



Univerzitet u Novom Sadu

Fakultet tehničkih nauka

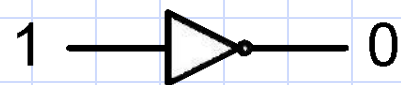
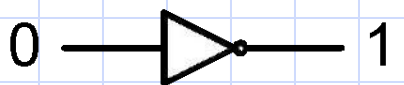
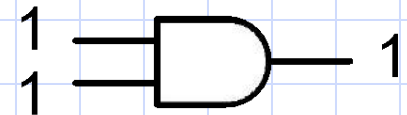
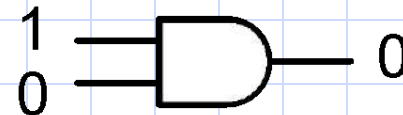
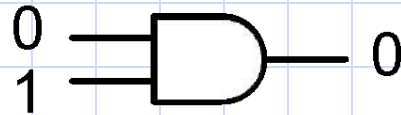
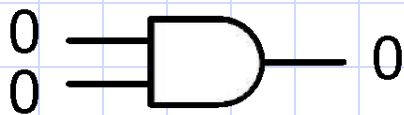
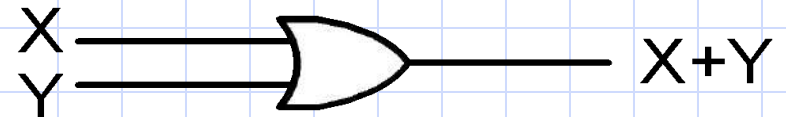
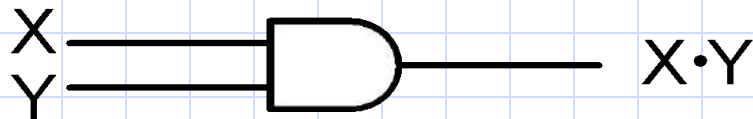
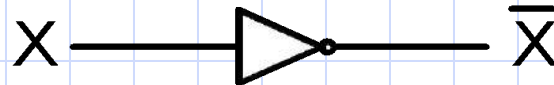
Odsek za računarsku tehniku i
računarske komunikacije



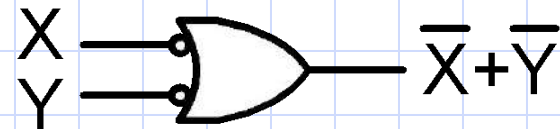
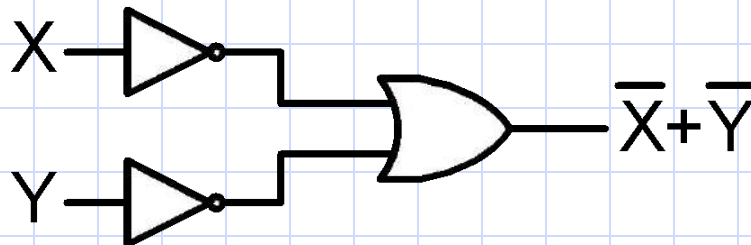
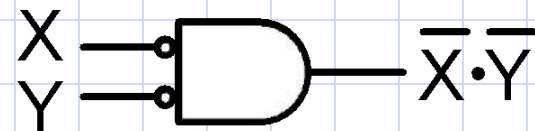
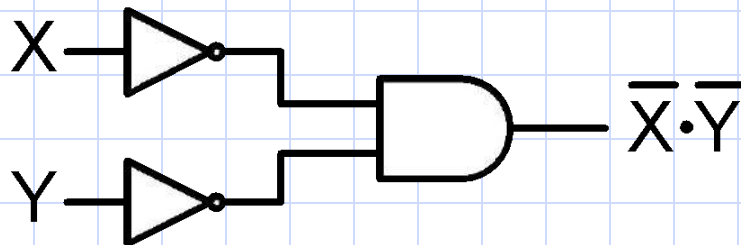
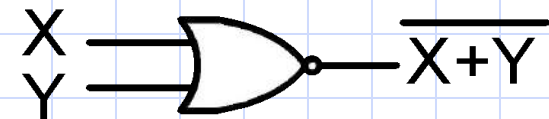
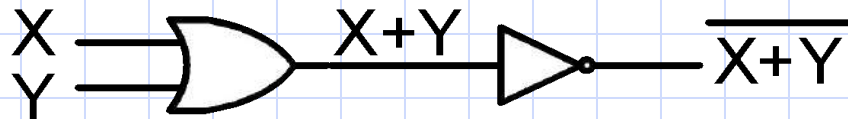
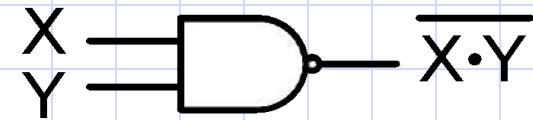
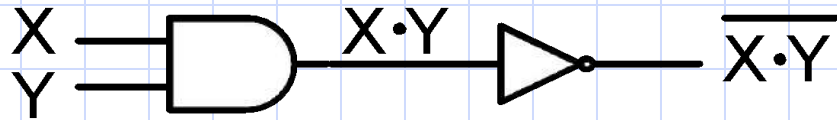
Standardne kombinacione mreže

Dekoderi, koderi, multiplekseri, demultiplekseri,
iterativne mreže i mreže tipa stablo

