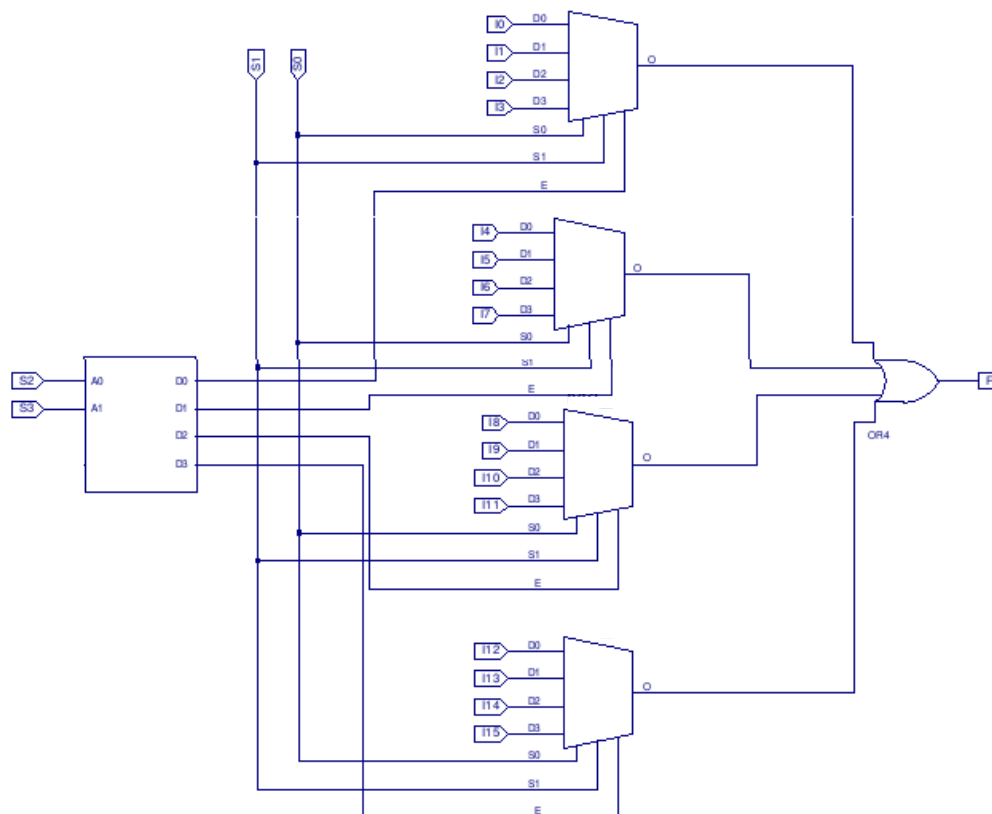


Laboratório 5: (resolução)

Exercício 16: Realização de funções utilizando *Multiplexers*

PARTE I – Projete um multiplexer de 4 entradas de seleção utilizando um decodificador e 4 multiplexers de 2 entradas de seleção com entrada de habilitação, e lógica combinatória elementar adicional.

S3	S2	S1	S0	F	
0	0	0	0	I0	Mux0
0	0	0	1	I1	
0	0	1	0	I2	
0	0	1	1	I3	
0	1	0	0	I4	Mux1
0	1	0	1	I5	
0	1	1	0	I6	
0	1	1	1	I7	
1	0	0	0	I8	Mux2
1	0	0	1	I9	
1	0	1	0	I10	
1	0	1	1	I11	
1	1	0	0	I12	Mux3
1	1	0	1	I13	
1	1	1	0	I14	
1	1	1	1	I15	



