

Universidade Federal de Campina Grande - UFCG
Centro de Engenharia Elétrica e Informática - CEEI
Departamento de Engenharia Elétrica - DEE

Nome: Alysson Machado de Oliveira Barbosa

Email: alysson.barbosa@ee.ufcg.edu.br

Disciplina: Laboratório de Circuitos Lógicos

Professora: Fernanda Cecília Correia Lima Loureiro

Experimento 03 - Multiplexador e Demultiplexador e Bloco Lógico Universal

Este experimento consiste na realização de três montagens específicas e possui como objetivo geral o estudo das funções de multiplexação e demultiplexação, bem como, na utilização do multiplexador como um Bloco Lógico Universal - BLU.

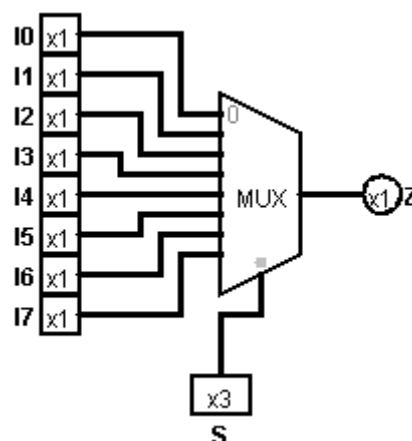
Objetivo 1

implementação e verificação do funcionamento do Sistema de Multiplexação 8:1, projetado como um seletor de dados de oito canais distintos.

Circuito 1 - a

Realizando a especificação de um MUX 8:1 que implementa a função $Z = f(S, I)$, em que Z é a saída, S representa as entradas de controle de seleção e I são as entradas de dados, teremos:

Figura 01 - Implementação de um MUX 8:1



Fonte: Imagem autoral.

Desse modo, analisando o circuito acima, podemos perceber que Multiplexadores ou só MUX, são circuitos combinacionais que têm como função principal selecionar através de chaves qual de suas entradas será transmitidas por

sua única saída. Assim, teremos que as entradas de controle de seleção estão dispostas, para ativar as entradas na saída, do seguinte modo (observe a tabela verdade).

Tabela Verdade 1 - a

S2	S1	S0	Saída
0	0	0	Z = I0
0	0	1	Z = I1
0	1	0	Z = I2
0	1	1	Z = I3
1	0	0	Z = I4
1	0	1	Z = I5
1	1	0	Z = I6
1	1	1	Z = I7

Expressões Lógicas 1 - a

Observando a tabela verdade acima, podemos obter as seguintes expressões lógicas:

$$Z_0 = \sim S_2 * \sim S_1 * \sim S_0 * I_0$$

$$Z_1 = \sim S_2 * \sim S_1 * S_0 * I_1$$

$$Z_2 = \sim S_2 * S_1 * \sim S_0 * I_2$$

$$Z_3 = \sim S_2 * S_1 * S_0 * I_3$$

$$Z_4 = S_2 * \sim S_1 * \sim S_0 * I_4$$

$$Z_5 = S_2 * \sim S_1 * S_0 * I_5$$

$$Z_6 = S_2 * S_1 * \sim S_0 * I_6$$

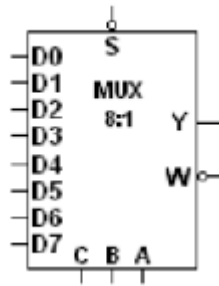
$$Z_7 = S_2 * S_1 * S_0 * I_7$$

$$Z = Z_0 + Z_1 + Z_2 + Z_3 + Z_4 + Z_5 + Z_6 + Z_7 + Z_8$$

Bloco Lógico 1 - b

Será considerado agora um multiplexador com uma entrada de habilitação S (NBA), três entradas (C,B,A) de controle de seleção, oito entradas de dados (D0, D1, ..., D7), uma saída de dados Y nominal, e uma saída de dados W invertida. Observe a figura 02 que ilustra esse problema:

Figura 02 - Bloco Lógico da MUX 8:1



Para esse problema, observe a tabela verdade abaixo. Em seguida, conforme a análise da tabela verdade, podemos obter uma expressão lógica que descreve a saída do multiplexador 8:1.

