

ARQ1 _ Aula_06

Tema: Introdução à linguagem Verilog e simulação em Logisim

Simplificações de expressões lógicas e circuitos

Expressões lógicas e seus circuitos equivalentes podem ser, muitas vezes, simplificados. Entretanto, nem todas as expressões podem ser minimizadas e, caso o forem, nem sempre haverá uma única expressão mínima global.

Dentre os métodos para simplificações destacam-se:

- as aplicações de regras da álgebra de Boole, principalmente, as que envolvem tautologias, contradições, negações, absorções e distribuições.
- métodos tabulares:
 - 1.) baseados em uma diferença (distância de Hamming = 1) ou $(x.y + x.y')$
 - mapa de Veitch-Karnaugh (poucas variáveis)
 - Quine-McCluskey (muitas variáveis)
 - 2.) baseados em diferenças (distância de Hamming = 2) ou $(x.y + x'.y')$ ou $(x'.y + x.y')$
 - Reed-Müller
 - 3.) inspeção visual da tabela e interpretação algorítmica
- métodos envolvendo representações por grafos (BDD's) ou redes de conexões.

Exemplo:

Dada a expressão lógica $(x+y'+z).(x+y+z)$ é possível encontrar uma simplificação mediante aplicações de regras da álgebra de Boole.

- por distribuições:

$$\begin{aligned} & (x+y'+z).x+(x+y'+z).y+(x+y'+z).z \\ &= (x.x+y'.x+z.x)+(x.y+y'.y+z.y)+(x.z+y'.z+z.z) \end{aligned}$$

- por identidades/contradições:

$$= (x.1+x.y'+x.z)+(x.y+0+y.z)+(x.z+y'.z+z)$$

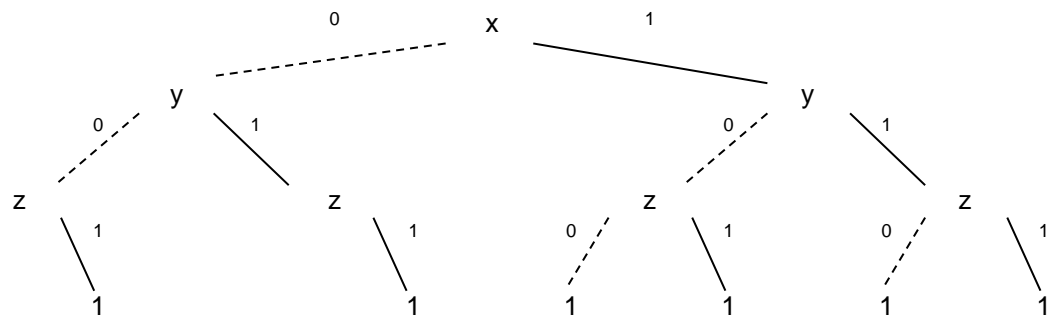
- por distribuições/identidades

$$= x.(1+y'+z+y+z)]+z.(y+y'+1) = x.1 + z.1 = x+z$$

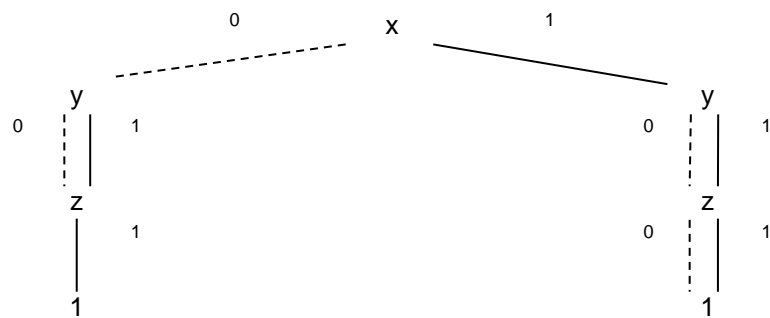
Resultado semelhante ao se considerar a diferença unitária, como nos métodos tabulares:

$$(x+y'+z).(x+y+z) = [(x+z)+y'] . [(x+z)+y] = (x+z).(y'+y) = (x+z).1 = x+z$$

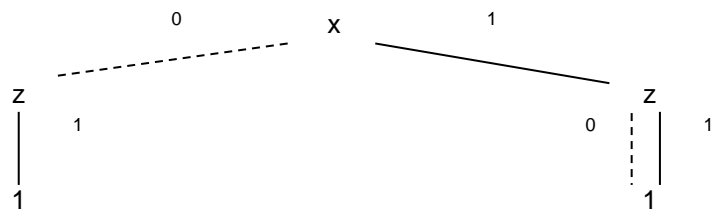
Se houver uma representação por grafo (BDD) também haverá simplificação:



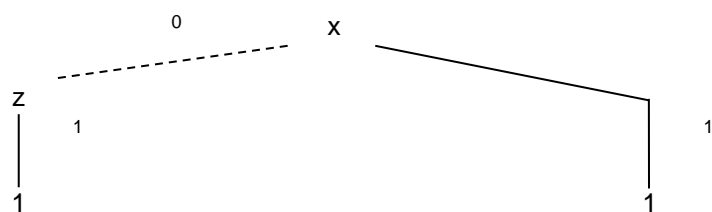
Ao se reagrupar, nota-se que não há dependência de y, pois qualquer caminho leva a z:



Ao se reorganizar o grafo, considerando as importâncias de x e de z:



E, ao considerar que se houver x não haverá dependência de z:



Portanto, depende apenas de x, ou de forma homóloga a partir de z:



Atividades

Preparação

Como preparação para o início das atividades, recomendam-se

- a.) leitura prévia do resumo teórico, do detalhamento na apostila e referências recomendadas
- b.) estudo e testes dos exemplos
- c.) assistir aos seguintes vídeos:

<https://www.youtube.com/watch?v=SvcTSNCB4zI>

<https://www.youtube.com/watch?v=VkJ71Js3QDw>

<https://www.youtube.com/watch?v=JUVeGaJkuDY>

Orientação geral:

Atividades previstas como parte da avaliação

Apresentar todas as soluções em apenas um arquivo com formato texto (.txt).
Sugere-se usar como nome Guia_xx.txt, onde xx indicará o guia, exemplo Guia_01.txt.

Todos os arquivos deverão conter identificações iniciais com o nome e matrícula,
no caso de programas, usar comentários.

As implementações e testes dos exemplos em Verilog (.v) fornecidos como pontos de partida,
também fazem parte da atividade e deverão ter os códigos fontes entregues **separadamente**,
a fim de que possam ser compilados e testados.

Sugere-se usar como nomes Guia_01yy.v, onde yy indicará a questão, exemplo Guia_0101.v

As saídas de resultados, opcionalmente, poderão ser copiadas ao final do código,
como comentários.

Atividades extras e opcionais

Outras formas de solução serão **opcionais**; não servirão para substituir as atividades
a serem avaliadas. Caso entregues, poderão contar apenas como atividades extras.

Os resultados poderão ser anexados ao código, ao final, como comentários.

Os *layouts* de circuitos deverão ser entregues no formato (.circ), identificados internamente.
Figuras exportadas pela ferramenta serão aceitas apenas como arquivos para visualização,
mas não terão validade para fins de avaliação. Separar versões completas (a) e simplificadas (b).

Arquivos em formato (.pdf), fotos, cópias de tela ou soluções manuscritas também serão aceitos
como recursos suplementares para visualização, e **não** terão validade para fins de avaliação.

Atividade: Simplificação de circuitos

01.) Funções lógicas podem ser simplificadas diretamente do mapa de Veitch-Karnaugh:

		E			
		xy\z	z'	z	
A	x'	00	(0)	(1)	y' D
	x'	01	(2)	(3)	y C
B	x	11	(6)	(7)	y
	x	10	(4)	(5)	y' D

$$A = x' \cdot f(z)$$

$$B = x \cdot f(z)$$

$$C = y \cdot f(z)$$

$$D = y' \cdot f(z)$$

$$E = f(x,y)$$

Exemplo:

Dado o mapa de Veitch-Karnaugh abaixo, determinar a função simplificada equivalente.

		E			
		xy\z	z'	z	
A	x'	00	(0)	1 (1)	y' D
	x'	01	1 (2)	1 (3)	y C
B	x	11	(6)	(7)	y
	x	10	1 (4)	(5)	y' D

$$f(x,y,z) = \sum m(1, 2, 3, 4) = x' \cdot y' \cdot z + x' \cdot y \cdot z' + x' \cdot y \cdot z + x \cdot y' \cdot z'$$

$$\text{Com (1, 3): } x' \cdot z \quad (A)$$

$$\text{Com (2, 3): } x' \cdot y \quad (E)$$

$$f(x,y,z) = \sum m(1, 2, 3, 4) = x' \cdot z + x' \cdot y + x \cdot y' \cdot z'$$

Montar o mapa de Veitch-Karnaugh e simplificar as funções lógicas abaixo e verificar pelas respectivas tabelas-verdades implementadas em Verilog e/ou Logisim:

a) $f(x,y,z) = \sum m(1, 5, 7)$

b) $f(x,y,z) = \sum m(0, 4, 6)$

c) $f(x,y,z) = \sum m(2, 4, 6, 7)$

d) $f(x,y,z) = \sum m(1, 2, 3, 6)$

e) $f(x,y,z) = \sum m(0, 2, 5, 7)$

02.) O mapa de Veitch-Karnaugh também serve para simplificar produtos de somas (MAXTERMOS):

		E			
		Z	Z'		
		0	1		
A	X \ yz	(0)	(1)	Y	D
	X 00	(2)	(3)	Y'	C
B	X' 11	(6)	(7)	Y'	
	X' 10	(4)	(5)	Y	D

$$A = X + F(Z)$$

$$B = X' + F(Z)$$

$$C = Y' + F(Z)$$

$$D = Y' + F(Z)$$

$$E = F(X, Y)$$

Exemplo:

Dado o mapa de Veitch-Karnaugh abaixo, determinar o produto das somas (PoS) simplificado.

		E			
		Z	Z'		
		0	1		
A	X \ yz	0 (0)	1 (1)	Y	D
	X 01	1 (2)	1 (3)	Y'	C
B	X' 11	0 (6)	0 (7)	Y'	
	X' 10	1 (4)	0 (5)	Y	D

$$F(X, Y, Z) = \prod M(0, 5, 6, 7) = (X+Y+Z) \cdot (X'+Y+Z') \cdot (X'+Y'+Z) \cdot (X'+Y'+Z')$$

$$\text{Com (5, 7): } (X'+Z') \quad (B)$$

$$\text{Com (6, 7): } (X'+Y') \quad (E)$$

$$F(X, Y, Z) = \prod M(0, 5, 6, 7) = (X+Y+Z) \cdot (X'+Z') \cdot (X'+Y')$$

Montar o mapa de Veitch-Karnaugh e simplificar as funções lógicas abaixo por MAXTERMOS e verificar pelas respectivas tabelas-verdades implementadas em Verilog e/ou Logisim:

a) $F(X, Y, Z) = \prod M(4, 6, 7)$

b) $F(X, Y, Z) = \prod M(1, 5, 6)$

c) $F(X, Y, Z) = \prod M(1, 2, 4, 6)$

d) $F(X, Y, Z) = \prod M(1, 3, 5, 7)$

e) $F(X, Y, Z) = \prod M(2, 3, 6, 7)$

03.) Se o número de variáveis aumentar, o mapa deve ser modificado ligeiramente, a fim de que as vizinhanças mantenham apenas uma diferença entre elas.

		E		F		
		w'	w'	w	w	
xy\wz		00	01	11	10	
A	x'	00	(1)	(3)	(2)	y'
	x'	01	(4)	(5)	(6)	y
	x	11	(12)	(13)	(14)	y
B	x	10	(8)	(9)	(10)	y'
		z'	z	z	z'	
		H	G		H	

$$\begin{aligned}
 A &= x' \cdot f(w,z) & E &= g(x,y) \cdot w' & (A,B) &= f(w,z) \\
 B &= x \cdot f(w,z) & F &= g(x,y) \cdot w & (C,D) &= f(w,z) \\
 C &= y \cdot f(w,z) & G &= g(x,y) \cdot z & (E,F) &= g(x,y) \\
 D &= y' \cdot f(w,z) & H &= g(x,y) \cdot z' & (G,H) &= g(x,y)
 \end{aligned}$$

Exemplo:

Dado o mapa de Veitch-Karnaugh abaixo, determinar a função simplificada equivalente.

		E		F		
		w'	w'	w	w	
xy\wz		00	01	11	10	
A	x'	0 (0)	1 (1)	0 (3)	1 (2)	y'
	x'	0 (4)	0 (5)	1 (7)	1 (6)	y
	x	1 (12)	0 (13)	0 (15)	1 (14)	y
B	x	0 (8)	1 (9)	0 (11)	0 (10)	y'
		z'	z	z	z'	
		H	G		H	

$$\begin{aligned}
 f(x,y,w,z) &= \sum m(1, 2, 6, 7, 9, 12, 14) \\
 &= x' \cdot y' \cdot w' \cdot z + x' \cdot y' \cdot w \cdot z' + x' \cdot y \cdot w \cdot z' + x' \cdot y \cdot w \cdot z + x \cdot y' \cdot w' \cdot z + x \cdot y \cdot w' \cdot z' + x \cdot y \cdot w \cdot z'
 \end{aligned}$$

$$\begin{aligned}
 \text{Com (2, 6): } &x' \cdot w \cdot z' & (A) & & \text{Com (1,9): } &y' \cdot w' \cdot z & (D) \\
 \text{Com (6, 7): } &x' \cdot y \cdot w & (F) & & \text{Com (12,14): } &x \cdot y \cdot z' & (H)
 \end{aligned}$$

$$f(x,y,w,z) = x' \cdot w \cdot z' + x' \cdot y \cdot w + y' \cdot w' \cdot z + x \cdot y \cdot z'$$

Construir os mapas de Veitch-Karnaugh e simplificar as funções lógicas abaixo e verificar pelas respectivas tabelas-verdades implementadas em Verilog e/ou Logisim:

- $f(x,y,w,z) = \sum m(1, 2, 5, 8, 10, 12)$
- $f(x,y,w,z) = \sum m(0, 1, 3, 5, 7, 9, 13)$
- $f(x,y,w,z) = \sum m(0, 1, 2, 6, 7, 8, 14, 15)$
- $f(x,y,w,z) = \sum m(2, 4, 5, 7, 10, 13, 14)$
- $f(x,y,w,z) = \sum m(0, 1, 3, 7, 8, 9, 11, 15)$

04.) O mapa de Veitch-Karnaugh também serve para simplificar produtos de somas (MAXTERMOS):

		E		F			
		W	W	W'	W'		
XY\WZ		00	01	11	10		
A	X	00	(0)	(1)	(3)	(2)	Y D
	X	01	(4)	(5)	(7)	(6)	Y' C
B	X'	11	(12)	(13)	(15)	(14)	Y'
	X'	10	(8)	(9)	(11)	(10)	Y D
		Z	Z'	Z'	Z		
		H	G		H		

$$\begin{aligned}
 A &= X + F(W,Z) & E &= G(X,Y) + W & (A,B) &= F(W,Z) \\
 B &= X' + F(W,Z) & F &= G(X,Y) + W' & (C,D) &= F(W,Z) \\
 C &= Y' + F(W,Z) & G &= G(X,Y) + Z' & (E,F) &= G(X,Y) \\
 D &= Y + F(W,Z) & H &= G(X,Y) + Z & (G,H) &= G(X,Y)
 \end{aligned}$$

Exemplo:

Dado o mapa de Veitch-Karnaugh abaixo, determinar o produto de somas (PoS) simplificado.

		E		F			
		W	W	W'	W'		
XY\WZ		00	01	11	10		
A	X	00	1 (0)	1 (1)	1 (3)	1 (2)	Y D
	X	01	1 (4)	0 (5)	0 (7)	1 (6)	Y' C
B	X'	11	1 (12)	0 (13)	0 (15)	1 (14)	Y'
	X'	10	0 (8)	1 (9)	1 (11)	0 (10)	Y D
		Z	Z'	Z'	Z		
		H	G		H		

$$F(X,Y,W,Z) = \prod M(5, 7, 8, 10, 13, 15)$$

$$= (X+Y'+W+Z') \cdot (X+Y'+W'+Z') \cdot (X'+Y'+W+Z') \cdot (X'+Y'+W'+Z') \cdot (X'+Y+W'+Z') \cdot (X'+Y+W'+Z)$$

$$\text{Com } (5, 7): (X+Y'+Z') \quad (G) \qquad \text{Com } (8,10): (X'+Y+Z) \quad (H)$$

$$\text{Com } (13,15): (X'+Y'+Z') \quad (G) \qquad \text{Com } (5,7,13,15): (Y'+Z') \quad (C)$$

$$F(X,Y,W,Z) = (Y'+Z') \cdot (X'+Y+Z)$$

Construir os mapas de Veitch-Karnaugh e simplificar as funções lógicas abaixo por MAXTERMOS e verificar pelas respectivas tabelas-verdades implementadas em Verilog e/ou Logisim:

a) $F(X,Y,W,Z) = \prod M(2, 5, 7, 13)$

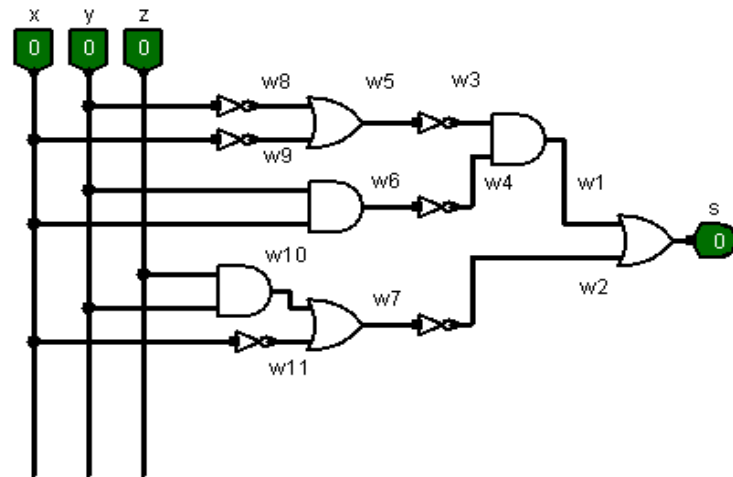
b) $F(X,Y,W,Z) = \prod M(4, 8, 9, 12, 15)$

c) $F(X,Y,W,Z) = \prod M(4, 8, 11, 13, 15)$

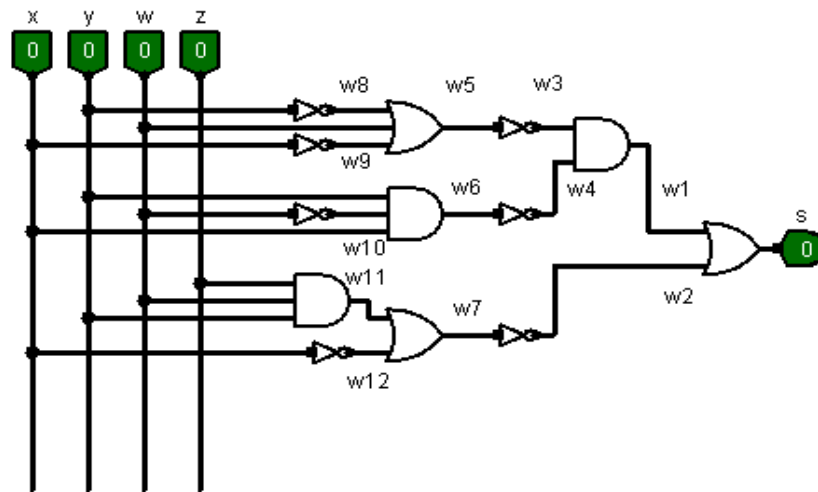
d) $F(X,Y,W,Z) = \prod M(1, 3, 5, 13, 14, 15)$

e) $F(X,Y,W,Z) = \prod M(3, 5, 6, 7, 13, 14)$

- 05.) Identificar a equação característica do circuito lógicos abaixo e simplifica-la pelo mapa de Veitch-Karnaugh usando mintermos. Descrever e simular o circuito simplificado em Verilog e Logisim.



- 06.) Identificar as equações características dos circuitos lógicos abaixo e simplifica-las pelo mapa de Veitch-Karnaugh usando MAXTERMOS. Descrever e simular o circuito simplificado em Verilog e Logisim.



Extras

07.) Dada a expressão em Verilog abaixo, verificar se há simplificação e montar o circuito equivalente:

```
module FXYZ (output S1, input X, input Y, input Z);  
  assign s1 = ( X | ~Y | ~Z ) & ( ~X | Y | ~Z ) & ( ~X | ~Y | Z );  
endmodule // FXYZ
```

08.) Dada a expressão em Verilog abaixo, verificar se há simplificação e montar o circuito equivalente:

```
module fwxyz (output s1, input w, input x, input y, input z);  
  assign s1 = (w & x & y & z) | (~w & ~x & ~y & ~z) | (w & x & ~y & ~z) | (w & ~x & y & ~z);  
endmodule // fwxyz
```