

LE2: Circuitos Combinacionais

Dispositivos Combinacionais (Aula 3-1)

1. (0.5p) 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. $/C = \text{not } C$; $/D = \text{not } D$; $/E = \text{not } E$;))

Figura 1

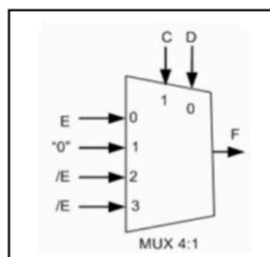


Figura 2

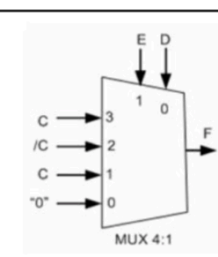


Figura 3

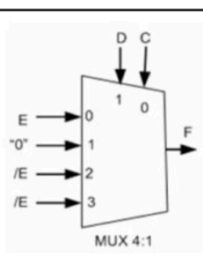
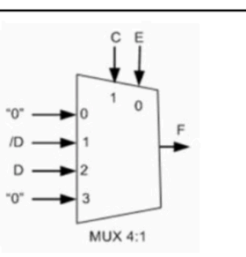


Figura 4



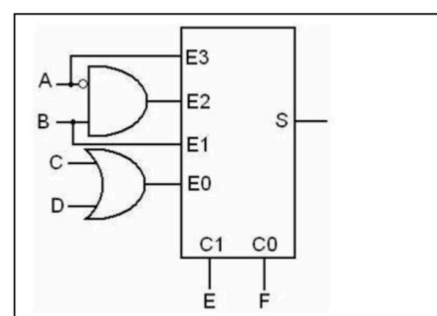
Escolher uma resposta.

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

2. (0.5p) 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?

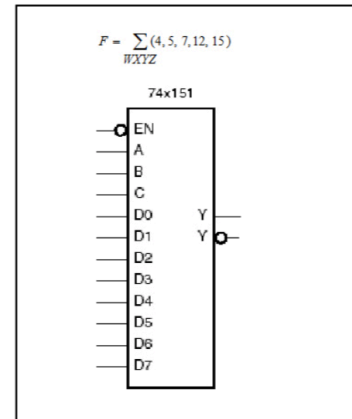
Resposta: a) Verdadeiro b) Falso



3. **(0.5p)** 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.