Tabela Verdade 1 - b

S (NBA)	C	B	A	Y	W
0	0	0	0	D0	$\sim Y$
0	0	0	1	D1	$\sim Y$
0	0	1	0	D2	$\sim Y$
0	0	1	1	D3	$\sim Y$
0	1	0	0	D4	$\sim Y$
0	1	0	1	D5	$\sim Y$
0	1	1	0	D6	$\sim Y$
0	1	1	1	D7	$\sim Y$
1	0	0	0	0	$\sim Y$
1	0	0	1	0	$\sim Y$
1	0	1	0	0	$\sim Y$
1	0	1	1	0	$\sim Y$
1	1	0	0	0	$\sim Y$
1	1	0	1	0	$\sim Y$
1	1	1	0	0	$\sim Y$
1	1	1	1	0	$\sim Y$

Expressões Lógicas 1 - b

$$Y0 = \sim S * \sim C * \sim B * \sim A * D0$$

$$Y1 = \sim S * \sim C * \sim B * A * D1$$

$$Y2 = \sim S * \sim C * B * \sim A * D2$$

$$Y3 = \sim S * \sim C * B * A * D3$$

$$Y4 = \sim S * C * \sim B * \sim A * D4$$

$$Y5 = \sim S * C * \sim B * A * D5$$

$$Y6 = \sim S * C * B * \sim A * D6$$

$$Y7 = \sim S * C * B * A * D7$$

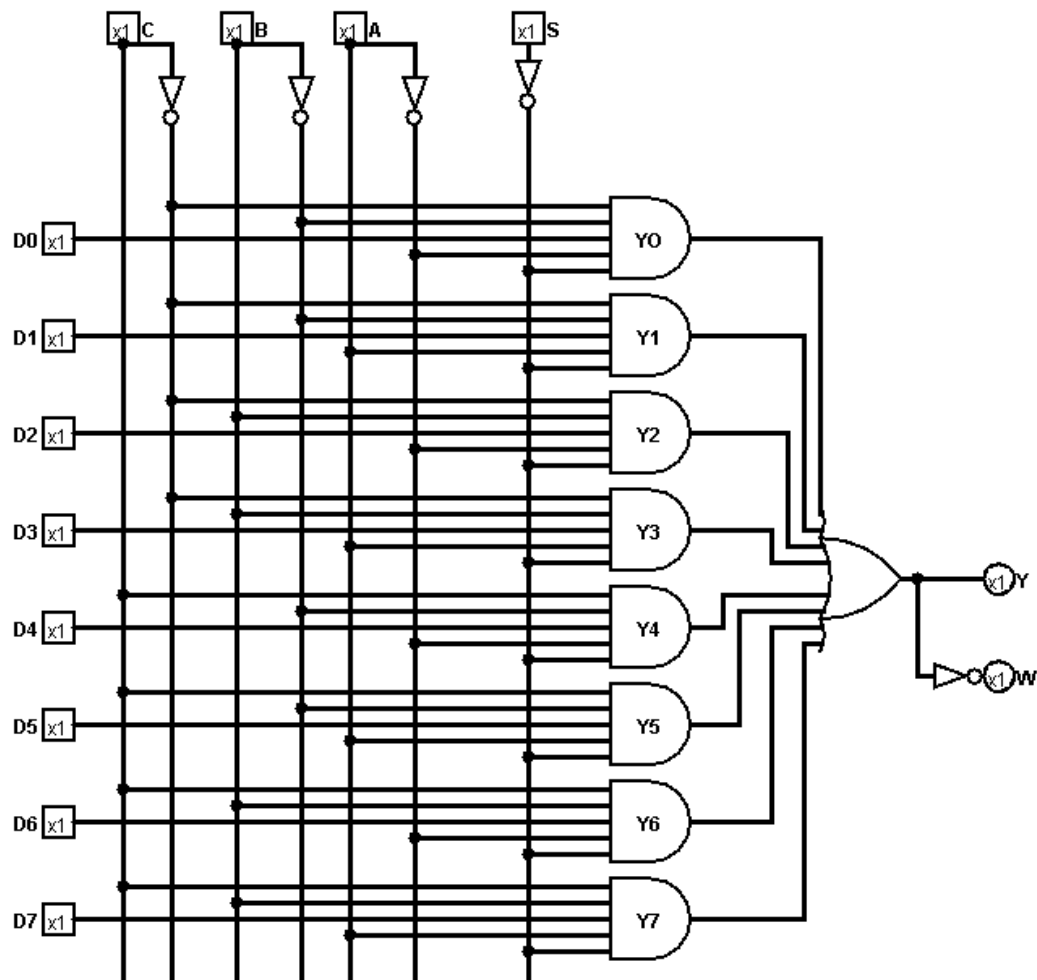
$$Y = Y0 + Y1 + Y2 + Y3 + Y4 + Y5 + Y6 + Y7 + Y8$$

