

Experimento 6:

IMPLEMENTAÇÃO DE CIRCUITOS COMBINACIONAIS COM MULTIPLEXADORES

1 OBJETIVO

Os conceitos de multiplexação e demultiplexação são apresentados, bem como sua utilização para implementar funções lógicas. Multiplexadores de 4 entradas usando portas lógicas serão projetados. Além disso, um somador será implementado a partir do multiplexador. Finalmente, será implementada uma função lógica de 7 variáveis usando-se um multiplexador e um decodificador.

2 INTRODUÇÃO TEÓRICA

2.1 Multiplexação e demultiplexação

Multiplexar significa selecionar dados dentre diversas fontes. A figura 1 mostra o esquema funcional generalizado de um multiplexador lógico. Nesse dispositivo, os terminais de seleção determinam o terminal de entrada de dados que terá seu conteúdo transferido para a saída.

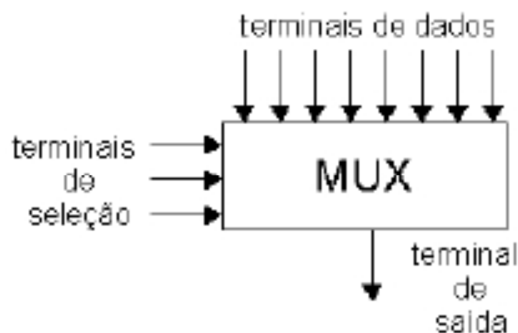


Figura 1 - Dispositivos eletrônicos de manuseio de dados: multiplexador.

A operação inversa é denominada demultiplexação. Como será mostrado adiante, o demultiplexador lógico (figura 2) é quase equivalente a um decodificador.

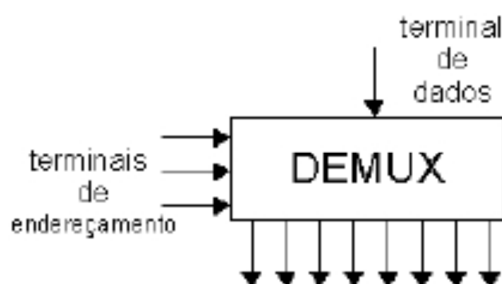


Figura 2 - Dispositivos eletrônicos de manuseio de dados: Demultiplexador.

As operações de multiplexação e demultiplexação são realizadas quando diversas fontes de dados compartilham de uma mesma unidade de processamento ou canal de transmissão (figura 3).

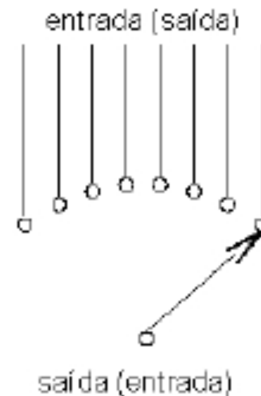


Figura 3 - Equivalente mecânico dos dispositivos eletrônicos de manuseio de dados.

2.2 Exemplos ilustrativos

Os multiplexadores e demultiplexadores também podem ser encarados simplesmente como circuitos combinacionais com diversos terminais de entrada e um de saída, ou vice-versa. O conceito de seleção de dados é mais uma questão de aplicação e ponto de vista do que de funcionamento.

Exemplo 1: Multiplexador com 4 terminais de dados (MUX-4)

Esse circuito é mostrado na figura 4. A tabela 1 mostra a tabela-verdade desse multiplexador em forma compactada. A tabela completa teria 64 linhas e, portanto, não seria uma maneira eficiente de exprimir seu funcionamento. Se, por exemplo, $E_1 = 1$ $E_2 = 0$, tem-se $S = D_1$. Esse circuito pode ser visto como um selecionador de dados.

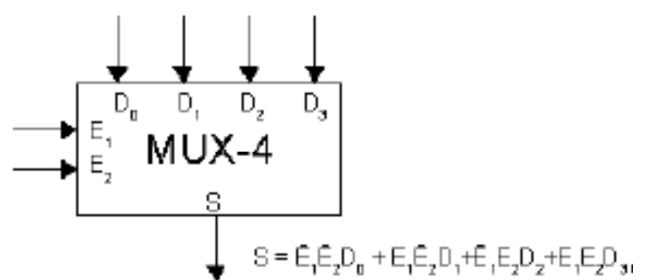


Figura 4 - MUX-4 e sua expressão booleana.

Tabela 1 - Tabela verdade compactada do MUX-4.

