

ETF U BEOGRADU, KATEDRA ZA ELEKTRONIKU

Predmet: OSNOVI DIGITALNE ELEKTRONIKE

OCENA _____

Ispit: 08.06.2011.

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

DEŽURNI:

Sala _____
Vreme početka _____
Vreme završetka _____
Potpis _____

KANDIDAT:

Ime _____
Prezime _____
Broj indeksa _____
Potpis _____

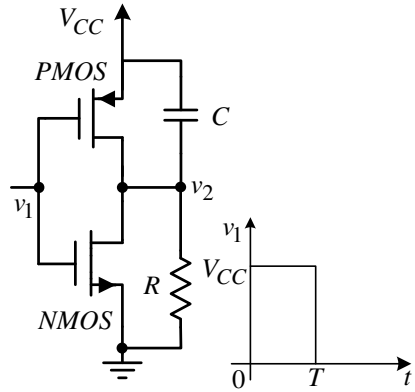
USLOVI ISPITA

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

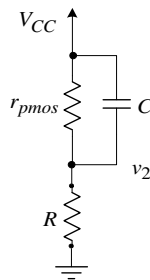
R.Br.	1	2	3	4	Total
Max	20	20	20	20	80
Dobijeno					

1. [20] Na slici je prikazano kolo CMOS invertora sa NMOS i PMOS tranzistorima i kondenzatorom C u paralelnoj vezi sa PMOS tranzistorom i otpornikom R u paralelnoj vezi sa NMOS tranzistorom. Vremenski oblik napona v_1 je takođe prikazan na slici. Izračunati vrednost napona v_2 u trenutku $t = 2T$. Smatrati da je u provodnom stanju otpornost NMOS tranzistora r_{nmos} konstantna i iznosi $r_{nmos} = R$, dok u neprovodnom stanju NMOS tranzistor ima beskonačnu otpornost. Otpornost PMOS tranzistora u provodnom stanju r_{pmos} je konstantna i iznosi $r_{pmos} = 2R$, dok u neprovodnom stanju PMOS tranzistor ima beskonačnu otpornost. ($V_{CC} > |V_T|$, $|V_T|$ – napon uključenja NMOS, PMOS tranzistora).



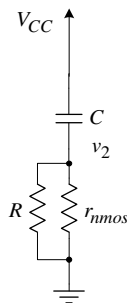
Rešenje:

Do momenta promene napona v_1 , u kolu CMOS invertora je uspostavljeno stacionarno stanje. NMOS tranzistor je zakočen, PMOS tranzistor vodi i ekvivalentna šema kola izgleda kao na sledećoj slici:



Napon v_2 je jednak naponu $v_2 = \frac{R}{R + r_{pmos}} V_{CC} = \frac{R}{R + 2R} V_{CC} = \frac{1}{3} V_{CC}$ jer kroz ekvivalentnu

otpornost PMOS tranzistora ne teče struja, struja teče samo kroz paralelnu otpornost R . Kada napon v_1 promeni vrednost i postane $v_1 = V_{CC}$, PMOS tranzistor se zakoči a NMOS tranzistor provede i ekvivalentno kolo CMOS invertora izgleda kao na sledećoj slici.



Promena v_2 napona od momenta $t_1 = 0^+$ pa na dalje se može opisati kao

$$v_2(t) = v_2(\infty) - (v_2(\infty) - v_2(0)) e^{-\frac{t}{C \frac{r_{nmos} R}{r_{nmos} + R}}}. \text{ Kako je } v_2(0) = \frac{2}{3} V_{CC} :$$

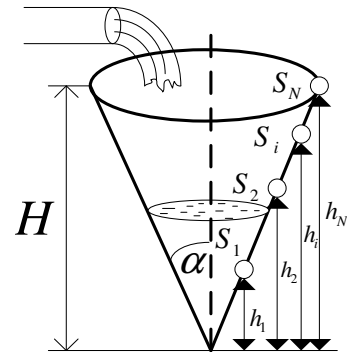
$$v_2(t) = \frac{1}{3}V_{CC} e^{-\frac{t}{C \frac{r_{nmos} R}{r_{nmos} + R}}}.$$

