4.0 Minimizacija logičkih funkcija

- Kod minimizacije logičkih funkcija se polazi od činjenice da se ista logička funkcija može napisati na više različitih načina koji, iako definišu istu funkciju, ne moraju biti podjednako pogodni za praktičnu realizaciju.
- Minimizacija logičkih funkcija se obavlja u cilju smanjenja broja logičkih kola u mrežama kojima se date logičke funkcije realizuju.

- Metodi minimizacije:

1. Algebarski,
2. Tablični,
3. Grafički,
4. Programskim metodama.

- Od grafičkih metoda najčešće se koristi minimizacija pomoću Karnoove karte.

4.1 Minimizacija pomoću Karnoovih karti

- Postupak minimizacije Karnoovom kartom:

1. Nacrtati Kornoovu kartu odgovarajuće dimenzije i popuniti je na osnovu date logičke funkcije.
2. Formirati *veće pravougaone* površine od 2^k *susednih* polja koje obuhvataju samo jedinice (k=0,1, ... ,n).
3. Napisati rezultujući izraz u obliku sume proizvoda izostavljajući promenljive koje u istoj pravougaonoj površini imaju različite vrednosti .

- Prilikom formiranja pravougaonih površina treba se držati sledećih pravila:

1. Prvo se izdvaja jedna ili više *što većih* površina koje u tabeli obuhvatju neku jedinicu koja nije obuhvaćena nijednom drugom površinom.
2. Kada se izdvoje sve takve površine, sve preostale jedinice u tabeli se takođe grupišu u *što veće* pravougaone površine.
3. Po *potrebi* iste jedinice se mogu grupisati *više puta* tj. mogu pripadati većem broju pravougaonih površina.

Primer 1. Pomoću Karnoove karte izvršiti minimizaciju logičke funkcije date sumom proizvoda

$$F=\overline{A}\,\overline{B}\,\overline{C}\,\overline{D}+\overline{A}B\overline{C}\,\overline{D}+\overline{A}\,\overline{B}C\overline{D}+\overline{A}BC\overline{D}+ABC\overline{D}+ABCD+A\overline{B}CD+A\overline{B}C\overline{D}$$

AB \ CD	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	0	0	1	1
10	0	0	1	1

$$F=\overline{A}\,\overline{D}+AC$$

Primer 2. Pomoću Karnoove karte izvršiti minimizaciju logičke funkcije date sumom proizvoda

$$F = \overline{A}\,\overline{B}\,\overline{C}D + \overline{A}B\overline{C}D + \overline{A}BCD + \overline{A}BC\overline{D} + \overline{A}\,\overline{B}C\overline{D} + ABC\overline{D} + A\overline{B}C\overline{D}$$

AB \ CD	00	01	11	10
00	0	1	0	1
01	0	1	1	1
11	0	0	0	1
10	0	0	0	1

$$F = \overline{A}\,\overline{C}D + C\overline{D} + \overline{A}BC$$

Primer 3. Pomoću Karnoove karte izvršiti minimizaciju logičke funkcije date skupom indeksa

$$F(1) = (0,2,4,8,10,12,15)$$

AB \ CD	00	01	11	10
00	1	0	0	1
01	1	0	0	0
11	1	0	1	0
10	1	0	0	1

$$F = \overline{C}\,\overline{D} + ABCD + \overline{B}\,\overline{D}$$

Primer 4.

