije.

Prilikom unosa algebarskog oblika za stanja bistabila koristiti oznake Qj (gdje je j broj bistabila; npr. Q2), odnosno U za pobudu. Primjer jednog takvog rješenja:

$Q2$ and not $Q1$ and U or not $Q0$ and not U . Prilikom očitavanja izlaza stroja bit izlaza $O0$ promatran je kao bit najmanje težine.

B2.J $\text{not } Q1 \text{ or } (Q0 \text{ and not } U)$

B2.K $(Q1 \text{ and } Q0) \text{ or } (Q1 \text{ and not } U) \text{ or } (Q0 \text{ and not } U) \text{ or } (\text{not } Q1 \text{ and not } Q0 \text{ and } U)$

B1.J $(Q2 \text{ and } U) \text{ or } (Q2 \text{ and not } Q0)$

B1.K $(Q2 \text{ and not } U) \text{ or } (\text{not } Q2 \text{ and } U) \text{ or } (\text{not } Q0 \text{ and not } U)$

B0.J $(Q2 \text{ and not } Q1) \text{ or } (\text{not } Q2 \text{ and } Q1) \text{ or } (\text{not } Q1 \text{ and not } U)$

B0.K $(Q2 \text{ and } Q1) \text{ or } (Q2 \text{ and not } U) \text{ or } (\text{not } Q2 \text{ and not } Q1)$

$O2$ $(Q2 \text{ and } Q1 \text{ and } Q0) \text{ or } (Q2 \text{ and } Q0 \text{ and } U) \text{ or } (Q1 \text{ and not } Q0 \text{ and } U) \text{ or } (\text{not } Q2 \text{ and not } Q1 \text{ and not } Q0 \text{ and } U)$

$O1$ $(Q2 \text{ and not } Q1 \text{ and } U) \text{ or } (\text{not } Q2 \text{ and not } Q0 \text{ and } U) \text{ or } (\text{not } Q2 \text{ and not } Q1 \text{ and not } Q0 \text{ and } U)$

$O0$ $(Q2 \text{ and } Q0 \text{ and } U) \text{ or } (\text{not } Q1 \text{ and } Q0 \text{ and } U) \text{ or } (\text{not } Q2 \text{ and not } Q0 \text{ and } U)$

6.

Stroj s konačnim brojem stanja zadan je tablicom u nastavku. Stroj ima jedan 1-bitni ulaz, te jedan 1-bitni izlaz.