Kada se napon v_1 ponovo promeni u $t_2 = T$ ekvivalentno kolo CMOS invertora izgleda opet kao na prvoj slici iz rešenja, i promena napona v_2 napona od momenta $t_2 = T$ pa na dalje se može opisati

$$\text{kao } v_2(t) = \frac{1}{3}V_{CC} - \left(\frac{1}{3}V_{CC} - v_2(T)\right) e^{-\frac{t-T}{C \frac{r_{pmos} R}{r_{pmos} + R}}}. \text{ U trenutku } t_3 = 2T, \text{ napon } v_2 \text{ ima vrednost}$$

$$v_2(2T) = \frac{1}{3}V_{CC} - \left(\frac{1}{3}V_{CC} - \frac{1}{3}V_{CC} e^{-\frac{T}{C \frac{r_{nmos} R}{r_{nmos} + R}}}\right) e^{-\frac{T}{C \frac{r_{pmos} R}{r_{pmos} + R}}} = \frac{1}{3}V_{CC} (1 - (1 - e^{-2})e^{-3/2}) = 0.268V_{CC}.$$

2. [20] Na slici je prikazan rezervoar za tečne materije u obliku kupe. Tečnost se uliva u rezervoar sa otvora na vrhu. Za utvrđivanje popunjenosti rezervoara koriste se N senzora $S_1, S_2, S_3, \dots, S_N$. Visina rezervoara je H , dok je ugao pri vrhu 2α . Ukoliko je neki od senzora potopljen u tečnost, na svom izlazu generiše logičku jedinicu, ukoliko se nalazi van tečnosti generiše logičku nulu.



a) Kako treba rasporediti senzore (na koju visinu od vrha kupe ih montirati) po obodu rezervoara tako da zapremina tečnosti između svaka dva susedna senzora bude identična (za svaki senzor S_i , odrediti visinu h_i).

b) Senzori za detektovanje zapremine u rezervoaru su povezani na ulaz digitalnog kola koje upravlja procesom punjenja rezervoara. Na izlazu digitalnog kola se dobija binarna vrednost u opsegu od $0-M$, koja je obrnuto proporcionalna zapremini tečnosti koja se nalazi u rezervoaru (rezervoar prazan izlaz ima vrednost M , rezervoar pun izlaz ima vrednost 0). Izlaz digitalnog kola se dovodi na ulaz ventila kojim se reguliše protok tečnosti kojom se puni rezervoar (M -maksimalan protok, 0 -ventil zatvoren).

b1) Ukoliko je $M = 255$, odrediti potreban broj senzora za utvrđivanje popunjenosti rezervoara kao i broj izlaza upravljačkog kola.

b2) Šta predstavlja (koju kombinacionu mrežu) digitalno kolo koje upravlja radom ventila? Prikazati simboličku realizaciju ove mreže ako su na raspolaganju NILI kola.

Rešenje:

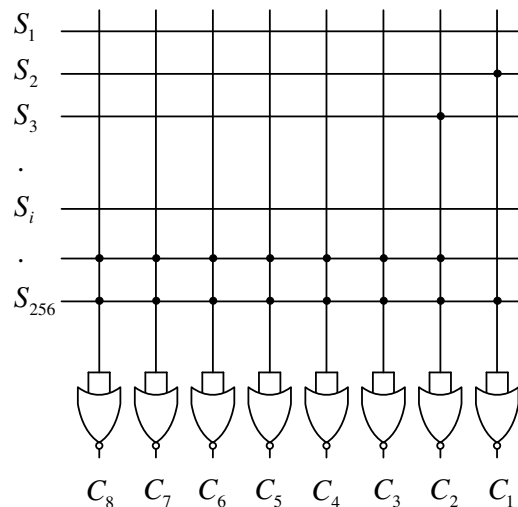
a) Ukupna zapremina rezervoara je data izrazom

$V = \frac{1}{3} \pi R^2 H = \frac{1}{3} \pi t g^2 \alpha H^3$, da bi zapremina tečnosti između svaka dva susedna senzora bila ista

treba da vazi $V_i = \frac{i}{N} V$, odavde se zaključuje da treba da važi $h_i = H \sqrt[3]{\frac{i}{N}}$

b) Ukupan broj senzora mora biti jednak broju M uvećan za jedan, dakle $N = M + 1 = 256$, jer je to broj mogućih nivoa tečnosti u rezervoaru. Broj izlaza koji ima kombinaciona mreža može se odrediti kao $n = \lceil \log_2(M) \rceil + 1 = 8$ (maksimalna binarna vrednost zapisana sa 8 bita iznosi 255).

