

CIRCUITOS LÓGICOS

CIRCUITOS COMBINACIONAIS BÁSICOS

(...)

Marco A. Zanata Alves

EXERCÍCIO COMPARADOR

Faça um comparador binário com 3 saídas, maior, menor e igual.

- 1 bit de entrada s/ sinal
- 2 bits de entrada s/ sinal
- 8 bits de entrada s/ sinal

EXERCÍCIO COMPARADOR

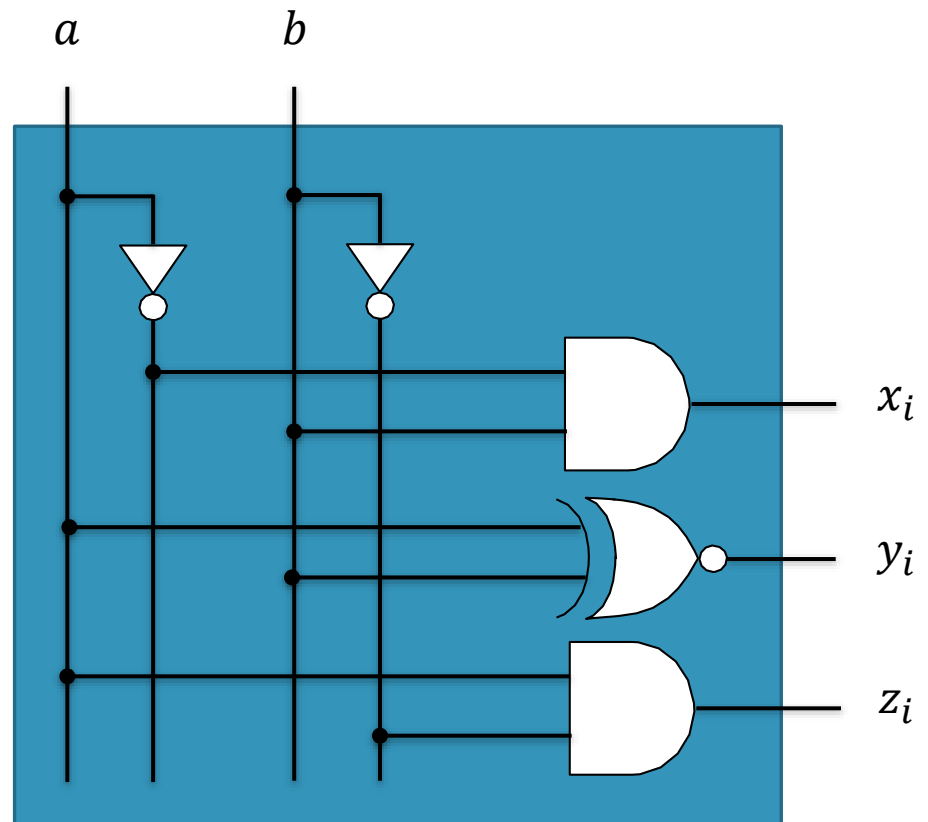
Faça um comparador binário com 3 saídas, maior, menor e igual.

