Lógica Digital

Aula-02: Mapas de Karnaugh

Eliseu César Miguel

Departamento de Ciência da Computação
Universidade Federal de Alfenas

August 2, 2022

Organização da Aula

1. Introdução

Organização da Aula

Organização da Aula

1. Introdução
2. Conceitos Básicos
3. Representação de expressões

Organização da Aula

Introdução

Considerações Preliminares

Este material não pretende ser completo quanto à amplitude do assunto. Aqui pretende-se apenas organizar os pontos relevantes para as aplicações dos conceitos da Lógica de Boole na disciplina de Lógica Digital, gerando um guia de estudos. Destarte, sempre consulte livros e apostilas para alcançar bons resultados em seus estudos.

Também, este material não é, em sua totalidade, de minha autoria. Ao contrário, ele contempla conteúdos de sítios de Internet e conteúdos de livros. Para tanto, cito bibliografias de textos aqui encorporados.

Boa leitura!

Sobre Maurice Karnaugh

Maurice Karnaugh (Nova Iorque, 4 de outubro de 1924) é um físico, cientista da computação e engenheiro de telecomunicações norte-americano.

Estudou física e matemática no *City College of New York*, de 1944 a 1948, quando foi transferido para a Universidade de *Yale*, onde obteve pós-graduação em 1949 e doutorado em física em 1952. É atualmente governador emérito do ICCC. Trabalhou como pesquisador no *Bell Labs*, de 1952 a 1966, e no centro de pesquisa da IBM, de 1966 a 1993. Lecionou ciência da computação no Instituto Politécnico da Universidade de Nova Iorque, de 1980 a 1999, e desde 1975 é membro do IEEE, por suas contribuições sobre o uso de métodos numéricos em telecomunicações.

Sua criação que o tornou famoso foi o mapa de Karnaugh, amplamente utilizado em álgebra booleana.

https://pt.wikipedia.org/wiki/Maurice_Karnaugh

Introdução: Mapas de Karnaugh

Objetivos dos Mapas de Karnaugh

Os mapas de Karnaugh, assim como os outros elementos estudados, servem para representar expressões lógicas.

Mas, muito além disso, é uma estratégia elaborada para a simplificação de expressões lógicas.

Inicialmente, faremos a representação e, em seguida, as simplificações.

Representação de expressões lógicas por Karnaugh

Basicamente, os mapas de Karnaugh são uma reorganização na representação de expressões lógicas canônicas.

Uma observação importante é que dois termos canônicos são simplificados quando há variação de apenas uma variável.

Sejam: a expressão $f_{(a,b)} \equiv 1$ e t, o conjunto de todos os termos de $f_{(a,b)}$.
Então, a sequência s de t, tal que

$$s = [\overline{a}.\overline{b}, \overline{a}.b, a.b, a.\overline{b}] \equiv [00, 01, 11, 10]$$

tem a propriedade de variação de apenas uma variável $\forall s_{(i)}$ e $s_{(i+1)}$.

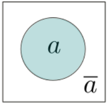
em que i é calculado como: $\forall n \in \mathbb{N},\ i = (n\ mod\ |t|)$

<u>Neste caso</u> se $s_{(i)} = 10 \rightarrow s_{(i+1)} = 00$, ou seja, se i=3, $s_{(i+1)} = s_{(0)}$

Karnaugh aproveitou esta observação para criar *vizinhança* entre termos que podem ser simplificados.

Representação: Expressões a uma variável

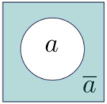
a	X
0	0
1	1



a	0	1
		1

$\equiv X$

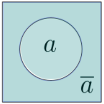
a	Y
0	1
1	0



a	0	1
	1	

$\equiv Y$

a	K
0	1
1	1

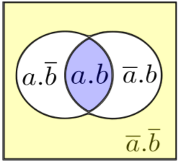


a	0	1
	1	1

$\equiv K$

Representação: Expressões a duas variável

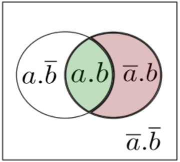
a	b	Z
0	0	1
0	1	0
1	0	0
1	1	1



$b \backslash a$	0	1
0	1	
1		1

$\equiv \quad Z$

a	b	M
0	0	0
0	1	1
1	0	0
1	1	1

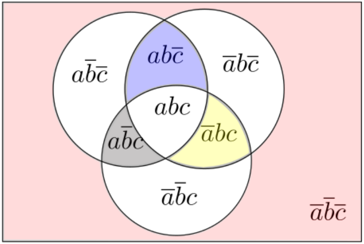


$b \backslash a$	0	1
0		
1	1	1

$\equiv \quad M$

Representação: Expressões a três variáveis

a	b	c	W
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0



$bc \backslash a$	0	1
00	1	
01		1
11	1	
10		1

$\equiv W \equiv$

$c \backslash ab$	00	01	11	10
0	1		1	
1		1		1

Representação: Expressões a quatro variáveis

a	b	c	d	V
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
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

$cd \backslash ab$	00	01	11	10
00	1			1
01		1	1	
11	1			1
10		1	1	

$\equiv V$

Exercícios: Represente em mapas de Karnaugh

1. $A_{(x,y,z,k)} = \{0000, 0001, 0011, 1001, 1011, 1111\}$
2. $H_{(a,b,c,d)} = \overline{a.b} + \overline{b.c} \to \overline{a}.b + c.\overline{d}$ (nota: $X \to Y \equiv \overline{X} + Y$)
3. $$Z_{(a,d,m,k)} \equiv \begin{cases} 0 & \text{se} \quad a \equiv d \text{ e } k \equiv 0 \\ 1 & \text{caso} \quad \text{contrário} \end{cases}$$

	00	01	11	10
00				
01				
11				
10				

$\equiv$

	00	01	11	10
00				
01				
11				
10				

$\equiv$

	00	01	11	10
00				
01				
11				
10				

$\equiv$

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	C_1	C_0
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	C_1	C_0
0	0	0		
0	0	1	0	
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	C_1	C_0
0	0	0		
0	0	1	0	1
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	C_1	C_0
0	0	0		
0	0	1	0	1
0	1	0	1	
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	C_1	C_0
0	0	0		
0	0	1	0	1
0	1	0	1	1
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	C_1	C_0
0	0	0		
0	0	1	0	1
0	1	0	1	1
0	1	1		
1	0	0	1	
1	0	1		
1	1	0		
1	1	1		

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	C_1	C_0
0	0	0		
0	0	1	0	1
0	1	0	1	1
0	1	1		
1	0	0	1	0
1	0	1		
1	1	0		
1	1	1		

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	C_1	C_0
0	0	0	×	×
0	0	1	0	1
0	1	0	1	1
0	1	1	×	×
1	0	0	1	0
1	0	1	×	×
1	1	0	×	×
1	1	1	×	×

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	C_1	C_0
0	0	0	×	×
0	0	1	0	1
0	1	0	1	1
0	1	1	×	×
1	0	0	1	0
1	0	1	×	×
1	1	0	×	×
1	1	1	×	×

$C_1 \equiv \overline{B_3}.B_2.\overline{B_1} + B_3.\overline{B_2}.\overline{B_1}$ e $C_0 \equiv \overline{B_3}.\overline{B_2}.B_1 + \overline{B_3}.B_2.\overline{B_1}$

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	C_1	C_0
0	0	0	×	×
0	0	1	0	1
0	1	0	1	1
0	1	1	×	×
1	0	0	1	0
1	0	1	×	×
1	1	0	×	×
1	1	1	×	×

$C_1 \equiv \overline{B_3}.B_2.\overline{B_1} + B_3.\overline{B_2}.\overline{B_1}$ e $C_0 \equiv \overline{B_3}.\overline{B_2}.B_1 + \overline{B_3}.B_2.\overline{B_1}$

Veremos que: $C_1 \equiv \overline{B_1}$ e $C_0 \equiv \overline{B_3}$

Simplificação por Karnaugh

Para fazer as simplificações, Karnaugh agrupa termos vinzinhos. A forma em que a vizinhança foi formada garante que tais termos ofereçam simplificação entre si.

Técnicas para formar grupos

- Os grupos são formados entre termos da expressão vizinhos;

Regras de simplificação

Simplificação por Karnaugh

Para fazer as simplificações, Karnaugh agrupa termos vinzinhos. A forma em que a vizinhança foi formada garante que tais termos ofereçam simplificação entre si.

Técnicas para formar grupos

- Os grupos são formados entre termos da expressão vizinhos;
- Termos são vizinhos entre colunas ou linhas, mas não na diagonal do mapa;

Regras de simplificação

Simplificação por Karnaugh

Para fazer as simplificações, Karnaugh agrupa termos vinzinhos. A forma em que a vizinhança foi formada garante que tais termos ofereçam simplificação entre si.

Técnicas para formar grupos

- Os grupos são formados entre termos da expressão vizinhos;
- Termos são vizinhos entre colunas ou linhas, mas não na diagonal do mapa;
- A quantidade de termos em cada grupo deve ser $z \mid z = 2^x$, $x \geq 0$ e $x \in \mathbb{N}$.

Regras de simplificação

Simplificação por Karnaugh

Para fazer as simplificações, Karnaugh agrupa termos vinzinhos. A forma em que a vizinhança foi formada garante que tais termos ofereçam simplificação entre si.

Técnicas para formar grupos

- Os grupos são formados entre termos da expressão vizinhos;
- Termos são vizinhos entre colunas ou linhas, mas não na diagonal do mapa;
- A quantidade de termos em cada grupo deve ser $z \mid z = 2^x$, $x \geq 0$ e $x \in \mathbb{N}$.

Regras de simplificação

- Cada grupo gera um termo simplificado;

Simplificação por Karnaugh

Para fazer as simplificações, Karnaugh agrupa termos vinzinhos. A forma em que a vizinhança foi formada garante que tais termos ofereçam simplificação entre si.

Técnicas para formar grupos

- Os grupos são formados entre termos da expressão vizinhos;
- Termos são vizinhos entre colunas ou linhas, mas não na diagonal do mapa;
- A quantidade de termos em cada grupo deve ser $z \mid z = 2^x$, $x \geq 0$ e $x \in \mathbb{N}$.

Regras de simplificação

- Cada grupo gera um termo simplificado;
- Toda variável que tem valores 0 e 1 dentro do grupo são elimiadas;

Simplificação por Karnaugh

Para fazer as simplificações, Karnaugh agrupa termos vinzinhos. A forma em que a vizinhança foi formada garante que tais termos ofereçam simplificação entre si.

Técnicas para formar grupos

- Os grupos são formados entre termos da expressão vizinhos;
- Termos são vizinhos entre colunas ou linhas, mas não na diagonal do mapa;
- A quantidade de termos em cada grupo deve ser $z \mid z = 2^x,\ x \geq 0$ e $x \in \mathbb{N}$.

Regras de simplificação

- Cada grupo gera um termo simplificado;
- Toda variável que tem valores 0 e 1 dentro do grupo são elimiadas;
- A quantidade de variáveis eliminadas por grupo é $x \mid x = \log_2 z,\ x \in \mathbb{N}$;

Simplificação por Karnaugh

Para fazer as simplificações, Karnaugh agrupa termos vinzinhos. A forma em que a vizinhança foi formada garante que tais termos ofereçam simplificação entre si.

Técnicas para formar grupos

- Os grupos são formados entre termos da expressão vizinhos;
- Termos são vizinhos entre colunas ou linhas, mas não na diagonal do mapa;
- A quantidade de termos em cada grupo deve ser $z \mid z = 2^x$, $x \geq 0$ e $x \in \mathbb{N}$.

Regras de simplificação

- Cada grupo gera um termo simplificado;
- Toda variável que tem valores 0 e 1 dentro do grupo são elimiadas;
- A quantidade de variáveis eliminadas por grupo é $x \mid x = \log_2 z$, $x \in \mathbb{N}$;
- Quanto maiores os grupos, mais variáveis serão simplificadas;

Simplificação por Karnaugh

Para fazer as simplificações, Karnaugh agrupa termos vinzinhos. A forma em que a vizinhança foi formada garante que tais termos ofereçam simplificação entre si.

Técnicas para formar grupos

- Os grupos são formados entre termos da expressão vizinhos;
- Termos são vizinhos entre colunas ou linhas, mas não na diagonal do mapa;
- A quantidade de termos em cada grupo deve ser $z \mid z = 2^x$, $x \geq 0$ e $x \in \mathbb{N}$.

Regras de simplificação

- Cada grupo gera um termo simplificado;
- Toda variável que tem valores 0 e 1 dentro do grupo são elimiadas;
- A quantidade de variáveis eliminadas por grupo é $x \mid x = \log_2 z$, $x \in \mathbb{N}$;
- Quanto maiores os grupos, mais variáveis serão simplificadas;
- Quanto menos grupos, mais simplificada é a expressão final.

Simplificação por Karnaugh

Para fazer as simplificações, Karnaugh agrupa termos vinzinhos. A forma em que a vizinhança foi formada garante que tais termos ofereçam simplificação entre si.

Técnicas para formar grupos

- Os grupos são formados entre termos da expressão vizinhos;
- Termos são vizinhos entre colunas ou linhas, mas não na diagonal do mapa;
- A quantidade de termos em cada grupo deve ser $z \mid z = 2^x$, $x \geq 0$ e $x \in \mathbb{N}$.

Regras de simplificação

- Cada grupo gera um termo simplificado;
- Toda variável que tem valores 0 e 1 dentro do grupo são elimiadas;
- A quantidade de variáveis eliminadas por grupo é $x \mid x = \log_2 z$, $x \in \mathbb{N}$;
- Quanto maiores os grupos, mais variáveis serão simplificadas;
- Quanto menos grupos, mais simplificada é a expressão final.

Simplificação por Karnaugh

Para fazer as simplificações, Karnaugh agrupa termos vinzinhos. A forma em que a vizinhança foi formada garante que tais termos ofereçam simplificação entre si.

Técnicas para formar grupos

- Os grupos são formados entre termos da expressão vizinhos;
- Termos são vizinhos entre colunas ou linhas, mas não na diagonal do mapa;
- A quantidade de termos em cada grupo deve ser $z \mid z = 2^x$, $x \geq 0$ e $x \in \mathbb{N}$.

Regras de simplificação

- Cada grupo gera um termo simplificado;
- Toda variável que tem valores 0 e 1 dentro do grupo são elimiadas;
- A quantidade de variáveis eliminadas por grupo é $x \mid x = \log_2 z$, $x \in \mathbb{N}$;
- Quanto maiores os grupos, mais variáveis serão simplificadas;
- Quanto menos grupos, mais simplificada é a expressão final.

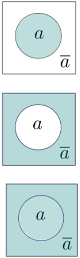
Sugestão: Não finalize a simplificação sem rever os grupos formados!
Muitos grupos são extintos, se todos os seus termos já participam de outros grupos.

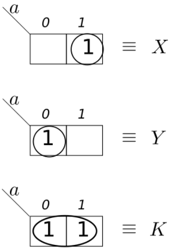
Exemplo: Simplificação por Karnaugh

a	X
0	0
1	1

a	Y
0	1
1	0

a	K
0	1
1	1





1. Expressão $X \equiv a$
   - 1 grupo com 1 termo
   - eliminamos 0 variáveis ($1 = 2^0$)
   - nenhuma variável troca de valor.
2. $Y \equiv \overline{a}$
   - 1 grupo com 1 termo
   - eliminamos 0 variáveis ($1 = 2^0$)
   - nenhuma variável troca de valor.
3. $K \equiv 1$
   - 1 grupo com 2 termos
   - eliminamos 1 variáveis ($2 = 2^1$)
   - a troca de valor

Exemplo: Simplificação por Karnaugh

$W \equiv$

$c \backslash ab$	00	01	11	10
0		1	1	
1			1	1

Exemplo: Simplificação por Karnaugh

$W \equiv$

$c \backslash ab$	00	01	11	10
0		1	1	
1			1	1

Regras de simplificação

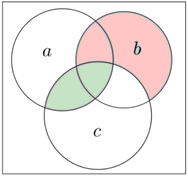
- Cada grupo gera um termo simplificado;
- Toda variável que tem valores 0 e 1 dentro do grupo são elimiadas;
- A quantidade de variáveis eliminadas por grupo é $x \mid x = \log_2 z,\ x \in \mathbb{N}$;
- Quanto maiores os grupos, mais variáveis serão simplificadas;
- Quanto menos grupos, mais simplificada é a expressão final.

Exemplo: Simplificação por Karnaugh

$W \equiv$

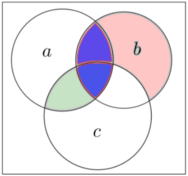
c \ ab	00	01	11	10
0		1	1	
1			1	1

$b.\overline{c}$
$a.c$



c \ ab	00	01	11	10
0		1	1	
1			1	1

$b.\overline{c}$
$a.b$
$a.c$

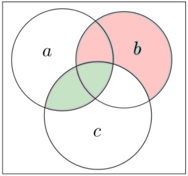


Exemplo: Simplificação por Karnaugh

$W \equiv$

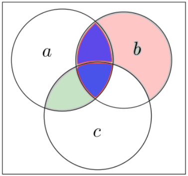
$c \backslash ab$	00	01	11	10
0		1	1	
1			1	1

$b.\overline{c}$
$a.c$



$c \backslash ab$	00	01	11	10
0		1	1	
1			1	1

$b.\overline{c}$
$a.b$
$a.c$



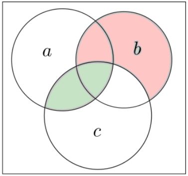
- A simplificação esperada é $W \equiv b.\overline{c} + a.c$.

Exemplo: Simplificação por Karnaugh

$W \equiv$

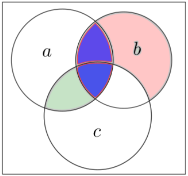
$c \backslash ab$	00	01	11	10
0		1	1	
1			1	1

$b.\overline{c}$
$a.c$



$c \backslash ab$	00	01	11	10
0		1	1	
1			1	1

$b.\overline{c}$
$a.b$
$a.c$



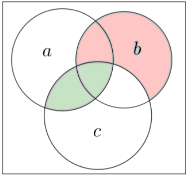
- A simplificação esperada é $W \equiv b.\overline{c} + a.c$.
- O termo $a.b$ sobrepõe-se às áreas do diagrama de Venn já hachuradas.

Exemplo: Simplificação por Karnaugh

$W \equiv$

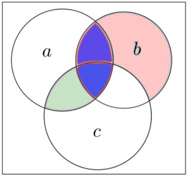
$c \backslash ab$	00	01	11	10
0		1	1	
1			1	1

$b.\overline{c}$
$a.c$



$c \backslash ab$	00	01	11	10
0		1	1	
1			1	1

$b.\overline{c}$
$a.b$
$a.c$



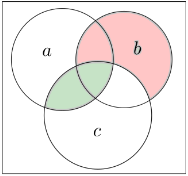
- A simplificação esperada é $W \equiv b.\overline{c} + a.c$.
- O termo $a.b$ sobrepõe-se às áreas do diagrama de Venn já hachuradas.
- Tal termo não altera logicamente W.

Exemplo: Simplificação por Karnaugh

$W \equiv$

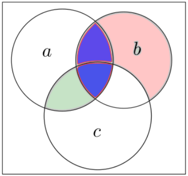
$c \backslash ab$	00	01	11	10
0		1	1	
1			1	1

$b.\overline{c}$
$a.c$



$c \backslash ab$	00	01	11	10
0		1	1	
1			1	1

$b.\overline{c}$
$a.b$
$a.c$



- A simplificação esperada é $W \equiv b.\overline{c} + a.c$.
- O termo $a.b$ sobrepõe-se às áreas do diagrama de Venn já hachuradas.
- Tal termo não altera logicamente W.
- Mas, a expressão $W \equiv b.\overline{c} + a.c + a.b$ pode ser desejada. Por que?

Exemplo: Simplificação por Karnaugh

$c \backslash ab$	00	01	11	10
0		1	1	
1			1	

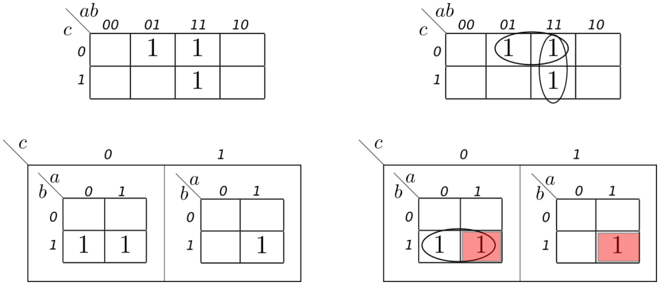
$c = 0$

$b \backslash a$	0	1
0		
1	1	1

$c = 1$

$b \backslash a$	0	1
0		
1		1

Exemplo: Simplificação por Karnaugh



Exemplo: Simplificação por Karnaugh

$cd \backslash ab$	00	01	11	10
00	1			1
01		1	1	
11	1			1
10		1	1	

$\equiv$

$kz \backslash xy$	00	01	11	10
00	1	1	1	
01	1	1		1
11	1	1	1	1
10		1	1	

$\equiv$

$cd \backslash ab$	00	01	11	10
00		1		
01	1			1
11		1	1	1
10		1	1	

$\equiv$

Exemplo: Simplificação por Karnaugh

$cd \backslash ab$	00	01	11	10
00	1			1
01		1	1	
11	1			1
10		1	1	

$\equiv$

$cd \backslash ab$	00	01	11	10
00	1			1
01		1	1	
11	1			1
10		1	1	

$$\equiv \overline{b}.\overline{c}.\overline{d} + b.\overline{c}.d + \overline{b}.c.d + b.c.\overline{d}$$

$kz \backslash xy$	00	01	11	10
00	1	1	1	
01	1	1		1
11	1	1	1	1
10		1	1	

$\equiv$

$cd \backslash ab$	00	01	11	10
00		1		
01	1			1
11		1	1	1
10		1	1	

$\equiv$

Exemplo: Simplificação por Karnaugh

$cd \backslash ab$	00	01	11	10
00	1			1
01		1	1	
11	1			1
10		1	1	

$\equiv$

$cd \backslash ab$	00	01	11	10
00	1			1
01		1	1	
11	1			1
10		1	1	

$$\equiv \bar{b}.\bar{c}.\bar{d} + b.\bar{c}.d + \bar{b}.c.d + b.c.\bar{d}$$

$kz \backslash xy$	00	01	11	10
00	1	1	1	
01	1	1		1
11	1	1	1	1
10		1	1	

$\equiv$

$kz \backslash xy$	00	01	11	10
00	1	1	1	
01	1	1		1
11	1	1	1	1
10		1	1	

$$\equiv y.\bar{z} + \bar{y}.z + \bar{x}.\bar{k} + y.k$$

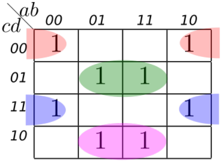
$cd \backslash ab$	00	01	11	10
00		1		
01	1			1
11		1	1	1
10		1	1	

$\equiv$

Exemplo: Simplificação por Karnaugh

cd \ ab	00	01	11	10
00	1			1
01		1	1	
11	1			1
10		1	1	

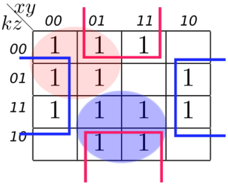
$\equiv$



$$\equiv \overline{b}.\overline{c}.\overline{d} + b.\overline{c}.d + \overline{b}.c.d + b.c.\overline{d}$$

kz \ xy	00	01	11	10
00	1	1	1	
01	1	1		1
11	1	1	1	1
10		1	1	

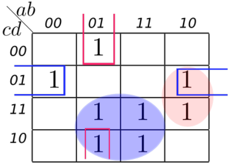
$\equiv$



$$\equiv y.\overline{z} + \overline{y}.z + \overline{x}.\overline{k} + y.k$$

cd \ ab	00	01	11	10
00		1		
01	1			1
11		1	1	1
10		1	1	

$\equiv$



$$\equiv \overline{a}.b.\overline{d} + \overline{b}.\overline{c}.d + b.c + a.\overline{b}.d$$

Simplificação de expressões: Mapas de Karnaugh

Como simplificar expressões quando alguns termos são indefinidas na prática?

Em algumas situações, certas combinações de variáveis (linhas da tabela ou células do mapa de Karnaugh) não são definidas para uma expressão lógica.

Estes termos são usados sempre que podem contribuir para a simplificação.

Termos indefinidos que não melhoram a simplificação são desconsiderados.

No exemplo que segue, todos os termos indefinidos são refenciados por x.

$cd \backslash ab$	00	01	11	10
00	1	1		x
01	1		x	1
11		x	x	x
10	x	x	x	x

$\equiv$

Simplificação de expressões: Mapas de Karnaugh

Como simplificar expressões quando alguns termos são indefinidas na prática?

Em algumas situações, certas combinações de variáveis (linhas da tabela ou células do mapa de Karnaugh) não são definidas para uma expressão lógica.

Estes termos são usados sempre que podem contribuir para a simplificação.

Termos indefinidos que não melhoram a simplificação são desconsiderados.

No exemplo que segue, todos os termos indefinidos são refenciados por x.

$cd \backslash ab$	00	01	11	10
00	1	1		x
01	1		x	1
11		x	x	x
10	x	x	x	x

$\equiv$

$cd \backslash ab$	00	01	11	10
00	1	1		x
01	1		x	1
11		x	x	x
10	x	x	x	x

$\equiv \overline{a}.\overline{d} + \overline{b}.\overline{c}$

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	C_1	C_0
0	0	0	x	x
0	0	1	0	1
0	1	0	1	1
0	1	1	x	x
1	0	0	1	0
1	0	1	x	x
1	1	0	x	x
1	1	1	x	x

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	C_1	C_0
0	0	0	x	x
0	0	1	0	1
0	1	0	1	1
0	1	1	x	x
1	0	0	1	0
1	0	1	x	x
1	1	0	x	x
1	1	1	x	x

B_1 \ $B_3.B_2$	00	01	11	10
0	x	1	x	1
1		x	x	x

$C_1 \equiv \overline{B_1}$

B_1 \ $B_3.B_2$	00	01	11	10
0	x	1	x	
1	1	x	x	x

$C_0 \equiv \overline{B_3}$

Exercícios: Represente em mapas de Karnaugh

Exercício de elaboração de circuito.

Um sistema de experimento com degustação oferece três produtos, P_1, P_2 e P_3, a serem experimentados pelo cliente. O cliente, após exprimentar os produtos, deverá informar qual produto mais o agradou. Para isso, o sistema oferece três botões, B_1, B_2 e B_3, referentes aos produtos P_i, $1 \leq i \leq 3$, respectivamente. A funcionalidade é simples: Caso o cliente tenha gostado de P_1, ele aperta o botão B_1, e o sistema armazena o código 01. Caso ele tenha gostado de P_2, ele aperta B_2 e o sistema armazena o código 11. Finalmente, caso ele tenha gostado de P_3, ele aperta o botão B_3 e o sistema armazena o código 10. Sabendo-se disso, faça o circuito lógico para solucionar esse problema

B_3	B_2	B_1	C_1	C_0
0	0	0	x	x
0	0	1	0	1
0	1	0	1	1
0	1	1	x	x
1	0	0	1	0
1	0	1	x	x
1	1	0	x	x
1	1	1	x	x

B_1 \ $B_3.B_2$	00	01	11	10
0	x	1	x	1
1		x	x	x

$$C_1 \equiv \overline{B_1}$$

B_1 \ $B_3.B_2$	00	01	11	10
0	x	1	x	
1	1	x	x	x

$$C_0 \equiv \overline{B_3}$$

Como chegar nestes resultados partindo de $\overline{C_1}$ e $\overline{C_0}$?

Simplificação de expressões: Mapas de Karnaugh a 5 variáveis

$e = 0$

cd \ ab	00	01	11	10
00				1
01		1	1	
11	1			1
10		1	1	

$e = 1$

cd \ ab	00	01	11	10
00	1	1	1	
01	1	1		1
11	1	1		1
10		1	1	

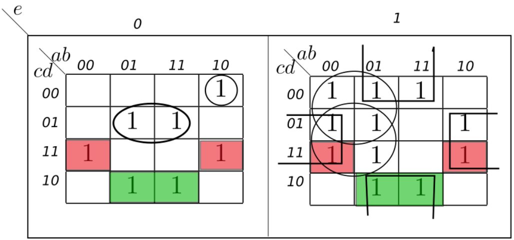
Simplificação de expressões: Mapas de Karnaugh a 5 variáveis

e = 0

cd \ ab	00	01	11	10
00				1
01		1	1	
11	1			1
10		1	1	

e = 1

cd \ ab	00	01	11	10
00	1	1	1	
01	1	1		1
11	1	1		1
10		1	1	



Por que equivalência e não igualdade?

$cd \backslash ab$	00	01	11	10
00	1	1		1
01		1		
11		1	1	
10			1	1

$\equiv$

Por que equivalência e não igualdade?

$cd \backslash ab$	00	01	11	10
00	1	1		1
01		1		
11		1	1	
10			1	1

$\equiv$

$cd \backslash ab$	00	01	11	10
00	1	1		1
01		1		
11		1	1	
10			1	1

$\equiv$

Por que equivalência e não igualdade?

cd \ ab	00	01	11	10
00	1	1		1
01		1		
11		1	1	
10			1	1

$\equiv$

cd \ ab	00	01	11	10
00	1	1		1
01		1		
11		1	1	
10			1	1

$\equiv$

cd \ ab	00	01	11	10
00	1	1		1
01		1		
11		1	1	
10			1	1

$\equiv$

Exercícios: Simplificação

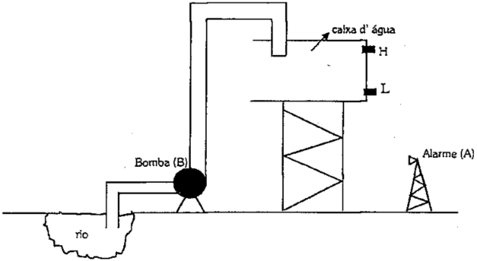
Uma estrutura recebe um código KPX e precisa ser convertida para MTV, como segue na tabela. Apresente o conversor necessário usando portas lógicas.

KPX				MTV		
0	0	0	0	1	1	0
0	0	0	1	1	0	0
0	0	1	0	0	1	1
0	0	1	1	1	0	1
0	1	0	0	1	1	0
0	1	0	1	0	0	0
0	1	1	0	0	1	0
0	1	1	1	1	1	1
1	0	0	0	0	1	0
1	0	0	1	1	1	0

Exercícios: Projeto

Exemplo de Aplicação: Controle de Bombeamento de Água

O desenho a seguir mostra um processo simples para encher uma caixa d'água a partir do bombeamento da água de um rio próximo.



Os sensores de nível alto (H) e de nível baixo (L) são utilizados para determinar o acionamento da bomba (B) e do alarme (A). Os sensores funcionam da seguinte forma:

$H = L = 0 \longrightarrow$ sensor desacionado, ou seja, a água está abaixo dele.

$H = L = 1 \longrightarrow$ sensor acionado, ou seja, a água está sobre ou acima dele.

↓

Simplificação por Mapas de Karnaugh

Exercícios

Faça os exercícios da lista relativos ao assunto desta aula

Agradecimentos Especiais

Agradeço especialmente a *Till Tantau* por ter escrito o *Beamer* para LaTeX e que, consequentemente, possibilitou a escrita desta aula.

Unifal-MG Universidade Federal de Alfenas