Experimento 2
Portas Lógicas: NAND, NOR e XOR

Isaac Lopes, 12/0120801
Lucas Mafra Chagas, 12/0126443
Marcelo Giordano Martins Costa de Oliveira, 12/0037301

[1]Dep. Ciência da Computação – Universidade de Brasília (UnB)
CiC 116351 - Circuitos Digitais - Turma C

{giordano.marcelo, chagas.lucas.mafra, isaaclopinho}@gmail.com

Abstract. *This essay has the intuition to give first contact with logical gates NAND e NOR, and discuss De Morga's theorem and fan-in and fan-out concepts.*

Resumo. *Esse relatório tem o intuito de dar um contato com as portas NAND e NOR, além de discutir o teorema de De Morgan e os conceitos de fan-in e fan-out.*

Objetivos

Analisar experimentalmente as portas lógicas NAND, NOR e XOR, mediante o estudo de suas respectivas tabelas da verdade e equivalências lógicas. Além disso, pretende-se verificar o caráter universal das portas NAND e NOR. Por fim, são observados e discutidos o teorema de De Morgan e os conceitos de fan-in e fan-out.

Materiais

- Painel Digital;
- *protoboard*;
- Ponta Lógica;
- Fios;
- Portas NAND e XOR.

Introdução

Portas NAND, NOR e XOR:

Quando implementamos circuitos, existem certos conjuntos de portas que são universais, ou seja, eles são capazes de representar qualquer expressão lógica sozinhos. As portas NAND e as portas NOR são dois exemplos de conjuntos de portas universais. A porta NAND representa a negação da expressão lógica AND, e portanto, ela tem a tabela verdade com saídas inversas à da tabela verdade da expressão AND. O mesmo ocorre para a porta lógica NOR, que é a negação da expressão lógica OR.

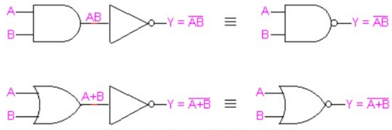

Figura 1 – Portas NAND e NOR

Porta NAND		
Entradas		Saída
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

Porta NOR		
Entradas		Saída
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

Figure 1. Portas NAND e NOR

Além das portas NAND e NOR, existem outras portas que, apesar de não serem universais tem aplicação muito útil: as portas XOR e XNOR. A primeira porta é conhecida como OU-exclusivo. Ela compara dois bits e a saída será 1 se e somente se os bits forem diferentes. Se houver mais de duas entradas, aí será 1 se houver um numero ímpar de valores verdadeiros com entrada. Já a porta XNOR, que produz a tabela verdae complementar da porta XOR, irá ter saída 1 quando as duas entradas forem iguais, ou, para o caso de mais de duas entradas, quando o número de entradas 1 for par.

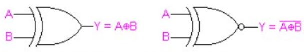

Figura 2 – Portas XOR e XNOR

Porta XOR		
Entradas		Saída
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

Porta XNOR		
Entradas		Saída
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

Figure 2. Portas XOR e XNOR

A expressão dasportas XOR e XNOR podem ser escritas em termos das portas AND, OR e NOT: $A \oplus B$ = (NOT A . B) + (NOT B . A) NOT($A \oplus B$) = (AB) + (NOT A . NOT B)

Fatores de Carga (fan-in, fan-out):

O tempo de chaveamento de uma porta lógica depende do número de portas alimentadas pela saída. O Fan-out de uma porta é o número de portas que pode ser alimentado na saída e depende de como a porta é utilizada na sequência lógica. Ele representa o número máximo de entradas lógicas que uma saída pode acionar com segurança. Se o valor estabelecido pelo FAN-OUT for excedido, a tensão de nível lógico de saída não poderá ser mais garantida. Este conceito se aplica quando ocorre o consumo de energia das portas ligadas na saída. Seu valor depende da tecnologia empregada:

• TTL: 2 a 10 • CMOS: 50 a 100

Além disso, o termo Fan-in é utilizado para representar o número máximo de entradas que uma porta lógica possui. Para a para a série TTL 74XX, utilizada nos experimentos da matéria, tem-se:

1 unidade de carga TTL = 40 μA, no nível lógico 1. = 1,6 mA, no nível lógico 0.