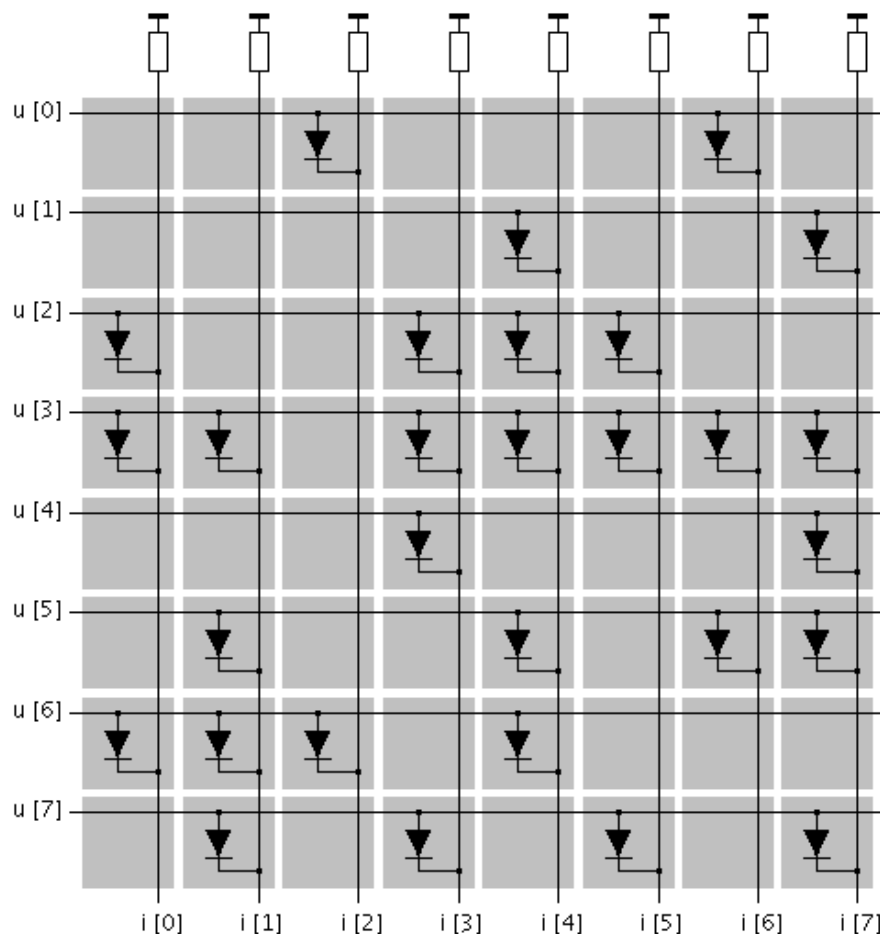
Trenutno stanje	Pobuda U	Sljedeće stanje	Izlaz
S4	0	S5	0
S4	1	S5	0
S6	0	S2	1
S6	1	S3	0
S0	0	S3	0
S0	1	S3	0
S2	0	S6	0
S2	1	S4	1
S1	0	S6	1
S1	1	S6	0
S7	0	S5	1
S7	1	S1	0
S3	0	S7	0
S3	1	S3	0
S5	0	S6	0
S5	1	S4	1

Provjerite je li zadani stroj s konačnim brojem stanja Mealyjev ili Mooreov? Dobro razmislite o vašem odgovoru!

- ☐ Stroj s konačnim brojem stanja je Mealyjev, ali mu izlaz ne ovisi o trenutnim ulazima
- ☐ Stroj s konačnim brojem stanja je Mooreov, ali mu izlaz ovisi i o trenutnim ulazima
- ☐ Stroj s konačnim brojem stanja je Mooreov
- ☒ Stroj s konačnim brojem stanja je Mealyjev

7.

Permanentna memorija je zadana slikom.

