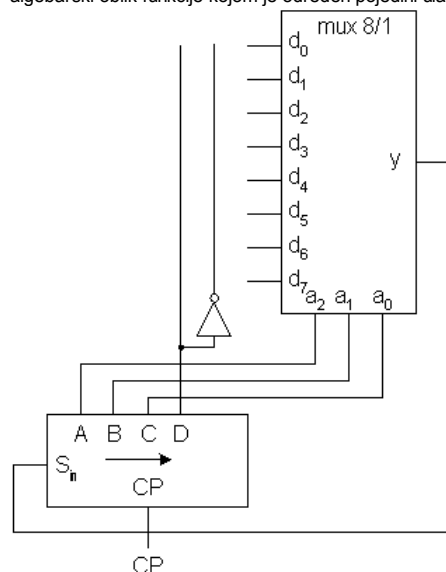


5. domaća zadaća (0910Z)

1. Djelomično točnoVaše rješenje | Točno rješenje

Relativni doprinos: 0.875/1.0

Potrebno je projektirati sklop koji će prolaziti kroz sljedeća stanja: (10, 13, 14, 7, 3, 1, 0, 8, 12, 6, 11, 5, 10). Sklop je potrebno ostvariti uporabom strukture prikazane na slici (posmačni registar + multiplexsor). Nespecificirana stanja treba tako riješiti da sklop najbrže stigne u stanje 10. Seleksijski ulazi u multiplexsor su: A, B, C. Što treba dovesti na podatkovne ulaze multiplexsora? U polja za unos odgovora je potrebno unijeti algebarski oblik funkcije kojom je određen pojedini ulaz multiplexsora. Prilikom očitavanja stanja, izlaz A tretirati kao bit najveće težine.



D0 NOT D

D1 NOT D

D2 1

D3 NOT D

D4 NOT D

D5 NOT D

D6 D

D7 0

Važna napomena: U svako polje za unos treba unijeti samo jedan algebarski oblik funkcije koja se dovodi na pojedini ulaz multiplexsora. Ulaze treba realizirati bez varijabli dovedenih na seleksijske ulaze, te ukoliko se to ne poštuje, zadatak će biti ocijenjen kao netočan. Dodatna napomena: za konstantne vrijednosti su dozvoljena dva načina unosa vrijednosti: 1 se tretira jednako kao i true, a 0 je ekvivalentna sa false.

2. Točno

Relativni doprinos: 1.0/1.0

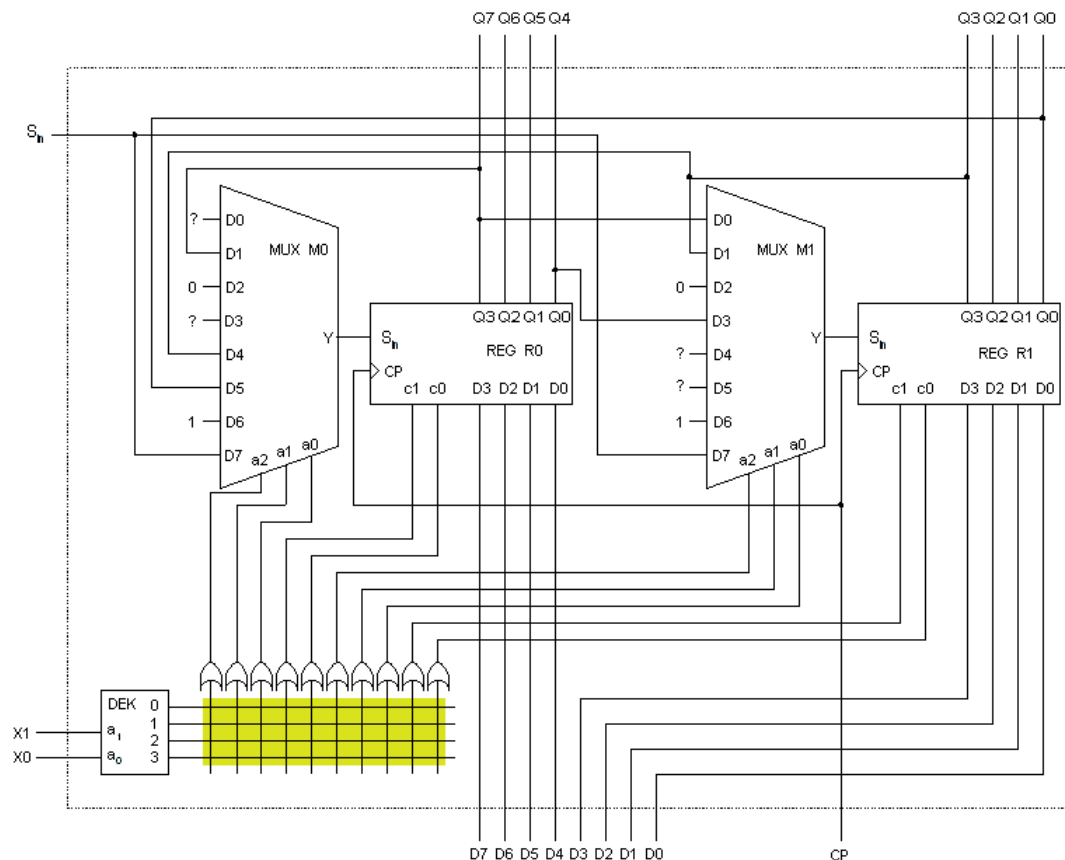
Funkcija 4 bitnih registrara prikazanih na slici definirana je sljedećom tablicom.