AB \ CD	00	01	11	10
00	0	0	1	0
01	1	0	0	1
11	0	0	1	0
10	0	0	1	1

$$F = \overline{A}B\overline{D} + ACD + \overline{B}CD + A\overline{B}C$$

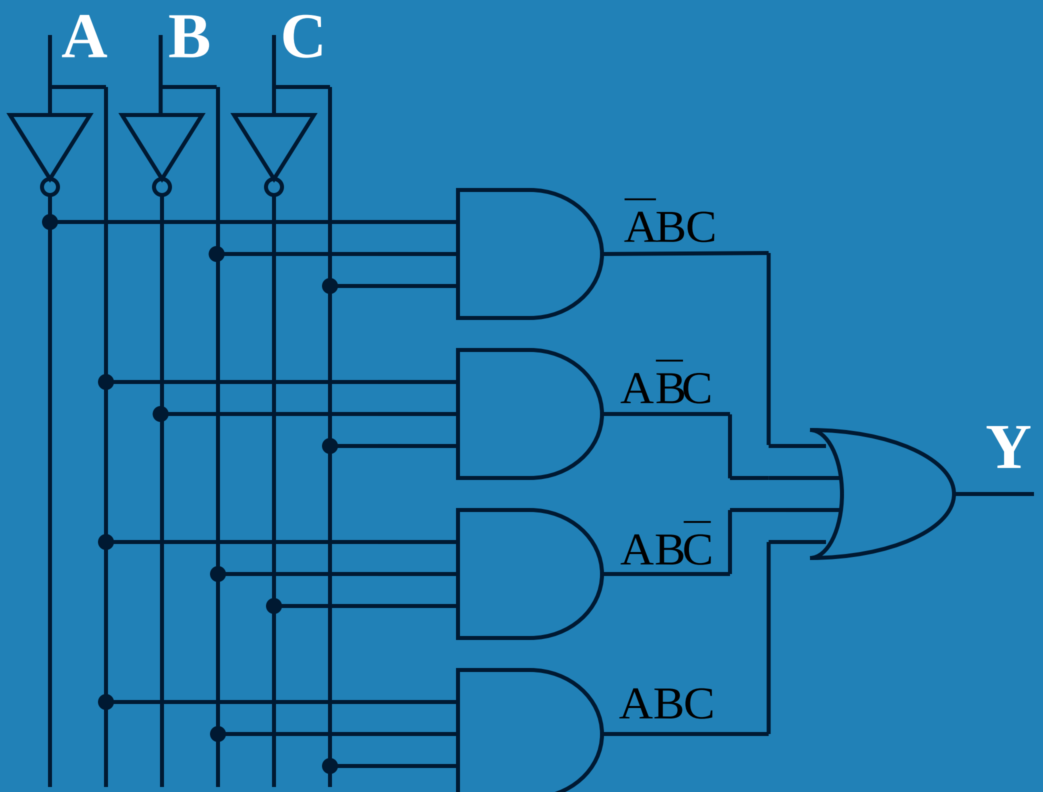
5.0 Realizacija logičkih funkcija

- Logičke funkcije se realizuju pomoću prekidačkih mreža. Prekidačke mreže su osnovne komponente savremenih digitalnih sistema. One predstavljaju skup logičkih kola (I, ILI, NE ...) tako povezanih da realizuju željenu logičku funkciju.
- Prekidačke mreže mogu biti kombinacione i sekvencijalne prekidačke mreže.
- Kod kombinacionih prekidačkih mreža vrednost funkcije na izlazu zavisi samo od trenutnog stanja na ulazu (tj. od trenutne vrednosti promenljivih), a kod sekvencijalnih zavisi i od prethodnog stanja u kome se mreža nalazila.

Primer 1. Realizovati prekidačku mrežu koja realizuje funkciju većinske logike

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$$Y = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$

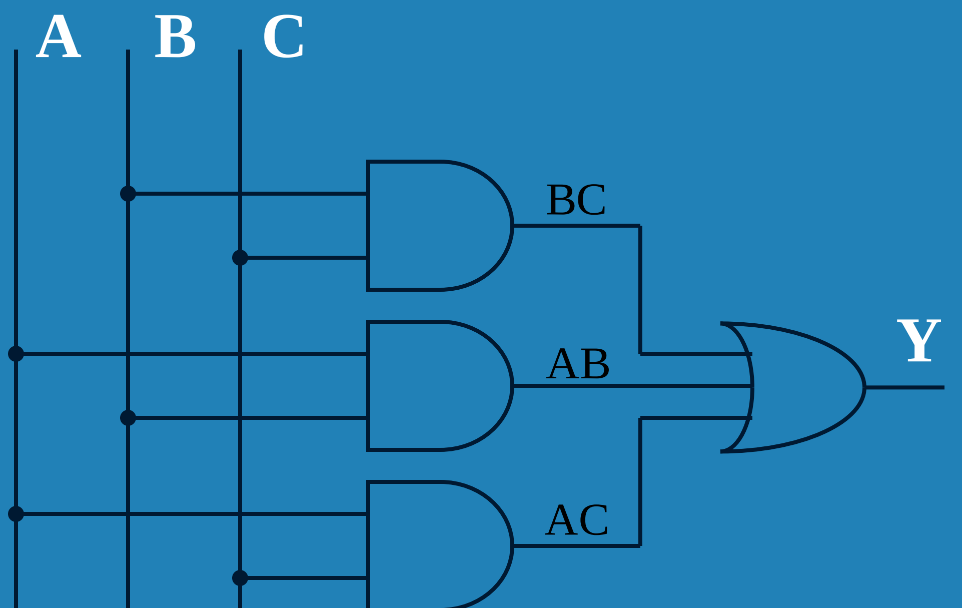


Primer 2. Realizovati prekidačku mrežu koja realizuje prethodno minimizovanu logičku funkciju iz primera 1

$$Y = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$

AB \ C	0	1
00	0	0
01	0	1
11	1	1
10	0	1

$$Y = BC + AB + AC$$



5.1 Koderi

- Da bi neka informacija mogla da se obrađuje digitalnim sistemom, potrebno je da se predstavi određenom kombinacijom nula i jedinica, odnosno treba da bude kodovana.
- Kombinaciona mreža koja obavlja ovu operaciju naziva se koder. Koderi imaju više ulaza i više izlaza.
- Koderi mogu biti potpuni, kada imaju 2^n ulaza i n izlaza, i nepotpuni, kada je za n izlaza broj ulaza manji od 2^n. Dakle kod potpunog kodera je $BrojUlaza=2^{BrojIzlaza}$, a kod nepotpunog je $BrojUlaza<2^{BrojIzlaza}$.

