

ETF U BEOGRADU, KATEDRA ZA ELEKTRONIKU

Predmet: OSNOVI DIGITALNE ELEKTRONIKE

OCENA \_\_\_\_\_

Ispit: 10.10.2010.

Odgovorni nastavnik i asistenti: Dragan Vasiljević, Goran Savić i Lazar Karbunar

DEŽURNI:

KANDIDAT:

Sala \_\_\_\_\_  
Vreme početka \_\_\_\_\_  
Vreme završetka \_\_\_\_\_  
Potpis \_\_\_\_\_

Ime \_\_\_\_\_  
Prezime \_\_\_\_\_  
Broj indeksa \_\_\_\_\_  
Potpis \_\_\_\_\_

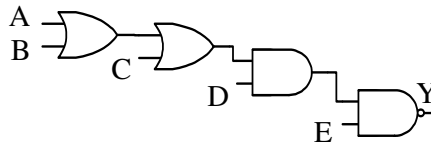
USLOVI ISPITA

1. Trajanje ispita 180 minuta.
2. Ispit se polaže na formularu.
3. Dozvoljeni su kalkulator i olovka.
4. Ocenjuju se rad kandidata i razumevanje gradiva.
5. Traži se koncizan, jasan, čitak odgovor napisan u predviđenom prostoru (linija, boks, crtež).

OCENJIVANJE

R.Br.	1	2	3	4	Total
Max	<b>20</b>	<b>20</b>	<b>20</b>	<b>20</b>	<b>80</b>
Dobijeno					

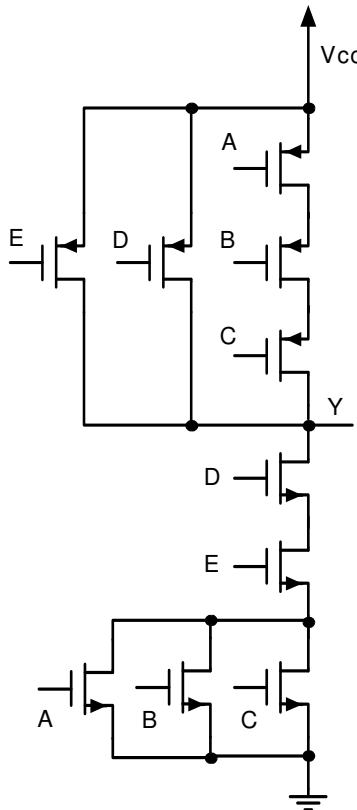
1. [20] a) Koristeći minimalan broj NMOS i PMOS tranzistora, isprojektovati najprostije statičko CMOS kolo koje realizuje bulovu funkciju kao kolo sa slike.



b) Ukoliko se na izlaz isprojektovanog CMOS kola iz tačke a) veže kondenzator kapacitivnosti  $C$ , merenjem se može ustanoviti da vremenska konstanta punjena tog kondenzatora u najboljem slučaju (kondenzator se najbrže puni) iznosi  $\tau_1$ . Ukoliko svi NMOS i PMOS tranzistori imaju istu otpornost kada su uključeni, na osnovu rezultata ovog merenja odrediti tu otpornost.

### Rešenje:

Analizom prikazane šeme se može utvrditi da je funkcija koju obavlja kombinaciona mreža  $Y = \overline{(A + B + C)} \cdot D \cdot E$ . Kombinacijom elementarnih NI, NILI struktura dobija se tražena realizacija.



b) Kondenzator se najbrže puni kada su svi ulazi logičkog kola na logičkoj nuli. U tom slučaju je ekvivalentna otpornost kojom je kondenzator vezan na  $V_{cc}$  napon  $R_e = R \parallel R \parallel 3R = \frac{3}{7}R$ , najmanja.

Vremenska konstanta u tom slučaju je  $\tau_1 = \frac{3}{7}RC$ , pa je na osnovu toga  $R = \frac{7}{3} \frac{\tau_1}{C}$ , što predstavlja traženu otpornost uključenih tranzistora.

2. Logičku funkciju  $F = \overline{C} \cdot B \cdot A + C \cdot \overline{A} + C \cdot \overline{B}$  realizovati korišćenjem:

a) [10] Multipleksera 4/1 bez dodatnih logičkih kola;

b) [10] Demultipleksera 1/8 i minimalnog broja potrebnih logičkih kola.