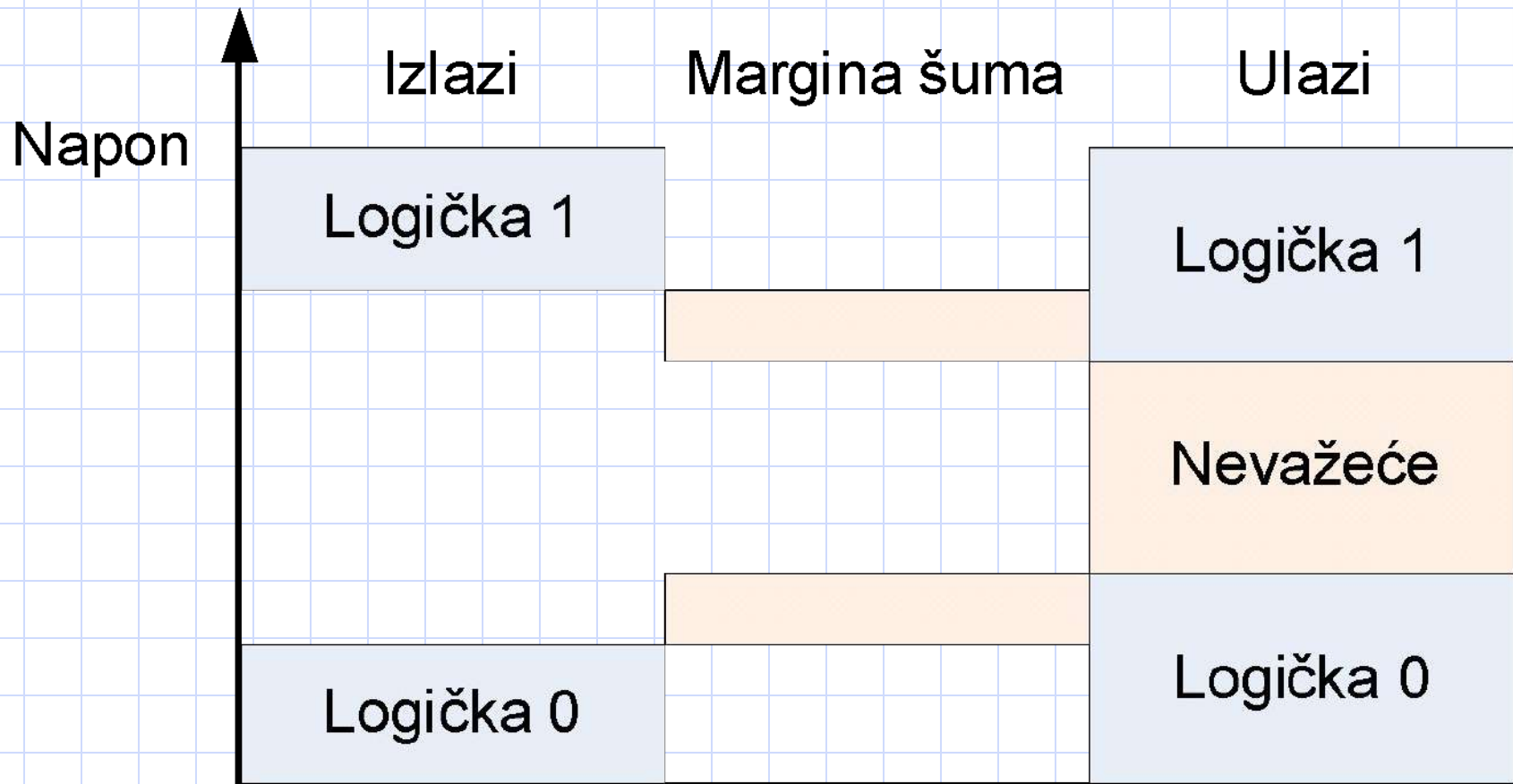
Logička kola



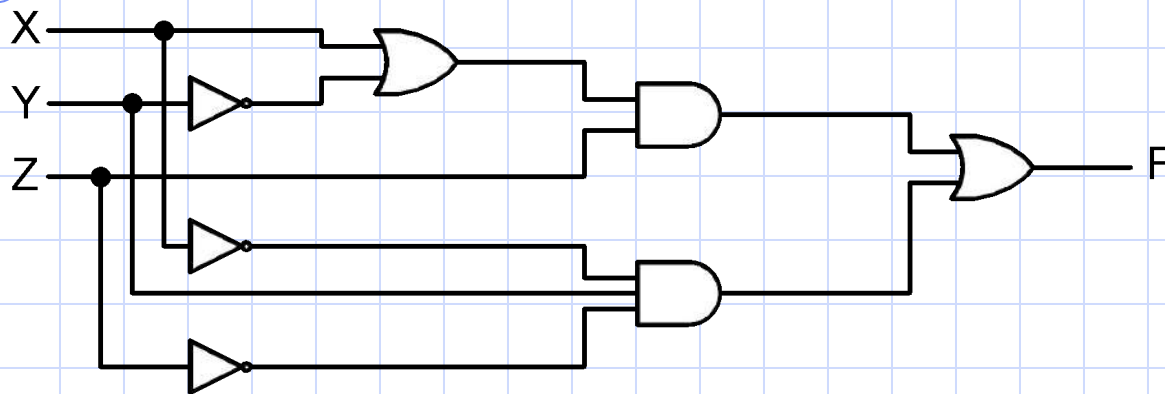
Oznake ekvivalentnih logičkih kola



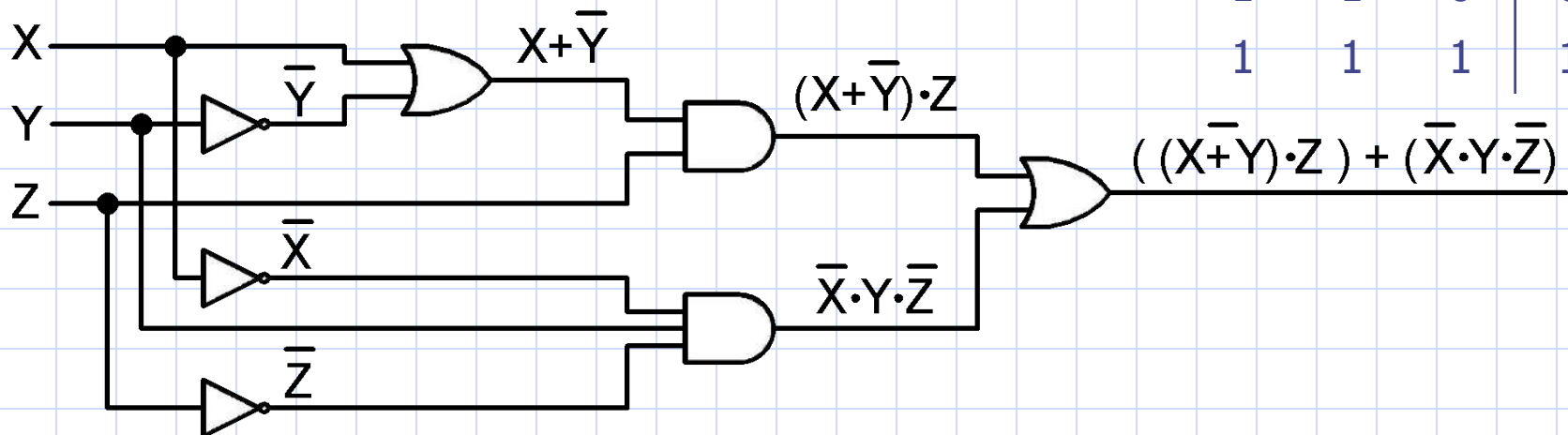
Naponski nivoi



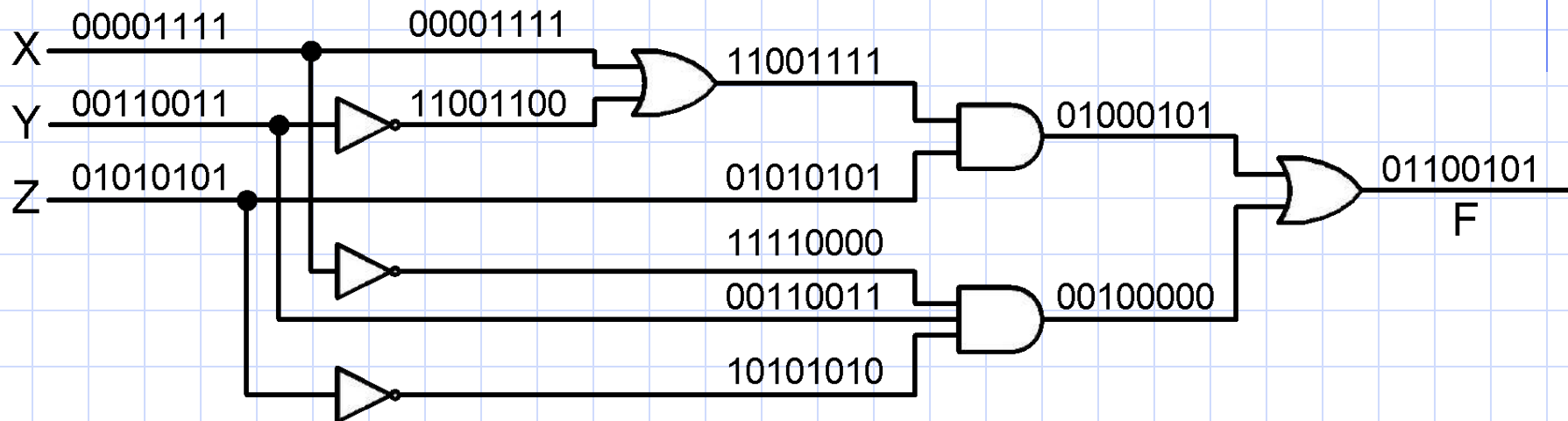
Primer kombinacione mreže



X	Y	Z	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1



Ilustracija rada kombinacione mreže

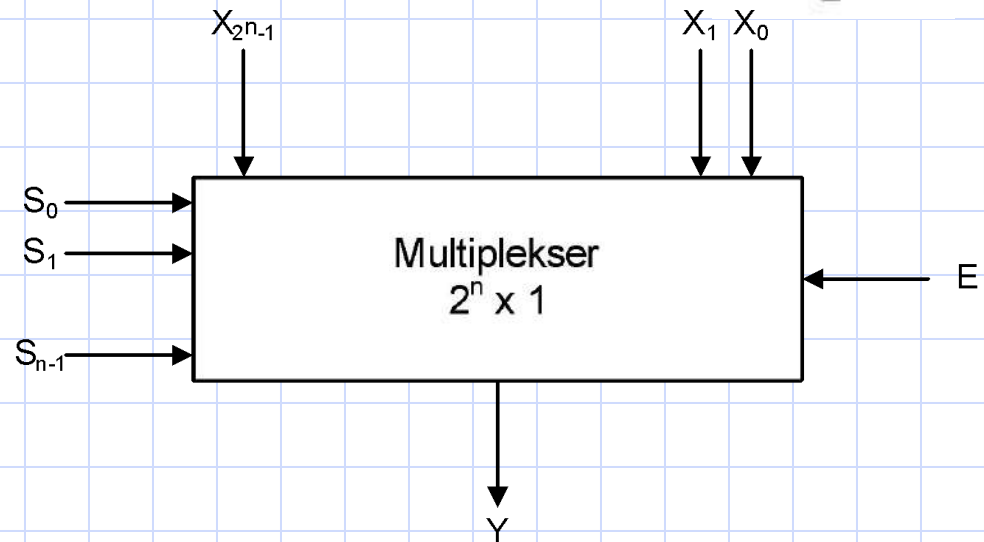


Propagacija signala u realnom vremenu!