- **Kombinaciona tablica i funkcije izlaza potpunog kodera sa 8 ulaza (potpuni koder sa 8 ulaza ima 3 izlaza).**

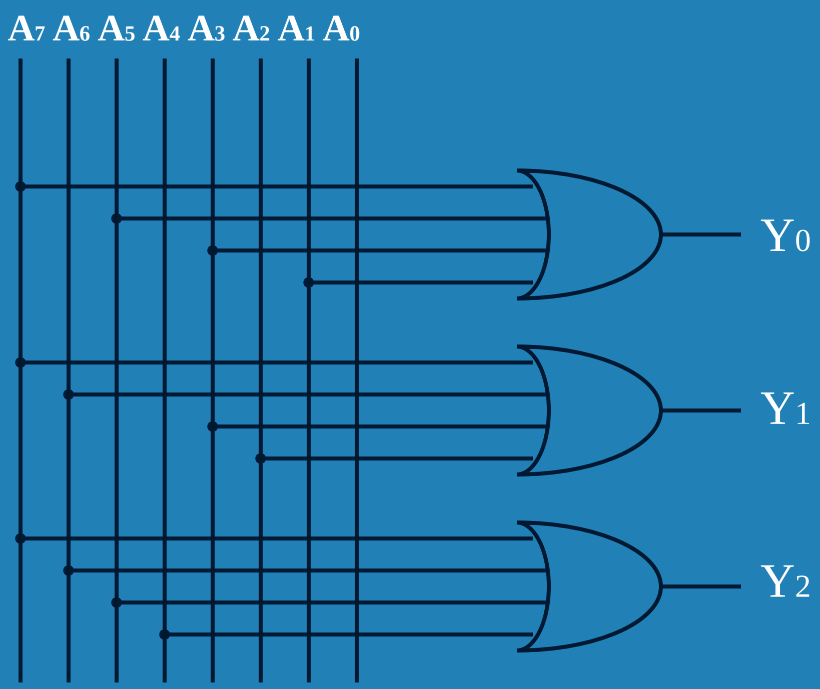
A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0		Y_2	Y_1	Y_0
0	0	0	0	0	0	0	1		0	0	0
0	0	0	0	0	0	1	0		0	0	1
0	0	0	0	0	1	0	0		0	1	0
0	0	0	0	1	0	0	0		0	1	1
0	0	0	1	0	0	0	0		1	0	0
0	0	1	0	0	0	0	0		1	0	1
0	1	0	0	0	0	0	0		1	1	0
1	0	0	0	0	0	0	0		1	1	1

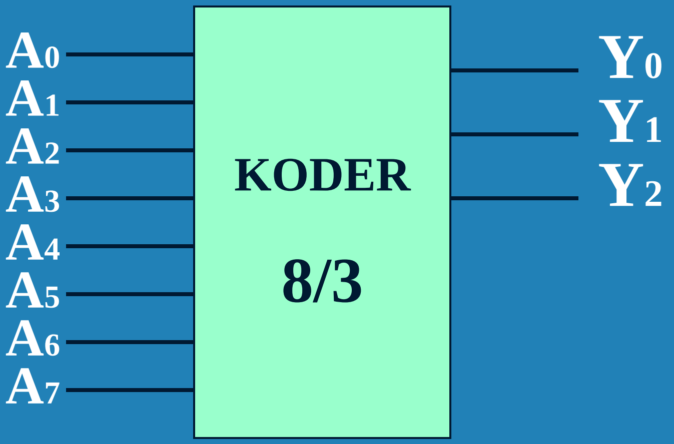
$$Y_0 = A_1 + A_3 + A_5 + A_7$$
$$Y_1 = A_2 + A_3 + A_6 + A_7$$
$$Y_2 = A_4 + A_5 + A_6 + A_7$$

- Kao što se iz prethodne tablice vidi na ulazu je aktivan jedan i samo jedan od 2^n signala, koji na izlazu koduje binarni broj od n bita.
- U slučaju da je istovremeno aktivno dva ili više ulaznih signala, koder će na izlazu generisati pogrešan kod.

Realizacija potpunog kodera sa 8 ulaza

$$Y_0 = A_1 + A_3 + A_5 + A_7$$
$$Y_1 = A_2 + A_3 + A_6 + A_7$$
$$Y_2 = A_4 + A_5 + A_6 + A_7$$

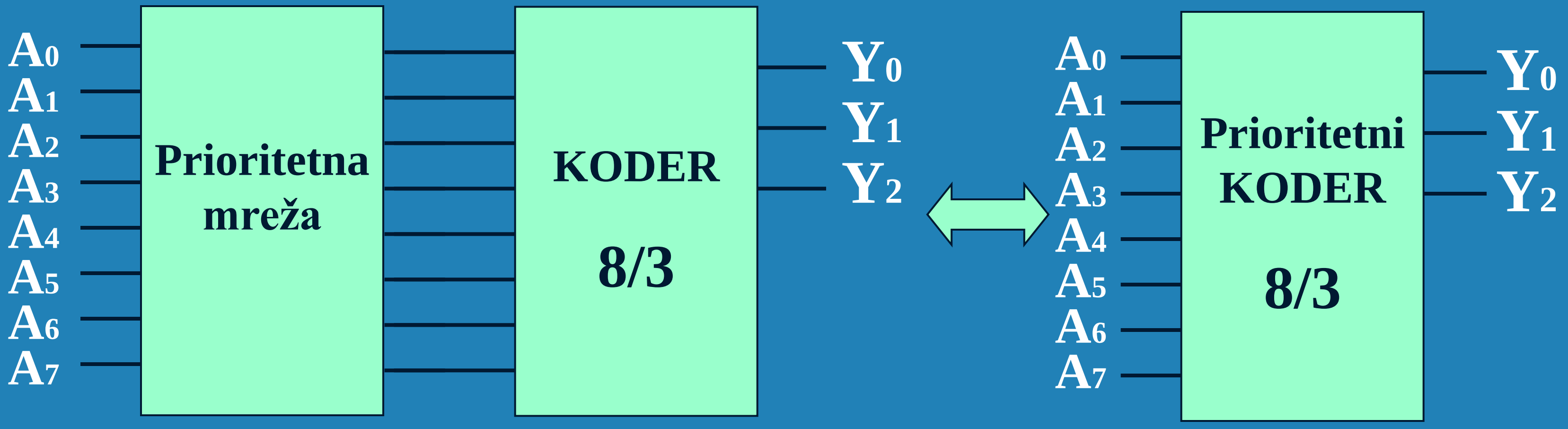




5.2 Prioritetni koder

- Za upotrebu u digitalnim sistemima gde postoji mogućnost da se na ulazu kodera istovremeno pojavi više od jednog signala do sada opisani koder se ne može koristiti.
- Potrebno je modifikovati mrežu kodera tako da se ulaznim linijama dodeli prioritet. Tada se u slučaju pojave više ulaza istovremeno, na izlazu generiše kod ulaza sa najvećim prioritetom. Ovako modifikovani koder se naziva *prioritetni koder.*

- Prioritetni koder se može realizovati korišćenjem običnog kodera i prioritetne mreže. Prioritetna mreža se vezuje između ulaznih signala i kodera.
- Prioritetna mreža treba da obezbedi da bez obzira na broj aktivnih ulaznih signala, na izlazu prioritetne mreže postoji jedan i samo jedan aktivan signal.



A0
A1
A2
A3
A4
A5
A6
A7
Prioritetna mreža
KODER
8/3
Y0
Y1
Y2
A0
A1
A2
A3
A4
A5
A6
A7
Prioritetni KODER
8/3
Y0
Y1
Y2

Prioritetna mreža

- Ako usvojimo da je ulazni signal A_7 najvišeg prioriteta, tada za prioritetnu mrežu važi sledeća kombinaciona tablica.

A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0		AP_7	AP_6	AP_5	AP_4	AP_3	AP_2	AP_1	AP_0
1	b	b	b	b	b	b	b		1	0	0	0	0	0	0	0
0	1	b	b	b	b	b	b		0	1	0	0	0	0	0	0
0	0	1	b	b	b	b	b		0	0	1	0	0	0	0	0
0	0	0	1	b	b	b	b		0	0	0	1	0	0	0	0
0	0	0	0	1	b	b	b		0	0	0	0	1	0	0	0
0	0	0	0	0	1	b	b		0	0	0	0	0	1	0	0
0	0	0	0	0	0	1	b		0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	1		0	0	0	0	0	0	0	1

- Na osnovu kombinacione tablice možemo da napišemo sledeće jednačine za prioritetnu mrežu:

$$AP_0 = A_0\overline{A_1}\,\overline{A_2}\,\overline{A_3}\,\overline{A_4}\,\overline{A_5}\,\overline{A_6}\,\overline{A_7}$$

$$AP_1 = A_1\overline{A_2}\,\overline{A_3}\,\overline{A_4}\,\overline{A_5}\,\overline{A_6}\,\overline{A_7}$$

$$AP_2 = A_2\overline{A_3}\,\overline{A_4}\,\overline{A_5}\,\overline{A_6}\,\overline{A_7}$$

$$AP_3 = A_3\overline{A_4}\,\overline{A_5}\,\overline{A_6}\,\overline{A_7}$$

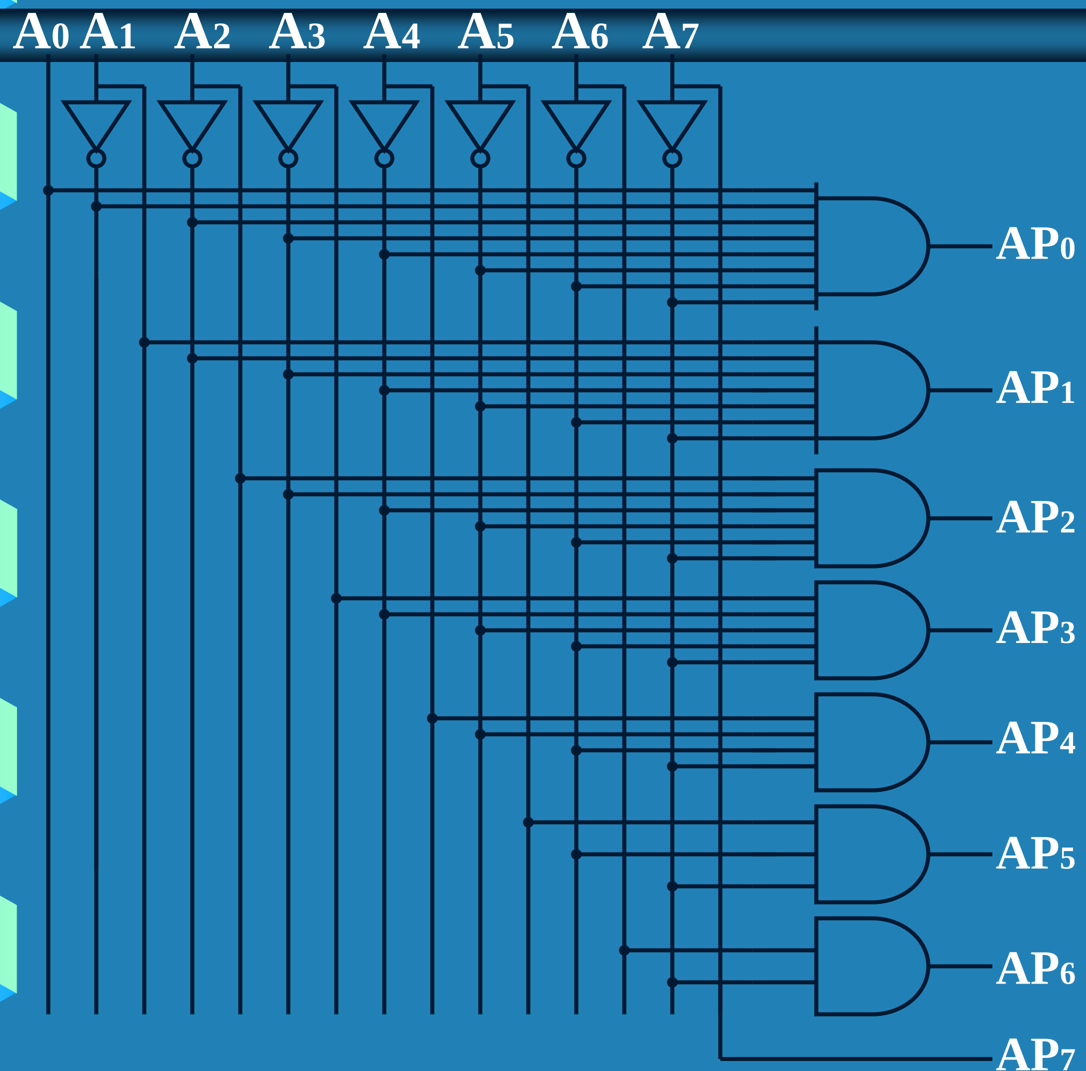
$$AP_4 = A_4\overline{A_5}\,\overline{A_6}\,\overline{A_7}$$

$$AP_5 = A_5\overline{A_6}\,\overline{A_7}$$

$$AP_6 = A_6\overline{A_7}$$

$$AP_7 = A_7$$

Realizacija prioritetne mreže za koder sa 8 ulaza



$$AP_0 = A_0\overline{A_1}\,\overline{A_2}\,\overline{A_3}\,\overline{A_4}\,\overline{A_5}\,\overline{A_6}\,\overline{A_7}$$

$$AP_1 = A_1\overline{A_2}\,\overline{A_3}\,\overline{A_4}\,\overline{A_5}\,\overline{A_6}\,\overline{A_7}$$

$$AP_2 = A_2\overline{A_3}\,\overline{A_4}\,\overline{A_5}\,\overline{A_6}\,\overline{A_7}$$

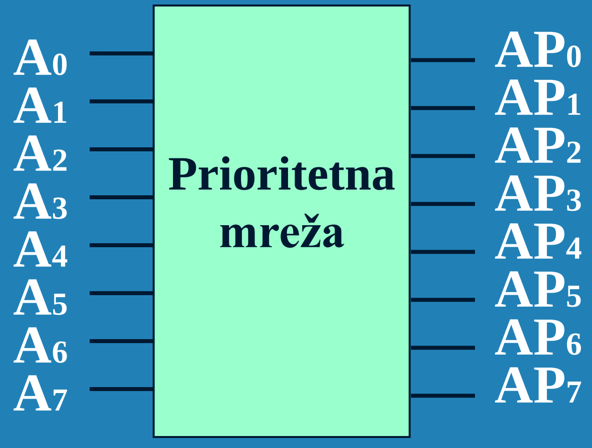
$$AP_3 = A_3\overline{A_4}\,\overline{A_5}\,\overline{A_6}\,\overline{A_7}$$

$$AP_4 = A_4\overline{A_5}\,\overline{A_6}\,\overline{A_7}$$

$$AP_5 = A_5\overline{A_6}\,\overline{A_7}$$

$$AP_6 = A_6\overline{A_7}$$

$$AP_7 = A_7$$



5.3 Dekoderi

- **Dekoderi spadaju u grupu kombinacionih prekidačkih mreža koje dekoduju binarno-kodovanu informaciju.**
- **Oni imaju više ulaza i više izlaza gde svaka *dozvoljena* kombinacija ulaznih promenjivih aktivira jedan i samo jedan izlaz.**
- **Dekoderi mogu biti *potpuni* (oni u kojima za n ulaznih promenljivih postoji 2^n izlaza) i *nepotpuni* (oni kod kojih je broj izlaza manji od 2^n, odnosno kod kojih se određene kombinacije ulaznih signala ne mogu pojaviti).**

- Kod potpunih dekodera na ulaz dovodimo binarno kodovane brojeve, a za svaku kombinaciju ulaznih promenljivih aktivan je *jedan i samo jedan* izlaz iz dekodera.
- Na ulazu je kodovan podatak predstavljen pomoću n promenljivih (bita). Za svaku kombinaciju ulaznih promenljivih predviđen je jedan od 2^n izlaza.

Kombinaciona tablica i funkcije izlaza potpunog dekodera sa tri ulaza.

A	B	C		Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0
0	0	0		0	0	0	0	0	0	0	1
0	0	1		0	0	0	0	0	0	1	0
0	1	0		0	0	0	0	0	1	0	0
0	1	1		0	0	0	0	1	0	0	0
1	0	0		0	0	0	1	0	0	0	0
1	0	1		0	0	1	0	0	0	0	0
1	1	0		0	1	0	0	0	0	0	0
1	1	1		1	0	0	0	0	0	0	0

$$Y_0 = \overline{A}\,\overline{B}\,\overline{C}$$

$$Y_1 = \overline{A}\,\overline{B}C$$

$$Y_2 = \overline{A}B\overline{C}$$

$$Y_3 = \overline{A}BC$$

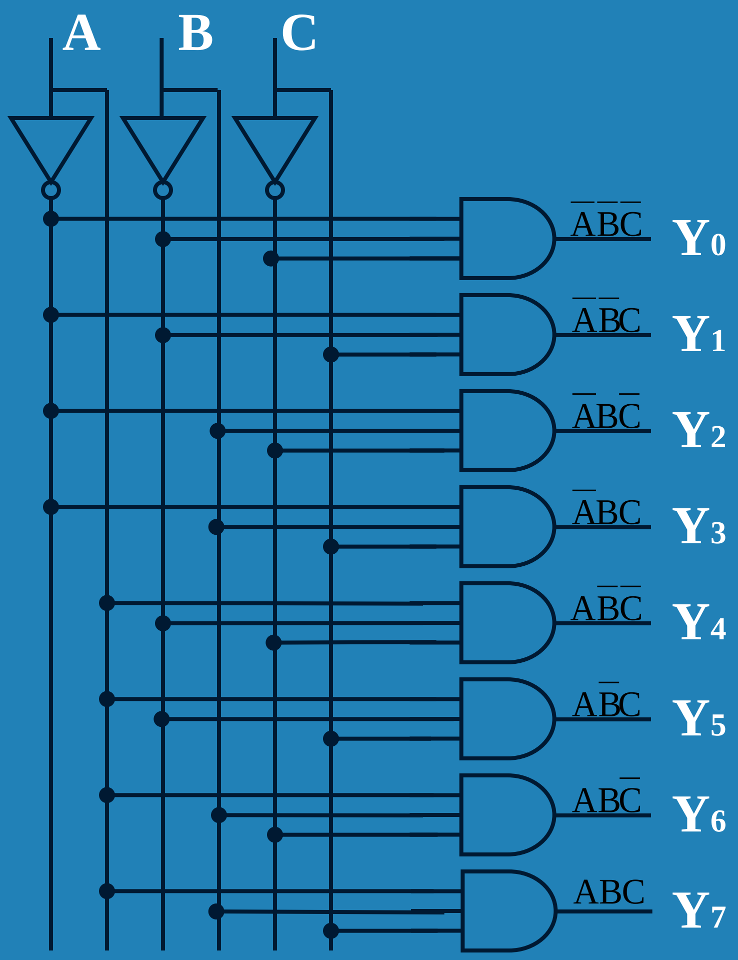
$$Y_4 = A\overline{B}\,\overline{C}$$

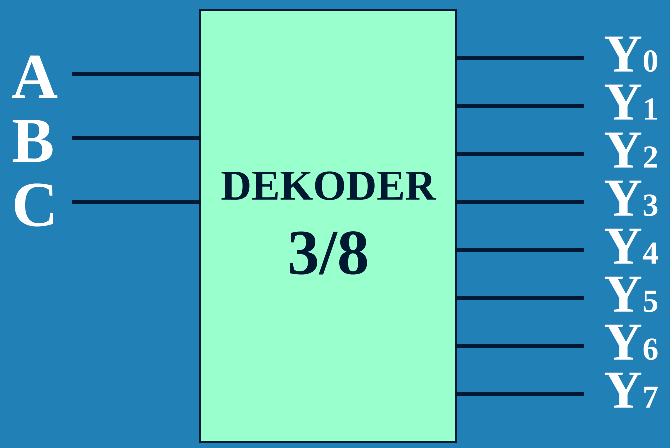
$$Y_5 = A\overline{B}C$$

$$Y_6 = AB\overline{C}$$

$$Y_7 = ABC$$

Realizacija potpunog dekodera sa 3 ulaza





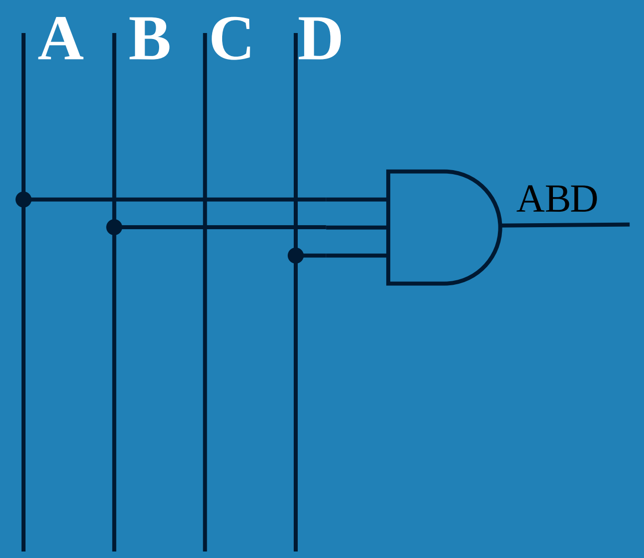
Vežbe

Izvršiti minimizaciju date logičke funkcije i tako dobijenu funkciju realizovati logičkim kolima

$$F = AB\bar{C}D + ABCD$$

AB \ CD	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	0	1	1	0
10	0	0	0	0

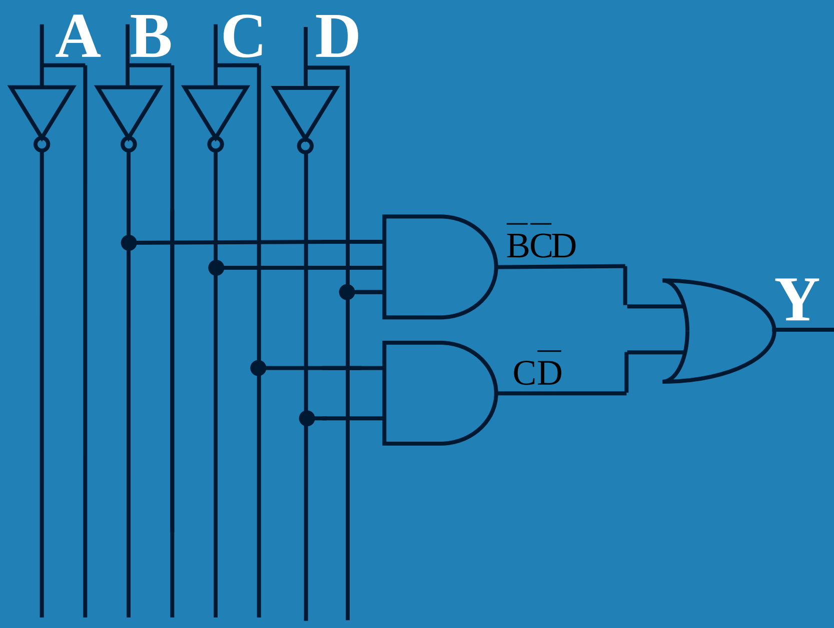
$$F = ABD$$



$$F=\overline{A}\,\overline{B}\,\overline{C}D+A\overline{B}\,\overline{C}D+\overline{A}\,\overline{B}C\overline{D}+\overline{A}BC\overline{D}+ABC\overline{D}+A\overline{B}C\overline{D}$$

AB \ CD	00	01	11	10
00	0	1	0	1
01	0	0	0	1
11	0	0	0	1
10	0	1	0	1

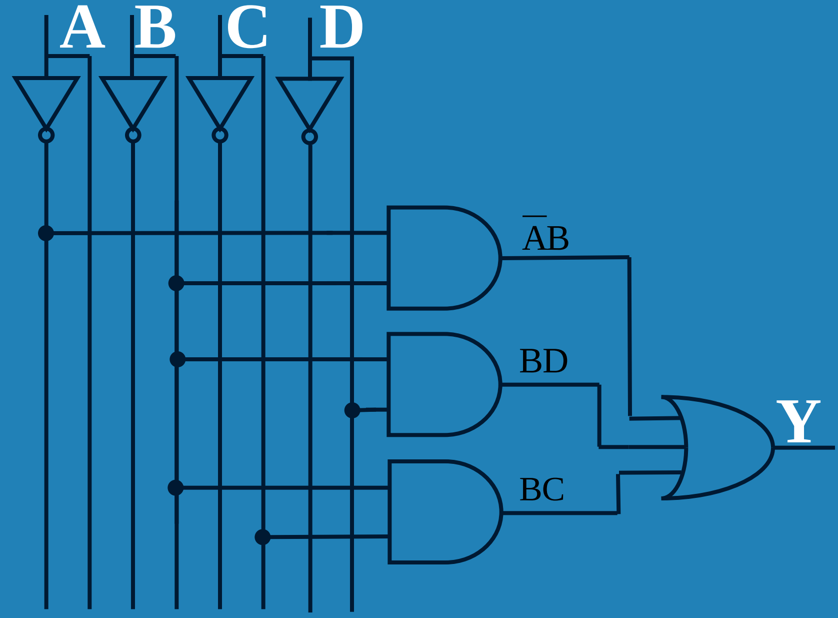
$$F=\overline{B}\,\overline{C}D+C\overline{D}$$



$$F(1) = (4,5,6,7,13,14,15)$$

AB \ CD	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	0	1	1	1
10	0	0	0	0

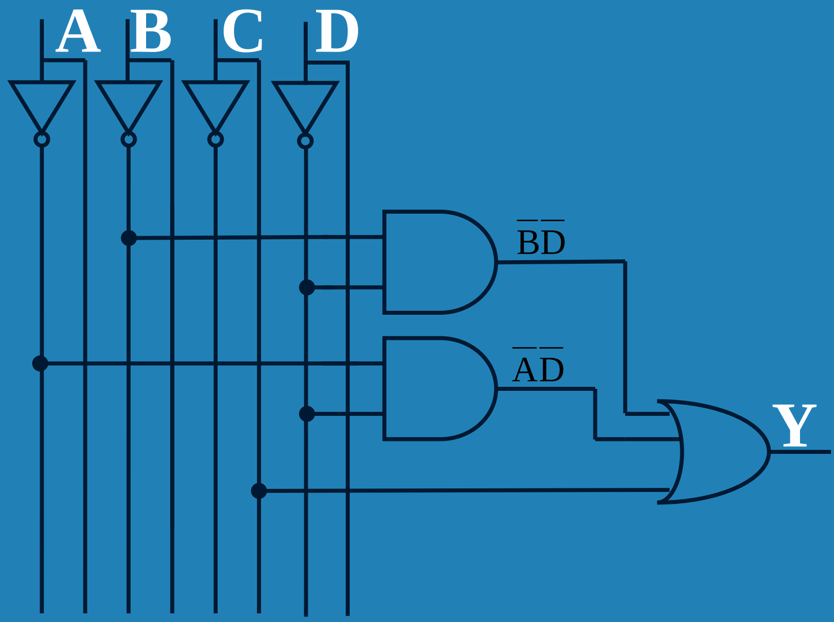
$$F = \overline{A}B + BD + BC$$



$$F(0) = (1,5,9,12,13)$$

AB \ CD	00	01	11	10
00	1	0	1	1
01	1	0	1	1
11	0	0	1	1
10	1	0	1	1

$$F = \overline{B}\,\overline{D} + \overline{A}\,\overline{D} + C$$

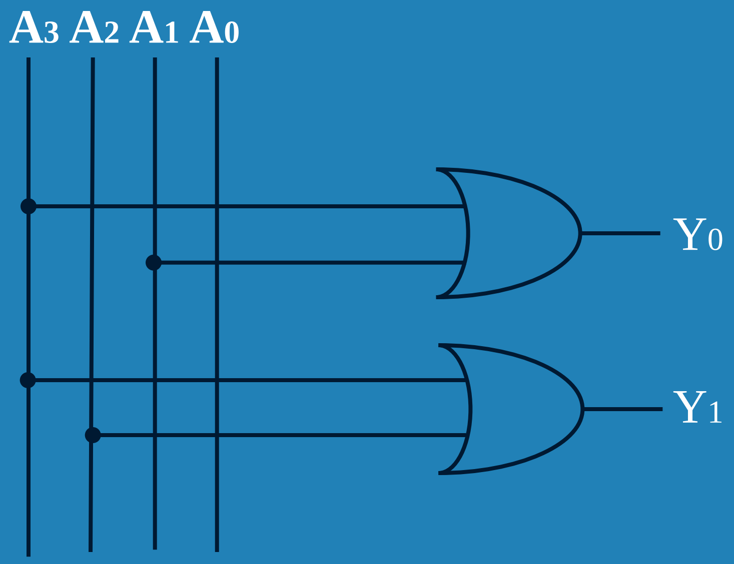


Realizovati koder 4/2

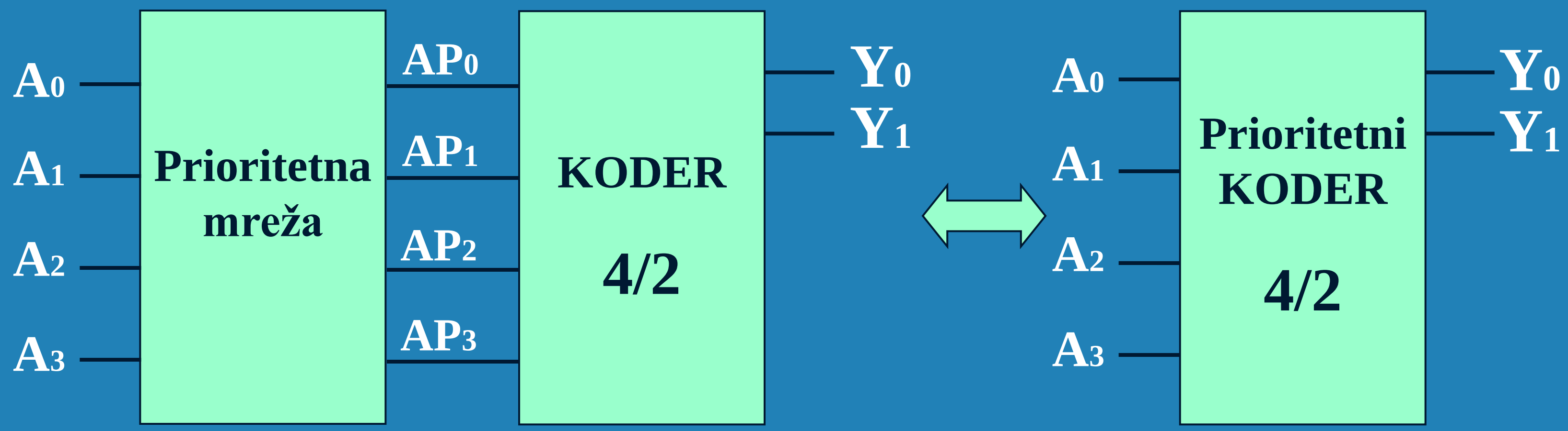
$$Y_0 = A_1 + A_3$$

$$Y_1 = A_2 + A_3$$

A_3	A_2	A_1	A_0	Y_1	Y_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1



Realizovati prioritetnu mrežu za koder 4/2



A_3	A_2	A_1	A_0	AP_3	AP_2	AP_1	AP_0
1	b	b	b	1	0	0	0
0	1	b	b	0	1	0	0
0	0	1	b	0	0	1	0
0	0	0	1	0	0	0	1

$$AP_0 = A_0\overline{A_1}\,\overline{A_2}\,\overline{A_3}$$

$$AP_1 = A_1\overline{A_2}\,\overline{A_3}$$

$$AP_2 = A_2\overline{A_3}$$

$$AP_3 = A_3$$

Realizacija logičke funkcije pomoću dekodera

- **Pomoću dekodera realizovati sledeću logičku funkciju:**

$$Y = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$