E_2	E_1	S
0	0	D0
0	1	D1
1	0	D2
1	1	D3

Exemplo 2: Demultiplexador com 4 saídas (DEMUX-4)

Esse circuito, mostrado na figura 5, é um decodificador onde D é um terminal de ativação. A tabela verdade é apresentada na tabela 2.

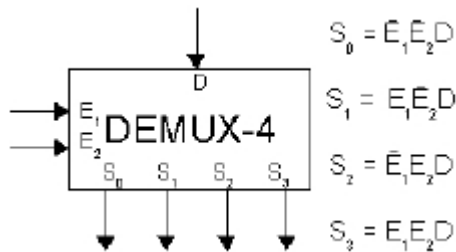


Figura 5 - DEMUX-4 com suas expressões booleanas.

Tabela 2 - Tabela verdade compactada do DEMUX-4.

D	E_2	E_1	S_0	S_1	S_2	S_3
0	X	X	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1

2.3 Implementação de multiplexadores

Os multiplexadores e demultiplexadores podem ser implementados com as portas lógicas conhecidas (figuras 6 e 7).

Dada sua grande aplicação na prática, os multiplexadores também são fabricados em forma integrada, com 2, 4, 8 ou mais terminais de entrada de dados. Os demultiplexadores são decodificadores com um terminal de ativação e, portanto também são encontrados em forma de circuitos integrados em média escala (MSI).

3 APLICAÇÃO DE MULTIPLEXADORES

Além de sua aplicação natural como selecionadores de dados, os multiplexadores podem ser utilizados para se implementar uma função booleana genérica. Eles são particularmente convenientes para tal fim quando a função a ser implementada é de natureza irregular, e não permite muita simplificação. Em um caso desses, o uso de

multiplexadores em lugar de portas lógicas convencionais resulta em um projeto mais fácil, mais compacto e mais flexível.

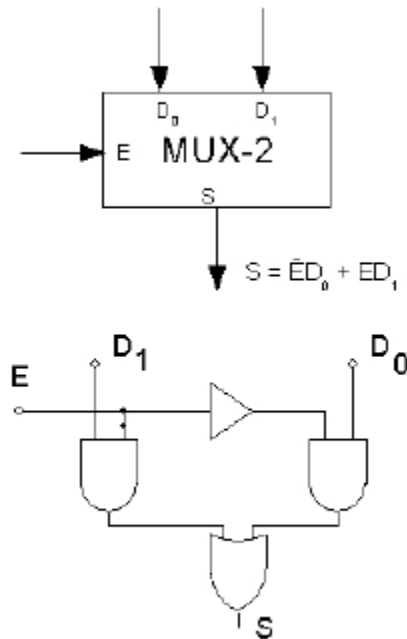


Figura 6 - Implementação de multiplexadores com componentes SSI.

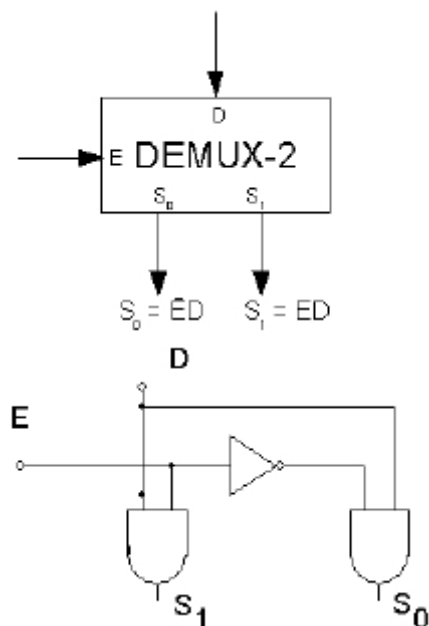


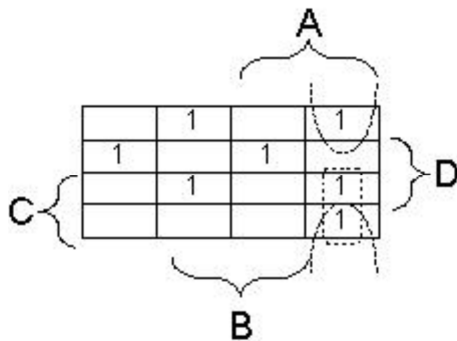
Figura 7 - Implementação de demultiplexadores com componentes SSI.

3.1 Exemplo

Projetar um circuito que realize a função f , descrita pela tabela-verdade apresentada na tabela 3, pelo mapa de Karnaugh apresentado na figura 8, e pela equação lógica apresentada na equação 1.