Smatrati da su ulazi i izlazi multipleksera i demultipleksera aktivni na logičkoj jedinici.

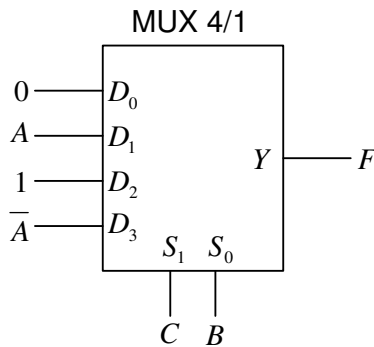
**Rešenje:**

a) Data logička funkcija se primenom pravila Bulove algebre može transformisati na sledeći način:

$$F = \overline{C} \cdot B \cdot A + C \cdot \overline{A} + C \cdot \overline{B} = \overline{C} \cdot B \cdot A + C \cdot \overline{B} + C \cdot \overline{A} \cdot (B + \overline{B}) = \overline{C} \cdot B \cdot A + C \cdot \overline{B} + C \cdot B \cdot \overline{A}$$

$$F = \overline{C} \cdot B \cdot A + C \cdot \overline{B} \cdot 1 + C \cdot B \cdot \overline{A} + \overline{C} \cdot \overline{B} \cdot 0$$

Ako se na selekzione ulaze multipleksera dovedu signali  $B$  i  $C$  tražena realizacija ima izgled kao na sledećoj slici:



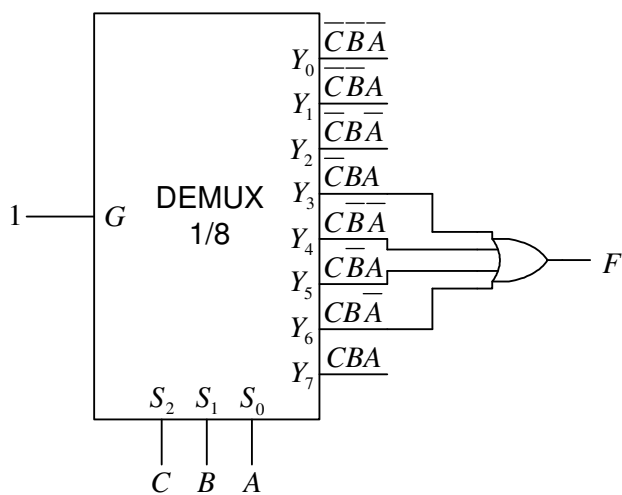
b) Ako se data logička funkcija predstavi preko sume potpunih logičkih proizvoda, imaće sledeći oblik:

$$F = \overline{C} \cdot B \cdot A + C \cdot \overline{A} + C \cdot \overline{B} = \overline{C} \cdot B \cdot A + C \cdot \overline{B} \cdot (A + \overline{A}) + C \cdot \overline{A} \cdot (B + \overline{B})$$

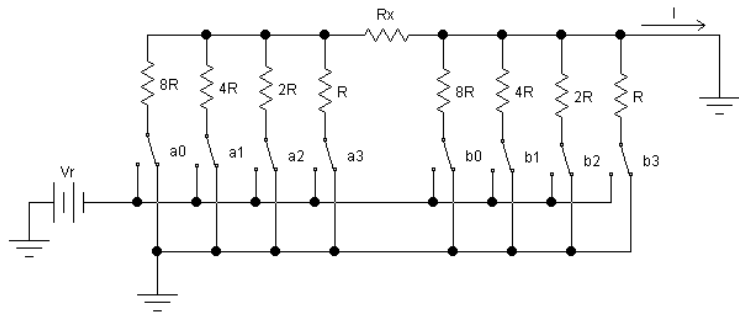
$$F = \overline{C} \cdot B \cdot A + C \cdot \overline{B} \cdot A + C \cdot \overline{B} \cdot \overline{A} + C \cdot B \cdot \overline{A} + C \cdot \overline{B} \cdot \overline{A} = \overline{C} \cdot B \cdot A + C \cdot \overline{B} \cdot A + C \cdot \overline{B} \cdot \overline{A} + C \cdot B \cdot \overline{A}$$

koji je pogodan za realizaciju pomoću demultipleksera.