Kada se nivo tečnosti podigne za jedan stepen, vrednost na izlazu digitalnog kola se mora umanjiti za jedan, na osnovu broja ulaza i izlaza kombinacionog kola može se zaključiti da je u pitanju koder prioriteta sa invertovanim izlazima. Simbolička realizacija ovog kodera sa NILI kolima je prikazana na slici.

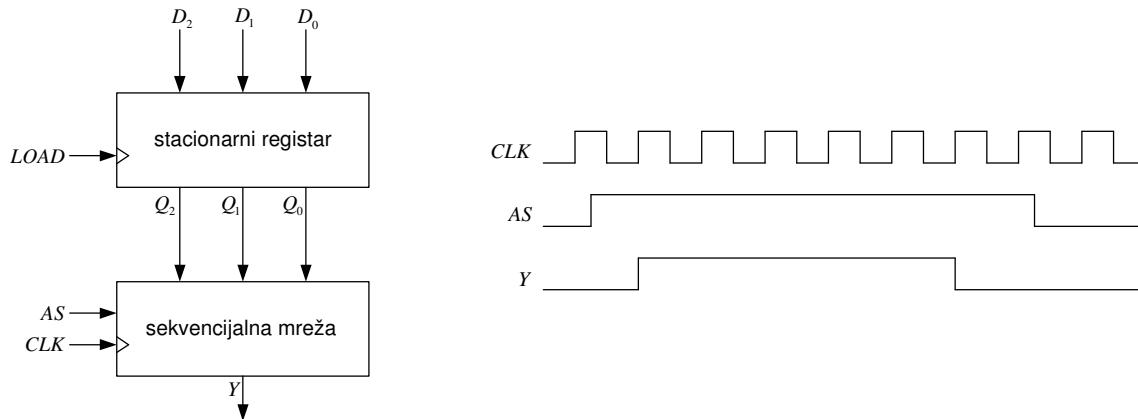


Matrica povezivanja odgovara matrici nula i jedinica, pri čemu vrste predstavljaju binarne osmobarne brojeve. Jedinici odgovara uspostavljena veza, a nuli raskunta veza u matrici povezivanja NILI kola i ulaza.

```
00000000
00000001
00000010
00000011
.....
.....
11111110
11111111
```

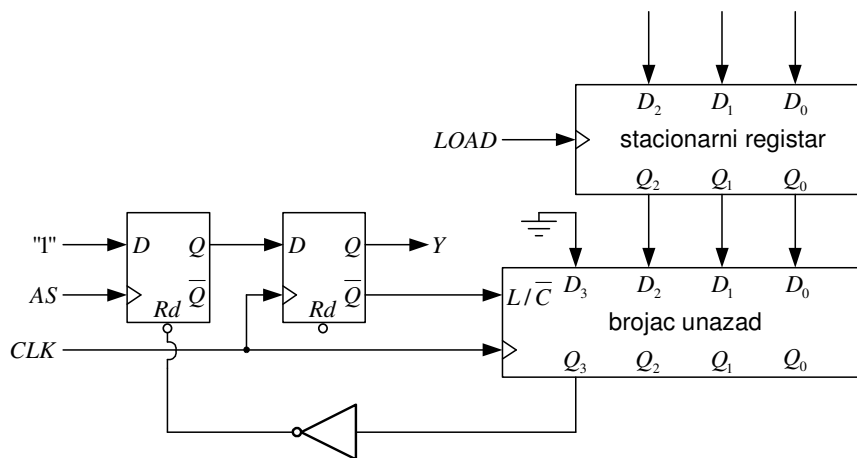
3. [20] Realizovati sekvencijalnu mrežu sa slike, koja treba da generiše izlazni impuls Y trajanja $(N+2) \cdot T_{CLK}$ nakon prve uzlazne ivice signala takta CLK od trenutka kada asinhroni signal AS postane logička jedinica. T_{CLK} je trajanje jedne periode signala takta CLK , a N je binarno kodovan broj upisan u stacionarni registar sa slike. Smatrati da je trajanje signala AS (dok je na nivou logičke jedinice) uvek duže od trajanja signala Y (dok je na nivou logičke jedinice). Na raspolaganju su D flip-flopovi sa ulazom za asinhroni reset, četvorobitni brojač unazad sa kontrolnim ulazom L/\bar{C} (ako se na ovaj kontrolni ulaz dovede logička jedinica brojač ne broji, već se u njega paralelno upisuje sadržaj sa njegovog ulaza za podatke; ako se na kontrolni ulaz dovede logička nula, brojač dekrementira svoj sadržaj na svaku uzlaznu ivicu signala takta) i potrebna logička kola. Koristiti minimalan broj potrebnih komponentata.