Tabela 3 - Tabela da função f .

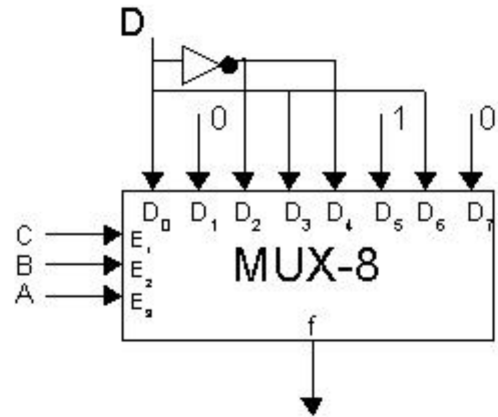
A	B	C	D	F
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0


Figura 8 - Mapa de Karnaugh da função f .

$$f = \overline{A}\overline{B}\overline{C}.D + \overline{A}.B.\overline{C}.\overline{D} + \overline{A}BCD + A.\overline{B}.\overline{C}.\overline{D} + \overline{A}BC\overline{D} + \overline{A}BCD + ABC\overline{D}$$

Equação 1 - Equação de Excitação da função f .

A figura 9 apresenta uma implementação da função f com um multiplexador de 8 entradas (MUX-8). Três das quatro variáveis independentes são escolhidas para acionar os terminais de seleção. Cada terminal de dado é então acionado por uma das quatro funções lógicas que a variável restante pode formar (D , \overline{D} , 1 ou 0).


Figura 9 - Implementação da função f com um MUX-8.

As três variáveis a serem ligadas nos terminais de seleção de dados (A, B e C, no exemplo) são escolhidas. Então, a tabela-verdade é dividida nos $2^3 = 8$ blocos onde essas variáveis são mantidas constantes. A saída f , dentro de cada um desses blocos, é função de D apenas e, portanto, existem somente quatro possibilidades: $f = 0$, $f = 1$, $f = D$ ou $f = \overline{D}$. Cada terminal de dados do MUX é acionado com a função do bloco correspondente.

Exercício: Realize a mesma função, porém, usando as variáveis B, C e D para acionar os terminais de seleção.

3.2 Técnica geral

A técnica de implementação introduzida acima pode ser generalizada para uma função de n variáveis. Considere por exemplo um multiplexador de 8 terminais de dados como o da Fig. 5. A expressão booleana da saída S é:

$$S = \overline{E_1}\overline{E_2}\overline{E_3}D_0 + \overline{E_1}\overline{E_2}E_3D_1 + \dots + E_1E_2E_3D_7$$

Qualquer função de $n > 3$ variáveis pode ser reescrita na forma

$$f(A, B, C, D, E, \dots) = \overline{A}\overline{B}\overline{C}F_0(D, E, \dots) + \overline{A}\overline{B}CF_1(D, E, \dots) + \dots + ABCF_7(D, E, \dots)$$

onde A, B e C são variáveis selecionadas arbitrariamente dentre as n variáveis, F_0, F_1, \dots, F_7 são funções das $(n-3)$ variáveis restantes, portanto mais simples que a função original f .

A identificação dessas duas expressões conduz à forma geral de implementação mostrada na Fig. 10. No caso particular em que $n = 4$, as funções F_0, F_1, \dots, F_7 são funções da única variável restante e existem apenas 4 possibilidades (1, 0, D , \overline{D}), como já foi visto. Também, se $n = 3$, então f estará na própria forma canônica de mintermos e, portanto, os fatores F_0, F_1, \dots, F_7 só podem ser ou identicamente iguais a 0 ou identicamente iguais a 1.

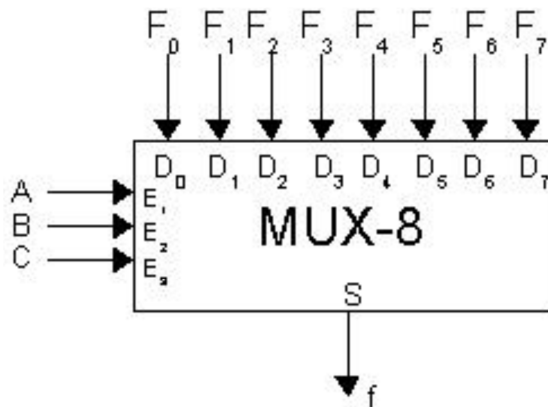
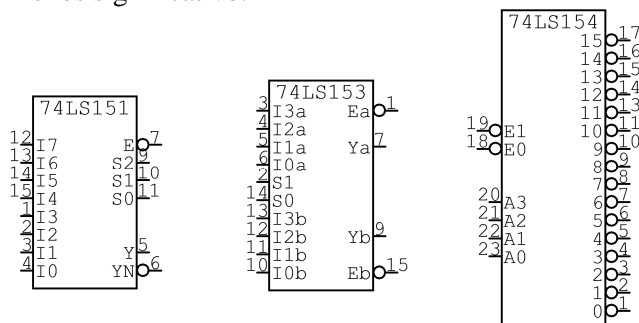