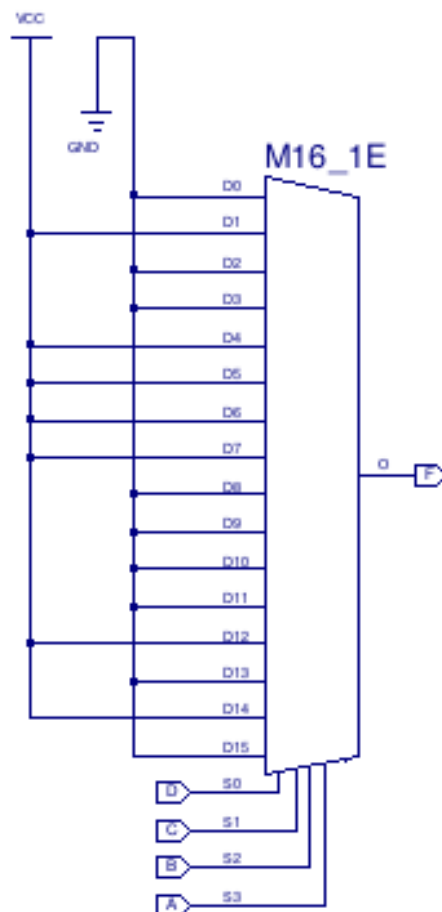
Utilizamos S3 e S2 como entradas de seleção do decodificador que servirão para selecionar qual o multiplexer que está ativo, atuando na entrada de habilitação (enable) de cada multiplexer.

Na implementação apresentada estamos a considerar um decodificador com saídas ativas a um e um multiplexer com enable ativo a um (Multiplexer “trabalha” quando Enable=1).

PARTE II – Considere a seguinte função de 4 variáveis: $F(A,B,C,D) = \Sigma(1,4,5,6,7,12,14)$

a) Realize a função usando o multiplexer projetado na primeira parte.

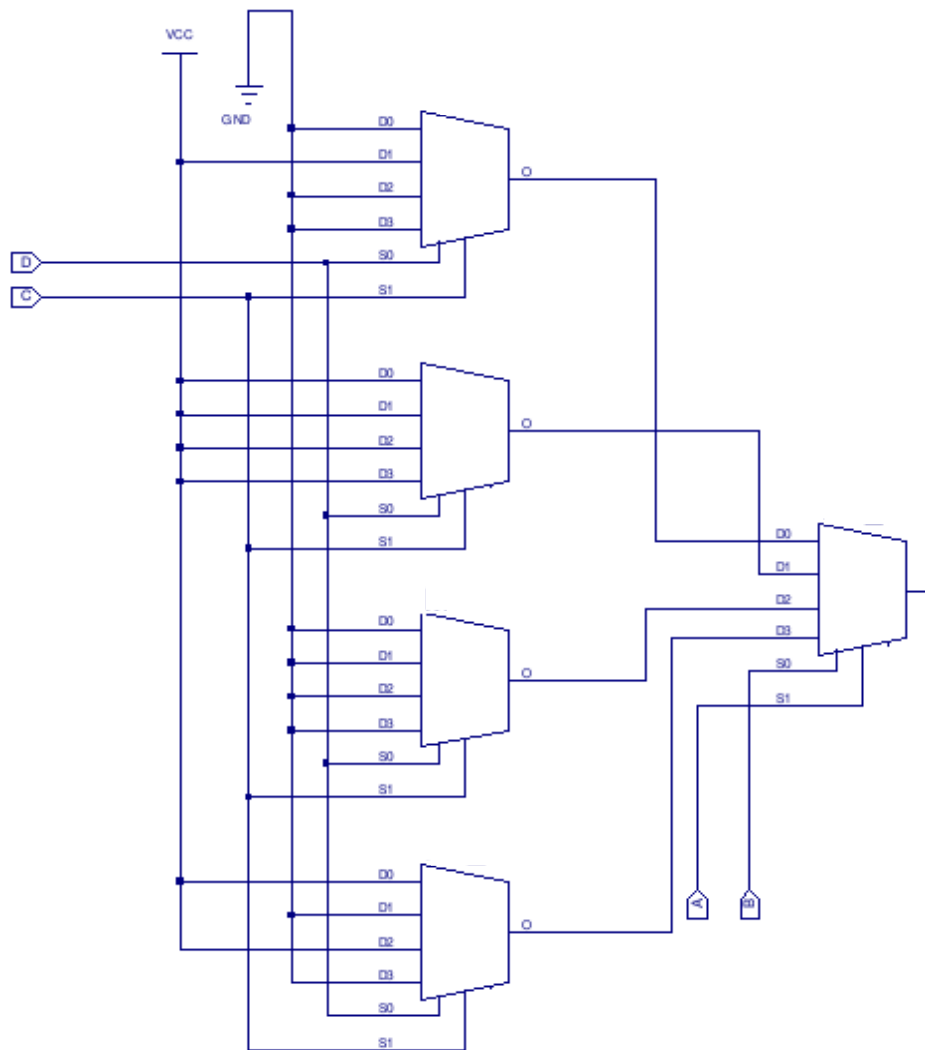
A	B	C	D	F
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0