C1	C0	Opis
0	0	Posmak u desno, punjenje sa Sin
0	1	Paralelni upis
1	0	Posmak u lijevo, punjenje sa Sin
1	1	NOP - nema nikakve promjene

Uporabom multiplexsora i dekodera s programirljivom ILI ravninom ostvarena je struktura koja se može programirati tako da se dobije 8-bitni registar s mnoštvom podržanih operacija. Vaš je zadatak programirati ovu strukturu tako da se dobije jedan 8-bitni registar čija je funkcija određena sljedećom tablicom.

X1	X0	Opis
0	0	Paralelni upis
0	1	Posmak u lijevo, punjenje sa Sin
1	0	Posmak u desno, punjenje sa Sin
1	1	Posmak u desno, punjenje s 1

Ulaze u multiplexsor označene upitnikom zabranjeno je koristiti. U polja za unos rješenja u nastavku potrebno je unijeti zarezima odvojen popis izlaza dekodera koje je potrebno spojiti na odgovarajući ILI sklop, kako bi se ostvarila potrebna funkcija. Ako se za neki ILI sklop ne definira niti jedan izlaz dekodera, taj ILI sklop na izlazu generira logičku nulu.



M0.a2 1,2,3

M0.a1 2,3

M0.a0 2

R0.c1	1
-------	---

R0.c0 0

M1.a2 1

M1.a1 1.2.3

M1.a0 1,2,3

R1.c1 1

R1.c0 0

3. **Točno**

Relativni doprinos: 1.0/1.0

Za realizaciju binarnog asinkronog brojala koje broji u skraćenom ciklusu duljine 30 stanja na raspolaganju su padajućim bridom okidani T bistabili s asinkronim ulazom za brisanje koji djeluju kada im se dovede logička jedinica (svi su spojeni zajedno; označimo tu točku oznakom X). Brojilo treba ostvariti minimalno potrebnim brojem bistabila, pri čemu stanje 0 treba pripadati ciklusu. Pobudu za točku X generira kombinacijski sklop. Koju funkciju taj sklop treba ostvarivati? Kao rješenje upišite algebarski oblik (npr. not Q2 or Q1). Prilikom očitavanja stanja izlaz Q0 smatra se izlazom naimanje težine.

Q4 AND Q3 AND Q2 AND Q1

4. **Djelomično točno** **Vaše rješenje** | **Točno rješenje**

Relativni doprinos: 0.333333333333333/1.0

Potrebno je realizirati sinkrono brojilo koje broji u ciklusu: (5, 7, 3, 1, 4, 2, 6, 0, 5), ako je na raspolaganju 3 JK bistabila. Prilikom očitavanja stanja izlaz Q0 bistabila B0 promatrati kao bit najmanje težine. U polja za unos rješenja unijeti minimizirane algebarske oblike funkcija.