Figura 10 - Ligações de um MUX-8 para a implementação de uma função de $n > 3$ variáveis. No caso $n = 3$, é mais econômico o uso de um MUX-4.

A mesma técnica pode ser naturalmente estendida a multiplexadores com mais entradas. Um multiplexador de 16 entradas de dados, por exemplo, pode implementar qualquer uma das 2^{32} funções diferentes de 5 variáveis, com apenas uma única porta inversora adicional.

4 PARTE EXPERIMENTAL

O primeiro projeto pedido é a implementação de um MUX com portas lógicas e é um exercício elementar de síntese de circuitos combinacionais. Os outros 2 projetos referem-se à aplicação de multiplexadores na implementação de funções booleanas. A seguir são apresentadas as pinagens dos principais circuitos integrados que serão usados neste experimento. Tenha muita atenção com a ordem de significância das entradas, saídas e bits de seleção, pois esse é o erro mais comum neste experimento! Como regra geral, o subscrito “0” se refere ao bit menos significativo.



4.1 Projete e monte um multiplexador com 4 terminais de entrada de dados usando 5 portas NÃO-E e 2 portas INVERSORAS. Teste seu funcionamento ligando um diodo emissor de luz à saída. Controle as entradas de seleção com 2 chaves lógicas e verifique se o diodo sempre acende ou

apaga de acordo com a entrada de dados selecionada. Preencha uma tabela-verdade compactada.

4.2 Os multiplexadores de 4 entradas de dados são encontrados como circuitos integrados em média escala (MSI). Cada CI 74153 tem dois MUX-4 com entradas de seleção em comum, como mostra a figura 11. Use um MUX-4 duplo e uma porta inversora para implementar um somador completo. O somador completo é um circuito com 3 terminais de entrada e 2 terminais de saída que são usados em operações aritméticas (figura 12). Os terminais de saída exprimem a soma binária das 3 entradas (tabela 4). Use diodos emissores de luz e chaves lógicas para testar seu circuito.

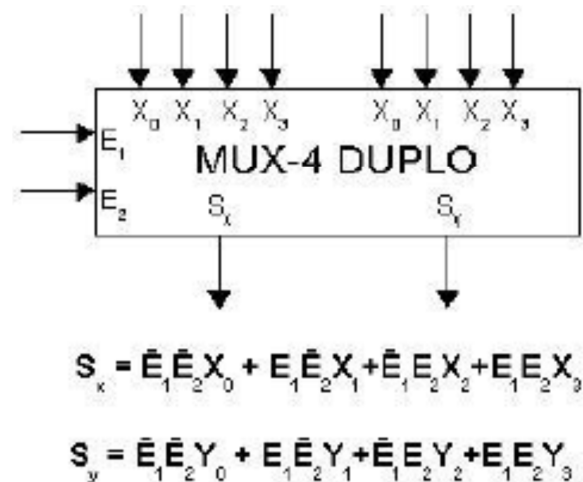


Figura 11 - Um MUX-4 duplo pode ser usado para implementar um somador completo.

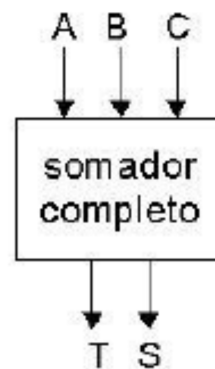


Figura 12 – Diagrama de um somador completo.

Tabela 4 – Tabela verdade do somador completo.