- 1 bit de entrada s/ sinal

$X: a < b$

$Y: a = b$

$Z: a > b$



EXERCÍCIO COMPARADOR

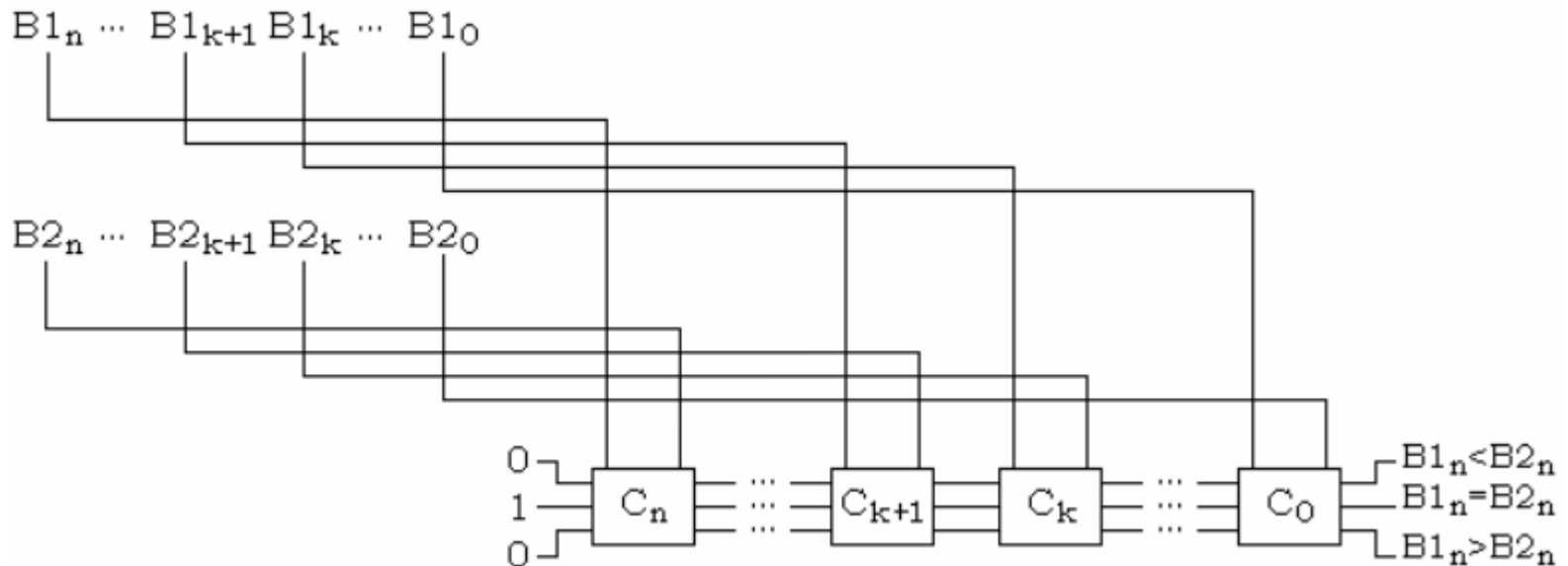
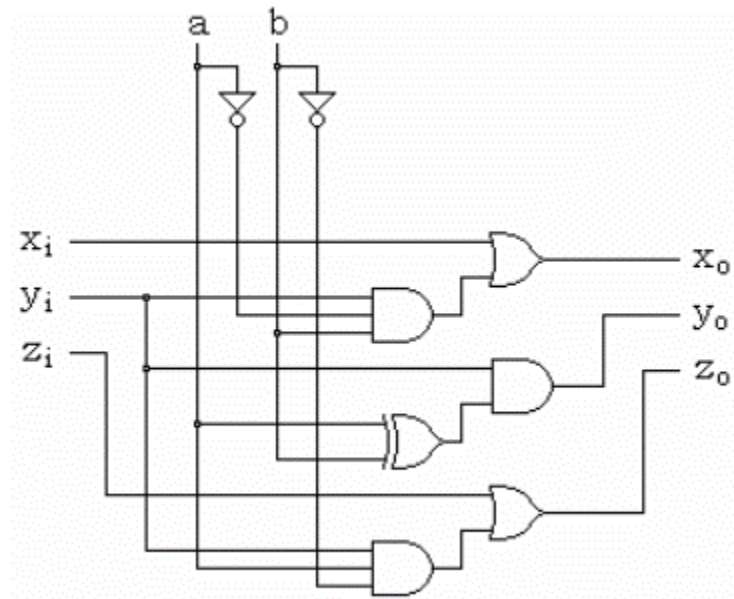
Faça um comparador binário com 3 saídas, maior, menor e igual.