J2 (NOT Q1) OR (NOT Q0)

K2 (Q1) OR (NOT Q0)

J1 q2

K1 (not q2 and q0) or (q2 and

J0 (NOT Q2 AND NOT Q1)

K0 (NOT Q2 AND NOT Q1)

Važna napomena: U polja za unos rješenja treba unijeti logičke izraze funkcija dovedenih na ulaze bistabila, uz koje će se ostvariti zadano brojiilo.

5. **Nije riješen** Vaše rješenje | Točno rješenje

Relativni doprinos: 0.0/1.0

Nacrtajte shemu 3-bitnog asinkronog binarnog brojila unatrag izvedenog rastućim bridom okidanih JK bistabila. Za taj sklop potrebno je nacrtati vremenske dijagrame do trenutka $T=1458$ ns. Na brojilo se dovodi signal takta periode 162 ns, pri čemu u trenutku $t=0$ nastupa rastući brid. Kašnjenje svakog bistabila iznosi 36 ns. Brojilo osim bistabila ne smije koristiti dodatne logičke sklopove. Pretpostaviti da su svi bistabili u trenutku prije $t=0$ u stanju 0. Ako s Q0 označimo izlaz bistabila najmanje težine, očitajte stanje svih izlaza u trenutku $t=1238$ ns.

Q0 0

Q1 0

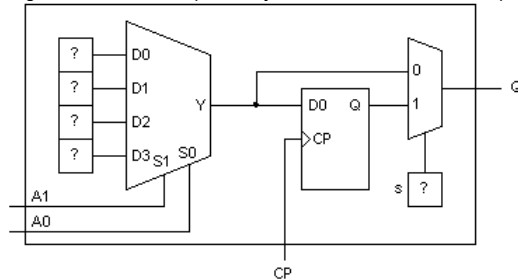
Q2 0

Važna napomena: vrijednosti koje se prihvaćaju su: 0, 1 (alternativno: true, false).

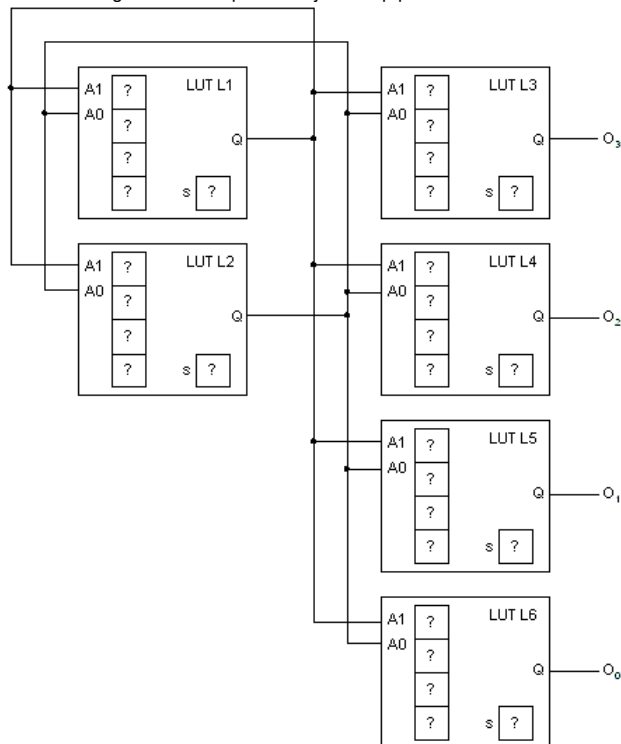
6. **Točno**

Relativni doprinos: 1.0/1.0

Logički blok FPGA sklopa temeljen na LUT-u, bistabilu i multiplexoru prikazan je na sljedećoj slici.



Više takvih logičkih blokova povezano je u sklop prikazan u nastavku.



Programirajte sve logičke blokove tako da se dobije sklop koji na izlazu generira sekvencu: 10, 12, 10, 10.

Pri tome izlaz O3 tretirati kao izlaz najveće, a izlaz O0 kao izlaz najmanje težine.

