

SIMPLIFICAÇÃO DE CIRCUITOS LÓGICOS COMBINACIONAIS: MAPAS DE KARNAUGH

BARROS, E. C.¹, NASCIMENTO, L. A. F.¹, MOURA, A. F. L.¹, EGOAVIL, C. J.²

¹Monitor (a) da disciplina DE SISTEMAS DIGITAIS - DAEE, Fundação Universidade Federal de Rondônia, Sala 206 - 2C, Porto Velho, Rondônia, Brasil (e-mail: eletronicamonitoria@gmail.com)

²Professor do Curso de Engenharia Elétrica - DAEE, Fundação Universidade Federal de Rondônia, Sala 206 - 2C, Porto Velho, Rondônia, Brasil (e-mail: ciro.egoavil@unir.br)

OBJETIVOS

- Estudar a simplificação de circuitos lógicos;
- Reduzir circuitos lógicos combinacionais a configurações mais simples;
- Utilizar *Mapas de Karnaugh* para simplificar os circuitos;

I. INTRODUÇÃO TEÓRICA

A simplificação de circuitos lógicos combinacionais pode se tornar intratável caso não se utilize ferramentas destinadas a tornar o processo de simplificação mais eficiente. As operações Booleanas fundamentam toda e qualquer operação destinada a simplificação destes circuitos [1].

Dentre as inúmeras formas de se realizar a simplificação de circuitos que realizam operações lógicas pode-se destacar as seguintes: *Soma de produtos canônicos* e os *Mapas de Karnaugh*. Que, em essência, são operações análogas, contudo, utilizando a segunda operação torna-se ainda mais simples simplificar tais circuitos, desde que hajam no máximo quatro variáveis de entrada. Para circuitos lógicos com cinco ou mais variáveis, utilizar *mapas de Karnaugh* para simplificá-los se torna um processo um tanto trabalhoso [1].

A. RESUMO SOBRE ÁLGEBRA DE BOOLE

George Boole, matemático e filósofo inglês desenvolveu a chamada lógica Booleana ainda no século XIX, porém apenas na década de 1930 um estudante de graduação da universidade de Michigan, Claude Shannon, fundou a lógica digital ao afirmar em sua dissertação de mestrado que era possível implementar a lógica booleana em componentes eletrônicos hoje chamados de portas lógicas. Com isto, Shannon abriu caminho para a revolução digital que possibilitou o advento de máquinas como o computador [2].

As propriedades mais importantes da Álgebra Booleana estão listadas abaixo. Todas elas podem ser escritas em termos de 3 operações lógicas básicas *NOT*, *OR* e *AND*:

- $A + B = B + A$ (Comutativa)
- $A.B = B.A$ (Comutativa)
- $A + (B + C) = (A + B) + C = A + B + C$ (Associativa)
- $A.(B.C) = (A.B).C = A.B.C$ (Associativa)
- $A.(B + C) = A.B + A.C$ (Distributiva)
- $(A + B)(C + D) = A.C + A.D + B.C + B.D$ (Distributiva)
- $A + AB = A$ (Absorção)
- $A + \bar{A}.B = A + B$ (Absorção)

Utilizando estas propriedades da *Álgebra de Boole* é possível estudar o primeiro processo de simplificação de circuitos lógicos combinacionais a *Soma de produtos Canônicos*.

B. SOMA DE PRODUTOS CANÔNICOS

Utilizar a soma de produtos canônicos para simplificar circuitos lógicos consiste em observar a tabela verdade da operação deste circuito e, escolhendo um valor lógico como referência, realizar operações lógicas Booleanas validas para simplificar uma determinada expressão [3].

1) Exemplo 1

Tabela 1. Tabela verdade para um circuito qualquer:

A	B	S
0	0	0
0	1	1
1	0	1
1	1	0

Para fazer a soma de produtos canônicos da tabela verdade (1) deve-se, primeiro, escolher como referência um valor lógico na saída, escolhendo o valor lógico 1, o passo seguinte é observar em que estado encontram-se as duas variáveis de entrada para todas as combinações de entrada que fornecem o valor lógico 1. Procedendo desta forma, obtém-se a equação (1):

$$S = \bar{A}.B + A\bar{B} = A \oplus B \quad (1)$$

É válido observar que pode-se proceder da mesma forma, porém escolhendo como referência os valores lógicos 0. Entretanto, caso seja mais eficiente proceder desta forma, há uma diferença quanto a equação que se obtém. Na verdade, esta operação é melhor definida como sendo o *produto das somas canônicas*. Na equação (2), pode-se observar o produto das somas canônicas da tabela (1). Ou seja, tomando como referência as saídas que apresentam valor lógico 0:

$$S = (\bar{A} + \bar{B}).(A + B) = A \oplus B \quad (2)$$

É claro que operando a equação (2), de acordo com as propriedades bolleanas, obtém-se novamente a equação (1).

Apesar de ser simples simplificar expressões booleanas utilizando os produtos canônicos, para certos circuitos, torna-se demasiado complexo proceder desta forma, por este motivo há de se definir os *Mapas de karnaugh*.

C. MAPAS DE KARNAUGH

A simplificação de circuitos lógicos combinacionais através de mapas de Karnaugh é um dos tópicos mais importantes em qualquer curso de Eletrônica Digital devido a eficiência envolvida na utilização desta ferramenta para a resolução de circuitos envolvendo portas lógicas. A Álgebra de Boole é o conjunto de operações que fundamentam esta operação em conjunto com o conhecimento das operações com produtos canônicos [4].

Antes de entrar nos conceitos e na resolução de mapas de karnaugh efetivamente, é necessário definir dois conceitos:

1) Mintermos

Os *Mintermos* são definidos com base na soma dos produtos canônicos, de acordo com a tabela (2) que tem uma saída arbitrária:

Tabela 2. Mintermos

A	B	C	Mintermo	Saída
0	0	0	$\bar{A}.\bar{B}.\bar{C}$	0
0	0	1	$\bar{A}.\bar{B}.C$	1
0	1	0	$\bar{A}.B.\bar{C}$	0
0	1	1	$\bar{A}.B.C$	0
1	0	0	$A.\bar{B}.\bar{C}$	1
1	0	1	$A.\bar{B}.C$	1
1	1	0	$A.B.\bar{C}$	1
1	1	1	$A.B.C$	1

Quando se deseja operar os *Mintermos* da tabela verdade de um circuito lógico, na verdade, está se definindo os

valores lógicos 1 na entrada como referência afim de se operar as combinações que fornecem este valor lógico na saída fazendo a soma dos seus produtos canônicos.

Há casos onde são dadas as posições que dos *Mintermos* ocupam no *Mapa de karnaugh*, nestes casos, as posições dos valores lógicos 1 são dadas como na equação (3):

$$\sum(1, 4, 5, 6, 7) \quad (3)$$

2) Maxtermos

Os *Maxtermos* são definidos com base no produto das somas canônicas, de acordo com a tabela (3), que tem saída igual a da tabela (2):

Tabela 3. Maxtermos

A	B	C	Maxtermo	Saída
0	0	0	$A + B + C$	0
0	0	1	$A + B + \bar{C}$	1
0	1	0	$A + \bar{B} + C$	0
0	1	1	$A + \bar{B} + \bar{C}$	0
1	0	0	$\bar{A} + B + C$	1
1	0	1	$\bar{A} + B + \bar{C}$	1
1	1	0	$\bar{A} + \bar{B} + C$	1
1	1	1	$\bar{A} + \bar{B} + \bar{C}$	1

Ou seja, os *Maxtermos* indicam as posições que os valores lógicos 0 ocupam no mapa de karnaugh. Quando forem fornecidos os *Maxterms* para a resolução de um problema, estes serão fornecidos como na equação (4):

$$\Pi(0, 2, 3) \quad (4)$$

Conhecido o fato de que os *Mintermos* estão associados a soma dos produtos canônicos e as posições dos valores lógicos 1 no mapa de Karnaugh, e os *Maxtermos* estão associados aos produtos das somas canônicas e as posições dos valores lógicos 0 no mapa de Karnaugh, é possível definí-los.

O mapa de Karnaugh consiste em dispor os *Mintermos* ou *Maxtermos* de forma a possibilitar uma manipulação algébrica das variáveis com mais facilidade do que se obteria se apenas se utilizassem os produtos canônicos.

Na figura (1) é possível observar o primeiro exemplo de *mapa de Karnaugh* para 3 variáveis, as posições dos *mintermos* e dos *maxtermos* são dadas pelas equações (3) e (4):

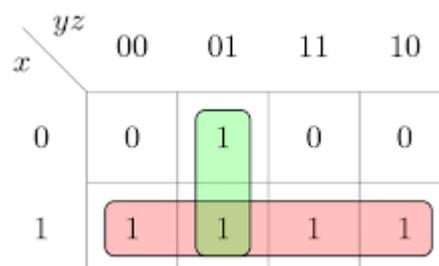


Figura 1. Mapa de karnaugh para 3 variáveis

Para compreender o procedimento envolvido na obtenção do *mapa de Karnaugh* da figura (1) deve-se primeiro compreender a relação existente entre a tabela verdade de um circuito lógico arbitrário e o *mapa de karnaugh*, esta relação pode ser observada nos *bits* binários que aparecem ao lado das linhas e colunas do *mapa de Karnaugh*, ou seja, considerando o número binário $S_2=100$ deve-se observar o valor lógico correspondente no *mapa de Karnaugh* a este número. Neste caso:

$$S = 100 = 1$$

Procedendo desta maneira, é possível obter uma tabela verdade correspondente a este *mapa de Karnaugh*:

Tabela 4. Tabela verdade para o mapa de Karnaugh da figura (1)

x	y	z	S
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

O procedimento para obter a expressão booleana do *mapa de Karnaugh* da figura (1), bem como da tabela verdade (2), (3) e (4) pode ser feito através dos procedimentos envolvendo os produtos ou somas canônicas descritos acima. Porém, como foi mencionado, o mapa de Karnaugh fornece um procedimento que simplifica as operações que teriam de ser realizadas caso não fosse possível utilizá-lo.

Para obter a expressão de saída do mapa de Karnaugh deve-se primeiro agrupar os *minterms* ou *maxterms* respeitando a seguinte condição: O número n , de termos agrupados deverá ser sempre uma potência de 2, ou seja:

$$n = 2^x$$

No *mapa de Karnaugh* da figura (1), os dois grupos formados estão representados e coloridos de modo a facilitar a visualização.

Após agrupados os termos, observa-se a influência das variáveis de entrada em cada um dos agrupamentos de modo a obter a expressão de saída, fazendo:

$$S = x \cdot \bar{y} \cdot \bar{z} \cdot z + \bar{x} \cdot x \cdot \bar{y} \cdot z \quad (5)$$

Simplificando a equação (5), obtém-se:

$$S = x + \bar{y} \cdot z$$

Que é a expressão lógica mínima para obtida através do *mapa de Karnaugh* da figura (1).

Assim como se utiliza os *mapas de Karnaugh* para dispor de uma melhor forma as saídas de um circuito lógico com duas variáveis de entrada, é possível estender este processo para um número maior de variáveis, no entanto torna-se demasiado complexo utilizar esta ferramenta para modelar

o comportamento de um circuito lógico com mais de cinco variáveis.

Na figura (2) é possível observar um *mapa de Karnaugh* para um circuito com quatro variáveis de entrada com os devidos agrupamentos realizados:

CD \ AB	00	01	11	10
00	1	0	1	0
01	0	0	0	0
11	0	1	1	1
10	1	1	1	1

Figura 2. Mapa de karnaugh para 4 variáveis

D. SIMPLIFICAÇÃO DE CIRCUITOS LÓGICOS UTILIZANDO MAPAS DE KARNAUGH

Simplificar um circuito lógico consiste em encontrar uma forma de implementar um circuito com a mesma saída, porém com complexidade de construção reduzida. As técnicas envolvidas na simplificação de circuitos lógicos combinacionais são, basicamente, as que envolvem as simplificações de expressões de álgebra booleana.

II. MATERIAIS UTILIZADOS

- Fonte de tensão DC
- 1 Circuito integrado 7408 (AND)
- 1 Circuito integrado 7404 (NOT)
- 1 Circuito integrado 7432 (OR)
- 2 Circuitos integrados 7400 (NAND)
- 1 Circuito integrado 7486 (OU EXCLUSIVO)
- Resistores de: 560Ω (4) e 220Ω (1)
- 1 Led

III. PARTE EXPERIMENTAL

A. IMPLEMENTAÇÃO DE CIRCUITOS LÓGICOS COMBINACIONAIS A PARTIR DA TABELA VERDADE:

Implementar um circuito lógico combinacional que apresente a saída correspondente àquela fornecida pela tabela (5):

Tabela 5. Tabela verdade a ser implementada

A	B	C	SAÍDA
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Para implementar o circuito correspondente represente a tabela verdade (5) em um *mapa de Karnaugh* de modo a obter a expressão booleana mínima e, desta forma, implementar o circuito com o menor número de portas lógicas para obter a saída correspondente a tabela (5).

B. UNIVERSALIDADE DAS PORTAS NAND

Para verificar as características universais da operação lógica *NAND*, deverá ser implementado um circuito que apresente saída correspondente a da tabela (5) apenas com portas lógicas *NAND*. Caso não haja disponibilidade de portas lógicas *NAND*, é possível obter esta configuração através da combinação de portas *AND* com portas *NOT*.

C. SIMPLIFICAÇÃO DE UM CIRCUITO LÓGICO COMBINACIONAL

Implemente o circuito da figura (3), obtenha a tabela verdade da operação deste circuito. Utilize o *mapa de Karnaugh* para obter a expressão mínima deste circuito e, consequentemente, o circuito mais simples que pode ser obtido de modo a obter a mesma saída:

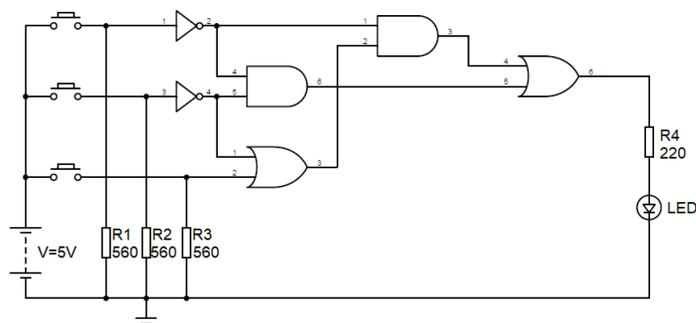


Figura 3.

D. MAPA DE KARNAUGH PARA QUATRO VARIÁVEIS

Implementar o circuito da figura (4) que apresenta certa complexidade de construção e obter a tabela verdade do seu funcionamento:

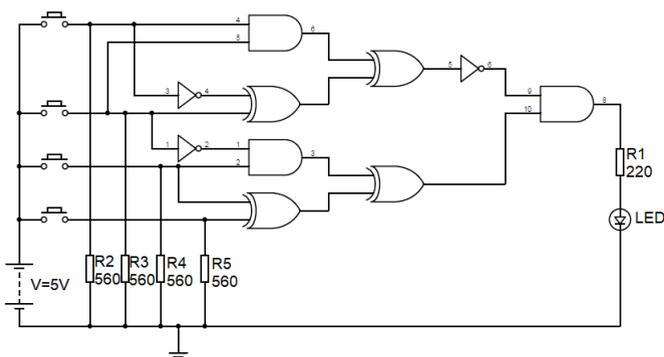


Figura 4.

Após verificado o funcionamento do circuito, represente a tabela verdade em um *mapa de Karnaugh* para quatro

variáveis de modo a verificar a possibilidade de se obter um circuito mais simples que forneça a mesma saída. Os circuitos mínimos implementados nos passos (A), (B), (C) e (D) deverão ser demonstrados no relatório.

Deverão ser apresentadas as simulações de cada circuito realizado experimentalmente em softwares como o Proteus PSIM ou Circuit Maker.

Referências

- [1] TOCCI, Ronald J, *Sistemas Digitais*, Pearson 11° Ed, Brasil, 2011.
- [2] CHRISTIAN, B. *O Humano Mais Humano: O que a inteligência artificial nos ensina sobre a vida*, Brasil, 2013.
- [3] SILVEIRA, D. *Circuitos Lógicos: Simplificação de circuitos lógicos com Mapa de Karnaugh*, Universidade Federal de Juiz de Fora, Brasil, 2015.
- [4] CHEUNG P. *Logic Simplification and Karnaugh Map*, Imperial College London, Londres, 2007.

...