- 2 bits de entrada s/ sinal
- 8 bits de entrada s/ sinal

$X: a < b$

$Y: a = b$

$Z: a > b$



EXERCÍCIO

COMPARADOR DO SINAL MAGNITUDE

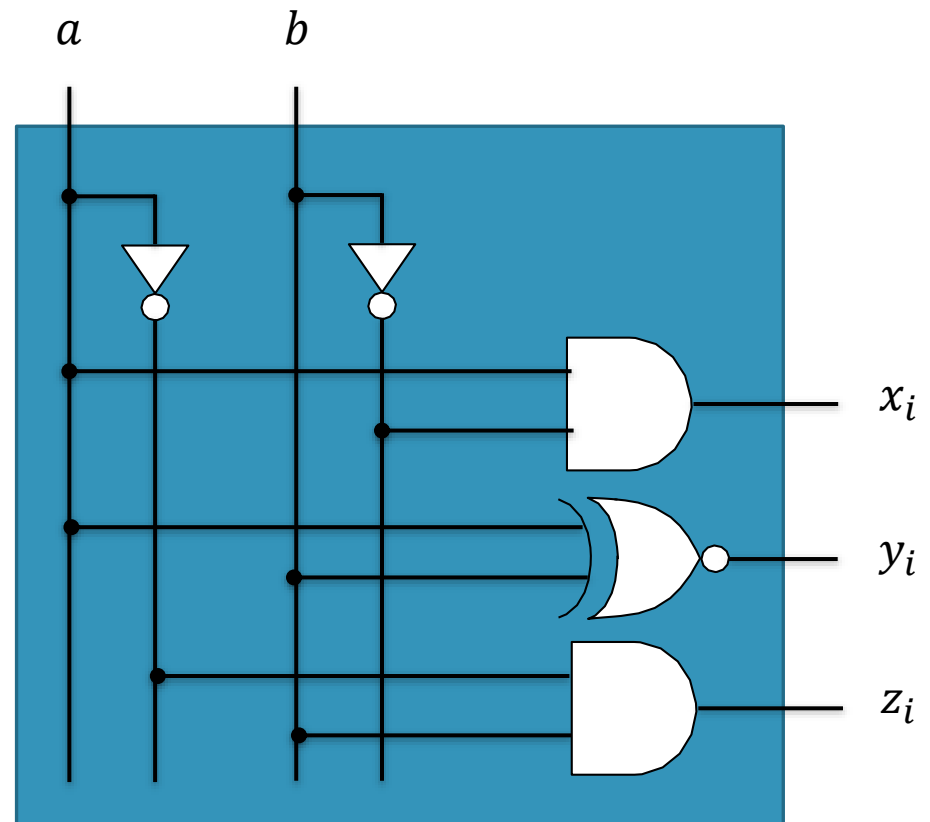
Faça um comparador binário com 3 saídas, maior, menor e igual.