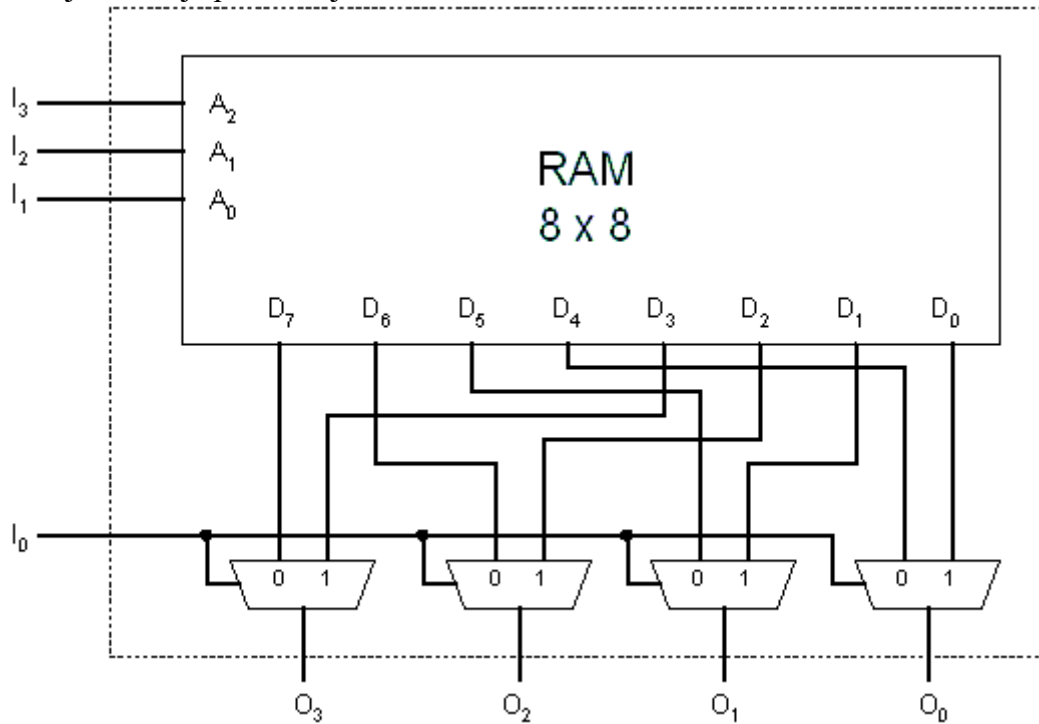


Očitajte sadržaj memorije po lokacijama. U polja za unos rješenja potrebno je unijeti vrijednost memorijske lokacije u heksadekaskom obliku (kao dvije heksadekadske znamenke); npr. E8 ili 2F. Pri tome bit i[0] tretirajte kao bit najveće težine.

Lokacija 0	<input type="text" value="22"/>
Lokacija 1	<input type="text" value="09"/>
Lokacija 2	<input type="text" value="9C"/>
Lokacija 3	<input type="text" value="DF"/>
Lokacija 4	<input type="text" value="11"/>
Lokacija 5	<input type="text" value="4B"/>
Lokacija 6	<input type="text" value="E8"/>
Lokacija	<input type="text" value="55"/>

8.

Zadana je uređena n-torka $P=(9, 2, 15, 8, 15, 15, 9, 10, 9, 13, 15, 11, 11, 0, 3, 8)$. Funkcija $F(i)$ vraća i -ti element od P (npr. $F(1) = 2$). Projektirati sklop koji ostvaruje ovu funkciju. Na raspolaganju je ispisna memorija 8×8 te 4 multipleksora 2×1 , spojenih prema slici. Prikazati sadržaj memorije po lokacijama, u heksadekadskom obliku.



Lokacija	0	92
Lokacija	1	F8
Lokacija	2	FF
Lokacija	3	9A
Lokacija	4	9D
Lokacija	5	FB
Lokacija	6	B0
Lokacija	7	38

Važna napomena: U svaki redak unijeti odgovarajuću vrijednost memorijske lokacije prikazane kao dvoznamenkasti heksadekadski broj. Primjerice, ako je $D7...D0 = 10110001$, tada je potrebno upisati B1. Unos nedvoznamenkastih brojeva neće biti prihvaćen.

9.

Zadana je funkcija $f(A,B,C,D) = m(1, 2, 3, 6, 7, 8, 10, 11, 12, 13, 15)$. Koliko ta funkcija ima **primarnih implikanata/minimalnih oblika**?

- ☐ 4/1
- ☐ 15/3
- ☒ 8/2
- ☐ 2/3

10.

Prilikom minimizacije Mooreovog stroja s konačnim brojem stanja, u nekom koraku analiziraju se dva stanja: S2 i S3. Dio tablice koji se odnosi na ta dva stanja prikazan je u nastavku.

Trenutno stanje	Pobuda	Sljedeće stanje	Trenutni izlaz
S2	0	S8	0
S2	1	S6	0
S3	0	S8	0
S3	1	S6	0