Logičko kolo unosi kašnjenje prilikom generisanje izlaza u odnosu na trenutak promene signala na njegovom ulazu

Multiplekser

Multiplekseri su kombinacione mreže sa 2^n ulaznih priključaka, n adresnih (upravljačkih $S_{n-1}, S_{n-2}, \dots, S_0$), ulazom dozvole \bar{E} i jednim izlaznim priključkom



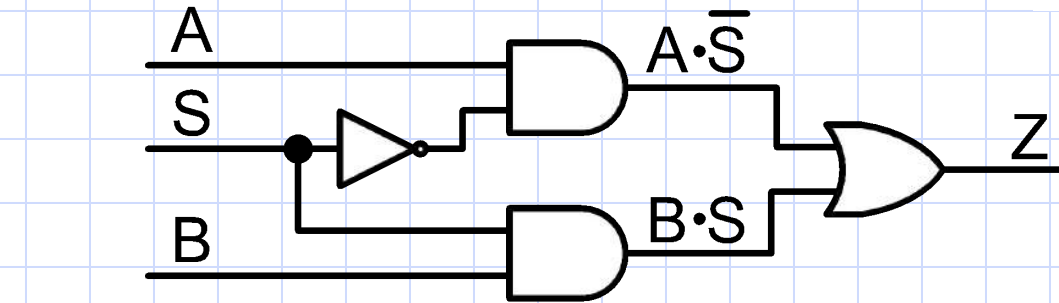
Multiplekser se analitički definiše u obliku:
$$Y = \begin{cases} x_s & \text{ako je } E = 1 \\ 0 & \text{ako je } E = 0 \end{cases}$$

gde je $s = \sum_{j=0}^{n-1} s_j 2^j$ odnosno $Y = E \cdot \sum_{i=1}^n x_i p_i(s)$

gde su p_i potpuni proizvodi (konstituente jedinice) adresnih promenljivih

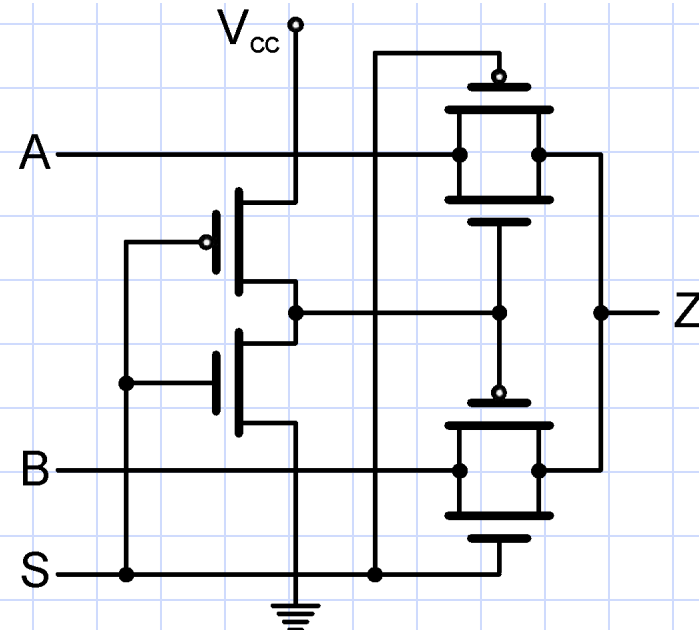
Multiplexer 2x1

S	A	B	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

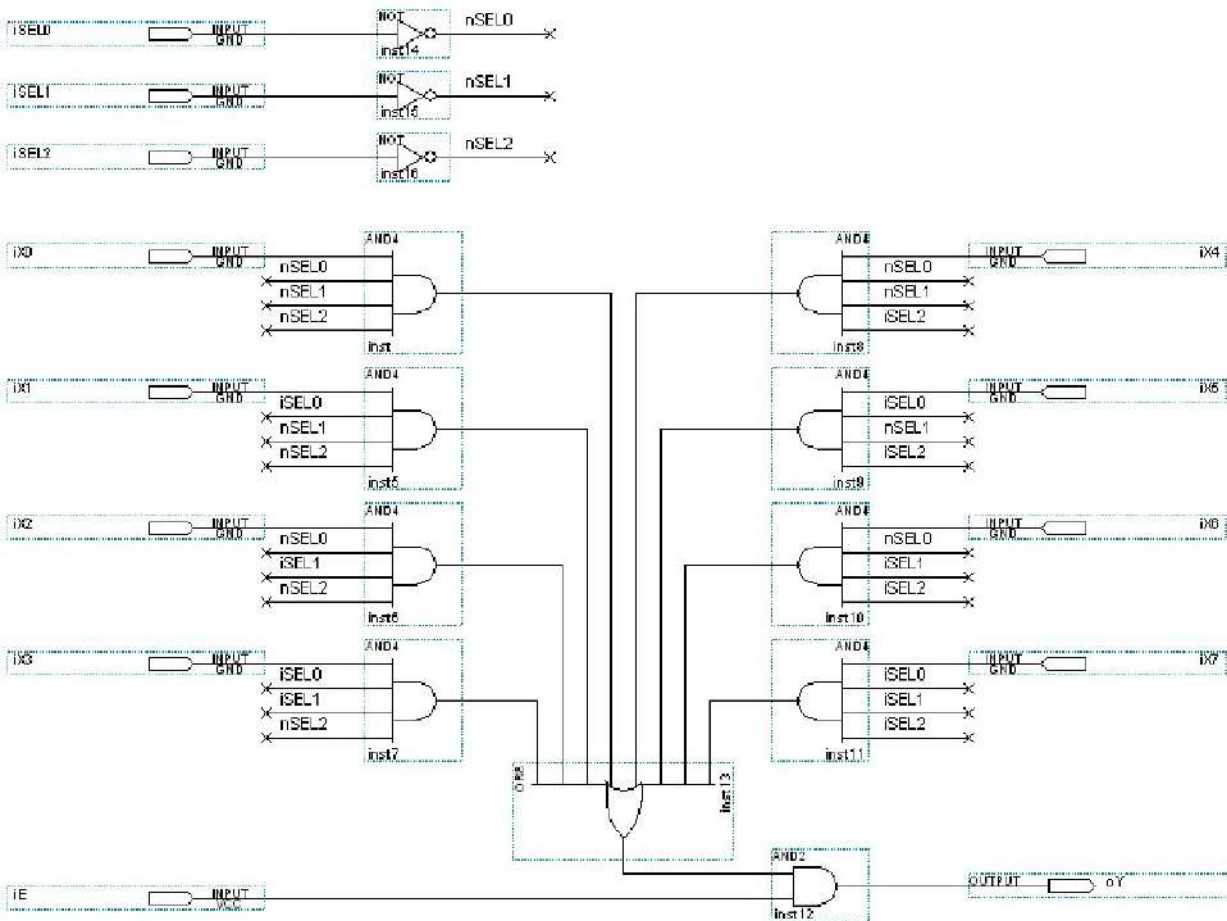


$Z \leq A$ when $S = '0'$ else B ;

$$Z = \bar{S} \cdot A + S \cdot B$$



Multiplexer 8x1



E	S ₂	S ₁	S ₀	Y
0	×	×	×	0
1	0	0	0	X ₀
1	0	0	1	X ₁
1	0	1	0	X ₂
1	0	1	1	X ₃
1	1	0	0	X ₄
1	1	0	1	X ₅
1	1	1	0	X ₆
1	1	1	1	X ₇

E – signal
dozvole
izlaza

Multiplexer 8x1

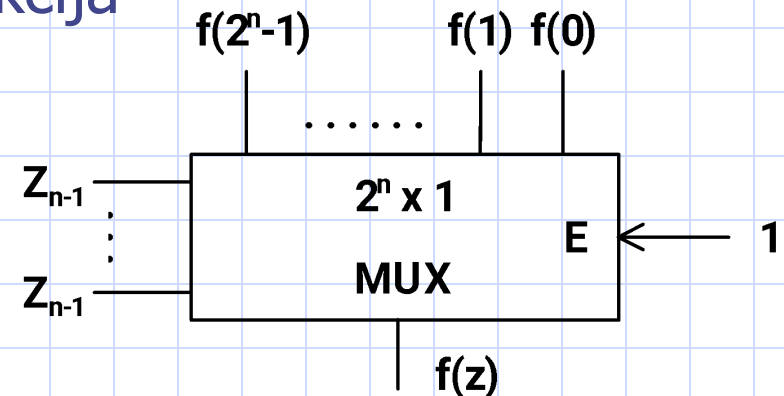
```
oY <= (iX(0) AND NOT(iSEL(2)) AND NOT(iSEL (1)) AND NOT(iSEL(0))) OR
(iX(1) AND NOT(iSEL(2)) AND NOT(iSEL (1)) AND (iSEL(0))) OR
(iX(2) AND NOT(iSEL(2)) AND (iSEL (1)) AND NOT(iSEL(0))) OR
(iX(3) AND NOT(iSEL(2)) AND (iSEL (1)) AND (iSEL(0))) OR
(iX(4) AND (iSEL(2)) AND NOT(iSEL (1)) AND NOT(iSEL(0))) OR
(iX(5) AND (iSEL(2)) AND NOT(iSEL (1)) AND (iSEL(0))) OR
(iX(6) AND (iSEL(2)) AND (iSEL (1)) AND NOT(iSEL(0))) OR
(iX(7) AND (iSEL(2)) AND (iSEL (1)) AND (iSEL(0))) OR

-- provera vrednosti signala dozvole rada
IF (iE = '1') THEN
    -- multipleksiranje dozvoljeno ->
    -- odredjivanje izlaznog signala
    -- u zavisnosti od vrednosti adresnog vektora
    CASE iSEL IS
        WHEN "000" => oY <= iX(0);
        WHEN "001" => oY <= iX(1);
        WHEN "010" => oY <= iX(2);
        WHEN "011" => oY <= iX(3);
        WHEN "100" => oY <= iX(4);
        WHEN "101" => oY <= iX(5);
        WHEN "110" => oY <= iX(6);
        WHEN OTHERS => oY <= iX(7);
    END CASE;
ELSE
    -- multipleksiranje nije dozvoljeno ->
    -- postavljanje definisane vrednosti na izlaz
    oY <= '0';
END IF;
```

Primena multipleksera

- ❖ Multiplekser kao selektor izvora
 - ❖ U slučaju da je izvor vektor sa proizvoljnim brojem bita koriste se vektorski multiplekseri