Ako se na ulaz za podatke demultipleksera dovede logička jedinica, a na kontrolne ulaze dovedu promenljive  $A$ ,  $B$  i  $C$ , na izlazima demultipleksera će biti formirani svi potpuni logički proizvodi promenljivih  $A$ ,  $B$  i  $C$ . Logičkim sabiranjem odgovarajućih potpunih proizvoda, realizuje se data logička funkcija. Odgovarajuća realizacija je prikazana na sledećoj slici:



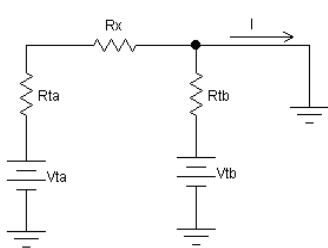
3. [20] Na slici je prikazana šema koja se koristi kao sastavni deo  $D/A$  konvertora sa 8 ulaza. Struja  $I$  predstavlja analogni ekvivalent digitalne vrednosti  $D_n$  koja je zapisana u binarno kodovanom decimalnom sistemu (NBCD) sa dve decimalne cifre  $a$ -lakša i  $b$ -teža cifra:  $I_i = kD_n$ .



- Odrediti vrednost otpornika  $R_x$  tako da se konvertor može koristiti u prirodnom BCD kodu. Ako je vrednost binarne promenljive 1, ( $a_i = 1, b_i = 1$ ) prekidač je spojen na referentni napon  $V_r$ , u suprotnom ( $a_i = 0, b_i = 0$ ), prekidač je na masi.
- Odrediti vrednost konstante  $k$  u izrazu za  $I_i$
- Uz pomoć operacionog pojačavača, minimalnog broja otpornika čije vrednosti je potrebno odrediti i prikazane lestvičaste strukture (sa izračunatom vrednošću  $R_x$ ) izvršiti sintezu mreže koja će na svom izlazu davati napon  $V_i = V_r D_n$ .

### Rešenje:

a) Naponi koji potiču od odgovarajuće cifre BCD broja se mogu ekvivalentirati odgovarajućim Tevenenovim generatorima  $V_{Ta}$  i  $V_{Tb}$  sa otpornostima  $R_{Ta}$  i  $R_{Tb}$ , kao na slici.



Otpornosti generatora su jednake i iznose  $R_{Ta} = R_{Tb} = \frac{8}{15} R = R_T$ .

Naponi tevenenovih generatora su :

$$V_{Ta} = -\frac{V_r}{15} (a_0 2^0 + a_1 2^1 + a_2 2^2 + a_3 2^3) \quad \text{i}$$

$$V_{Tb} = -\frac{V_r}{15} (b_0 2^0 + b_1 2^1 + b_2 2^2 + b_3 2^3)$$

$$\text{struja } I \text{ se dobija kao } I = \frac{V_{Ta}}{R_{Ta} + R_x} + \frac{V_{Tb}}{R_{Tb}} = \frac{1}{R_T + R_x} (V_{Ta} + (1 + \frac{R_x}{R_T}) V_{Tb}) = -\frac{V_r}{15(R_T + R_x)} (a + (1 + \frac{R_x}{R_T}) b),$$

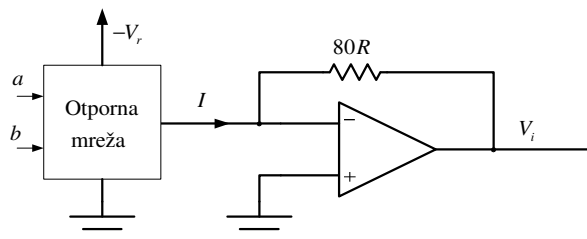
gde su  $a$  i  $b$  decimalne vrednosti binarnih ekvivalenata. Za BCD kod potrebno je da težinski faktor za cifru desetice bude 10 ( $b$  je cifra desetica,  $a$  je cifra jedinica). Odatle je

$$(1 + \frac{R_x}{R_T}) = 10 \Rightarrow R_x = 9R_T = 4.8R. \text{ Izlazna struja u tom slučaju je:}$$

$$I = -V_r(a + 10b) / 150R_T \Rightarrow I = -\frac{V_r}{80R} (a + 10b)$$

b) Vrednost konstante  $k$  iznosi  $k = -\frac{V_r}{80R}$ .

c) Granu otporne mreže sa strujom  $I$  na izlazu je potrebno spojiti na virtuelnu masu operacionog pojačavača kao na slici.