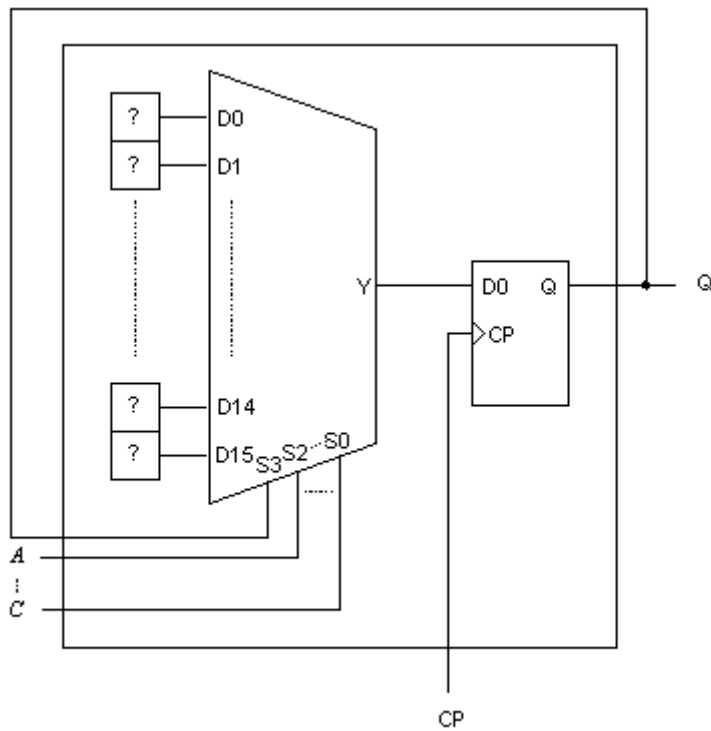
Što možemo sa sigurnošću zaključiti o ta dva stanja na temelju ovdje prikazanih podataka?

- ☐ stanja su možda ekvivalentna
- ☒ stanja nisu ekvivalentna
- ☐ stanja su ekvivalentna

11.

Na raspolaganju je 4-ulazni LUT s D bistabilom, prikazan slikom. Programirati taj bistabil tako da se dobije bistabil čija je funkcija opisana sljedećom tablicom.

A	B	C	Q_{n+1}
0	0	0	not Q_n
0	0	1	Q_n
0	1	0	not Q_n
0	1	1	Q_n
1	0	0	Q_n
1	0	1	Q_n
1	1	0	Q_n
1	1	1	not Q_n



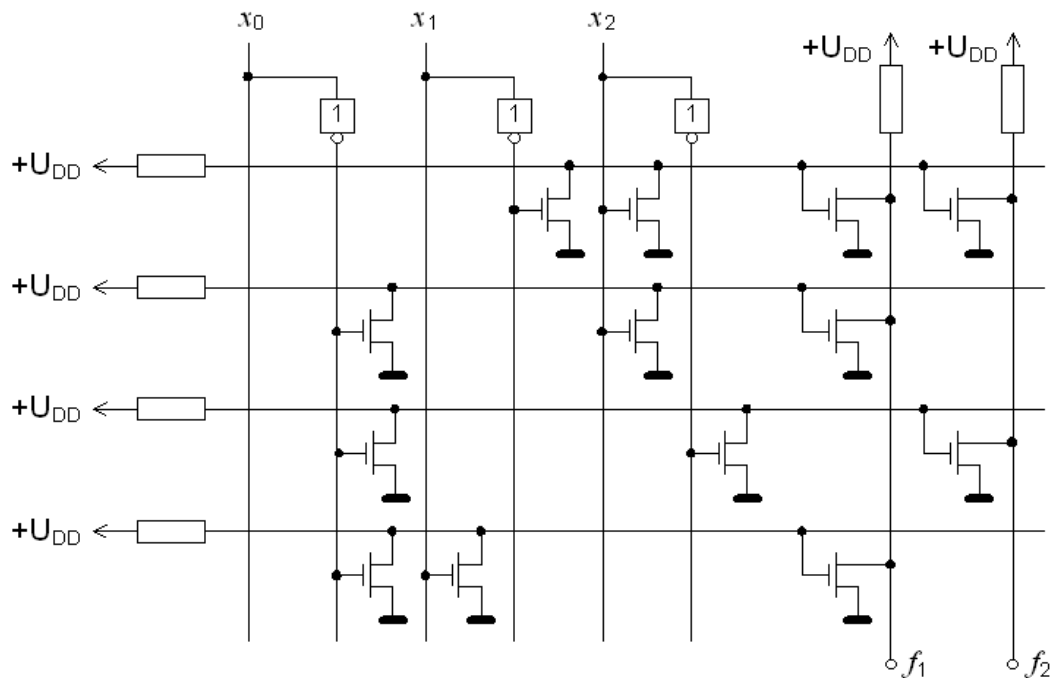
LUT_0	1
LUT_1	0
LUT_2	1
LUT_3	0
LUT_4	0
LUT_5	0
LUT_6	0