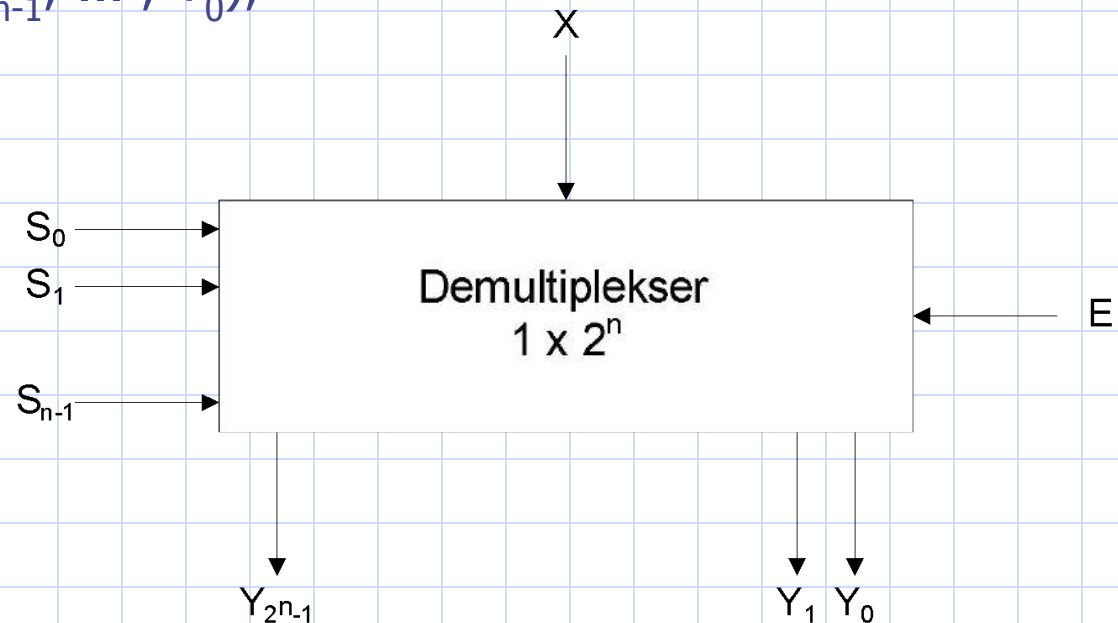
- ❖ Multiplekser kao generator funkcija



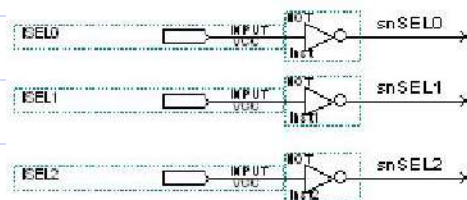
- ❖ Paralelno serijska konverzija
 - ❖ Informacije se dovode istovremeno na ulaze multipleksera, a kontinualnim menjanjem adresne informacije na ulazu izabiraju se izlazni priključci redom jedan za drugim, i spajaju na izlaz

Demultiplekser

- ❖ Demultiplekser je kombinaciona mreža sa
 - ❖ jednim ulazom X ,
 - ❖ n adresnih (selekcioni) promenljivih $S=(S_{n-1}, \dots, S_0)$,
 - ❖ priključkom za upravljanje (dozvola) E i
 - ❖ 2^n izlaza $Y = (Y_{2^n-1}, \dots, Y_0)$,



Demultiplexer



E	S ₂	S ₁	S ₀	Y ₇	Y ₆	Y ₅	Y ₄	Y ₃	Y ₂	Y ₁	Y ₀
0	X	X	X	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	X
1	0	0	1	0	0	0	0	0	0	X	0
1	0	1	0	0	0	0	0	0	X	0	0
1	0	1	1	0	0	0	0	X	0	0	0
1	1	0	0	0	0	0	X	0	0	0	0
1	1	0	1	0	0	X	0	0	0	0	0
1	1	1	0	0	X	0	0	0	0	0	0
1	1	1	1	X	0	0	0	0	0	0	0

$$Y_0 = \overline{S_2} \cdot \overline{S_1} \cdot \overline{S_0} \cdot E \cdot X$$

$$Y_1 = \overline{S_2} \cdot \overline{S_1} \cdot S_0 \cdot E \cdot X$$

$$Y_2 = \overline{S_2} \cdot S_1 \cdot \overline{S_0} \cdot E \cdot X$$

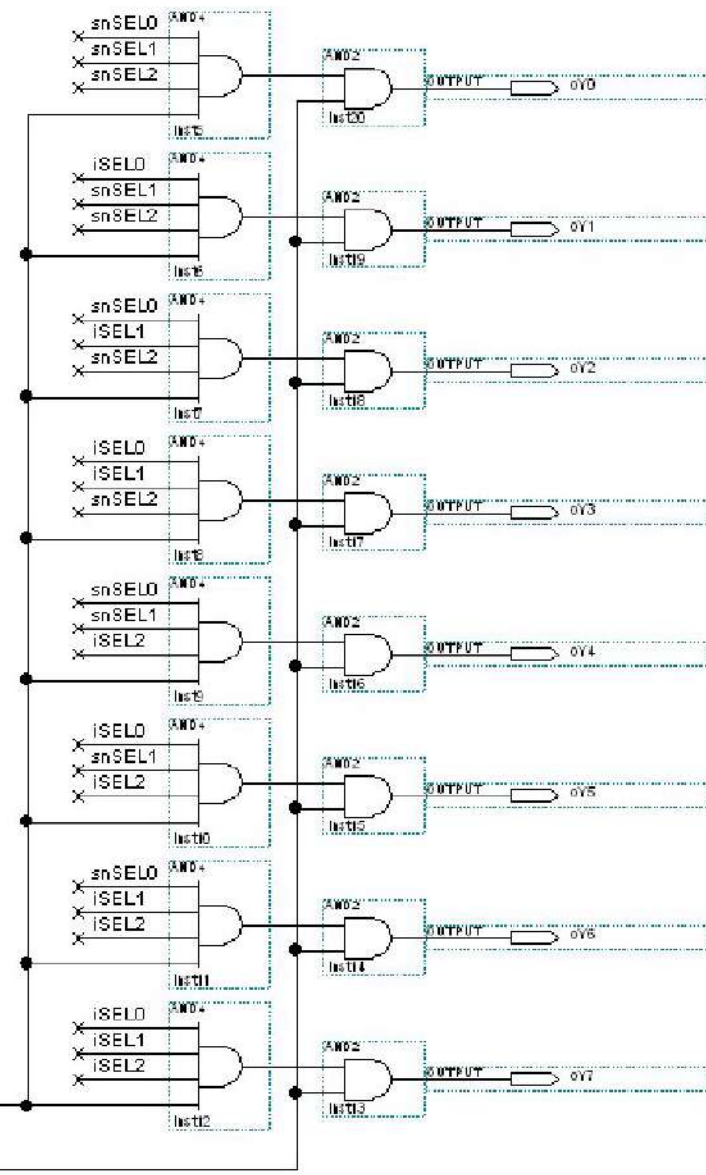
$$Y_3 = \overline{S_2} \cdot S_1 \cdot S_0 \cdot E \cdot X$$

$$Y_4 = S_2 \cdot \overline{S_1} \cdot \overline{S_0} \cdot E \cdot X$$

$$Y_5 = S_2 \cdot \overline{S_1} \cdot S_0 \cdot E \cdot X$$

$$Y_6 = S_2 \cdot S_1 \cdot \overline{S_0} \cdot E \cdot X$$

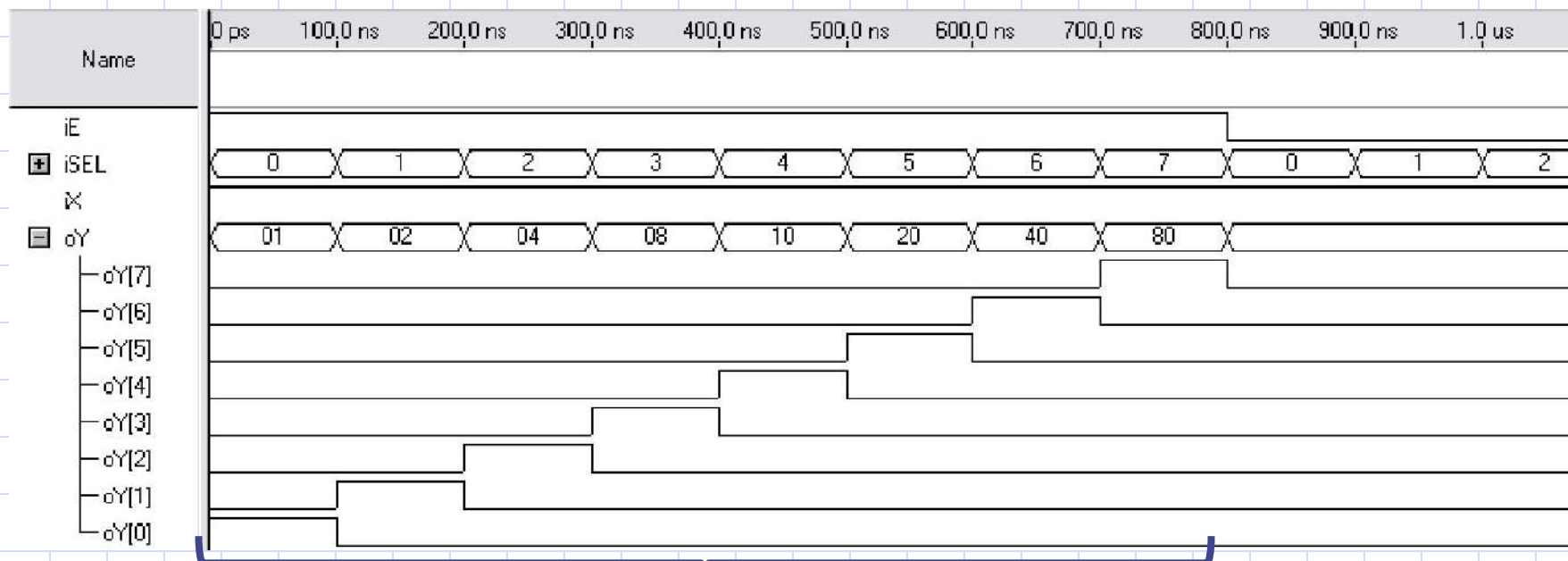
$$Y_7 = S_2 \cdot S_1 \cdot S_0 \cdot E \cdot X$$