- 1 bit de entrada representando o sinal magnitude

$X: a < b$

$Y: a = b$

$Z: a > b$



EXERCÍCIO

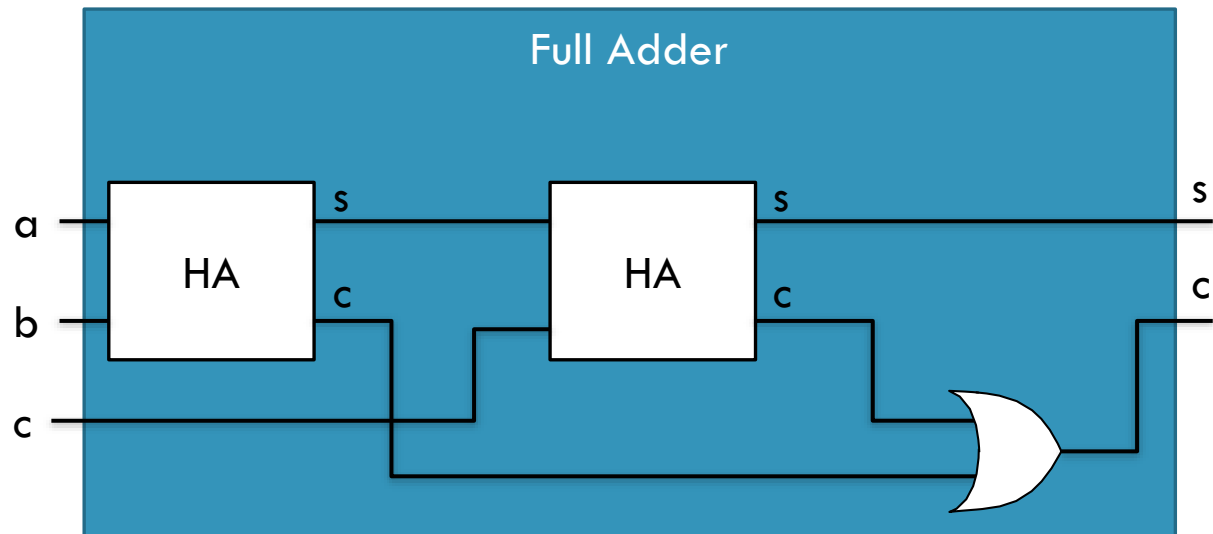
SOMADOR COMPLETO

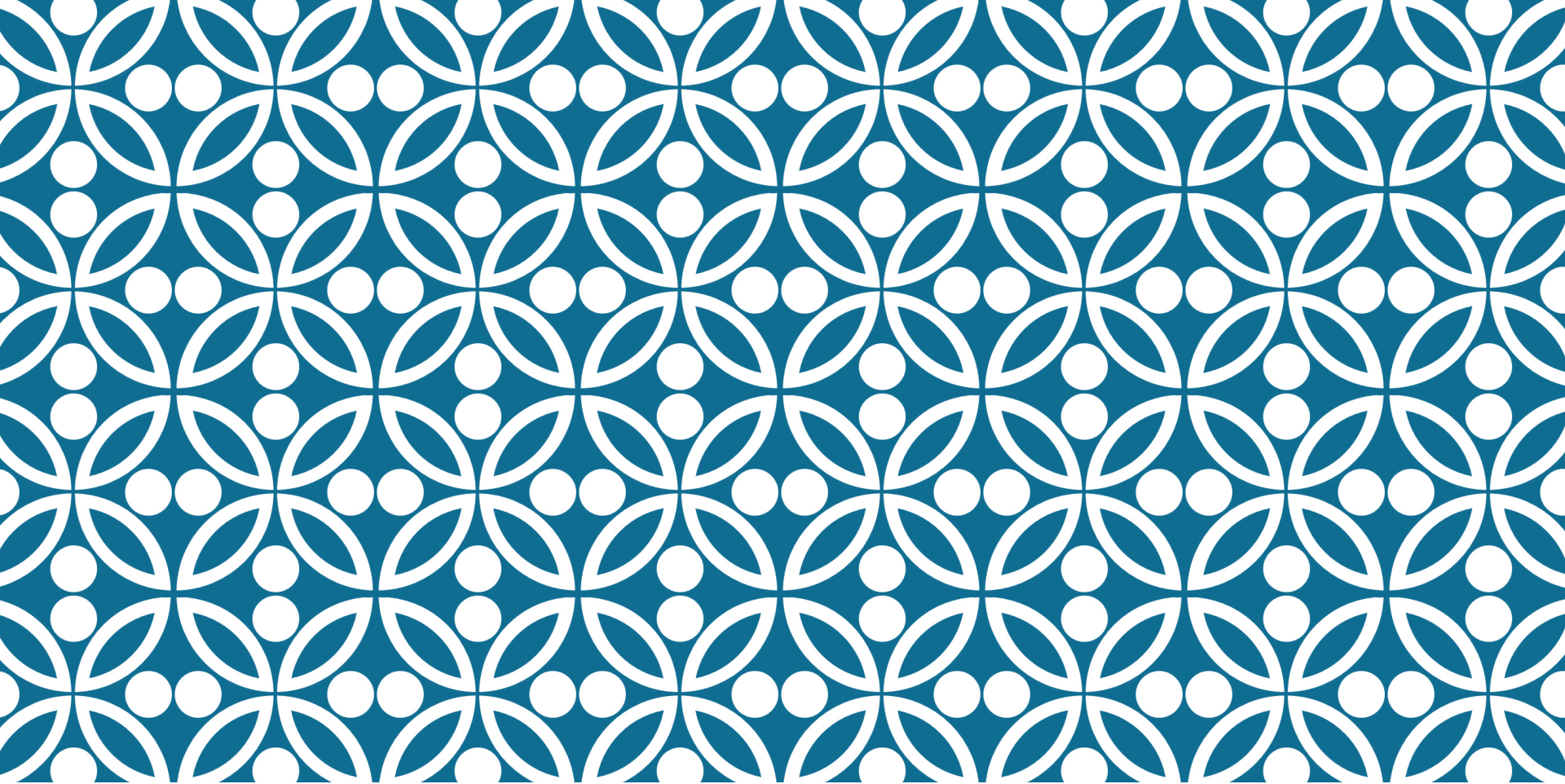
Faça um somador completo utilizando apenas 2 half-adders e portas lógicas.