L1.lut 0110

L1.s 1

L2.lut 1010

L2.s 1

L3.lut 1111

L3.s 0

L4.lut 0100

L4.s 0

L5.lut 1011

L5.s 0

L6.lut 0000

L6.s 0

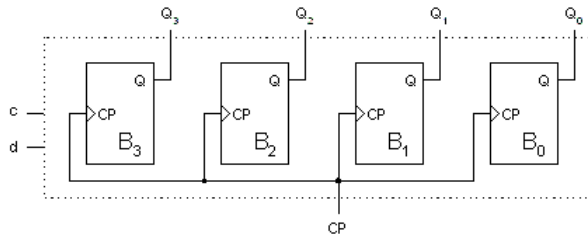
Važna napomena: U polja za unos u koje treba unijeti vrijednost pregledne tablice (Li.lut) unosi se popis zarezima odvojenih vrijednosti. Pri tome se najprije unosi vrijednost koja odgovara nultoj lokaciji pregledne tablice. U polja za unos u koje treba unijeti vrijednost adresnog ulaza multiplexora (Li.s) unosi se jedan broj.

7. **Djelomično točno** Vaše rješenje | **Točno rješenje**

Relativni doprinos: 0.25/1.0

Uporabom T bistabila realizirati 4-bitno brojilo koji broji ovisno o signalu d: ako je d=1, tad je sljedeće_stanje = trenutno_stanje+1, inače sljedeće_stanje = trenutno_stanje-4 (pod pojmom stanje misli se na binarno kodirani broj zapisan kroz bistabile, pri čemu je izlaz Q0 izlaz najmanje

težine). Sklop treba imati i sinkroni ulaz za brisanje c (kojeg bistabili nemaju). Koristiti minimalni broj osnovnih logičkih sklopova. U svako polje za unos potrebno je unijeti algebarski zapis funkcije tog bistabila. Prilikom očitavanja stanja izlaz Q0 bistabila B0 tretirati kao izlazni bit najmanje težine. Prilikom unosa algebarskog oblika za stanja bistabila koristiti oznake Qj (gdje je j broj bistabila; npr. Q2). Primjer jednog takvog rješenja: c and Q2 and not Q1 or not c and d and not Q0.



B3.T (NOT c AND NOT d AND

B2.T (NOT c AND NOT d) OF

B1.T q0 and d and not c or q1

B0.T (NOT c AND d) OR (c AND

8. **Točno**

Relativni doprinos: 1.0/1.0

Za neki 11-bitni pretvornik sa sukcesivnom aproksimacijom poznato je da se ulazni napon iznosa 3.6V pretvara 48 ms. Koliko iznosi vrijeme pretvorbe za ulazni napon iznosa 10.5V? Podrazumijevana mjerna jedinica je ms.

48 ms

9. **Netočno** **Vaše rješenje** | **Točno rješenje**

Relativni doprinos: 0.0/1.0

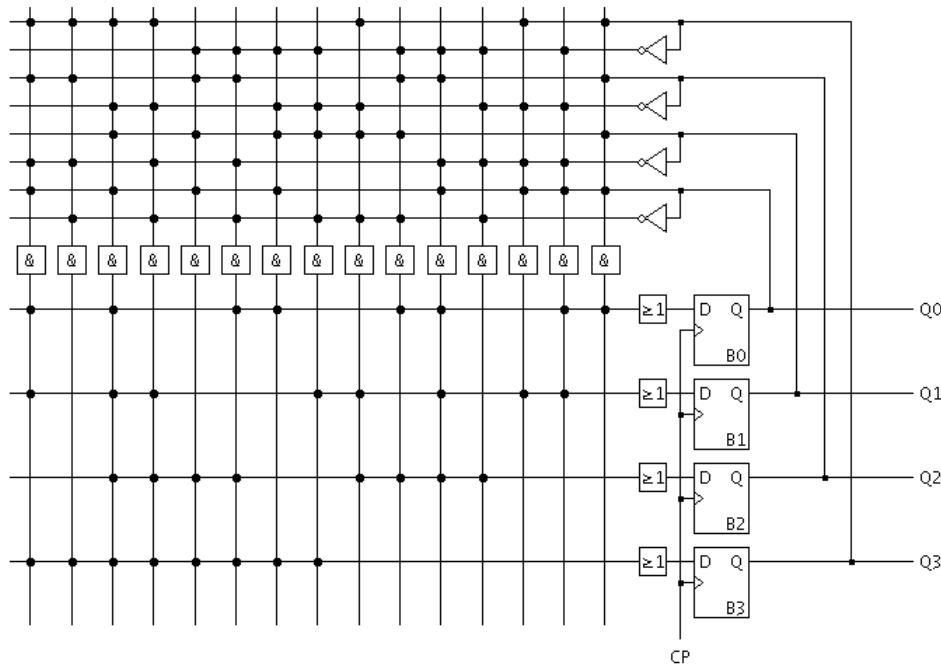
Za realizaciju binarnog asinkronog brojlara koje broji u skraćenom ciklusu duljine 37 stanja na raspolaganju su padajućim bridom okidani T bistabili s asinkronim ulazom za brisanje koji su svi spojeni zajedno. Pri tome je utrošen minimalno potreban broj bistabila. Stanje 0 treba biti sastavni dio ciklusa. Koje stanje treba dekodirati kako bi se realiziralo to brojilo? Kao rješenje unesite broj stanja u dekadskom sustavu (npr. 12).