b) Realize a função apresentada com base em multiplexers de **2 bits** de seleção.

A	B	C	D	F	
0	0	0	0	0	Mux0
0	0	0	1	1	
0	0	1	0	0	
0	0	1	1	0	
0	1	0	0	1	Mux1
0	1	0	1	1	
0	1	1	0	1	
0	1	1	1	1	
1	0	0	0	0	Mux2
1	0	0	1	0	
1	0	1	0	0	
1	0	1	1	0	
1	1	0	0	1	Mux3
1	1	0	1	0	
1	1	1	0	1	
1	1	1	1	0	

A	B	F
0	0	Mux0
0	1	Mux1
1	0	Mux2
1	1	Mux3



c) Realize a função utilizando **unicamente um** multiplexer de 2 bits de seleção e lógica adicional.

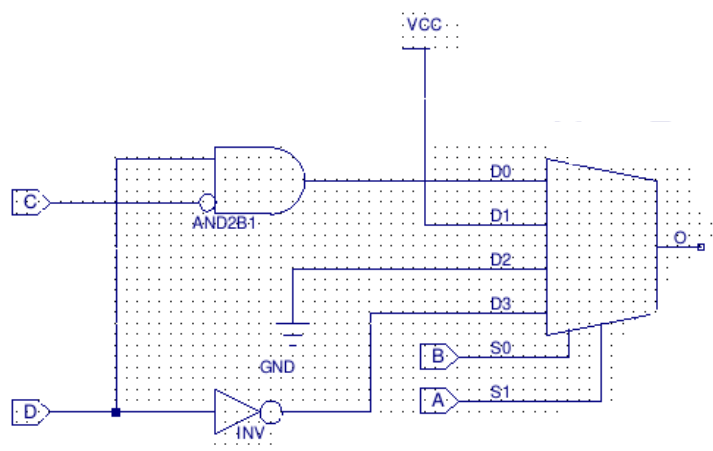
A	B	C	D	F
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