Demultiplekser u VHDL-u

```
PROCESS (iX, iE, iSEL) BEGIN
  oY <= "00000000"; -- inicijalizacija izlaznog vektora
  -- provera signala dozvole rada
  IF (iE = '1') THEN
    -- demultipleksiranje dozvoljeno ->
    -- prosledjivanje ulaznog signala na izlaz
    -- u zavisnosti od vrednosti adresnog vektora
    CASE iSEL IS
      WHEN "000" => oY(0) <= iX;
      WHEN "001" => oY(1) <= iX;
      WHEN "010" => oY(2) <= iX;
      WHEN "011" => oY(3) <= iX;
      WHEN "100" => oY(4) <= iX;
      WHEN "101" => oY(5) <= iX;
      WHEN "110" => oY(6) <= iX;
      WHEN OTHERS => oY(7) <= iX;
    END CASE;
  ELSE
    -- demultipleksiranje nije dozvoljeno ->
    -- dodela predefinisane vrednosti izlaznom signalu
    oY <= "00000000";
  END IF;
END PROCESS;
```

Vremenska simulacija rada demultipleksera



Aktivan signal dozvole izlaza

Signal dozvole izlaza neaktivan

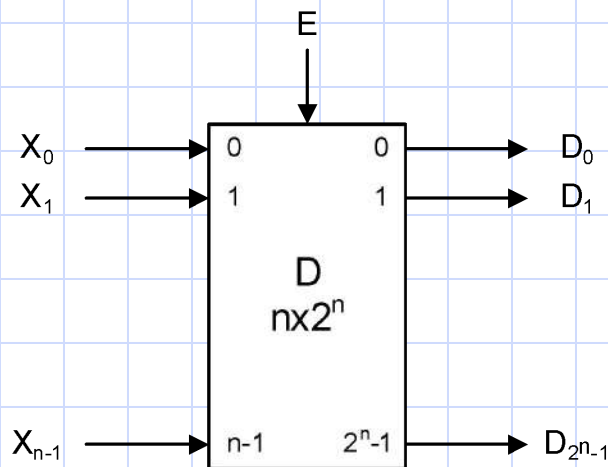
Dekoder

Dekoder je kombinaciona mreža koja poseduje n ulaznih i 2^n izlaznih priključaka.
Mreža realizuje 2^n Bulovih funkcija:

$$D_j(X) = \begin{cases} 1 & \text{za } x = j \\ 0 & \text{za } x \neq j \end{cases}$$

gde su $j = 0, 1, 2, \dots, 2^n - 1$; i

$$X = \sum_{i=1}^n x_i 2^{i-1} \quad x_i \in \{0, 1\}$$



$$D_0(X) = \bar{x}_{n-1} \bar{x}_{n-2} \dots \bar{x}_2 \bar{x}_1 \bar{x}_0$$

$$D_1(X) = \bar{x}_{n-1} \bar{x}_{n-2} \dots \bar{x}_2 \bar{x}_1 x_0$$

$$D_2(X) = \bar{x}_{n-1} \bar{x}_{n-2} \dots \bar{x}_2 x_1 \bar{x}_0$$

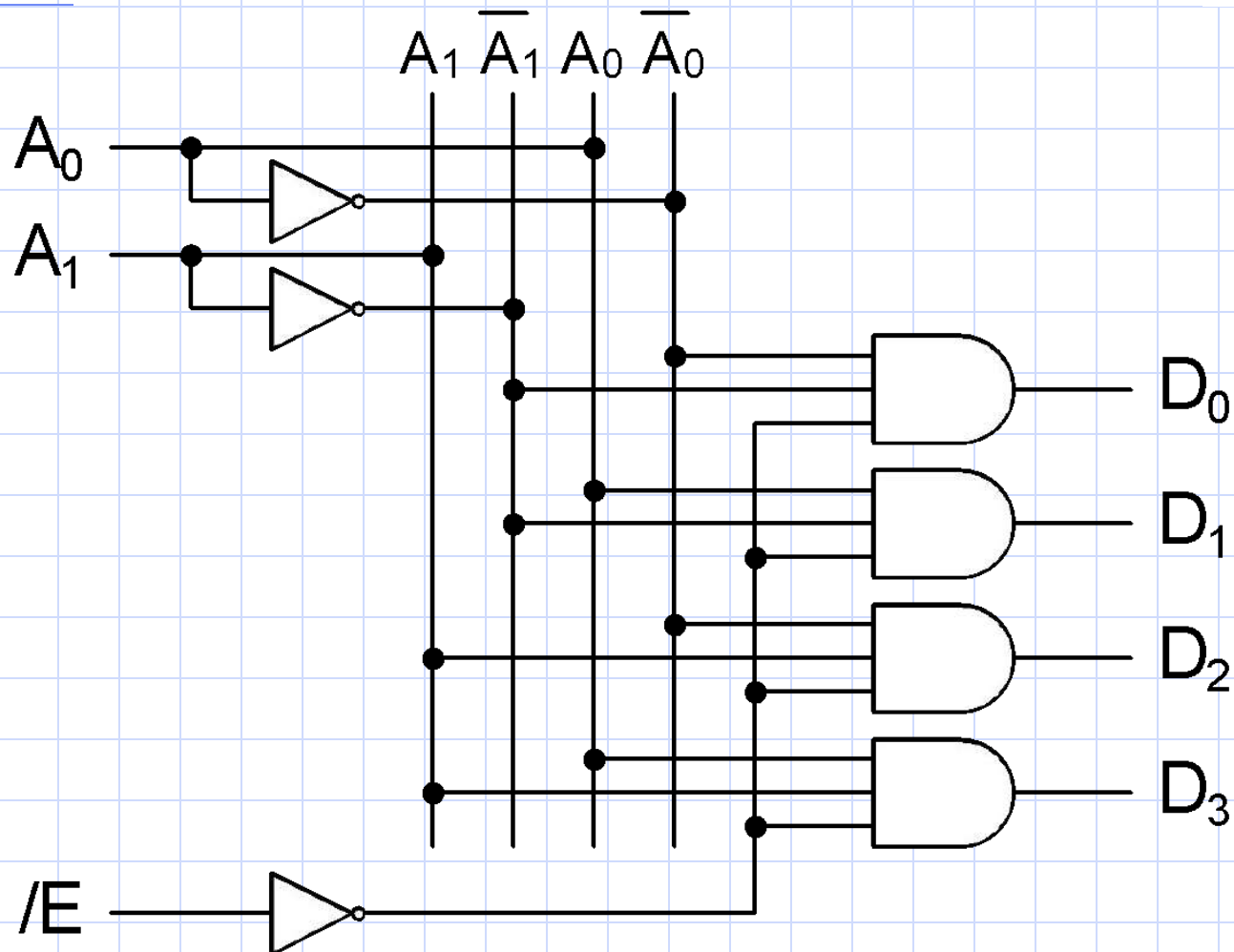
⋮

$$D_{2^n-2}(X) = x_{n-1} x_{n-2} \dots x_2 x_1 \bar{x}_0$$

$$D_{2^n-1}(X) = x_{n-1} x_{n-2} \dots x_2 x_1 x_0$$

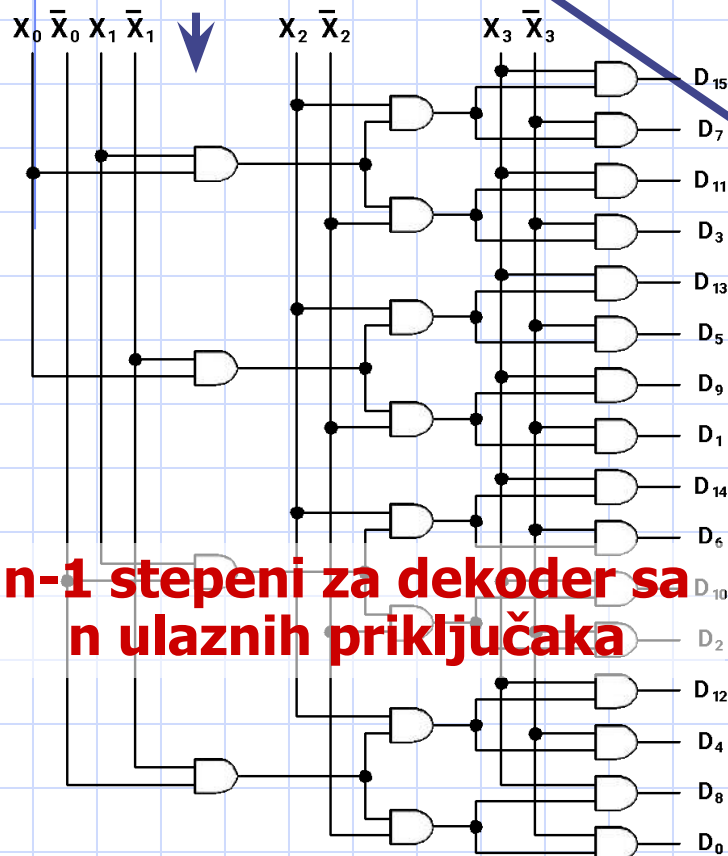
Dekoder transformiše n -elementarni binarni težinski kod u 2^n -elementarni kod, jer svakoj reči na ulazu dekodera odgovara jedinični signal samo na jednom izlaznom priključku.

Dekoder 2x4

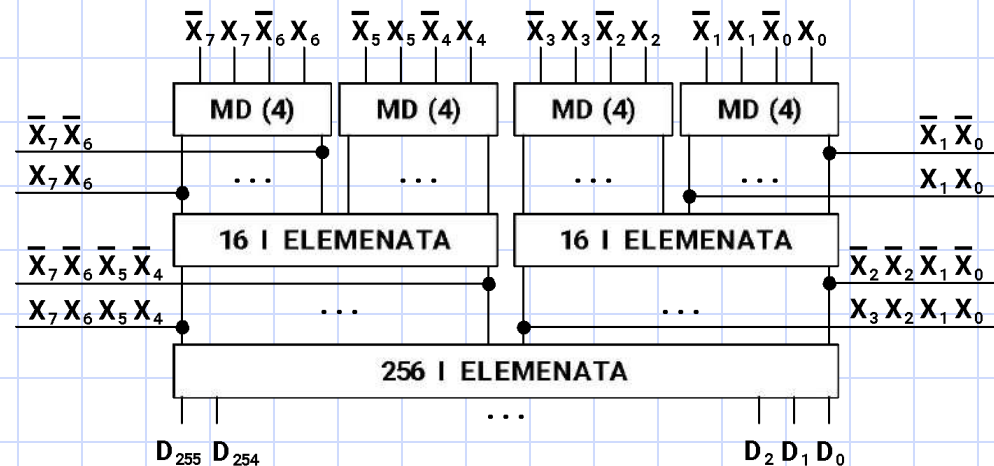
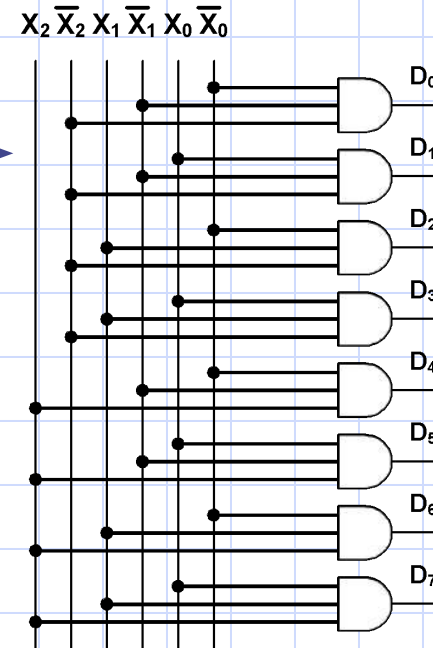


Tipovi dekodera

- ❖ Matrični (linearni),
- ❖ Kaskadni i
- ❖ Piramidalni

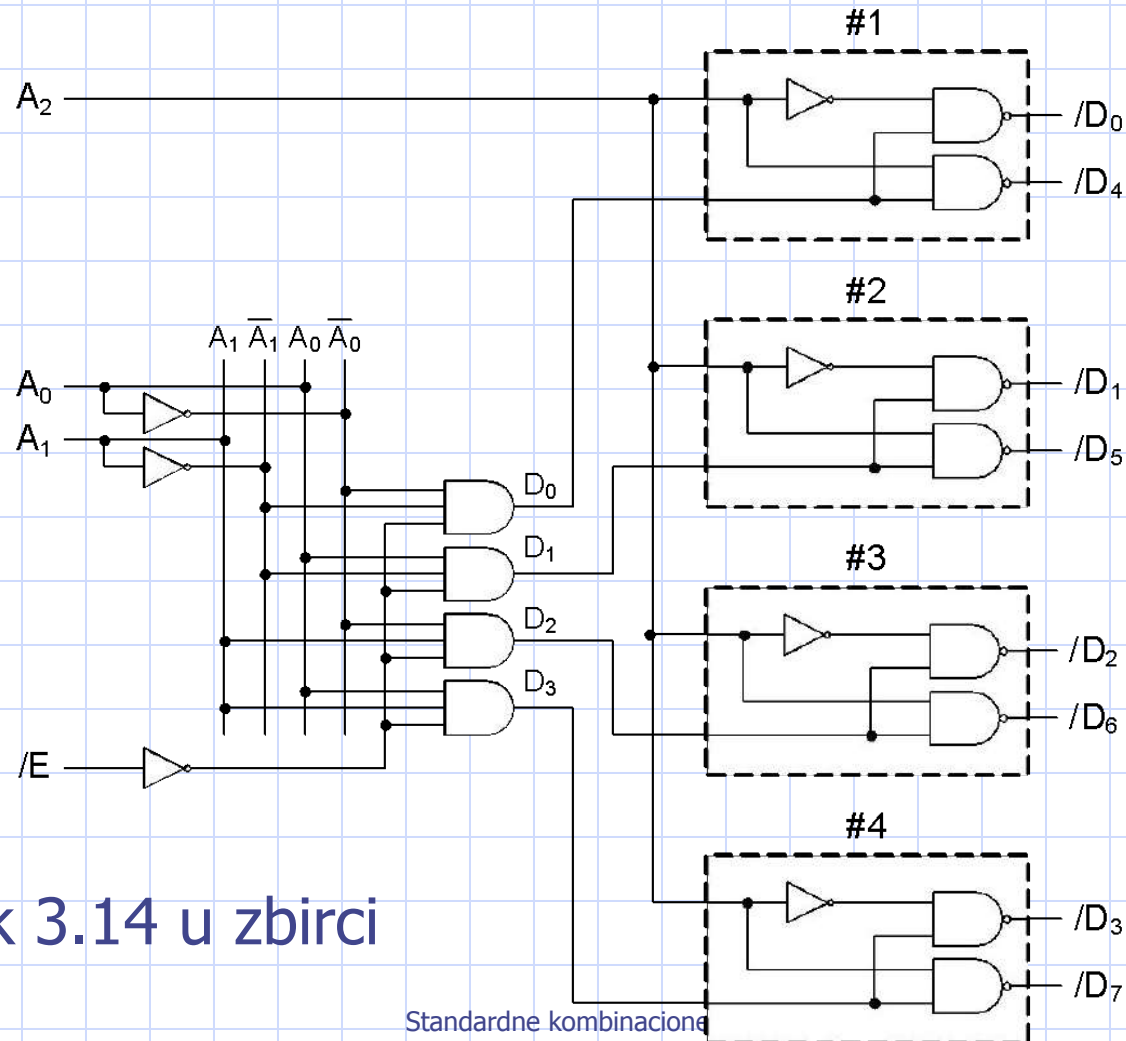


n-1 stepeni za dekoader sa n ulaznih priključaka



Ilustracija kaskadnog dekodera

Dekoder 3x8 formiran od dekodera 2x4 i 1x2



Zadatak 3.14 u zbirci

Signal dozvole dekodovanja

problem

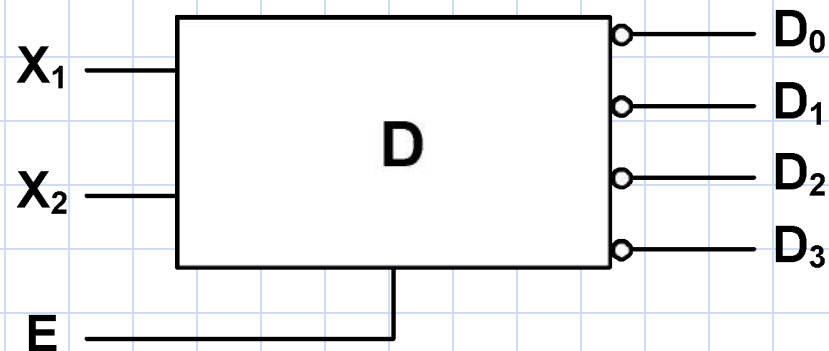
X1	X2	D0	D1	D2	D3
0	0	0	1	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

Kako prepoznati na osnovu stanja izlaznih signala kada je ulazni signal validan?

Signal dozvole dekodovanja

rešenje

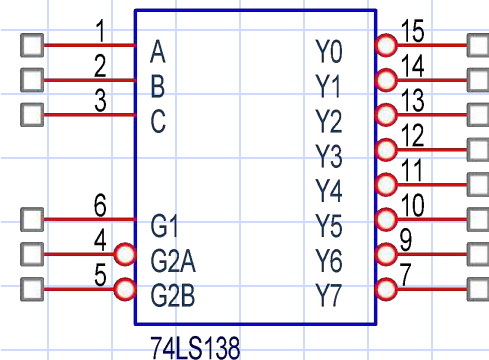
E	X1	X2	D0	D1	D2	D3
1	x	x	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0



Primer dekodera: IK: 74LS138

Matrični dekodер 3x8

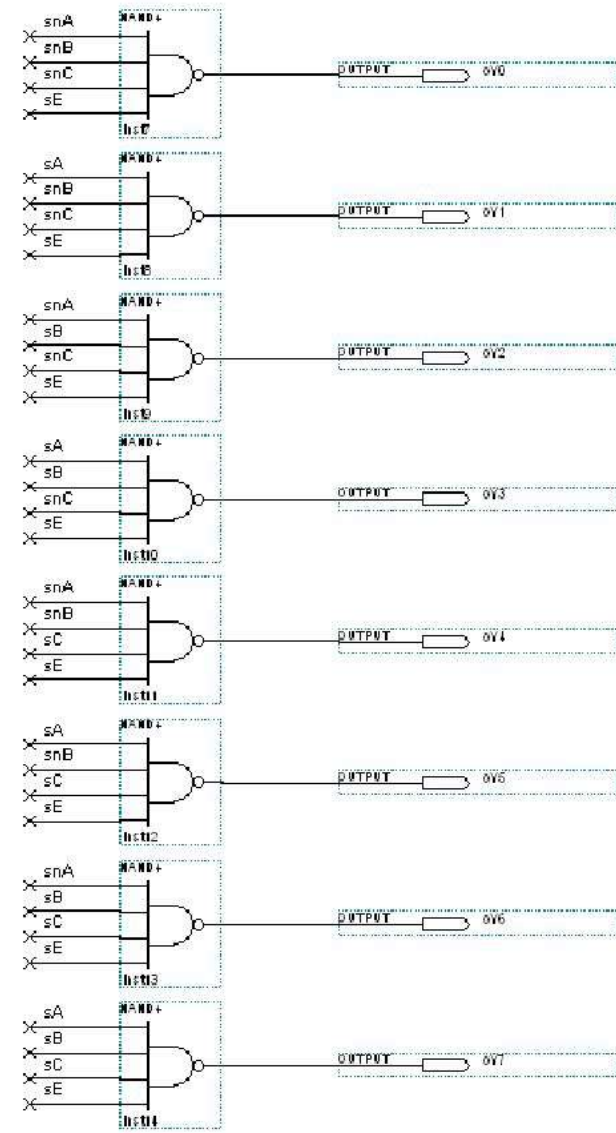
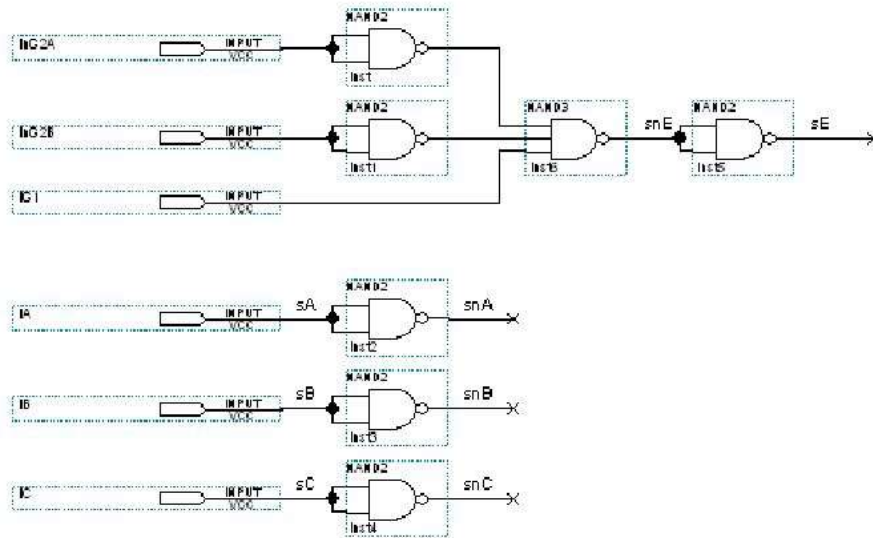
ULAZI DEKODERA					IZLAZI DEKODERA							
G_1	$\overline{G_2}^*$	C	B	A	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7
X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	H	H	H	H	H	L	H	H	H	H
H	L	H	L	L	H	H	H	H	L	H	H	H
H	L	H	L	H	H	H	H	H	H	L	H	H
H	L	H	H	L	H	H	H	H	H	H	L	H
H	L	H	H	H	H	H	H	H	H	H	H	L



$$E = G_1 \cdot \overline{G_2A} \cdot \overline{G_2B}$$

$$Y_0 = \overline{E} \cdot \overline{C} \cdot \overline{B} \cdot \overline{A}, \quad Y_1 = \overline{E} \cdot \overline{C} \cdot \overline{B} \cdot A, \quad Y_2 = \overline{E} \cdot \overline{C} \cdot B \cdot \overline{A}, \quad Y_3 = \overline{E} \cdot \overline{C} \cdot B \cdot A, \\ Y_4 = \overline{E} \cdot C \cdot \overline{B} \cdot \overline{A}, \quad Y_5 = \overline{E} \cdot C \cdot \overline{B} \cdot A, \quad Y_6 = \overline{E} \cdot C \cdot B \cdot \overline{A}, \quad Y_7 = \overline{E} \cdot C \cdot B \cdot A,$$

Logička šema 74LS138

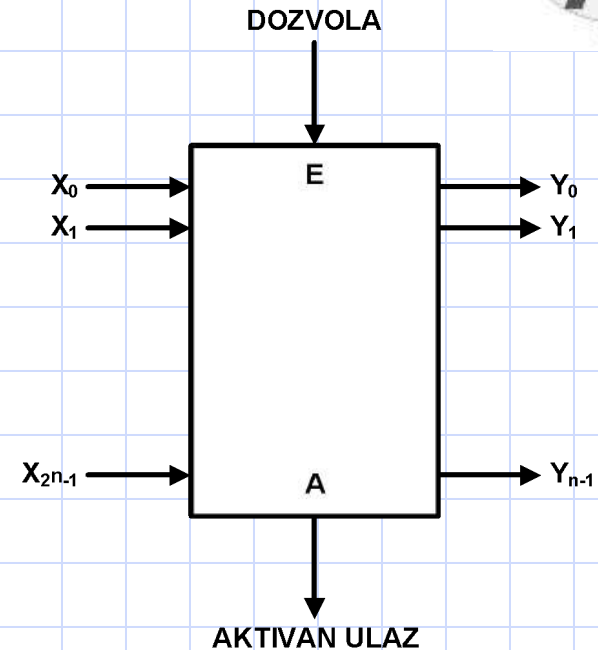
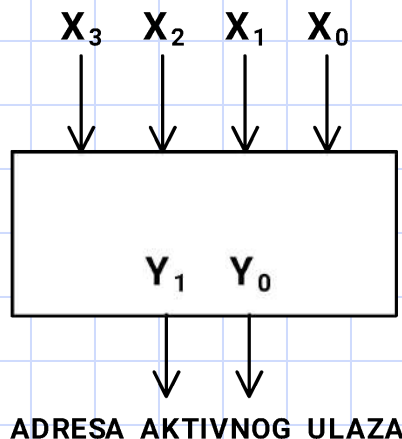


Dekoder u VHDL-u

```
PROCESS (iX, iE) BEGIN
    -- provera zadovoljenja uslova dozvole dekodovanja
    IF (iE = '0') THEN
        -- dekodovanje je dozvoljeno ->
        -- formira se izlaz dekodera
        CASE iX IS
            WHEN "00"    => oD <= "1110";
            WHEN "01"    => oD <= "1101";
            WHEN "10"    => oD <= "1011";
            WHEN "11"    => oD <= "0111";
            WHEN OTHERS => oD <= "1111";
        END CASE;
    ELSE
        -- dekodovanje nije dozvoljeno ->
        -- svi izlazi dekodera su postavljeni na 1
        oD <= "1111";
    END IF;
END PROCESS;
```


Koder

- ❖ Koderi su kombinacione mreže sa 2^n ulaza i n izlaza (izlazi se nazivaju adrese)
- ❖ Koder obavlja inverznu operaciju u odnosu na dekode, jer predstavlja pretvarač koda "jedan od 2^n " u binarni kod



X_3	X_2	X_1	X_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

Prioritetni koder

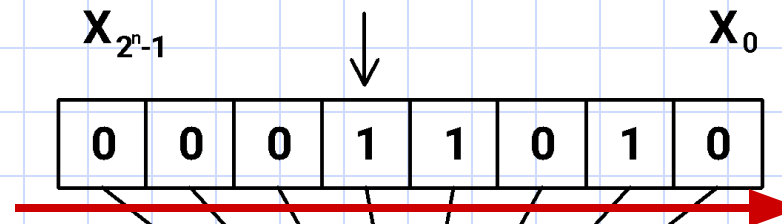
problem

**Kako definisati izlaz koda
u slučaju da je istovremeno
aktivno više od jednog ulaza ?**

rešenje

Definiše se lista prioriteta za ulazne signale

Ulaz najvećeg
prioriteta



Ulaz najnižeg
prioriteta



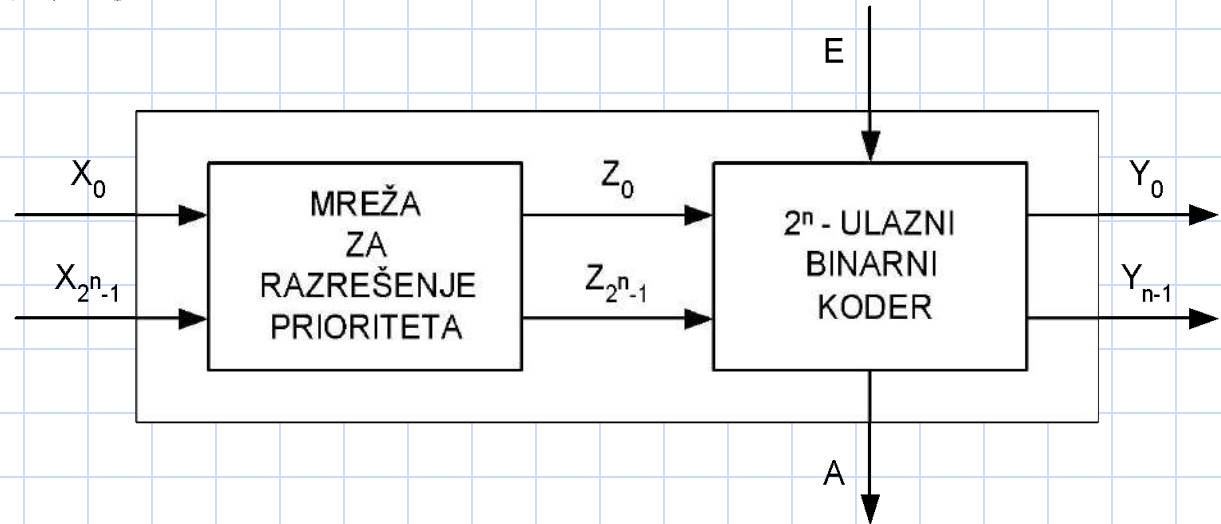
$Y_2 Y_1 Y_0$

$Y_2 Y_1 Y_0 = 100$

Mreža za razrešenje prioriteta

kombinaciona mreža sa 2^n ulaza i 2^n izlaza pri čemu je

$$Z_i = \begin{cases} 1 & \text{za } x_i = 1 \text{ i } x_k = 0 \text{ za } k > i \\ 0 & \text{u protivnom} \end{cases}$$



$Z(3) \leq X(3);$

$Z(2) \leq \text{NOT}(X(3)) \text{ AND } X(2);$

$Z(1) \leq \text{NOT}(X(3)) \text{ AND } \text{NOT}(X(2)) \text{ AND } X(1);$

$Z(0) \leq \text{NOT}(X(3)) \text{ AND } \text{NOT}(X(2)) \text{ AND } \text{NOT}(X(1)) \text{ AND } X(0);$

BCD Koder

```

PROCESS (iU) BEGIN
    sBCD_VECTOR <= "0000";
    IF (iU(0) = '1') THEN sBCD_VECTOR <= "0000"; END IF;
    IF (iU(1) = '1') THEN sBCD_VECTOR <= "0001"; END IF;
    IF (iU(2) = '1') THEN sBCD_VECTOR <= "0010"; END IF;
    IF (iU(3) = '1') THEN sBCD_VECTOR <= "0011"; END IF;
    IF (iU(4) = '1') THEN sBCD_VECTOR <= "0100"; END IF;
    IF (iU(5) = '1') THEN sBCD_VECTOR <= "0101"; END IF;
    IF (iU(6) = '1') THEN sBCD_VECTOR <= "0110"; END IF;
    IF (iU(7) = '1') THEN sBCD_VECTOR <= "0111"; END IF;
    IF (iU(8) = '1') THEN sBCD_VECTOR <= "1000"; END IF;
    IF (iU(9) = '1') THEN sBCD_VECTOR <= "1001"; END IF;
END PROCESS;

```

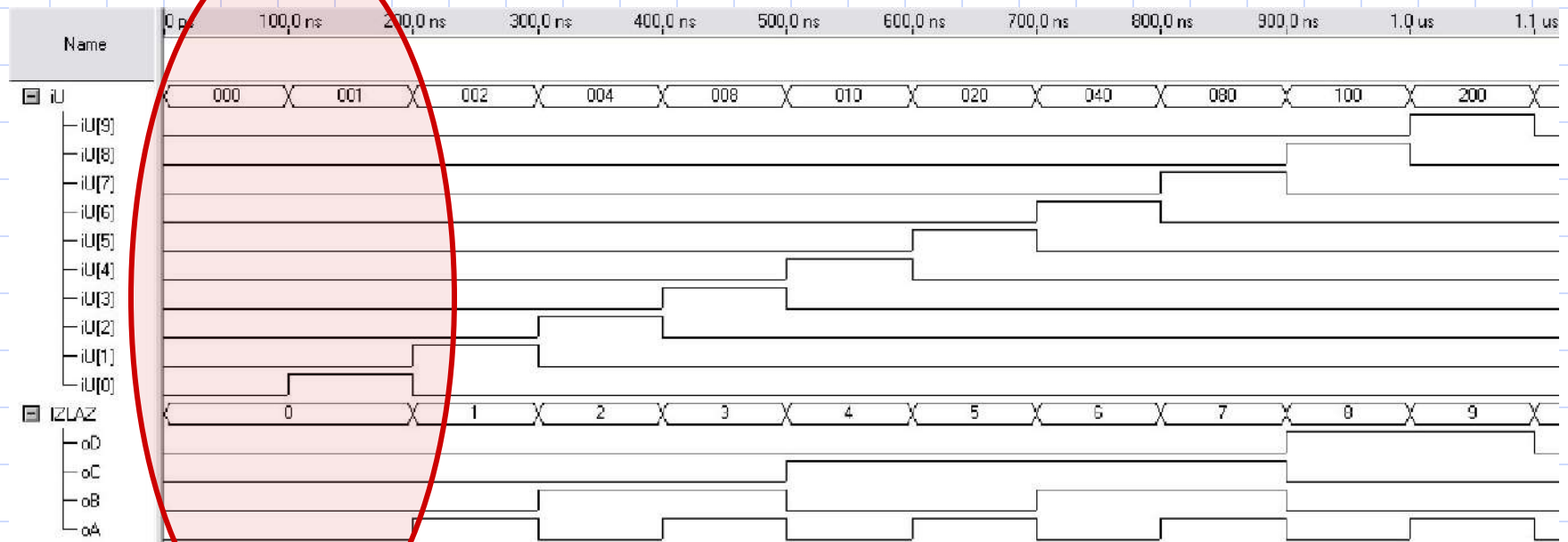
```

oD <= sBCD_VECTOR(3); -- bit sa težinom 8
oC <= sBCD_VECTOR(2); -- bit sa težinom 4
oB <= sBCD_VECTOR(1); -- bit sa težinom 2
oA <= sBCD_VECTOR(0); -- bit sa težinom 1

```

Koji ulazni signal je najvećeg prioriteta u navedenom pristupu implementaciji?

Vremenska simulacija rada BCD kodera

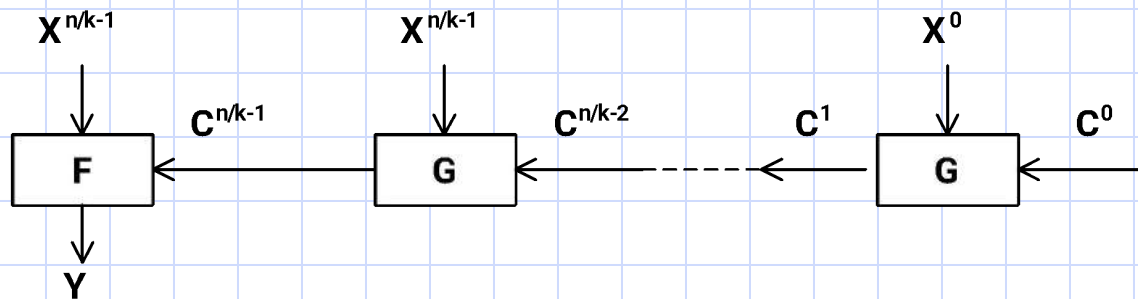


PROBLEM:

Ista vrednost izlaznog signala za dva različita ulazna vektora !!!

Iterativne mreže

- Mreže u obliku K-iterativnih mreža realizuju prekidačke funkcije n promenljivih pomoću $n/k-1$ identičnih podmreža (ćelija) G i F



- Ulazni vektor je podeljen u n/k grupa koje sadrže k-elemenata, svaka grupa se dovodi na pojedinačnu podmrežu (ćeliju)