37

10. **Točno**

Relativni doprinos: 1.0/1.0

Neko sinkrono brojilo prikazano je sljedećom slikom.



U kojem ciklusu broji to brojilo? Prilikom očitavanja stanja izlaz Q0 tretirati kao bit najmanje težine.

- ☐ (1, 14, 15, 0, 9, 10, 11, 2, 6, 5, 4, 7, 8, 3, 13, 12, 1)
- ☐ (15, 2, 1, 6, 0, 3, 10, 5, 4, 13, 12, 8, 14, 7, 11, 9, 15)
- ☒ (5, 7, 12, 8, 14, 0, 4, 13, 11, 15, 1, 3, 9, 2, 10, 6, 5)
- ☐ (9, 0, 1, 4, 12, 7, 10, 11, 3, 6, 2, 5, 15, 8, 13, 14, 9)

11. **Točno**

Relativni doprinos: 1.0/1.0

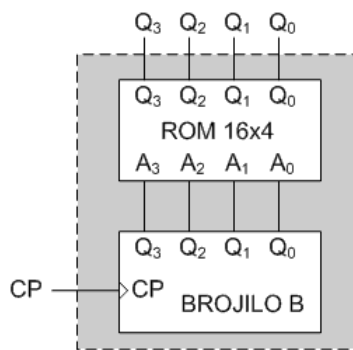
Za kod 1351 konstruiran je težinski D/A pretvornik s operacijskim pojačalom. Ako je najveći otpor u težinskoj mreži pretvornika 19 Oma , izračunajte iznos otpora Rf u povratnoj vezi operacijskog pojačala. Poznati su sljedeći podaci: ako se na ulaz pretvornika dovede broj 9, apsolutna vrijednost izlaznog napona je 3.6 V; iznos referentnog napona Uref = 6 V. Podrazumijevana mjerna jedinica je Om.

1,27

12. **Djelomično točno** **Vaše rješenje** | **Točno rješenje**

Relativni doprinos: 0.5/1.0

Na raspolaganju je sklop prikazan na slici.



Brojilo B koje je iskorišteno za brojanje broji u ciklusu 9 -> 13 -> 7 -> 12 -> 14 -> 8 -> 6 -> 3 -> 0 -> 5 -> 4 -> 1 -> 15 -> 2 -> 11 -> 10.

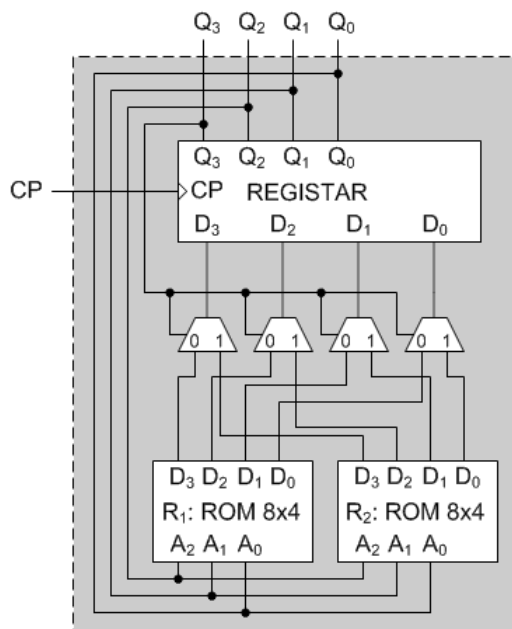
Programirajte memoriju tako da se čitav sklop ponaša kao standardno binarno brojilo unaprijed. Uključenjem na napajanje brojilo B ulazi u stanje 9, a izlaz čitavog sklopa treba poprimiti stanje 0 (prilikom očitavanja stanja Q3 tretirati kao bit najveće težine). Napomena: kao rješenje upišite sadržaj memorije po lokacijama, i to svaku lokaciju kao jednu heksadekadsku znamenku.