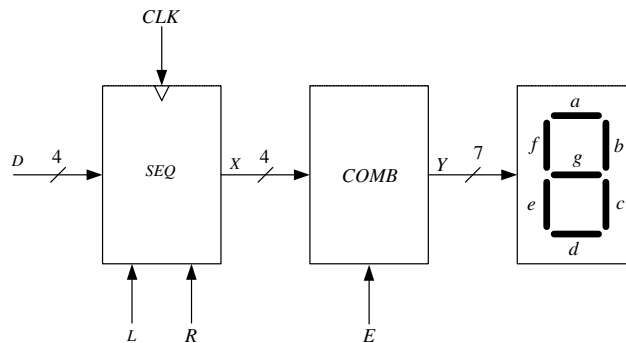


**KANDIDAT:** Ime \_\_\_\_\_ Prezime \_\_\_\_\_ Broj indeksa \_\_\_\_\_  
 Dežurni \_\_\_\_\_

**4.** U digitalnom sistemu na slici sekvencijalna mreža *SEQ* i kombinaciona mreža *COMB* su opisane odgovarajućim VHDL kodovima koji su dati u nastavku ovog teksta. Izlazi kombinacione mreže se koriste za pobudu sedmosegmentnog displeja sa zajedničkom anodom. Biti vektora koji predstavlja izlaz kombinacione mreže počev od bita najveće težine pa do bita najmanje težine pobuđuju redom segmente *a*, *b*, *c*, *d*, *e*, *f*, *g* displeja, respektivno.

**a) [5]** Na osnovu datog VHDL koda odrediti koju funkciju obavlja sekvencijalna mreža *SEQ*, a koju kombinaciona mreža *COMB*.

**b) [15]** Za date vremenske dijagrame signala *CLK*, *R*, *L*, *D* i *E*, odrediti stanja na magistralama *X* i *Y* i upisati ih na dati dijagram u heksadecimalnom formatu. Odrediti za svaku periodu takta *CLK* koji segmenti displeja su upaljeni, a koji ugašeni (upaljene segmente olovkom zatamniti na datoj slici ispod vremenskih dijagrama).



```

LIBRARY IEEE;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;

ENTITY seq IS PORT
(
  clk,r,l   : IN STD_LOGIC;
  d         : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
  x         : OUT STD_LOGIC_VECTOR(3 DOWNTO 0)
);
END seq;

ARCHITECTURE behav OF seq IS
  SIGNAL tmp : UNSIGNED(3 DOWNTO 0);
BEGIN
  PROCESS (clk) BEGIN
    IF rising_edge(clk) THEN
      IF r='1' THEN
        tmp <= (others => '0');
      ELSE
        IF l='1' THEN
          tmp <= UNSIGNED(d);
        ELSE
          tmp <= tmp-1;
        END IF;
      END IF;
    END IF;
  END PROCESS;
  x <= STD_LOGIC_VECTOR(tmp);
END behav;