A	B	C	T	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

4.3 Implemente a função de 7 variáveis dada abaixo, usando um multiplexador de 8 entradas de dados (74151), um decodificador de 4 entradas (74154) e algumas portas adicionais (um 7400 será suficiente).

$$f(A, B, C, D, E, F, G) = F \cdot G + A \cdot B \cdot C \cdot D \cdot \overline{E} \cdot \overline{F} \cdot G + \overline{A} \cdot \overline{B} \cdot \overline{C} \cdot \overline{D} \cdot \overline{E} \cdot \overline{F} \cdot G + \overline{A} \cdot \overline{B} \cdot \overline{C} \cdot \overline{D} \cdot \overline{E} \cdot F \cdot G + \overline{A} \cdot \overline{B} \cdot C \cdot E \cdot F \cdot G + \overline{A} \cdot B \cdot C \cdot D \cdot E \cdot \overline{F} \cdot G + A \cdot B \cdot C \cdot D \cdot E \cdot \overline{F} \cdot G + A \cdot B \cdot C \cdot D \cdot E \cdot F \cdot G$$

Orientação: Use o decodificador como um gerador de mintermos para o multiplexador. O decodificador também é um circuito útil à implementação de funções complicadas, pois cada uma de suas saídas constitui um dos mintermos das variáveis de entrada. Observe, por exemplo, que qualquer função de 4 variáveis pode ser implementada com um DECOD-4 e uma porta OU (figura 13).

$$f = \overline{A} \cdot \overline{B} \cdot \overline{C} \cdot \overline{D} + \overline{A} \cdot \overline{B} \cdot C \cdot D + \overline{A} \cdot B \cdot \overline{C} \cdot D + A \cdot \overline{B} \cdot \overline{C} \cdot D$$

Dicas: Ligue as chaves A, B, C e D no decodificador, como mostrado na Figura 13. Ligue as chaves E, F e G nos terminais de seleção do mux-8. Não se esqueça de ativar os terminais de “enable” do MUX e do decodificador (em ambos, os terminais são ativos em nível baixo), e também de alimentar todos os CIs com terra e 5V. Simule o seu circuito no Circuit Maker antes de vir ao laboratório. Imprima o circuito com a pinagem e traga para a aula.

Atenção! As saídas do decodificador 75154 são invertidas (a saída ativada ficará em nível baixo, e as demais em nível alto). Por isso, seguindo o teorema de DeMorgan, o invés de portas OU (como na Figura 13), você deverá usar portas NÃO-E para combinar os mintermos. Além disso, os mintermos do decodificador que seriam ligados direto ao MUX (sem passar por portas OU) devem sem ser invertidos antes de ligados ao MUX. Use portas NÃO-E de duas entradas para implementar a(s) inversora(s) que você precisar, para economizar CIs.

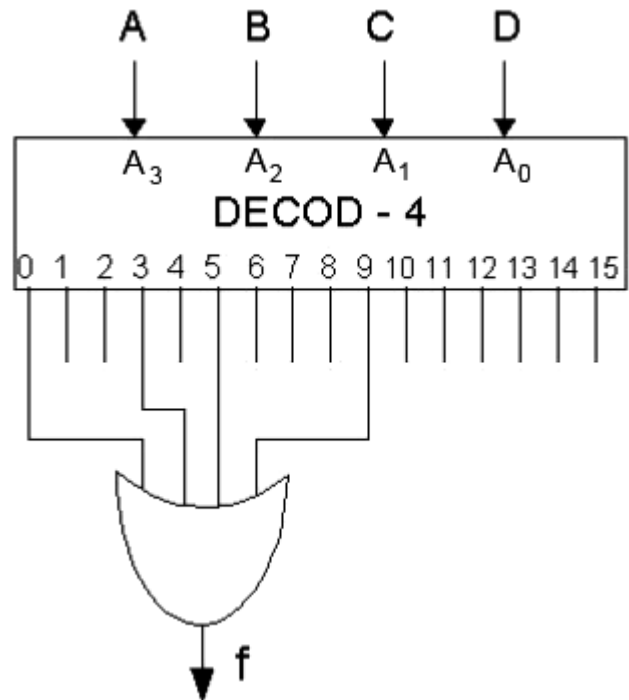


Figura 13 - Uso de um decodificador como gerador de mintermos para implementação de funções. Se o decodificador tiver saídas invertidas, a porta OU é substituída por uma porta NÃO-E.

5 EQUIPAMENTOS E MATERIAL

- Painel digital
- Portas lógicas
- MUX-4 duplo
- MUX-8
- Decodificador de 4 entradas

6 SUMÁRIO

A multiplexação e a demultiplexação são realizadas quando diversas fontes de dados compartilham de uma mesma unidade de processamento ou canal de transmissão.

Neste experimento, os multiplexadores são apresentados como circuitos combinacionais, e é mostrada a relação existente entre os demultiplexadores e decodificadores.

O enfoque principal é o da aplicação de multiplexadores na geração de funções Booleanas. Essa técnica de implementação é bem diferente e mais simples do que as convencionais.

7 INSTRUÇÕES PARA A EXECUÇÃO DO EXPERIMENTO

7.1 Projeto

Apresente os projetos dos três circuitos a serem implementados no laboratório (ver seções 4.1, 4.2 e 4.3). Apresente todos os elementos que forem

produzidos durante o projeto (quando se aplicar, apresente tabelas verdade, mapas de Karnaugh, equações, diagramas, esquemáticos com pinagens indicadas, etc.). Obs.: neste experimento, o mapa de Karnaugh é útil somente para o projeto do circuito da seção 4.1

7.2 Visto

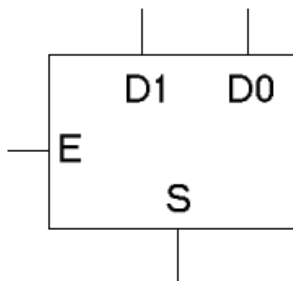
Os vistos serão dados se o grupo implementar os itens 4.1, 4.2 e 4.3 com sucesso. O item 4.1 valerá 2 pontos, enquanto os itens 4.2 e 4.3 valerão 3 pontos cada. Os demais 2 pontos corresponderão à nota de pré-relatório (projeto).

7.3 Relatório

O relatório é individual, deve ser feito à mão, e consiste em responder ao questionário abaixo. Cada questão vale 2 pontos.

1) Explique, com suas palavras o funcionamento dos multiplexadores. Você pode usar como exemplo um MUX de 4 entradas para facilitar a explicação.

2) Projete o MUX de 2 entradas ilustrado abaixo, e apresente o esquemático do circuito implementado somente com portas NÃO-E. Mostre o desenvolvimento (tabela verdade, mapa de Karnaugh e equação booleana).

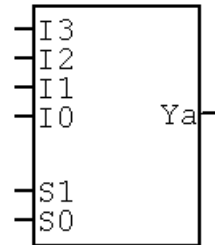


3) Mostre como implementar um MUX de 8 entradas usando um MUX duplo de 4 entradas (CI 74153) e mais

- a) somente quatro portas lógicas;
- b) somente duas portas lógicas e as entradas de “enable”.

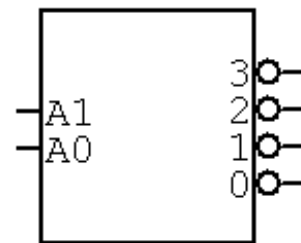
4) Usando o MUX de 4 entradas da figura abaixo e uma inversora, implemente a função lógica:

$$S = \overline{A} \cdot \overline{B} + \overline{A} \cdot B \cdot \overline{C} + A \cdot B \cdot C$$



Dica: faça S0 = A e S1 = B.

5) Usando o MUX de duas entradas da questão 2, duas portas NÃO-E de 2 entradas, e o decodificador binário-decimal de 2 bits ilustrado na figura a seguir, implemente a mesma função lógica da questão anterior.



Dica: faça A0 = A, A1 = B e E = C.

8 TESTE DE AUTO-AVALIAÇÃO (OPCIONAL)

Marque verdadeiro ou falso:

- a) () Os multiplexadores e demultiplexadores são circuitos combinacionais.
- b) () A tabela verdade completa de um multiplexador com 2 terminais de seleção de dados tem 64 linhas.
- c) () A tabela verdade completa de um multiplexador com 8 terminais de entrada de dados tem 2⁸ linhas.
- d) () A tabela verdade completa de um demultiplexador de 3 terminais de endereçamento tem 16 linhas e 8 colunas de saída.
- e) () A tabela verdade completa de um multiplexador tem exatamente o mesmo número de 0s e 1s.
- f) () O demultiplexador é um decodificador com um terminal de ativação. Assim sendo, os terminais de endereçamento do demultiplexador correspondem aos terminais de entrada do decodificador.



- g) () Continuando o item anterior, o terminal de ativação do decodificador corresponde ao terminal de dados do demultiplexador.
- h) () O multivibrador também é equivalente a um codificador.
- i) () Um multiplexador com 1 terminal de seleção pode implementar qualquer uma das 16 funções de 2 variáveis com apenas uma porta inversora adicional.
- j) () Um multiplexador com 4 terminais de dados pode implementar qualquer função de 4 variáveis com apenas uma porta inversora adicional.
- k) () Um multiplexador de 16 terminais de dados pode implementar qualquer função de 4 variáveis sem nenhuma porta adicional.