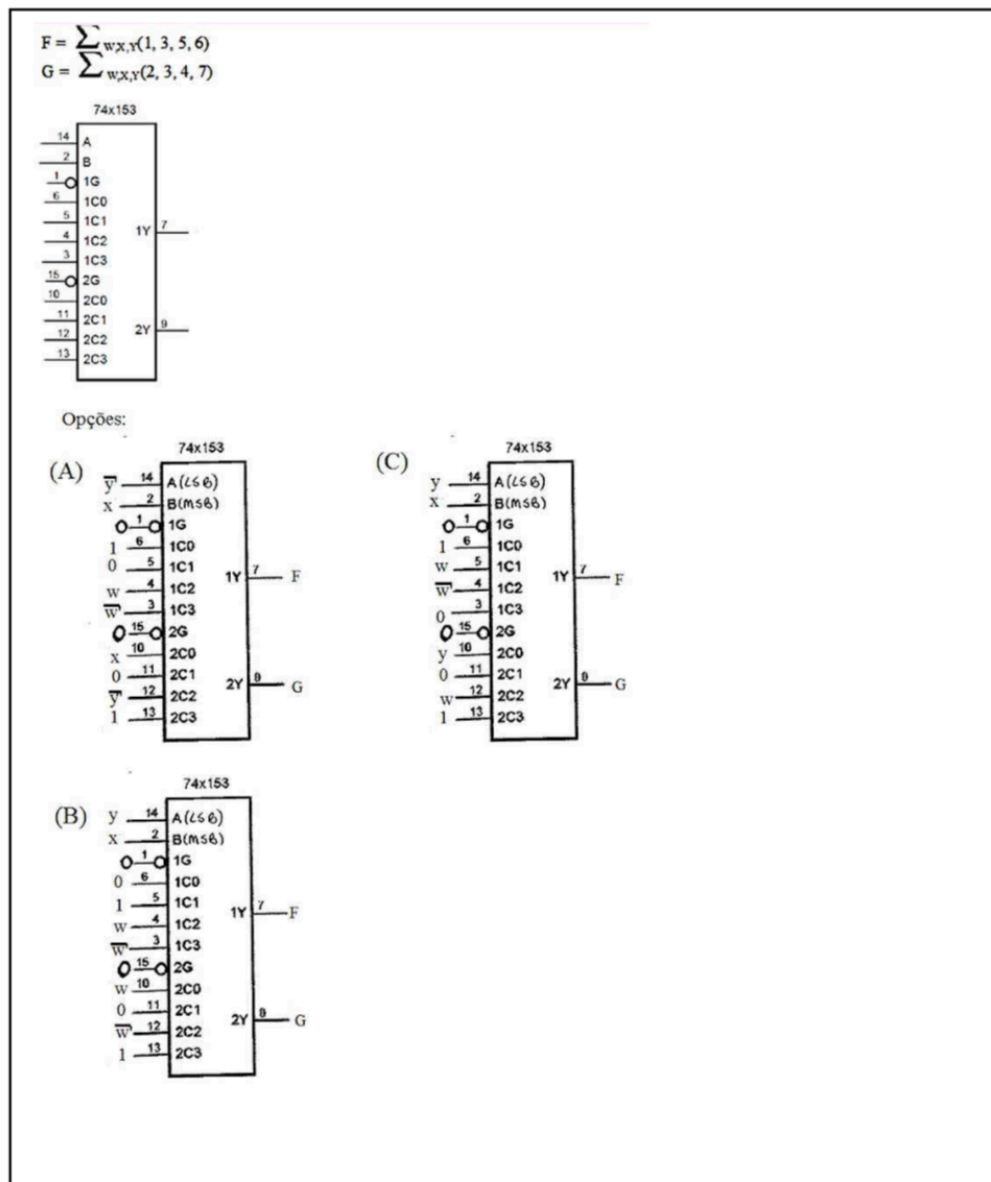
Escolher uma resposta

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



4. **(1p)** Projete as funções lógicas usando apenas o CI 74x153 (dois MUX 4-para-1). Não se esqueça de habilitar o CI.

Escolher uma resposta: a) Opção (B); b) Opção (A); c) Opção (C)



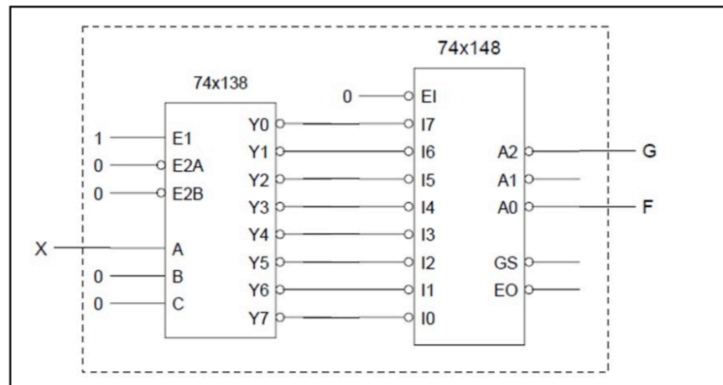
(De)Codificadores (Aula 3-2)

5. **(0.5p)** 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

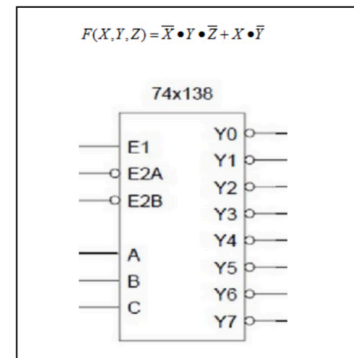


6. **(1p)** Projete a função lógica F usando apenas um CI 74x138 (decodificador 3-para-8) e uma porta lógica. Não se esqueça de habilitar o CI.

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

Escolher uma resposta

- a) E1=1; /E2A = /E2B = 0; A=Z; B=Y; C=X; F= NOT(/Y2 AND /Y6 AND /Y7)
- b) E1=1; /E2A = /E2B = 0; A=Z; B=Y; C=X; F= NOT(/Y2 AND /Y4 AND /Y5)
- c) E1=1; /E2A = /E2B = 0; A=Z; B=Y; C=X; F= NOT(/Y0 AND /Y3 AND /Y4)
- d) E1=1; /E2A = /E2B = 0; A=Z; B=Y; C=X; F= NOT(/Y2 AND /Y4 AND /Y6)
- e) E1=1; /E2A = /E2B = 0; A=Z; B=Y; C=X; F= NOT(/Y0 AND /Y2 AND /Y4)



7. **(0.5p)** Realize a conversão de
- a) $(1435)_{10}$ da base 10 para a base 8.
 - b) $(1101011)_2$ da base 2 para a base 16.
8. **(0.5p)** Realize a conversão de
- c) $(1101001)_2$ da base 2 para a base 10.
 - d) $(72)_8$ da base 8 para a base 16.

Unidade Lógica Aritmética (Aula 3-3)

9. **(0.5p)** 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 $OP=1$.
10. **(0.5p)** Fazer as operações de soma a seguir (mostrar todas as etapas intermediárias e os "carries")
- a) $4310 + 7721$ (Octal)
 - b) $4A34 + 8D$

11. **(0.5p)** 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.
- 6-3
 - 8-3
12. **(0.5p)** 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.
- 23-17
 - 32-70

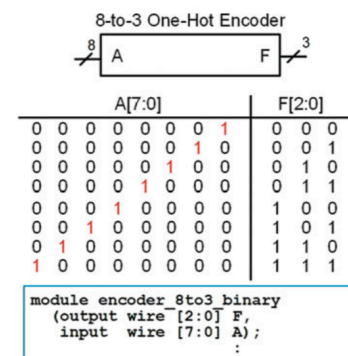
Aritmética de Ponte Flutuante (Aula 3-4)

13. **(0.5p)** Realize a conversão de
- $(00110,0111)_2$ da base 2 para a base 10. Considere sistema de numeração com sinal em complemento de 2 (parte inteira de 5 bits).
 - $(111001,0111)_2$ 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.

Dispositivos Programáveis (Aula 3-5)

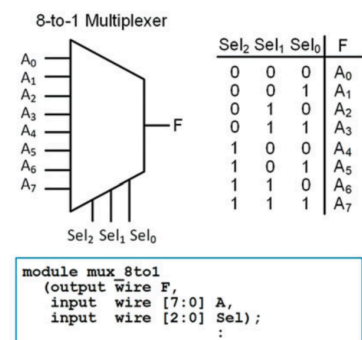
14. **(1p)** Projete um modelo Verilog para o codificador *one-hot* de oito-para-três mostrado na figura. Declare seu módulo e portas para coincidir com o diagrama de blocos fornecido

- Use atribuição contínua e operadores condicionais.
- Use atribuição contínua e operadores lógicos.
- Verifique e simule seus módulos com a ajuda de uma bancada de teste.



15. **(1p)** Projete um modelo Verilog para o multiplexador de oito-para-um mostrado na figura. Declare seu módulo e portas para coincidir com o diagrama de blocos fornecidos.

- Use atribuição contínua e operadores condicionais.
- Use atribuição contínua e operadores lógicos.
- Verifique e simule seus módulos com a ajuda de uma bancada de teste



16. **(0.5p)** (a) Encontre uma tabela PLA de linha mínima para implementar as seguintes equações:

(i) $x(A, B, C, D) = \sum m(0, 1, 4, 5, 6, 7, 8, 9, 11, 12, 14, 15)$

(ii) $y(A, B, C, D) = \sum m(0, 1, 4, 5, 8, 10, 11, 12, 14, 15)$

(iii) $z(A, B, C, D) = \sum m(0, 1, 3, 4, 5, 7, 9, 11, 15)$

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