$$W = \sim Y$$

Circuito 1 - b

Analisando todos os apontamentos acima, podemos determinar o diagrama lógico da seguinte forma:

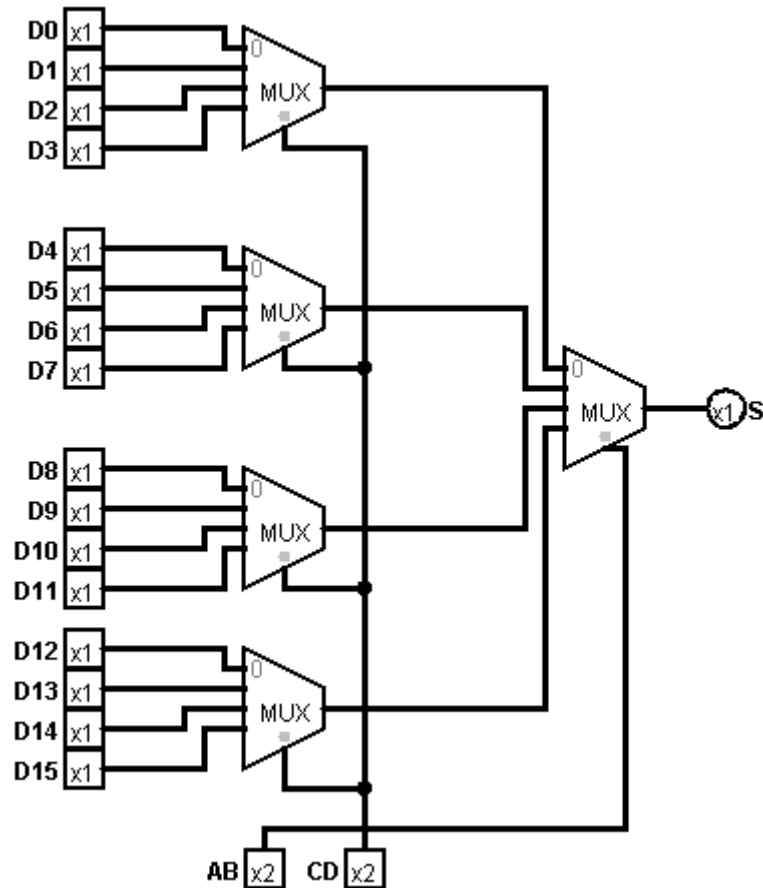
Figura 03 - Multiplexador implementado com portas lógicas



Fonte: Imagem Autoral

Circuitos MUX 4:1, podem ser utilizados para implementar um MUX 16:1, pois poderemos reutilizar plexers menores de modo a obter um maior. Para provar que isso está definido, observe o diagrama lógico abaixo:

Figura 04 - Multiplexador 16:1 usando multiplexadores 4:1



Fonte: Imagem Autoral

Objetivo 2

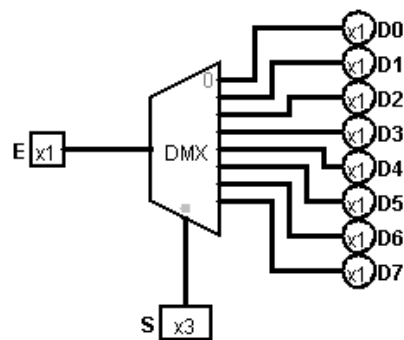
Especificação e verificação do funcionamento de um demultiplexador 1:8.