$\bar{C} \cdot D$

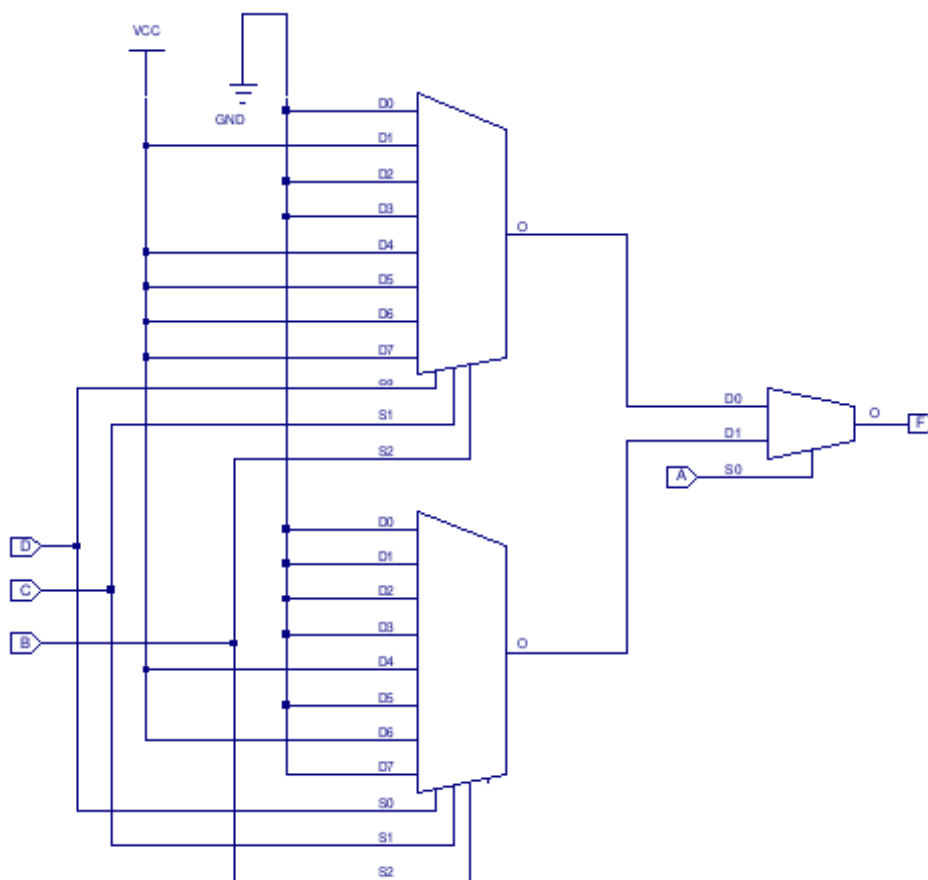
1

0

\bar{D}



d) Realize a função utilizando **pelo menos** um multiplexador de 1 bit de seleção.



Exercício 17: Comparadores

Pretende-se realizar um comparador de dois números X e Y. O resultado deve ser apresentado através de três saídas: Z(ero) que é activada quando $X=Y$, m(enor) que é ativada quando $X<Y$, e M(aior) que é ativada quando $X>Y$.

- a) Projete o circuito comparador para números X e Y de dois bits cada, simplificando as expressões através de mapas de Karnaugh.

Hipótese 1

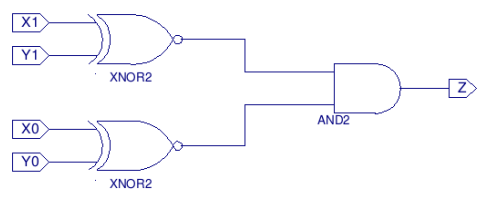
X1	X0	Y1	Y0	Z (X=Y)	M(X>Y)	m(X<Y)
0	0	0	0	1	0	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	0	1	0
0	1	0	1	1	0	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	0	1	0
1	0	0	1	0	1	0
1	0	1	0	1	0	0
1	0	1	1	0	0	1
1	1	0	0	0	1	0
1	1	0	1	0	1	0
1	1	1	0	0	1	0
1	1	1	1	1	0	0

Função transferência para Z

Y1 \ X1	X0		Y0	
	0	1	0	1
0	1	0	0	0
1	0	1	0	0
1	0	0	1	0
1	0	0	0	1

$$\begin{aligned}
 Z(X1, X0, Y1, Y0) &= \overline{X1} \cdot \overline{X0} \cdot \overline{Y1} \cdot \overline{Y0} + \overline{X1} \cdot X0 \cdot \overline{Y1} \cdot Y0 + X1 \cdot X0 \cdot Y1 \cdot Y0 + X1 \cdot \overline{X0} \cdot Y1 \cdot \overline{Y0} \\
 &= \overline{X1} \cdot \overline{Y1} \cdot (\overline{X0} \cdot \overline{Y0} + X0 \cdot Y0) + X1 \cdot Y1 \cdot (X0 \cdot Y0 + \overline{X0} \cdot \overline{Y0}) \\
 &= (X0 \cdot Y0 + \overline{X0} \cdot \overline{Y0}) \cdot (\overline{X1} \cdot \overline{Y1} + X1 \cdot Y1) \\
 &= (X0 \odot Y0) \cdot (X1 \odot Y1)
 \end{aligned}$$