0. 8
1. B
2. D
3. 7
4. A
5. 9
6. 6
7. 2
8. 5
9. 0
10. F
11. E
12. 3
13. 1
14. 4
15. C

13. Djelomično točno **Vaše rješenje** | **Točno rješenje**

Relativni doprinos: 0.625/1.0

Na raspolaganju je sklop prikazan na slici.



Memorije R1 i R2 potrebno je programirati tako da se dobije brojilo koje broji u ciklusu 2 -> 8 -> 3 -> 1 -> 10 -> 13 -> 4 -> 11 -> 15 -> 14 -> 7 -> 6 -> 12 -> 0 -> 9 -> 5 (prilikom očitavanja stanja Q3 tretirati kao bit najveće težine). Napomena: kao rješenje upišite sadržaj memorija po lokacijama, i to svaku lokaciju kao jednu heksadekadsku znamenku.

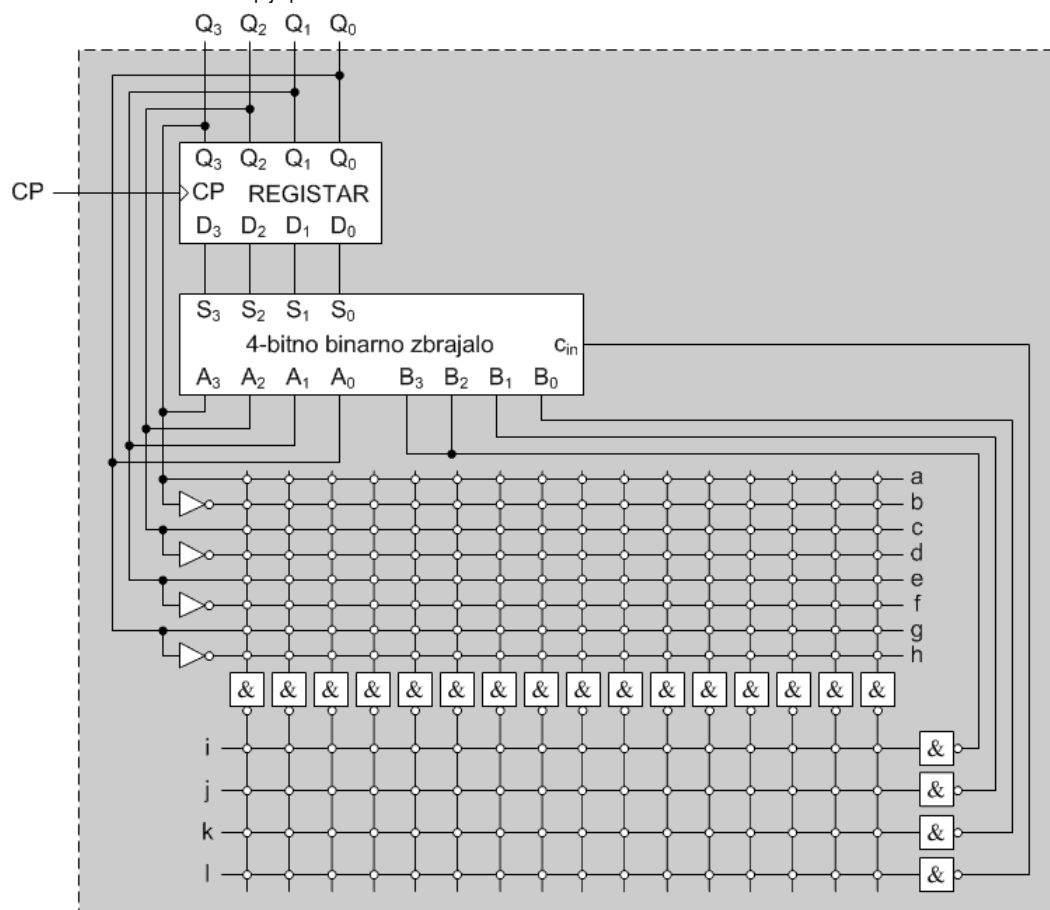
- R1: 9
- 0.
- R1: A
- 1.
- R1: 8
- 2.
- R1: 1
- 3.

R1: B
4. B
R1: 2
5. 2
R1: C
6. C
R1: 6
7. 6
R2: 3
0. 3
R2: 5
1. 5
R2: D
2. D
R2: F
3. F
R2: 0
4. 0
R2: 4
5. 4
R2: 7
6. 7
R2: E
7. E

14. **Točno**

Relativni doprinos: 1.0/1.0

Uporabom registra, binarnog zbrajala i sklopa PLA želi se ostvariti brojilo koje broji u ciklusu 0 -> 15 -> 14 -> 13 -> 12 -> 10 -> 9 -> 11 -> 8 -> 5 -> 6 -> 7 -> 4 -> 3 -> 2 -> 1. Sklop je prikazan u nastavku.



Programirajte sklop PLA tako da ostvarite zadano brojilo. Napomena: a, b, c, d, e, f, g, h, i, j, k, l čine polje programirljivih sklopki. Kao rješenje se za svaki redak unosi 1 na mjestu uključene sklopke, odnosno 0 na mjestu isključene sklopke. Primjerice, ukoliko prvi i drugi NI sklop koriste Q3, tada će redak 'a' biti 1100000000000000. Uočite kako konfiguracija svakog retka ima točno 16 znamenki.

a 0000000011111111
b 1111111100000000
c 0000111100001111
d 1111000011110000
e 0011001100110011
f 1100110011001100
g 0101010101010101
h 1010101010101010
i 1111100110111111

j 1111100001100111

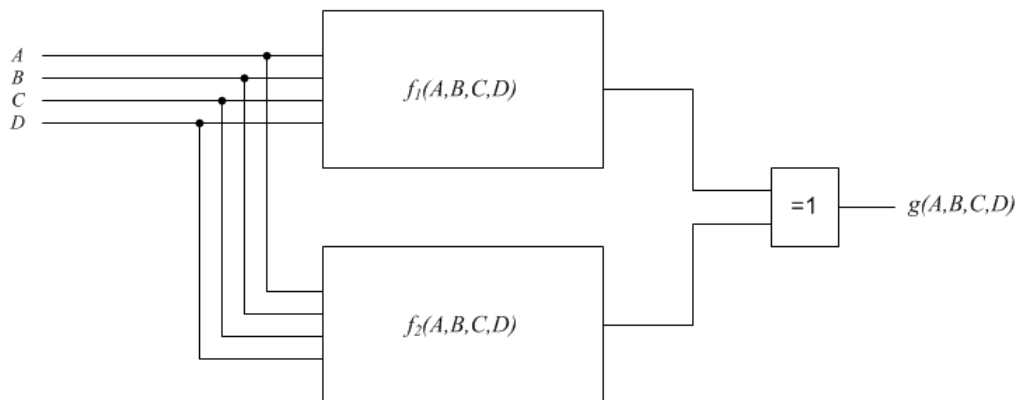
k 1000011000001000

l 0111100110111111

15. **Točno**

Relativni doprinos: 1.0/1.0

Kombinacijski sklop na slici označen s $f_1(A,B,C,D)$ obavlja funkciju $\text{suma_minterma}(0, 3)$. Čitav digitalni sklop na svom izlazu g treba obavljati funkciju $g(A,B,C,D) = \text{produkt_maksterma}(1, 2, 3, 5, 6, 7, 8, 9, 11, 13, 14, 15)$. Koju tada funkciju treba obavljati kombinacijski sklop $f_2(A,B,C,D)$? Kao odgovor je potrebno unijeti algebarski zapis te funkcije u obliku minimalne sume produkata.



(b and not c and not d) or (not a and not b and c and d) or (a and not b and c and not d)

[Povratak](#)