Circuito 2 - a

Demultiplexadores ou Demux, são responsáveis pela operação inversa ao multiplexador, ou seja, escolhe em qual saída a informação de sua única entrada irá sair.

Seja um DEMUX 1:8 que implementa a função $D = f(S, E)$, em que D são as saídas de dados, S são as entradas de controle de seleção e E é a entrada de dados, podemos representar um demultiplexador 1:8 da seguinte forma:

Figura 05 - Demultiplexador 8:1



Fonte: Imagem Autoral

Tabela Verdade 2 - a

Observe a tabela verdade obtida para esse problema:

S2	S1	S0	D0	D1	D2	D3	D4	D5	D6	D7
0	0	0	E	0	0	0	0	0	0	0
0	0	1	0	E	0	0	0	0	0	0
0	1	0	0	0	E	0	0	0	0	0
0	1	1	0	0	0	E	0	0	0	0
1	0	0	0	0	0	0	E	0	0	0
1	0	1	0	0	0	0	0	E	0	0
1	1	0	0	0	0	0	0	0	E	0
1	1	1	0	0	0	0	0	0	0	E

Expressões Lógicas 2 - a

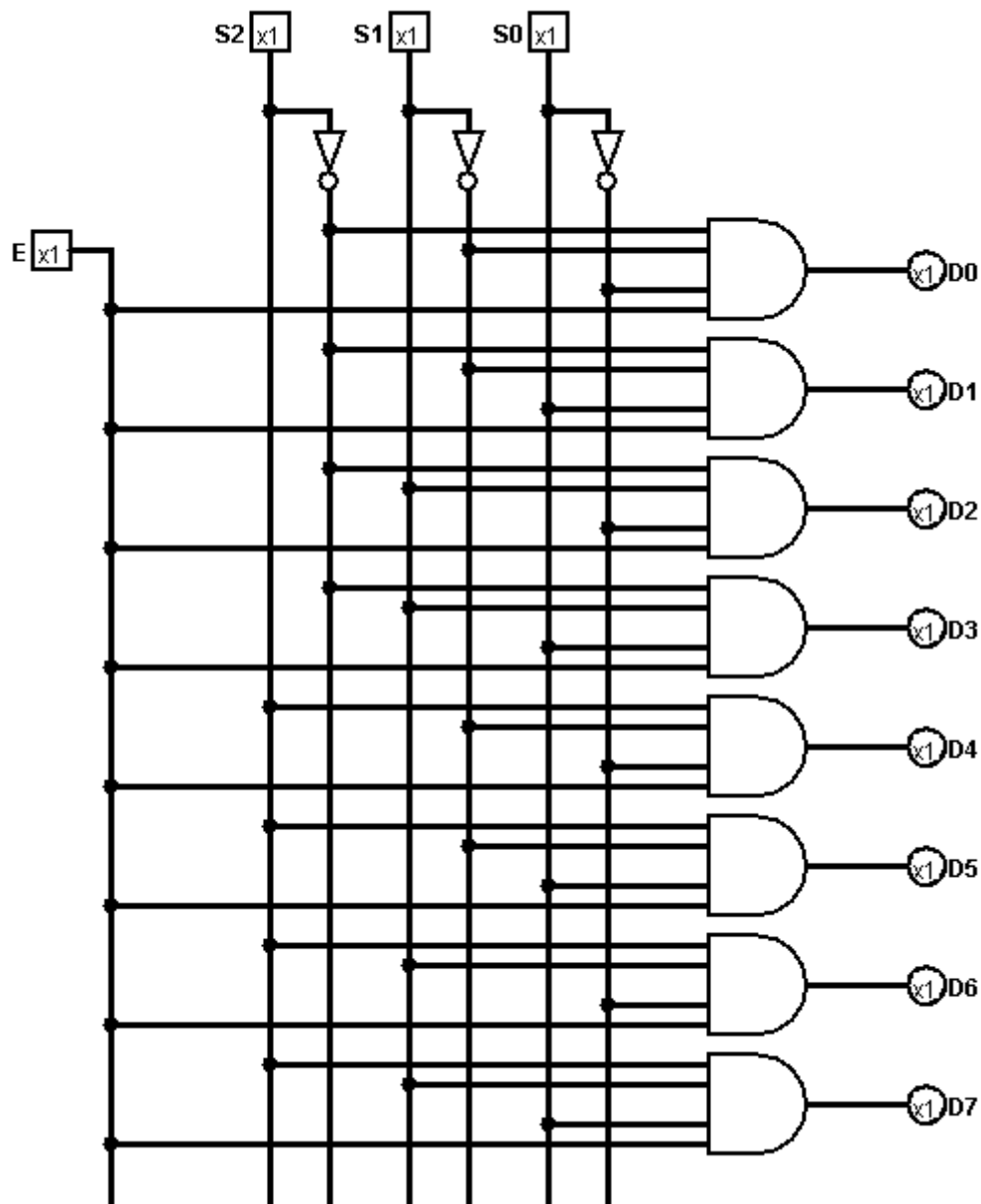
Analisando a tabela verdade para esse problema, podemos obter as seguintes expressões lógicas:

$$\begin{aligned}
 D0 &= \sim S2 * \sim S1 * \sim S0 * E \\
 D1 &= \sim S2 * \sim S1 * S0 * E \\
 D2 &= \sim S2 * S1 * \sim S0 * E \\
 D3 &= \sim S2 * S1 * S0 * E \\
 D4 &= S2 * \sim S1 * \sim S0 * E \\
 D5 &= S2 * \sim S1 * S0 * E \\
 D6 &= S2 * S1 * \sim S0 * E \\
 D7 &= S2 * S1 * S0 * E
 \end{aligned}$$

Circuito 2 - b

Implementando esse circuito usando portas lógicas teremos:

Figura 06 - Demux 8:1 usando portas lógicas



Fonte: Imagem autoral

Objetivo 3

Especificação, implementação e verificação do funcionamento de uma função lógica de quatro variáveis, com o projeto realizado a partir de um multiplexador 8:1, utilizado como Bloco Lógico Universal. Implemente um sistema Detector de Paridade Par, em função de quatro variáveis D, C, B, A . Apresentar a tabela funcional, expressão lógica e diagrama lógico para a função do Detector, $Y(D, C, B, A)$. Usar um

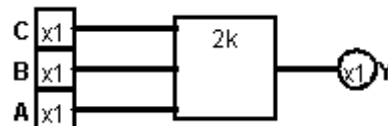
multiplexador 8:1, como bloco lógico universal, para implementar o Detector de Paridade Par.

O verificador de paridade é utilizado em transmissão para minimizar erros. Temos que 1 bit extra é anexado ao conjunto de bits para verificar a sua paridade, e com isso, seja o bit de paridade podendo ser 1 ou 0, poderemos analisar o conjunto do número de 1's contido no conjunto de bits do código.

Bloco Lógico 3

Vamos implementar um sistema de paridade par, em que o bit anexado serve para tornar o número total de 1's par. Seja $Y = (D,C,B,A)$, com C, B, A sendo as chaves seletoras do circuito, D as entradas de 8 bits e Y a saída do circuito, poderemos ter a seguinte tabela verdade para o problema do verificador de paridade par:

Figura 07 - Verificador de Paridade Par



Fonte: Imagem Autoral

Tabela Verdade 3

D	C	B	A	Y
1	0	0	0	1
0	0	0	1	0
0	0	1	0	0
1	0	1	1	1
0	1	0	0	0
1	1	0	1	1
1	1	1	0	1
0	1	1	1	0

Analisando a tabela verdade, poderemos ter a seguinte expressão para a saída Y:

Expressão Lógica 3

$$Y = (\sim C * \sim B * \sim A * D0) + (\sim C * B * A * D3) + (C * \sim B * A * D5) + (C * B * \sim A * D6)$$

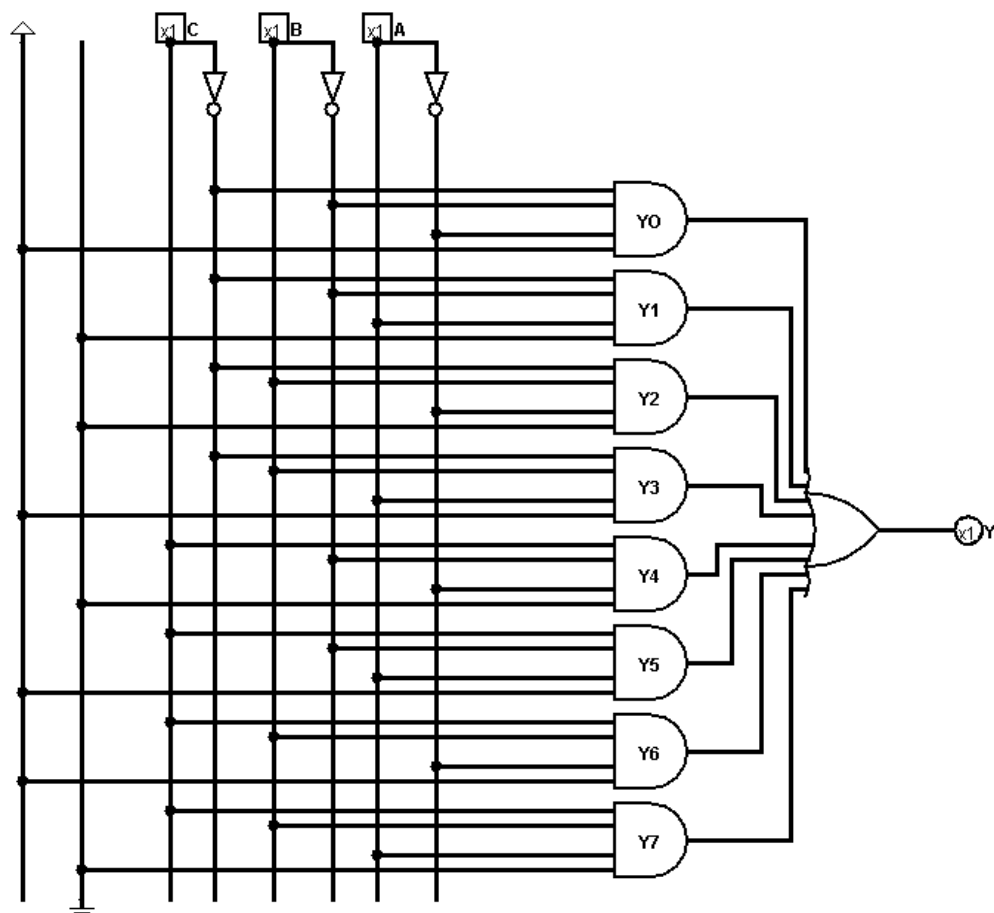
É simples perceber que esse detector de paridade par pode ser implementado com um MUX 8:1, esse que possui a seguinte expressão lógica:

$$\begin{aligned}
 Y0 &= \sim S * \sim C * \sim B * \sim A * D0 \\
 Y1 &= \sim S * \sim C * \sim B * A * D1 \\
 Y2 &= \sim S * \sim C * B * \sim A * D2 \\
 Y3 &= \sim S * \sim C * B * A * D3 \\
 Y4 &= \sim S * C * \sim B * \sim A * D4 \\
 Y5 &= \sim S * C * \sim B * A * D5 \\
 Y6 &= \sim S * C * B * \sim A * D6 \\
 Y7 &= \sim S * C * B * A * D7 \\
 Y &= Y0 + Y1 + Y2 + Y3 + Y4 + Y5 + Y6 + Y7
 \end{aligned}$$

Circuito 3

Observe que temos expressões lógicas equivalentes para o verificador de paridade par e o MUX 8:1, de modo que seja possível manipular as entradas da MUX 8:1 para transformá-lo em um verificador de paridade par. Reaproveitando o multiplexador da Figura 03, poderemos obter o seguinte diagrama lógico para esse problema:

Figura 08 - Detector de Paridade Par usando uma MUX 8:1



Fonte: Imagem Autoral