Primer vremenskih dijagrama signala CLK , AS i Y za slučaj $N=3$ je dat na slici.



Rešenje:

Tražena realizacija je prikazana na sledećoj slici:



Kada je izlaz sekvencijalne mreže $Y = 0$, izlaz \bar{Q} desnog D flip-flopa je $\bar{Q} = 1$. Time se u brojač upisuje sadržaj N stacionarnog registra. Pojavom uzlazne ivice signala AS izlaz Q levog D flip-flopa poprima nivo logičke jedinice. Nailaskom prve naredne uzlazne ivice signala takta, izlaz Y postaje jednak logičkoj jedinici, njegov komplement postaje jednak logičkoj nuli i brojač počinje da broji unazad sinhrono sa uzlaznom ivicom signala takta. Izlaz sekvencijalne mreže će biti $Y = 1$ ukupno $N+2$ periode signala takta (nakon $N+1$ periode signala takta od trenutka kada je postalo $Y = 1$, stanje brojača će postati 1111 i bitom najveće težine izlaza brojača će se preko invertora resetovati levi D flip-flop; nakon još jedne dodatne periode signala takta će i izlaz sekvencijalne mreže postati $Y = 0$, a brojač će da pređe u stanje upisa sadržaja stacionarnog registra).

4. [20] Koristeći metod dizajniranja na bazi opisa ponašanja sistema, napraviti VHDL model četvorobitnog brojača unapred koji broji u Grejovom kodu. Brojač treba da poseduje ulaz za sinhroni reset `rst` i ulaz koji predstavlja signal takta `clk`. Brojač broji sinhrono sa uzlaznom ivicom signala takta. Trenutno stanje brojača treba da je stalno prisutno na izlazima brojača `q`.

Svi vektori treba da imaju bite poredane u smeru MSB>LSB.

Napomena: Ocenjuje se precizna upotreba sintakse.

Rešenje:

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY counter_4bits IS PORT
(
    clk      : IN    STD_LOGIC;
    rst      : IN    STD_LOGIC;
    q        : OUT   STD_LOGIC_VECTOR(3 DOWNTO 0)
);
END counter_4bits;

ARCHITECTURE behav OF counter_4bits IS
    SIGNAL tmp : STD_LOGIC_VECTOR(3 DOWNTO 0);
BEGIN
    PROCESS (clk) BEGIN
        IF rising_edge(clk) THEN
            IF (rst='1') THEN
                tmp <= "0000";
            ELSE
                CASE tmp IS
                    WHEN "0000" => tmp <= "0001";
                    WHEN "0001" => tmp <= "0011";
                    WHEN "0011" => tmp <= "0010";
                    WHEN "0010" => tmp <= "0110";
                    WHEN "0110" => tmp <= "0111";
                    WHEN "0111" => tmp <= "0101";
                    WHEN "0101" => tmp <= "0100";
                    WHEN "0100" => tmp <= "1100";
                    WHEN "1100" => tmp <= "1101";
                    WHEN "1101" => tmp <= "1111";
                    WHEN "1111" => tmp <= "1110";
                    WHEN "1110" => tmp <= "1010";
                    WHEN "1010" => tmp <= "1011";
                    WHEN "1011" => tmp <= "1001";
                    WHEN "1001" => tmp <= "1000";
                    WHEN "1000" => tmp <= "0000";
                END CASE;
            END IF;
        END IF;
    END PROCESS;
    q <= tmp;
END behav;

```