EXERCÍCIO

SOMADOR COMPLETO

Faça um somador completo utilizando apenas 2 half-adders e portas lógicas.

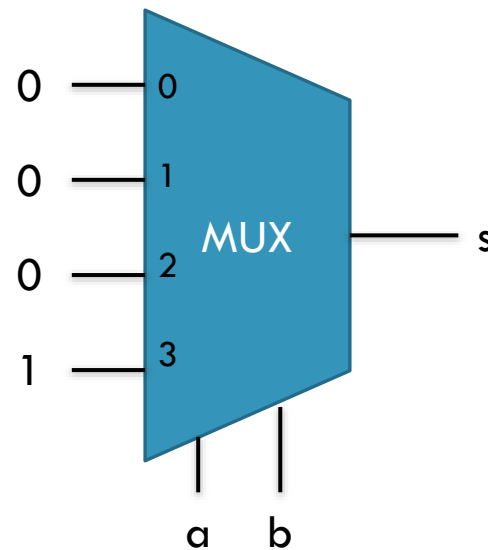




IMPLEMENTANDO FUNÇÕES COM MULTIPLEXADORES

EXERCÍCIO FX-MUX

Implemente a porta lógica AND de duas entradas com um MUX.

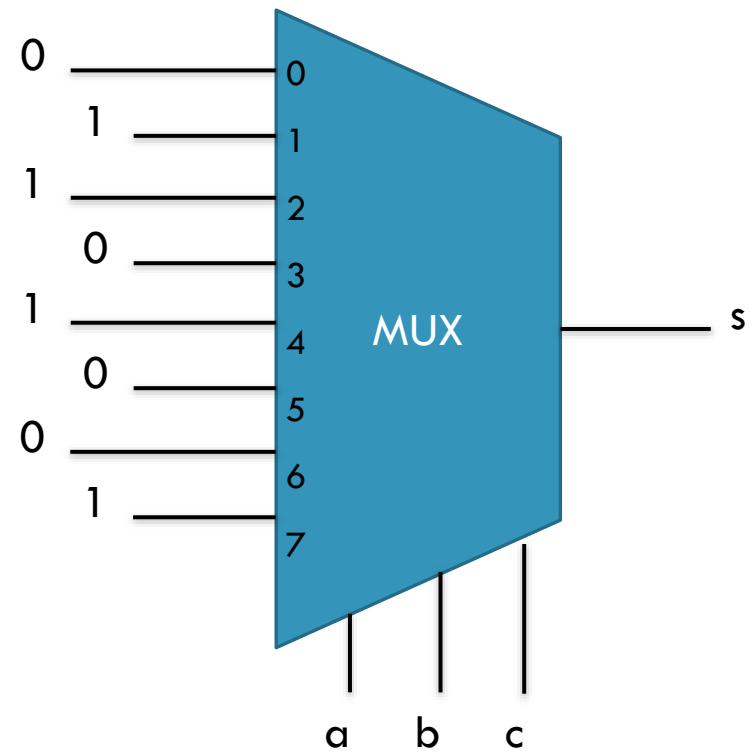


EXERCÍCIO FX-MUX

Implemente a porta lógica XOR de três entradas com um MUX 8:1.

EXERCÍCIO FX-MUX

Implemente a porta lógica XOR de três entradas com um MUX 8:1.



EXERCÍCIO FX-MUX

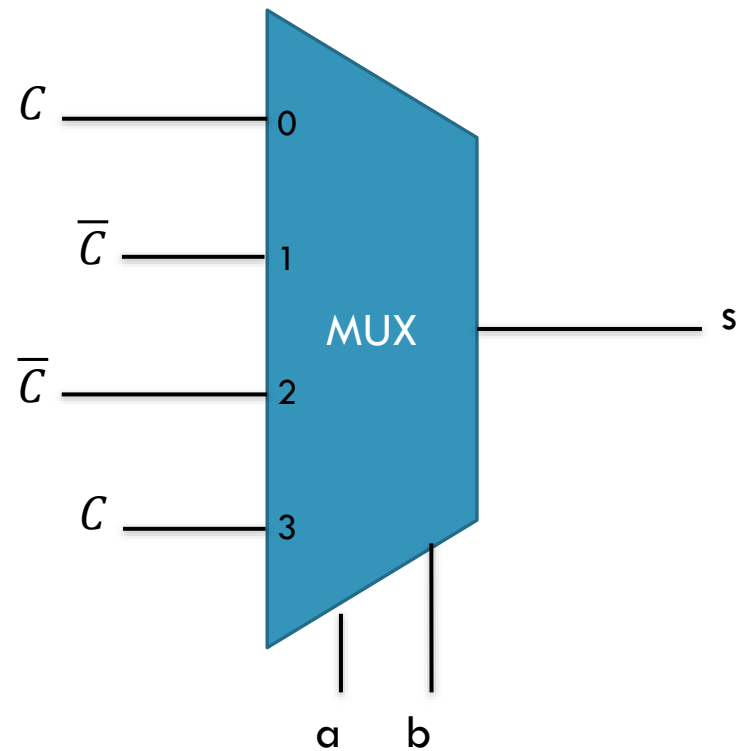
Implemente a porta lógica XOR de três entradas com um MUX 4:1.

Faça em função de C !

EXERCÍCIO FX-MUX

Implemente a porta lógica XOR de três entradas com um MUX 4:1.

Faça em função de C !



EXERCÍCIO FX-MUX

Implemente a seguinte função:

$$F = A'B'C'D' + A'B'CD + A'BC'D + A'BC'D' + AB'C'D + AB'CD' + ABC'D + ABC'D'$$

- Utilizando portas lógicas
- Utilizando um MUX 8:1

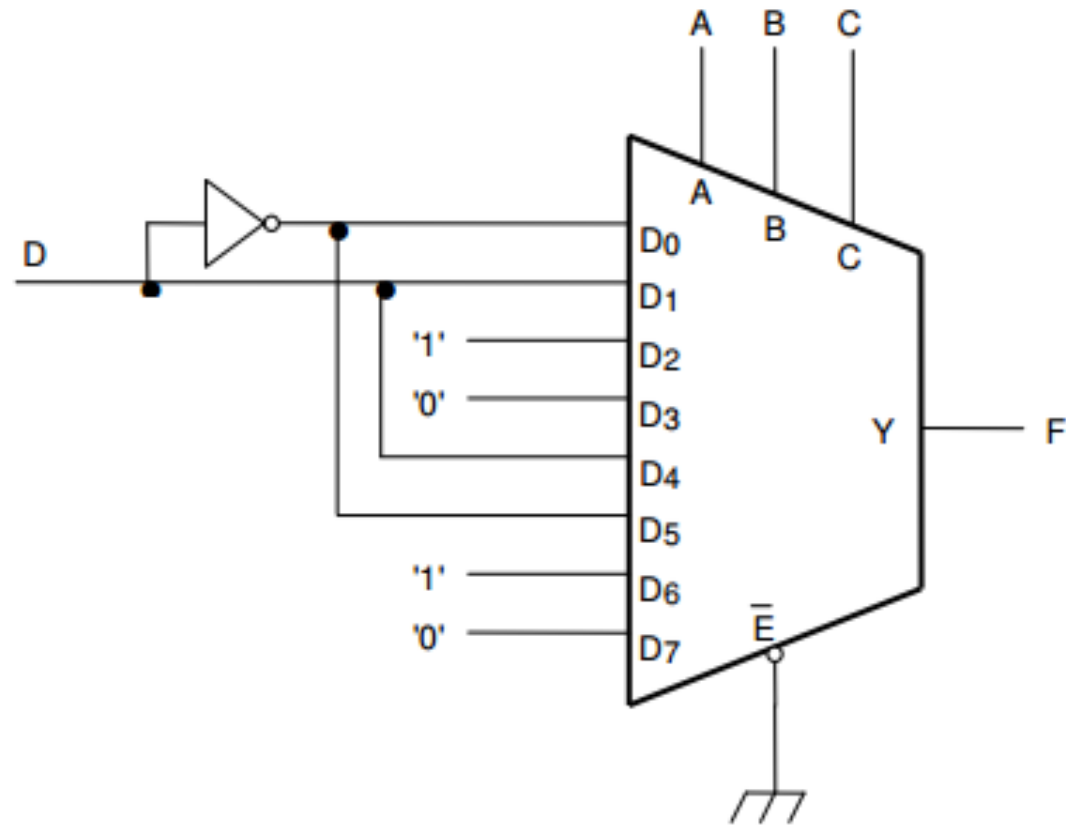
EXERCÍCIO FX-MUX

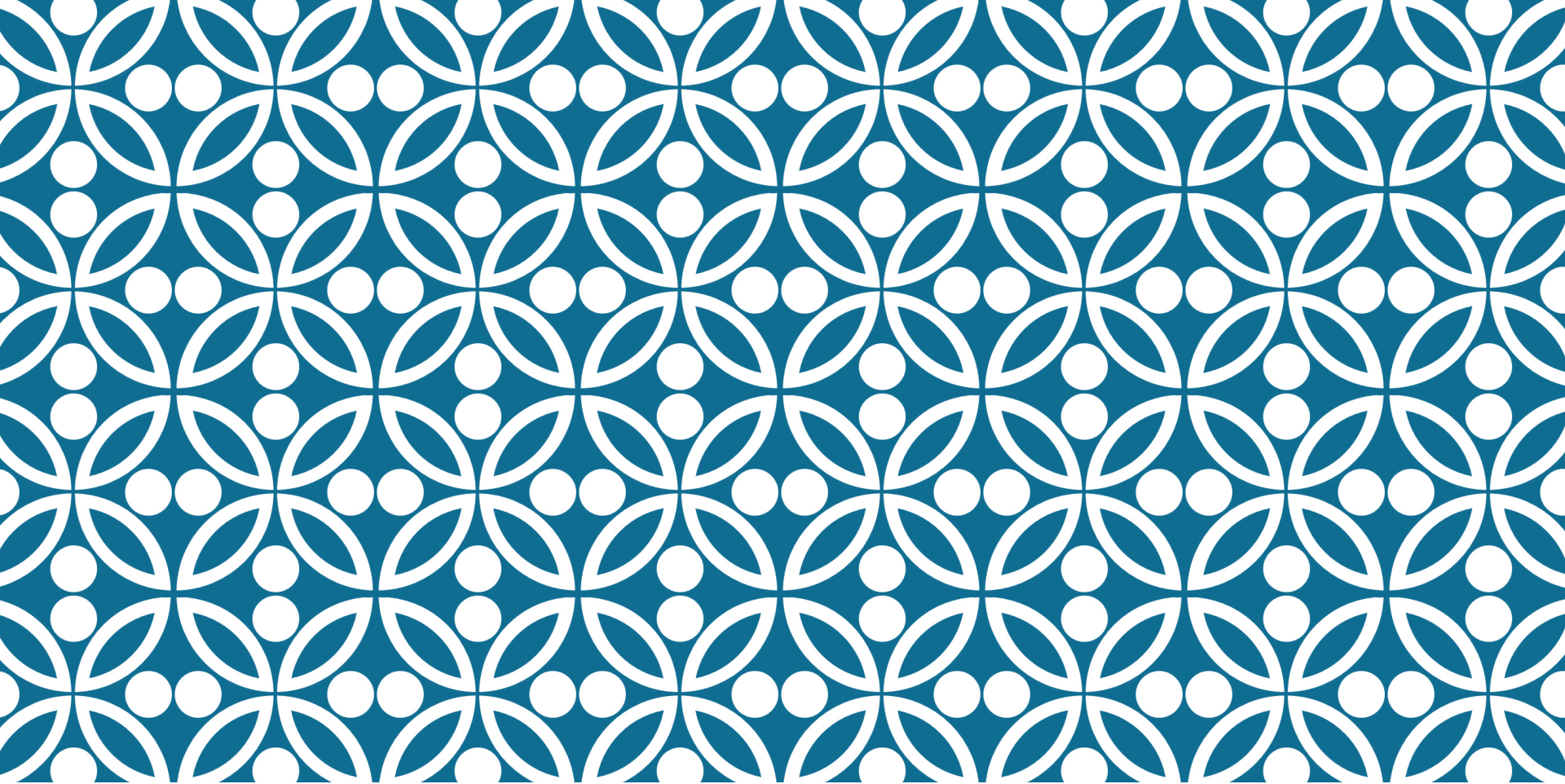
$$F = A'B'C'D' + A'B'CD + A'BC' + AB'C'D + AB'CD' + ABC'$$

Implemente a seguinte função:

$$F = A'B'C'D' + A'B'CD + A'BC'D + A'BC'D + AB'C'D + AB'CD' + ABC'D + ABC'D'$$

- Utilizando portas lógicas
- Utilizando um MUX 8:1





UTILIZANDO DON'T CARE

EXERCÍCIO DON'T CARE

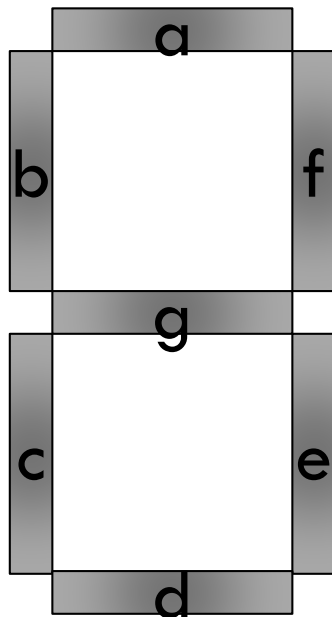
Simplifique com mapas de Karnaugh e implemente a seguinte lógica:

A	B	C	Saída
1	0	0	1
0	1	0	0
1	1	0	0
0	1	1	1
0	0	1	1

EXERCÍCIO

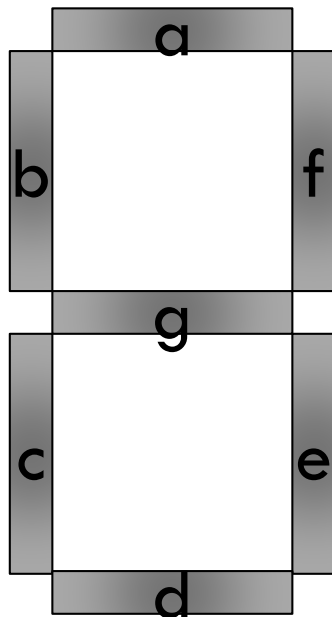
DON'T CARE

Simplifique com mapas de Karnaugh e implemente a lógica para acender o **led a** utilizando a codificação BCD (binário – decimal):



EXERCÍCIO DON'T CARE

Simplifique com mapas de Karnaugh e implemente a lógica para acender o **led a** utilizando a codificação BCD (binário – decimal):
(binário – decimal):



b_0	b_1	b_2	b_3	Led A
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	X
1	1	0	0	X
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X