LUT_7	1
LUT_8	0
LUT_9	1
LUT_10	0
LUT_11	1
LUT_12	1
LUT_13	1
LUT_14	1
LUT_15	0

Važna napomena: U svako polje za unos treba unijeti samo vrijednost pripadnog ulaza luta za navedenu funkciju. Dozvoljena su dva načina upisa vrijednosti: 1 se tretira jednako kao i true, a 0 je ekvivalentna sa false. Bilo koji oblik redundantnog unosa (dupliciranje nula, proširivanje jedinice sa vodećom nulom) povlači netočnost unosa.

12.

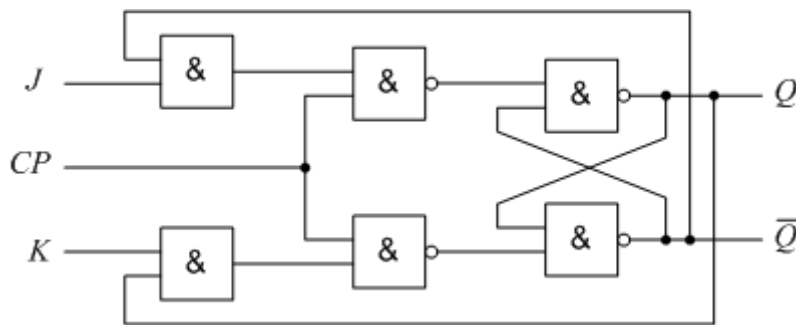
PLA strukturom u tehnologiji MOSFET ostvarene su funkcije f_1 i f_2 . O kojim se funkcijama radi?



- ☐ $f_1 = (x_1 \text{ AND } x_0) \text{ OR } (\text{NOT } x_1 \text{ AND } \text{NOT } x_0)$, $f_2 = (x_1 \text{ AND } \text{NOT } x_0) \text{ OR } (\text{NOT } x_2 \text{ AND } \text{NOT } x_1)$
- ☒ $f_1 = (x_2 \text{ AND } x_1) \text{ OR } (\text{NOT } x_1 \text{ AND } \text{NOT } x_0)$, $f_2 = (x_2 \text{ AND } \text{NOT } x_0) \text{ OR } (\text{NOT } x_2 \text{ AND } \text{NOT } x_1)$
- ☐ $f_1 = (x_2 \text{ AND } x_1) \text{ OR } (\text{NOT } x_1 \text{ AND } \text{NOT } x_0)$, $f_2 = (x_1 \text{ AND } \text{NOT } x_0) \text{ OR } (\text{NOT } x_2 \text{ AND } \text{NOT } x_1)$
- ☐ $f_1 = (x_1 \text{ AND } x_0) \text{ OR } (\text{NOT } x_1 \text{ AND } \text{NOT } x_0)$, $f_2 = (x_2 \text{ AND } \text{NOT } x_0) \text{ OR } (\text{NOT } x_2 \text{ AND } \text{NOT } x_1)$

13.

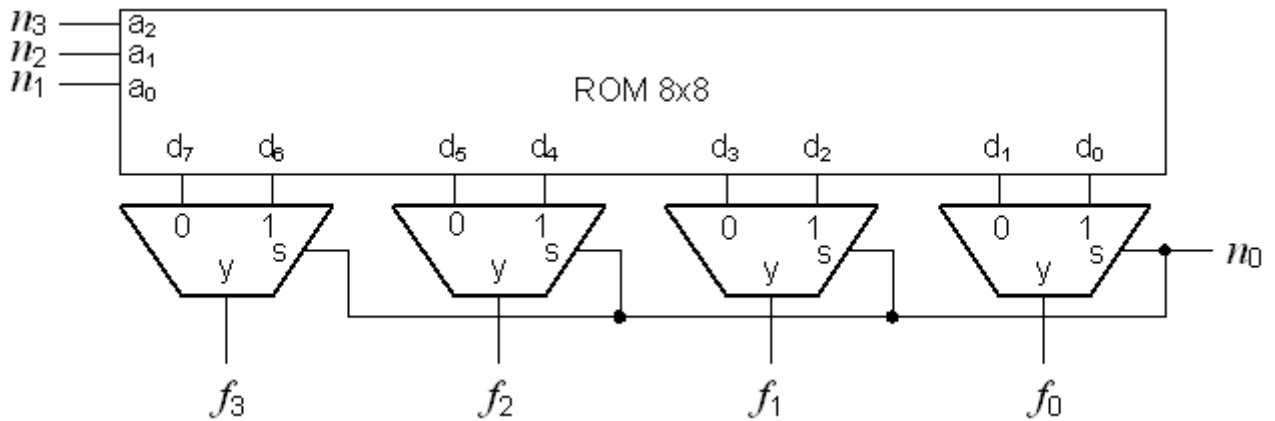
Na raspolaganju je SR-bistabil izveden s 4 sklopa NI i ulazom za signal takta. Taj je bistabil pomoću dva sklopa I pretvoren u JK-bistabil (vidi sliku). Ako sve ulaze ovog bistabila (J,K,CP) trajno spojimo na logičko 1, na izlazu ćemo primijetiti oscilacije. Ukoliko označimo s TL trajanje logičke 0, s TH trajanje logičke 1, a kašnjenje svakog logičkog sklopa iznosi 25ns, odredite iznose vremena TH i TL. Podrazumijevana mjerna jedinica je ns.



TL	50
TH	100

14.

Funkcija $f(n)$ svakom n iz skupa $\{0, \dots, 15\}$ pridružuje broj $(n \text{ XOR } m)$, gdje je m jednak broju n zarotiranom udesno za jedan bit. Ovu funkciju potrebno je ostvariti permanentnom memorijom kapaciteta 8×8 , i multipleksorima, prema slici. Što treba upisati u memoriju? U ponuđenim rješenjima prikazan je sadržaj memorijskih lokacija od 4 do 7, u heksadekadskom zapisu.



- ☒ b1, be, 8d, 82
- ☐ b2, bd, 8e, 81
- ☐ c6, fa, 36, 0a
- ☐ f1, fe, cd, c2

15.

Dvoulazni NI sklop modeliran je VHDL-om kao sklop nand2. Potom je napisan strukturni model sklopa sklop1. Sučelje sklopa nand2 te model sklopa sklop1 prikazani su u nastavku.

```
ENTITY nand2 IS
  PORT (
    a : OUT std_logic;
    b,c : IN std_logic);
END nand2;

ENTITY sklop1 IS PORT (d, e : IN std_logic;
  f : OUT std_logic);
END sklop1;
ARCHITECTURE ar OF sklop1 IS
  SIGNAL i : std_logic;
BEGIN
  s1: ENTITY work.nand2 PORT MAP (b<=i, c<=e, a<=f);
  s2: ENTITY work.nand2 PORT MAP (i,e,d);
END ar;
```

Ponašajni opis istovjetan opisu "ar" u tijelu bloka ARCHITECTURE sadržava sljedeći izraz:

- ☐ f <= NOT (d AND e) AND e;
- ☒ f <= NOT e OR d;
- ☐ f <= NOT d AND NOT e;
- ☐ f <= d OR e;