```

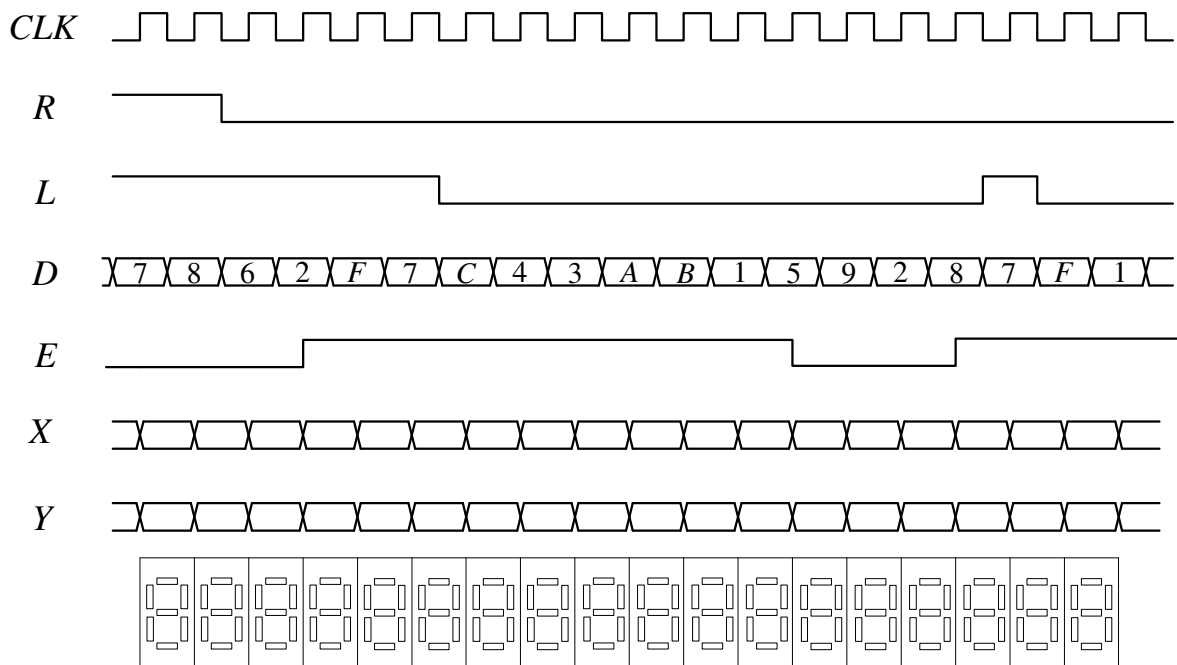
```

LIBRARY IEEE;
USE ieee.std_logic_1164.all;

ENTITY comb IS PORT
(
  e         : IN STD_LOGIC;
  x         : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
  y         : OUT STD_LOGIC_VECTOR(6 DOWNTO 0)
);
END comb;

ARCHITECTURE behav OF comb IS
  SIGNAL tmp : STD_LOGIC_VECTOR(6 DOWNTO 0);
BEGIN
  PROCESS (x,e,tmp) BEGIN
    CASE x IS
      WHEN "0000" => tmp <= "0111000";
      WHEN "0001" => tmp <= "0110000";
      WHEN "0010" => tmp <= "1000010";
      WHEN "0011" => tmp <= "0110001";
      WHEN "0100" => tmp <= "1100000";
      WHEN "0101" => tmp <= "0001000";
      WHEN "0110" => tmp <= "0000100";
      WHEN "0111" => tmp <= "0000000";
      WHEN "1000" => tmp <= "0001111";
      WHEN "1001" => tmp <= "0100000";
      WHEN "1010" => tmp <= "0100100";
      WHEN "1011" => tmp <= "1001100";
      WHEN "1100" => tmp <= "0000110";
      WHEN "1101" => tmp <= "0010010";
      WHEN "1110" => tmp <= "1001111";
      WHEN "1111" => tmp <= "0000001";
    END CASE;
    IF e='1' THEN
      y <= tmp;
    ELSE
      y <= "1111111";
    END IF;
  END PROCESS;
END behav;

```



**Rešenje:**

a) Mreža *SEQ* predstavlja četvorobitni sinhroni brojač unazad sa mogućnošću reseta i mogućnošću paralelnog upisa. Ako je signal *R* na nivou logičke jedinice, sekvencijalna mreža se resetuje nailaskom prve uzlazne ivice signala takta *CLK* (po resetu brojač ulazi u stanje “0000”), bez obzira na vrednosti ostalih ulaznih signala. U protivnom, ako je signal *L* na nivou logičke jedinice u sekvencijalnu mrežu se upisuje četvorobitna binarna vrednost *D* sa ulaza (sinhrono sa uzlaznom ivicom signala takta *CLK*), a ako je signal *L* na nivou logičke nule sekvencijalna mreža radi kao četvorobitni binarni brojač unazad (sinhrono sa uzlaznom ivicom signala takta *CLK*).

Kombinaciona mreža *COMB* predstavlja dekodler koji pri niskom logičkom nivou signala *E* generiše na izlazu sve jedinice (i time obezbeđuje da su svi segmenti displeja ugašeni, jer je u pitanju sedmosegmentni displej sa zajedničkom anodom), a pri visokom logičkom nivou signala *E* generiše na izlazu signale kojima se pale odgovarajući segmenti displeja prikazujući pritom heksadecimalnu cifru koja predstavlja prvi komplement četvorobitnog binarnog broja koji se nalazi na ulazu dekodera.

b) Tražena stanja na magistralama *X* i *Y* i upaljeni segmenti displeja tokom svih prikazanih perioda takta su predstavljeni na sledećoj slici:

