

Lista 2

quinta-feira, 25 de agosto de 2022 19:17

- 1. (1p)** Sem efetuar qualquer simplificação, qual dos seguintes esquemas implementa a função: $F(C,D,E) = \prod_{C,D,E} M(0,2,3,5,7)$ (C é o bit MSB e E é o bit LSB. OBS. /C=not C; /D=not D; /E=not E)

Figura 1

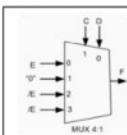


Figura 2

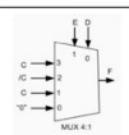


Figura 3

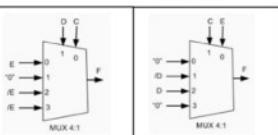
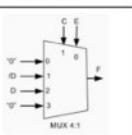


Figura 4



Escolher e justificar uma resposta.

- a) Figura 3 \rightarrow S1=D; S0=C; I0=E; I1="0"; I2=/E; I3=/
b) Figura 1 \rightarrow S1=C; S0=D; I0="0"; I2=E; I3=
c) Figura 2 \rightarrow S1=E; S0=D; I0="0"; I1=C; I2=C; I3=C
d) Figura 4 \rightarrow S1=C; S0=E; I0="0"; I1=/D; I2=D; I3="0"

Alternativa b

Row	C	D	E	F	Função que eu quero que seja			$\prod_{C,D,E} M(0,2,3,5,7)$
0	0	0	0	0				
1	0	0	1	1				
2	0	1	0	0				
3	0	1	1	0				
4	1	0	0	1				
5	1	0	1	0				
6	1	1	0	1	/E	C	D	
7	1	1	1	0	/E	C	D	

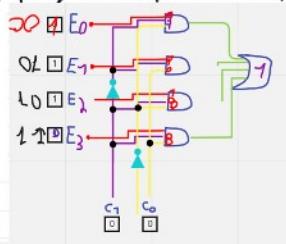
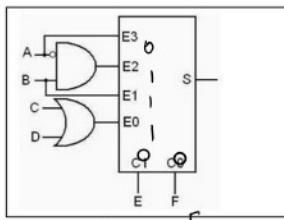
A alternativa é b já que C é o Bit mais significativo e Recebe 1

- 2. (1p)** A figura ilustra um circuito combinacional que utiliza um multiplexador 4-para-1. As linhas de controle são as linhas C1 e C0, em que C1 é o bit mais significativo. Considerando esta figura, julgue:

Se $A = 0$, $B = 1$, $C = 1$, $D = 0$, $C1 = 0$ e $C0 = 0$, então a saída S assumirá o valor 1?

Escolher e justificar: a) Verdadeiro b) Falso

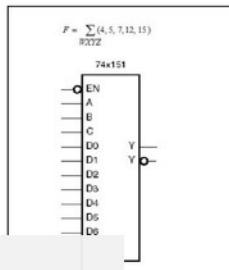
(Como os 0 bits de seleção são 0 e 0, ou seja, a saída sera E0 que recebe o valor 1, ja que na porta OR possui um com valor)



- 3. (1p)** Mostre como construir a função lógica F usando um multiplexador binário 74151 e portas lógicas. Não se esqueça de habilitar o CI. Obs. W é MSB e Z é LSB.

Escolher e justificar uma resposta

- a) /EN=0; A=Z; B=Y; C=X; D0=D1=D2=D6=0; D3=D4=/W; D5=D7=1; Y=F
b) /EN=0; A=Z; B=Y; C=X; D0=D1=D2=D3=D6=0; D4=D7=1; D5 = /W; Y=F
c) /EN=0; A=Z; B=Y; C=X; D0=D1=D7=0; D3=D4=/W; D2=D5=D6=1; Y=F



	W	X	Y	Z	S
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	1	1	1	0

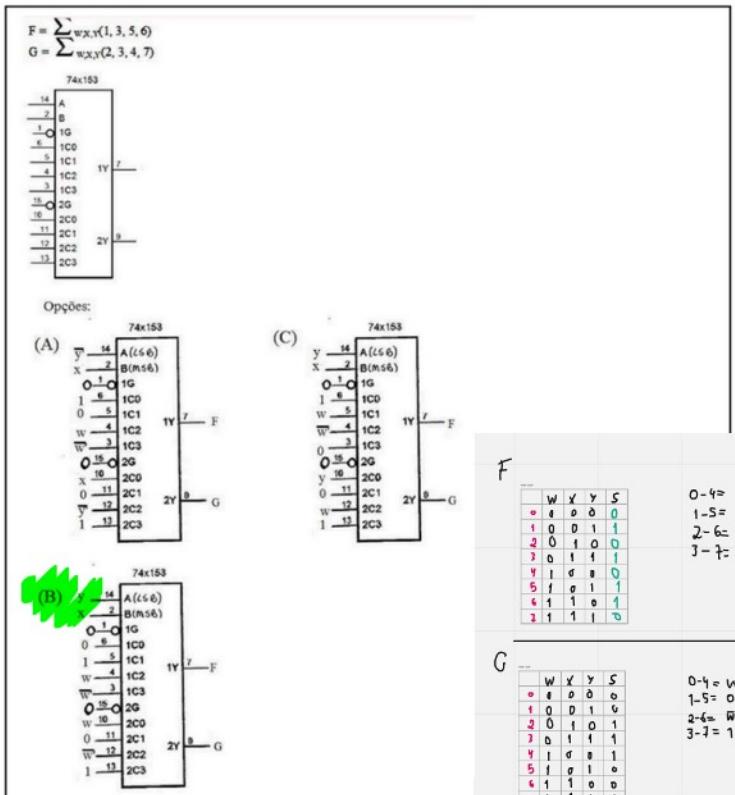
$$\begin{cases}
 D_0 = 0 - 8 = 0 \\
 D_1 = 1 - 9 = 0 \\
 D_2 = 2 - 10 = 0 \\
 D_3 = 3 - 11 = 0 \\
 D_4 = 4 - 12 = 1 \\
 D_5 = 5 - 13 = \overline{W} \\
 D_6 = 6 - 14 = 0 \\
 D_7 = 7 - 15 = 1
 \end{cases}$$

	w	x	y	z	s
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	7
6	0	1	1	0	0
7	0	1	1	1	7
8	1	0	0	0	6
9	1	0	0	1	8
10	1	0	1	0	0
11	1	1	1	1	0
12	1	1	0	0	1
13	1	1	0	1	0
14	1	1	1	0	0
15	1	1	1	1	1

$$\left\{ \begin{array}{l} D_0 = 0 - 8 = 0 \\ D_1 = 1 - 9 = 0 \\ D_2 = 2 - 10 = 0 \\ D_3 = 3 - 11 = 0 \\ D_4 = 4 - 12 = 1 \\ D_5 = 5 - 13 = W \\ D_6 = 6 - 14 = 0 \\ D_7 = 7 - 15 = 1 \end{array} \right.$$

A alternativa B é a unica que apresenta todas as características utilizadas

4. (1p) Projete as funções lógicas usando apenas o CI 74x153 (dois MUX 4-para-1). Escolher e justificar uma resposta: a) Opção (B); b) Opção (A); c) Opção (C)



L ₀	w	x	y	F	G	f	a	b	c	
L ₀	0	0	0	0	0	1	x	0	1	x
L ₁	0	0	1	1	0	0	x	1	w	x
L ₂	0	1	0	0	1	0	w	w	w	w
L ₃	0	1	1	1	1	1	w	w	0	x
L ₄	1	0	0	0	1	1	w	w	w	y
L ₅	1	0	1	1	0	0	w	0	0	v

L4	1	0	0	0	1	1	X'	W	Y'
L5	1	0	1	1	0	0, $\bar{w}, \bar{x}, \bar{y}$	0	0	0
L6	1	1	0	1	0	0, $\bar{w}, \bar{x}, \bar{y}$	\bar{y}	\bar{w}	\bar{w}
L7	1	1	1	0	1	1, w, x, y	1	1	1

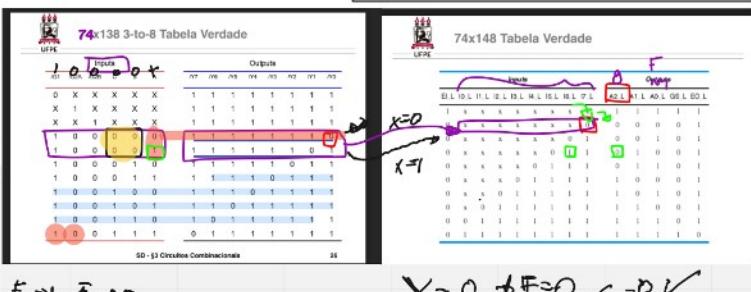
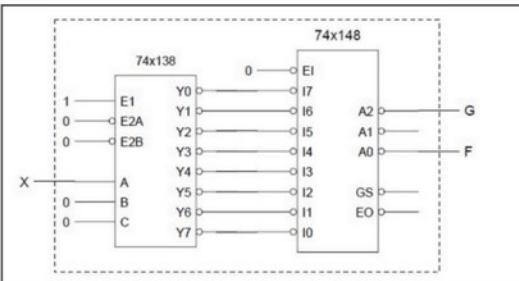
Além disso temos que obedece as funções e a tautologia b

5. (1p) As funções lógicas geradas nas saídas F e G são:
 $F = X$ e $G = 0'$

Resposta: a) Verdadeiro b) Falso

Obs.:

- 74x138 - decodificador binário-decimal 3-para-8
- 74x148 - codificador decimal-binário 8-para-3
- Enable: E1, /E2A, /E2B, /E1



$$E_1 \rightarrow 0, \bar{E}_2 \rightarrow 0$$

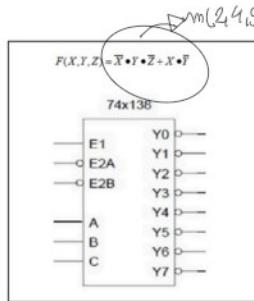
Assim vemos que sempre $G = 0$ e que F assume o mesmo valor de X
 ASSUMA A TAUTLOGIA E VERDADISTRA

6. (1p) Projete a função lógica F usando apenas um CI 74x138 (decodificador 3-para-8) e uma porta lógica. Não se esqueça de habilitar o CI.

Obs.: /E2A = NOT E2A; /E2B = NOT E2B; /Y0 = NOT Y0 /Y7 = NOT Y7

Escolher e justificar uma resposta

- E1=1; /E2A = /E2B = 0; A=Z; B=Y; C=X; F= NOT(/Y2 AND /Y6 AND /Y7) \cancel{F}
- (E1=1; /E2A = /E2B = 0; A=Z; B=Y; C=X; F= NOT(/Y2 AND /Y4 AND /Y5)) $\cancel{\text{verdadeira}}$ (2,4,5)
- c) E1=1; /E2A = /E2B = 0; A=Z; B=Y; C=X; F= NOT(/Y0 AND /Y3 AND /Y4) \cancel{F}
- d) E1=1; /E2A = /E2B = 0; A=Z; B=Y; C=X; F= NOT(/Y2 AND /Y4 AND /Y6) \cancel{F}
- e) E1=1; /E2A = /E2B = 0; A=Z; B=Y; C=X; F= NOT(/Y0 AND /Y2 AND /Y4) \cancel{F}



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

Usando a tabela-verdade e o mapa de Karnaugh,
 nota-se que a função é uma $F = \text{Jm}(2,4,5)$
 e a unica alternativa que apresenta é a 'b'

7. (0.4p) Realize a conversão de

- (1435)₁₀ da base 10 para a base 8.
- (1101011)₂ da base 2 para a base 16.
- (1101001)₂ da base 2 para a base 10.
- (72)₈ da base 8 para a base 16.

Q.

$2 \rightarrow 2633$

1435 | 8

Quesitos

$$\begin{array}{r}
 1435 | 8 \\
 (3) 179 | 8 \\
 (3) 22 | 8 \\
 (6) 2 | 8 \\
 (2) 0
 \end{array}$$

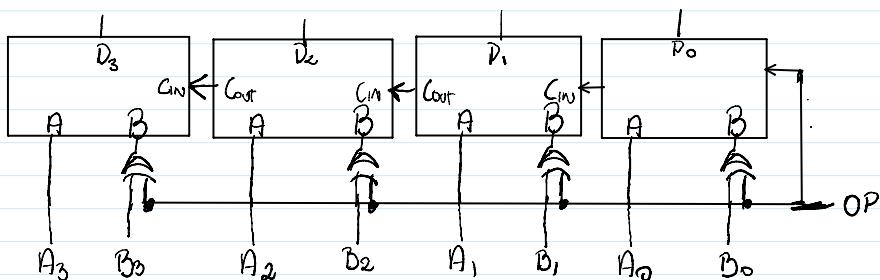
b) $\sum 6 \rightarrow B \rightarrow 6B$

c) $2^0 + 2^3 + 2^5 + 2^6 = 1 + 8 + 32 + 64 = 105$

d) $2 \cdot 8^0 + 7 \cdot 8^1 = 2 + 56 = 58$

$$\begin{array}{r}
 58 | 16 \\
 (10) 3 | 16 \\
 (13) 0
 \end{array} \rightsquigarrow 3A$$

8. (0.8p) Utilizando um somador completo de 4 bits e portas lógicas adicionais, implemente um circuito que recebe duas palavras de 4 bits A e B e um código de operação OP, efetuando a soma A+B quando OP=0 e a subtração A-B quando OP=1.



9. (0.4p) Usando complemento de 2, resolver as seguintes operações (todos os números estão em decimal). É necessário mostrar todas as etapas e indicar quando um overflow ocorrer. Use uma representação de 4 bits.

a) 6-3
b) -8-3 conv. c. 2,

a) $(+6) \xrightarrow{\text{conv.}} 0011$
 $+(-3) \xrightarrow{\text{conv.}} 1100$
 $\xrightarrow{+1} 1101 \xrightarrow{\text{Desconsidere o último bit e remova}} 0011$

b) $(-8) \xrightarrow{\text{conv.}} 1000$
 $+(-3) \xrightarrow{\text{conv.}} 1101$
 $\xrightarrow{-1} 1010 \rightsquigarrow \text{Overflow}$

10. (0.4p) Usando complemento de 2, resolver as seguintes operações (todos os números estão em decimal). É necessário mostrar todas as etapas e indicar quando um overflow ocorrer. Use uma representação de 8 bits.

c) 23-17
d) 32-70

c) $23 \xrightarrow{\text{conv.}} 0001\ 0001$
 $+(-17) \xrightarrow{\text{conv.}} 10111$
 $\xrightarrow{+6} 10001 \xrightarrow{\text{conv.}} 0001\ 0001$
 $\xrightarrow{+1} 1110\ 110 \xrightarrow{\text{Overflow}}$

d) $32 \xrightarrow{\text{conv.}} 001\ 00000$
 $+(-70) \xrightarrow{\text{conv.}} 1101\ 1210$
 $\xrightarrow{+1} 10111\ 001 \xrightarrow{\text{Overflow}}$

$$\begin{array}{r}
 \text{a)} \quad 32 \ 12 \ 17 \ 10 \ 12 \ 11 \ 10 \\
 + (-38) \quad (0) \ 16 \ 12 \quad (0) \ 35 \ 12 \quad (0) \ 17 \ 12 \quad (0) \ 8 \ 12 \\
 \hline
 00100000 \quad (0) \ 8 \ 12 \quad (0) \ 1 \ 12 \quad (0) \ 4 \ 12 \quad (0) \ 1 \ 12 \\
 + 10111010 \quad (0) \ 1 \ 12 \quad (1) \ 0 \quad (0) \ 2 \ 12 \quad (1) \ 0 \\
 \hline
 11011010
 \end{array}$$

$\hookrightarrow -38 \text{ em binário}$

11.(1p) Realize a conversão de

- a) $(00110, 0111)_2$ da base 2 para a base 10. Considere sistema de numeração com sinal em complemento de 2 (parte inteira de 5 bits).
- b) $(111001, 0111)_2$ da base 2 para a base 16. Considere sistema de numeração com sinal em complemento de 2 (Extensão da parte inteira de 6 bits até 8 bits). OBS. use letra maiúscula para a representação hexadecimal.

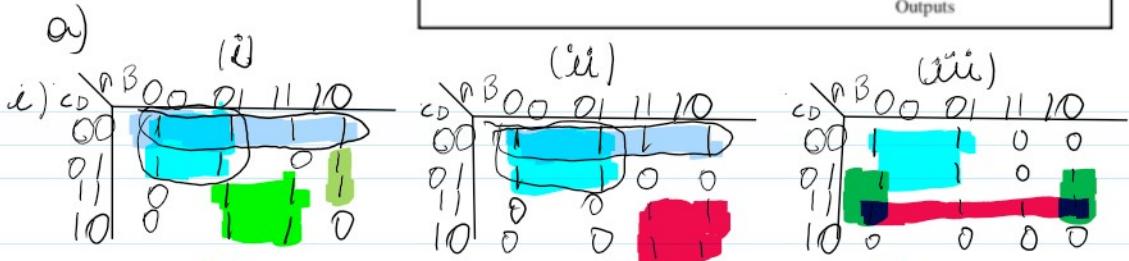
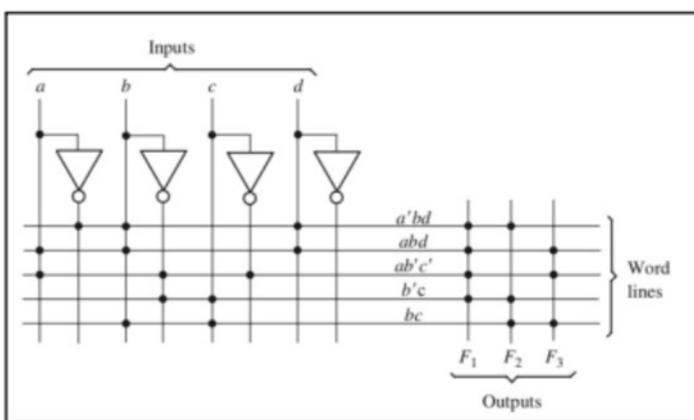
$$\text{a)} 2^0 + 2^1 + 2^2 + 2^{-3} + 2^{-4} \approx 4 + 2 + 0,25 + 0,125 + 0,0625 = 6,4375$$

$$\text{b)} (\underbrace{111}_{F}, \underbrace{100}_{9}, \underbrace{0111}_{7})_2 \rightarrow (F97)_{16}$$

12. (1p) (a) Encontre uma tabela PLA de linha mínima para implementar as seguintes equações:

$$\begin{aligned}
 \text{(i)} \quad x(A, B, C, D) &= \sum m(0, 1, 4, 5, 6, 7, 8, 9, 11, 12, 14, 15) \\
 \text{(ii)} \quad y(A, B, C, D) &= \sum m(0, 1, 4, 5, 8, 10, 11, 12, 14, 15) \\
 \text{(iii)} \quad z(A, B, C, D) &= \sum m(0, 1, 3, 4, 5, 7, 9, 11, 15)
 \end{aligned}$$

(b) Indique as conexões que serão feitas para programar um PLA para implementar sua solução na parte (a) em um diagrama semelhante à Figura.



$$F_1 + F_2 + F_3 \Rightarrow A\overline{BD} + \overline{AC} + BC + CD + \overline{CD} + AC$$