Esquemático para Z

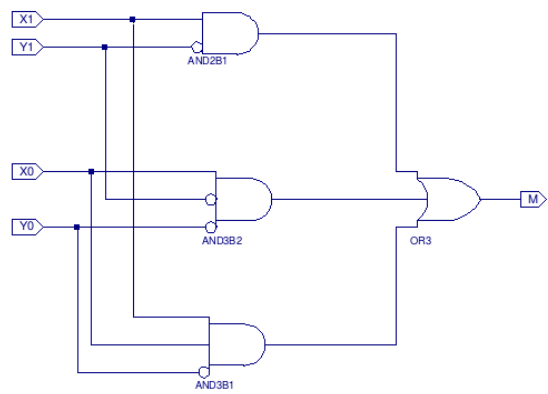


Função transferência para M

X1 \ X0 Y1	0 0	0 1	1 1	1 0
0 0	0	1	1	1
0 1	0	0	1	1
1 1	0	0	0	0
1 0	0	0	1	0

$M(X1,X0,Y1,Y0) = X1.\overline{Y1} + X0.\overline{Y1}.\overline{Y0} + X1.X0.\overline{Y0}$

Esquemático para M

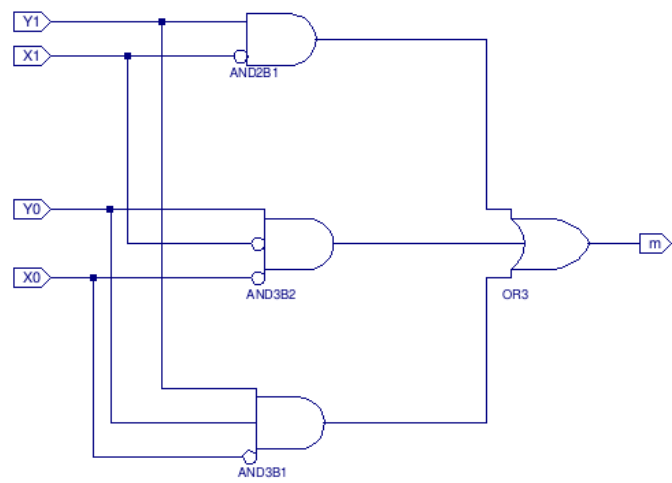


Função transferência para m

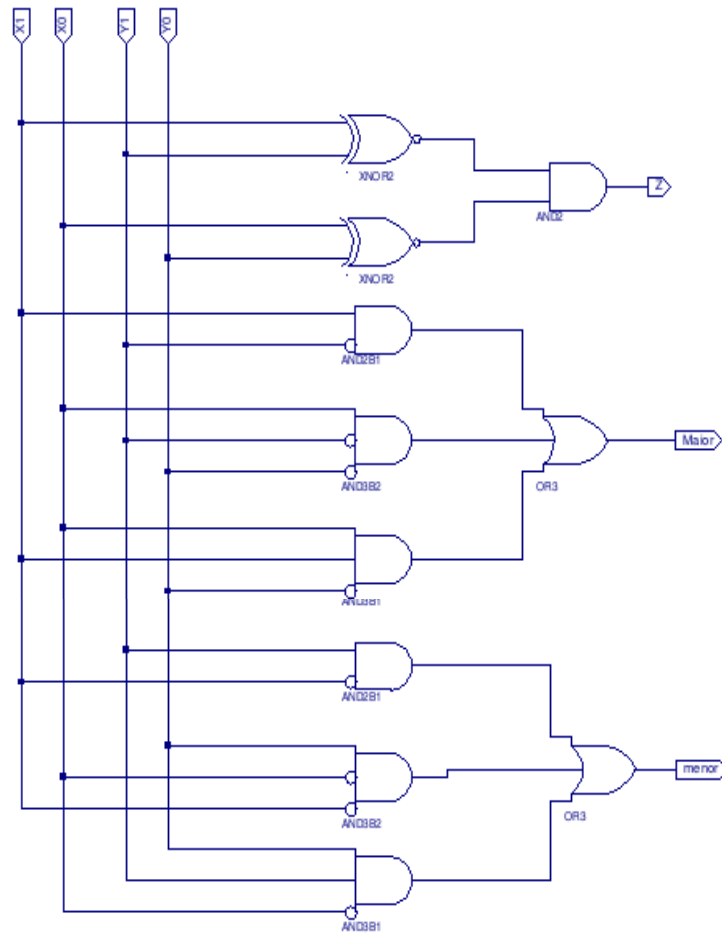
		X0		0		1		1		0	
Y1	X1	0		0		1		1		0	
		Y0		Y0		Y0		Y0		Y0	
0	0	0	0	0	0	0	0	0	0	0	0
0	1	1	1	0	5	0	13	0	9	0	0
1	1	1	2	1	6	1	15	0	10	1	0
1	0	1	3	1	7	1	14	0	11	0	0