Em outras palavras, uma porta 7400 que necessite de uma corrente de entrada máxima de IIL = 1,6 mA para o nível lógico 0 e uma corrente de entrada máxima IIH = 40 μA para o nível lógico 1 é especificada como tendo um fator de carga unitário. Isto é, possui um fan-in de 1. Por outro lado, a saída de uma porta 7400 absorverá 16 mA no nível lógico 0 e fornecerá 800 μA no nível lógico 1. Portanto, ela tem capacidade de acionar 10 portas no nível lógico 0 (pois 16 mA / 1,6 mA = 10). Isto é, possui um fan-out de 10 para o nível lógico 0. Da mesma forma, o fan-out para o nível lógico 1 é 800 μA / 40 μA = 20.

Teorema de De Morgan:

Um teorema muito útil e que comprova a universalidade de certos conjuntos de portas é o teorema de DeMorgan. Este teorema estabelece que:

1. $\overline{A} + \overline{B} = \overline{A.B}$
2. $\overline{A}\ .\ \overline{B} = \overline{A + B}$

Isso, em outras palavras, quer dizer que, para retirar a negação de uma expressão devemos negar as partes e intercambiar a expressão entre elas. A partir disso é possível concluir o porque da universalidade das portas NAND e NOR. Nessas portas temos uma expressão negada, ou seja, é possivel criar variáveis negadas e obter a expressão lógica não originalmente testada pela porta. Para as portas NAND temos a seguinte relação:

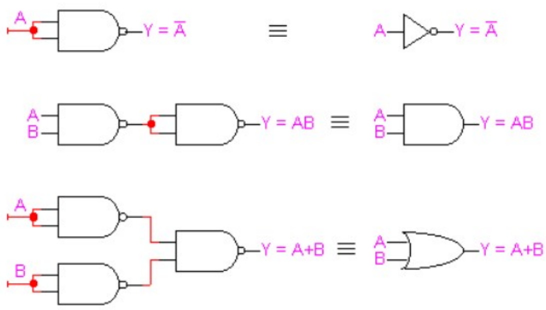

Figure 3. Universalidade da porta NAND

Já para a porta NOR, temos:

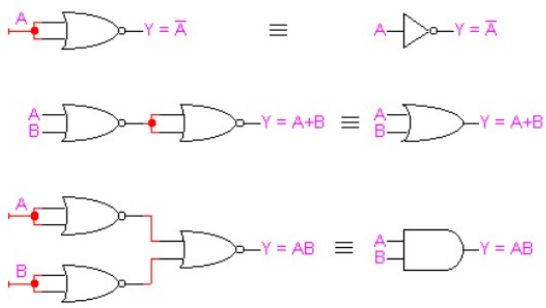

Figure 4. Universalidade da porta NOR

Neste experimento iremos provar que esta universalidade é de fato verdadeira, e iremos explorar o funcionamento das portas NAND, NOR e XOR.

Procedimentos

- Na primeira parte do experimento, foi implementado uma porta NAND de três entradas, utilizando-se três portas NAND de duas entradas:

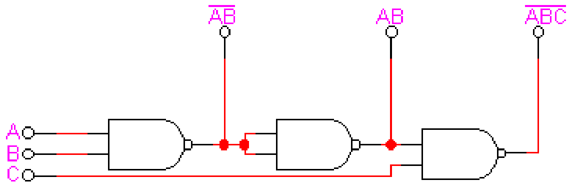

Figure 5. Esquema da porta NAND de três entradas

Após a devida implementação do circuito, foi preenchida a tabela da verdade para as saídas $\overline{A.B}$, A.B e $\overline{A.B.C}$ imagem abaixo apresenta o circuito montado:

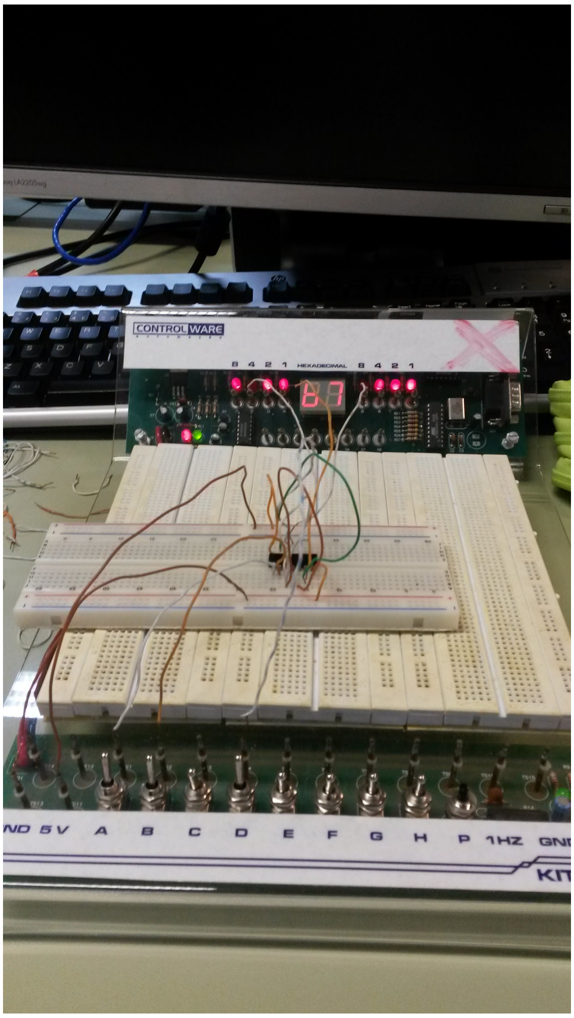

Figure 6. Porta NAND de três entradas

- Na segunda parte do experimento, implementou-se uma função XOR mediante a utilização de quatro portas NAND:

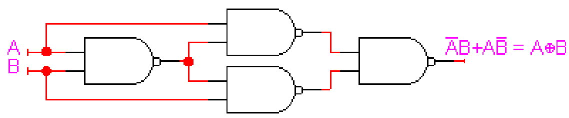

Figure 7. Esquema da função XOR com portas NAND.

Após a devida implementação do circuito, foi preenchida a tabela verdade correspondente à função XOR.

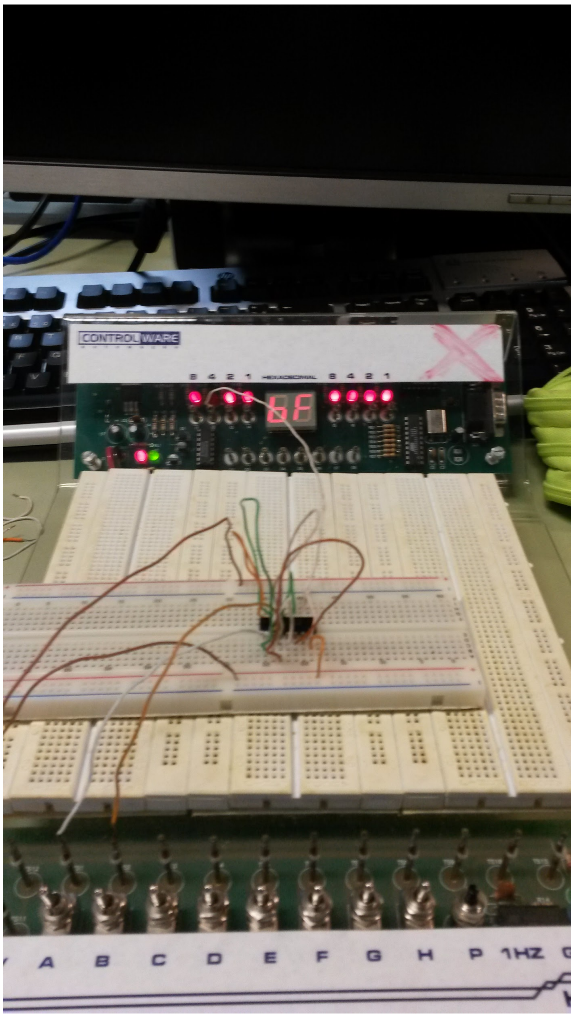

Figure 8. Função XOR com portas NAND.

- Na última parte do experimento, uma porta XOR de quatro entradas foi projetada e implementada a partir da utilização de três portas XOR de duas entradas.

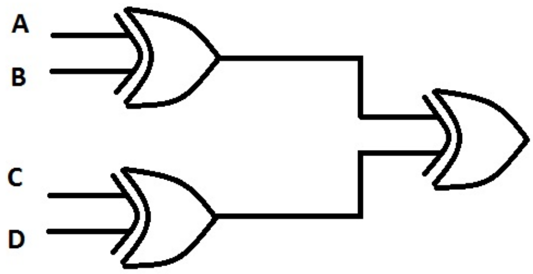

Figure 9. Função XOR de quatro entradas.

Após a devida projeção e implementação do circuito, foram verificados os casos nos quais a saída do circuito corresponde a 1.

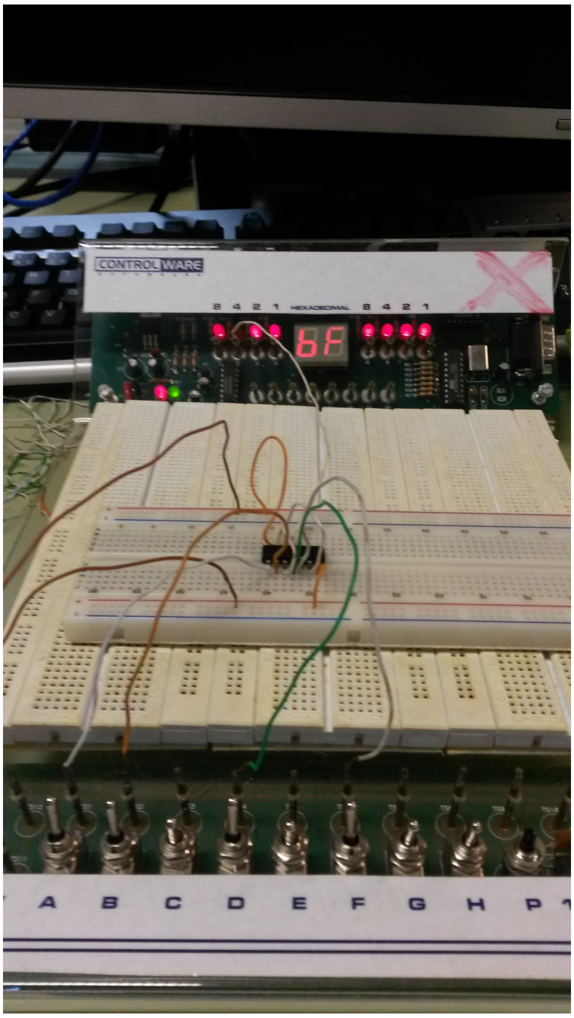

Figure 10. Função XOR com 4 entradas.

Implementação de uma porta NAND de 3 entradas.

A	B	C	$\overline{A.B}$	A.B	$\overline{A.B.C}$
0	0	0	1	0	1
0	0	1	1	0	1
0	1	0	1	0	1
0	1	1	1	0	1
1	0	0	1	0	1
1	0	1	1	0	1
1	1	0	0	1	1
1	1	1	0	1	0

Implementação da função XOR usando portas NAND.

A tabela verdade obtida após a construção do circuito encontra-se abaixo:

A	B	A(XOR)B
0	0	0
0	1	1
1	0	1
1	1	0

Verificação da função XOR usando a porta XOR (CI7486)

A tabela verdade obtida para esta parte do experimento foi a mesma obtida na parte 2.2 (acima).

Implementação de uma porta XOR de 4 entradas usando portas XOR de 2 entradas (CI 74LS86).

A	B	C	D	f(A,B,C,D)
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
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	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Análise dos Resultados

Para todas as partes deste experimento temos que os dados obtidos corresponderam ao esperado, sendo iguais aos resultados teóricos. Para ambas a parte 2.1 e a parte 2.2 vemos a aplicação direta do Teorema de DeMorgan, já que é possível obter uma função AND e uma função OR (que compoe a função XOR) a partir de uma negação da função AND. Para a parte 2.1 foi provado como a porta AND pode ser substituída por 2 portas NAND gerando a mesma tabela verdade. Temos:

f(A,B) = NOT(AB) Aplicando outra porta NAND, usando f(A,B) para as duas entradas, ficamos com: f(A,B) = NOT(NOT(AB) . NOT(AB) = NOT(NOT(AB)) De acordo com DeMorgan: f(A,B) = NOT (NOT A + NOT B) = NOT NOT A . NOT NOT B f(A,B) = AB Chegando a esse ponto, foi possivel implementar uma porta NAND de três entradas, incluindo f(A,B) e uma nova chave C, finalmente chegando à:

g(A,B,C) = NOT(f(A,B) . C) = NOT(ABC)

Para a parte 2.2 foi comprovada novamente essa universalidade da porta NAND, com a construção de uma porta XOR sendo feita apenas com a utilização da negação de AND. Temos, incialmente:

f(A,B) = NOT(AB) Por DeMorgan, f(A,B) = NOT A + NOT B Pegando este resultado e adicionando-o a uma nova porta NAND junto com a chave A: g(A, f(A,B)) = NOT ((NOT A + NOT B) . A) = NOT(A . NOT B) = NOT A + B Para a outra porta, onde o resultado é inserido em uma porta NAND com a chave B, temos: h(B, f(A,B)) = NOT ((NOT A + NOT B) . B) = NOT(B . NOT A) = NOT B + A Finalmente, juntando h e g em uma última porta NAND, ficaremos com:

F(A,B) = NOT((NOT A + B)(NOT B + A)) = NOT(NOT A + B) + NOT(NOT B + A) = (NOT NOT A . NOT B) + (NOT NOT B . NOT A) = (A . NOT B) + (B . NOT A)

Que nada mais é do que a função XOR.

Finalmente, na parte 2.4 vemos como a presença de apenas duas portas não deve nos limitar a utilizar apenas duas portas, pois é possível utilizar vários circuitos de duas portas para reproduzir circuitos com mais portas. Com 3 portas XOR foi possível reproduzir o resultado de caso a porta tivesse 4 entradas. Temos que, para duas portas XOR:

f(A,B) = (A . NOT B) + (B . NOT A) g(C,D) = (C . NOT D) + (C . NOT D) Juntando f e g em uma última porta XOR, ficaremos com: h(A,B,C,D) = (NOT((A . NOT B) + (B . NOT A)) . (C . NOT D) + (C . NOT D)) + (NOT (C . NOT D) + (C . NOT D) . (A . NOT B) + (B . NOT A))

Simplificando esta expressão, ficaremos com a soma dos minitermos indicando os oito casos em que a saída do circuito será 1:

h(A,B,C,D) = ABD(NOT C) + ABC(NOT D) + (NOT A)(NOT B)(NOT C)D + (NOT A)(NOT B)(NOT D)C + CD(NOT A)B + CD(NOT B)A + (NOT C)(NOT D)(NOT A)B + (NOT C)(NOT D)(NOT B)A

É possível notar que, nesta simplificação, serão verdadeiros apenas os casos em que temos um número ímpar de entradas 1.

Conclusão

Formos apresentados às portas NAND e XOR, pudemos fazer de ambas as partes a implementaçao das tabelas verdade. Além disso, verificamos o Teorema DeMorgan e a fundamentabilidade da porta NAND.

Auto-Avaliação

1. B
2. A
3. D
4. D
5. B