$m(X1,X0,Y1,Y0) = \overline{X1}.Y1 + \overline{X0}.Y1.Y0 + \overline{X1}.\overline{X0}.Y0$

Esquemático para m



O aspeto do comparador será o seguinte:



Hipótese 2

Na hipótese 2 é importante perceber que podemos obter cada uma das saídas em função das outras duas, faremos o mesmo exercício obtendo m em função de Z e M.

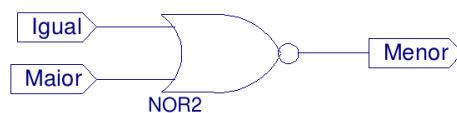
Fazendo a tabela de verdade para Z(X=Y) e M(X>Y):

X1	X0	Y1	Y0	Z (X=Y)	M(X>Y)
0	0	0	0	1	0
0	0	0	1	0	0
0	0	1	0	0	0
0	0	1	1	0	0
0	1	0	0	0	1
0	1	0	1	1	0
0	1	1	0	0	0
0	1	1	1	0	0
1	0	0	0	0	1
1	0	0	1	0	1
1	0	1	0	1	0
1	0	1	1	0	0
1	1	0	0	0	1
1	1	0	1	0	1
1	1	1	0	0	1
1	1	1	1	1	0

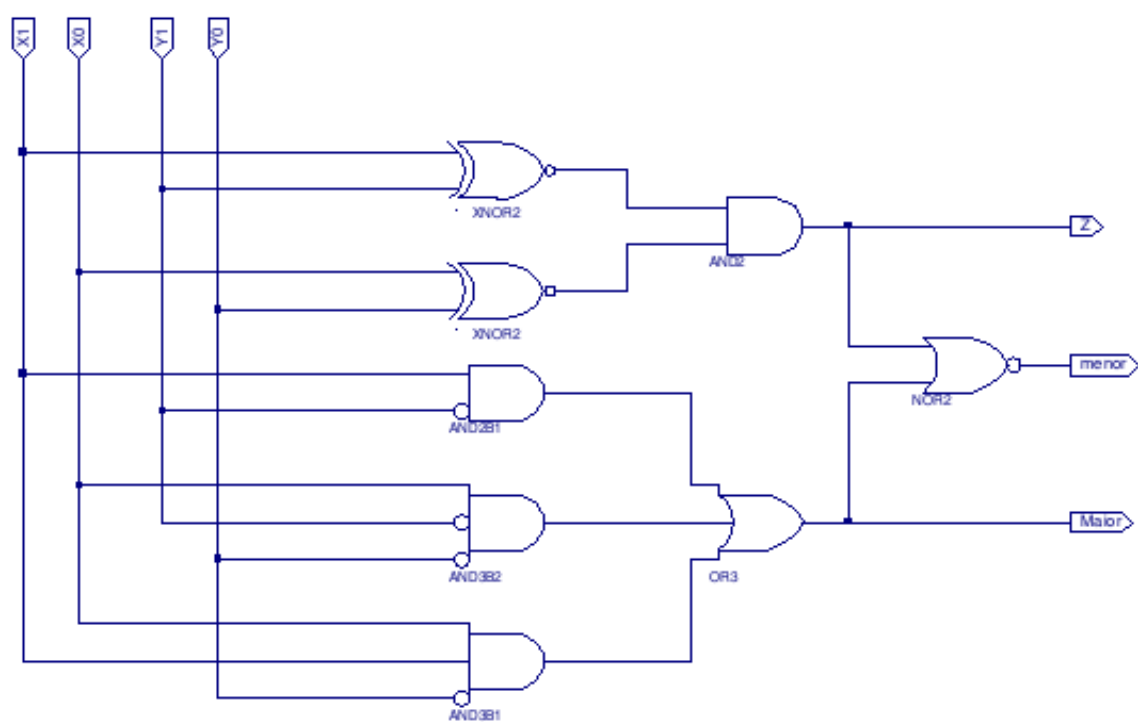
As funções transferência para Z e M foram calculadas na hipótese 1, para m(X<Y) faremos em função de Z e M:

Z(X=Y)	M(X>Y)	m(X<Y)
0	0	1
0	1	0
1	0	0
X	X	-

$$m(X < Y) = \overline{Z + M}$$

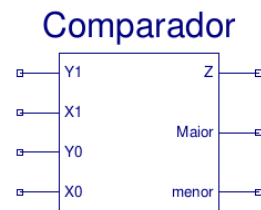


Ficando o projeto do comparador com o seguinte aspeto:



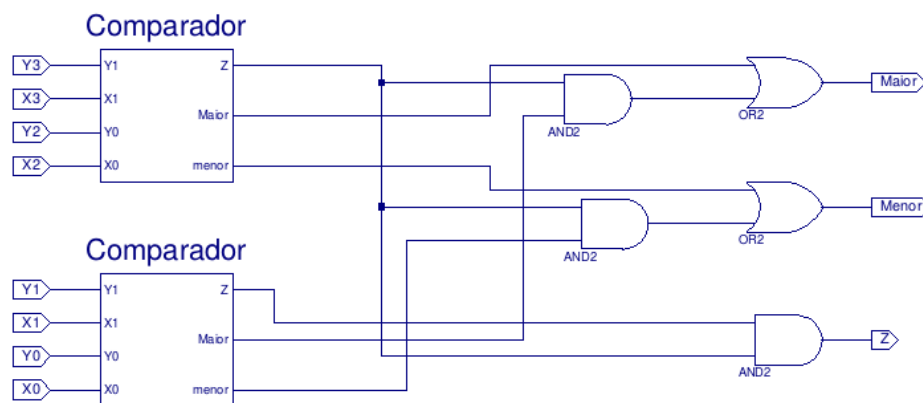
b) Utilizando blocos comparadores (do tipo projetado na alínea anterior), projete um comparador de dois números X e Y de quatro bits cada. Como resultado deverá apresentar o esquemático associado.

O comparador projetado na alínea anterior tem o seguinte aspeto:



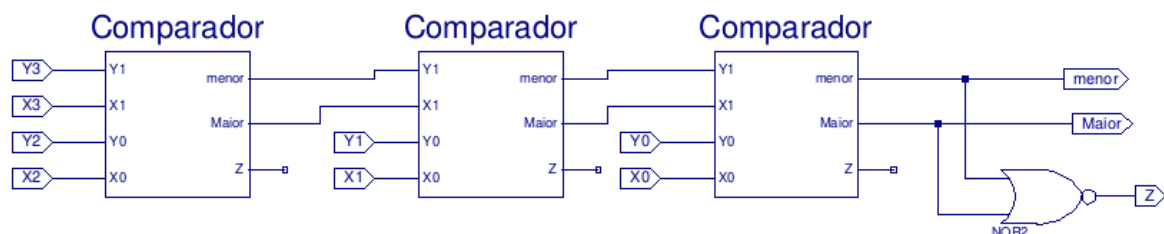
Hipótese 1

Como queremos projetar um comparador de números de 4 bits, poderemos usar dois comparadores deste tipo, ficando o esquemático da seguinte maneira:



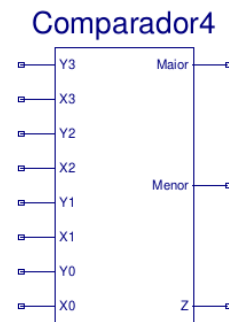
Hipótese 2

Utilizando comparadores em cascata:



c) Com base nos resultados anteriores, projete agora um comparador de dois números X e Y de 8 bits cada. Como resultado deverá apresentar o esquemático associado.

O Comparador projetado na alínea anterior tem o seguinte aspeto:



Como queremos projetar um comparador de números de 8 bits, teremos de usar dois comparadores de 4 bits, ficando o esquemático da seguinte maneira:

