Sistemas Digitais
Universidade Católica do Salvador
Professor Marco Antônio C. Câmara

Aula 03 – Simplificação de Expressões Lógicas

Roteiro da Aula :

Nesta aula conheceremos os métodos mais utilizados para simplificação de circuitos lógicos. Através do uso destes métodos, conseguiremos reduzir significativamente as expressões utilizadas para representar uma determinada tabela verdade, reduzindo o porte e o custo de projetos de circuitos digitais.

Na verdade, este trabalho já podia ser realizado com base na álgebra de *Boole* estudada na apostila anterior. No entanto, devido à relativa complexidade obtida para expressões de maior porte, foram criados dois métodos extremamente adequados. O primeiro é um método gráfico, muito interessante para simplificação de expressões de até 4 (quatro) variáveis. Já o segundo método, que é tabular, permite simplificar expressões com qualquer quantidade de variáveis, com a grande vantagem de permitir a implementação simplificada através de um programa de computador.

1. Diagramas de Veitch-Karnaugh

2. Processo Tabular de Quine-McCluskey

O processo de Quine-McCluster é totalmente tabular e consiste na execução de uma seqüência pré-determinada de passos que permite a simplificação de expressões lógicas com qualquer número de variáveis.

Apresentaremos abaixo os passos que devem ser executados. Para melhorar a didática, executaremos os procedimentos recomendados em cada um dos passos com base em uma expressão exemplo.

#	e4	e3	e2	e1	e0	S
0	0	0	0	0	0	0
1	0	0	0	0	1	1
2	0	0	0	1	0	1
3	0	0	0	1	1	1
4	0	0	1	0	0	0
5	0	0	1	0	1	1
6	0	0	1	1	0	0
7	0	0	1	1	1	0
8	0	1	0	0	0	0
9	0	1	0	0	1	1
10	0	1	0	1	0	1
11	0	1	0	1	1	1
12	0	1	1	0	0	0
13	0	1	1	0	1	0
14	0	1	1	1	0	0
15	0	1	1	1	1	0
16	1	0	0	0	0	0
17	1	0	0	0	1	0
18	1	0	0	1	0	1
19	1	0	0	1	1	1
20	1	0	1	0	0	1
21	1	0	1	0	1	1
22	1	0	1	1	0	0
23	1	0	1	1	1	1
24	1	1	0	0	0	0
25	1	1	0	0	1	1
26	1	1	0	1	0	1
27	1	1	0	1	1	1
28	1	1	1	0	0	0
29	1	1	1	0	1	0
30	1	1	1	1	0	0
31	1	1	1	1	1	0

Tabela 1 - Tabela Verdade (expressão exemplo)

2.1. Representar a função a ser simplificada no formato de uma soma de mintermos

Antes de começar, é necessário representar a expressão como um somatório de mintermos. Os mintermos são os números correspondentes às linhas da tabela verdade que tem saída igual a 1. Isto é particularmente simples se possuirmos a tabela verdade da expressão. Caso a expressão não esteja neste formato, devemos convertê-la antes de aplicar o método.

No nosso exemplo, temos uma tabela verdade (veja figura ao lado). Os mintermos encontram-se identificados em vermelho. Podemos dizer então que a função é representada da seguinte forma :

$$f(e4,e3,e2,e1,e0)=\sum m(1,2,3,5,9,10,11,18,19,20,21,23,25,26,27)$$

2.2. Agrupar os mintermos que possuem a mesma quantidade de bits iguais a 1

JUSTIFICATIVA : Os mintermos identificam as variáveis de entrada consideradas na expressão, tal como no formato canônico para lógica positiva. Sendo assim, o mintermo **5**, por exemplo, representa a inexistência das variáveis **e4**, **e3** e **e1** (veja a área identificada com uma elipse azul na tabela verdade ao lado). Em outras palavras, temos :

$$5=0x2^4+0x2^3+1x2^2+0x2^1+1x2^0$$

No passo seguinte, vamos formar pares de mintermos cuja diferença entre seus valores sejam potências de 2. Isto não seria possível na subtração de mintermos com diferenças na quantidade de bits maior que 1, já que as potências de 2 são sempre números binários de apenas um bit. Por isto os grupos são criados.

# Bits = 1	Mintermos
1	1
	2
2	3
	5
	9
	10
	18
	20
3	11
	19
	21
	25
	26
4	23
	27

Na tabela ao lado representamos o agrupamento dos mintermos com base na quantidade de bits de cada um. É interessante colocar os mintermos em ordem alfabética dentro dos grupos, e os grupos em ordem crescente de número de bits.

2.3. Formar pares entre os mintermos de um grupo e do grupo seguinte

Basta formar os pares observando a regra de só criar pares entre mintermos cuja diferença de valor seja uma potência de 2, considerando também que o mintermo do grupo posterior deve ter valor maior.

Os mintermos utilizados devem ser marcados para que seja possível a análise posterior dos mintermos não agrupados.

Para cada par formado, além de identificar o par de mintermos agrupados no formato **x**,**y** (**d**) (onde “x” e “y” são os mintermos agrupados e “d” é a diferença entre os mesmos). Todos os pares gerados na avaliação de dois grupos de mintermos adjacentes devem ser agrupados devem ser agrupados separadamente para cada conjunto de dois grupos, assim como fizemos para os mintermos com a mesma quantidade de bits iguais a 1.

As três setas na tabela abaixo representam os três pares iniciados pelo mintermo 1.

JUSTIFICATIVA : Ao criar estes pares, estamos na verdade eliminando uma variável de entrada (representada pela potência de 2 correspondente à diferença). Isto porque a soma entre dois mintermos que no formato canônico possuem apenas um bit diferente na prática elimina este bit. Basta lembrar que o agrupamento em pares na prática não altera o valor da função a ser simplificada, que nada mais é do que a soma <u>de todos</u> os mintermos (propriedade associativa da soma). Ex :

$$\bar{A}.B.\bar{C}.\bar{D}.E.\bar{F}.G.H + \bar{A}.B.\bar{C}.\bar{D}.E.\bar{F}.\bar{G}.H = \bar{A}.B.\bar{C}.\bar{D}.E.\bar{F}.H\overbrace{(G + \bar{G})}^{1}$$

Bit diferente

# Bits = 1	Mintermos		Pares
1	1	v	1,3 (2)
	2	v	1,5 (4)
2	3	v	1,9 (8)
	5	v	2,3 (1)
	9	v	2,10 (8)
	10	v	2,18 (16)
	18	v	3,11 (8)
	20	v	3,19 (16)
3	11	v	5,21 (16)
	19	v	9,11 (2)
	21	v	9,25 (16)
	25	v	10,11 (1)
	26	v	10,26 (16)
4	23	v	18,19 (1)
	27	v	18,26 (8)
			20,21 (1)
			11,27 (16)
			19,23 (4)
			19,27 (8)
			21,23 (2)
			25,27 (2)
			26,27 (1)

2.4. Formar quadras entre os novos grupos

Seguimos regras semelhantes ao item anterior. Agora só podemos agrupar pares resultantes da eliminação da mesma variável de entrada (números entre parênteses iguais). O resto é similar, ou seja, a diferença entre cada mintermo de um par e seu correspondente do outro par deve ser uma potência de 2 (a mesma para todos).

Continua valendo a regra de marcar todos os pares utilizados e também aquela que só permite a formação de quadras entre pares de grupos adjacentes, onde todos os mintermos do grupo posterior devem ser maiores que seus correspondentes no grupo anterior.

Para cada quadra no formato **x**,**y**,**z**,**t** (**dp**,**dq**) (onde **dp**= **y** - **x** = **t** - **z** e **dq** = **z** - **x** = **t** - **y**), todas as quadras geradas na avaliação de grupos de pares adjacentes devem ser agrupadas separadamente para cada conjunto de dois grupos, assim como fizemos para os mintermos com a mesma quantidade de bits iguais a 1.

Devem ser eliminadas quadras repetidas dos mesmos mintermos, mesmo que estas apresentem-nos em seqüências diferentes.

# Bits = 1	Mintermos	Pares	Quadra
1	1 v	1,3 (2) v	1,3,9,11 (2,8)
	2 v	1,5 (4)	~~1,9,3,11 (8,2)~~
2	3 v	1,9 (8) v	2,3,10,11 (1,8) v
	5 v	2,3 (1) v	2,3,18,19 (1,16) v
	9 v	2,10 (8) v	~~2,10,3,11 (8,1)~~
	10 v	2,18 (16) v	2,10,18,26 (8,16) v
	18 v	3,11 (8) v	~~2,18,3,19 (16,1)~~
	20 v	3,19 (16) v	~~2,18,10,26 (16,8)~~
3	11 v	5,21 (16)	3,11,19,27 (8,16) v
	19 v	9,11 (2) v	~~3,19,11,27 (16,8)~~
	21 v	9,25 (16) v	9,11,25,27 (2,16)
	25 v	10,11 (1) v	~~9,25,11,27 (16,2)~~
	26 v	10,26 (16) v	10,11,26,27 (1,16) v
4	23 v	18,19 (1) v	~~10,26,11,27 (16,1)~~
	27 v	18,26 (8) v	18,19,26,27 (1,8) v
		20,21 (1)	~~18,26,19,27 (8,1)~~
		11,27 (16) v	
		19,23 (4)	
		19,27 (8) v	
		21,23 (2)	
		25,27 (2) v	
		26,27 (1) v	

2.5. Formar grupos de 8, 16, 32, ... mintermos

Repete-se o procedimento anterior para os octetos e demais agrupamentos possíveis.

# Bits = 1	Mintermos	Pares	Quadra	Octetos
1	1 v	1,3 (2) v	1,3,9,11 (2,8)	2,3,10,11,18,19,26,27 (1,8,16)
	2 v	1,5 (4)	~~1,9,3,11 (8,2)~~	~~2,3,18,19,10,11,26,27 (1,16,8)~~
2	3 v	1,9 (8) v	2,3,10,11 (1,8) v	~~2,10,18,26,3,11,19,27 (8,16,1)~~
	5 v	2,3 (1) v	2,3,18,19 (1,16) v	
	9 v	2,10 (8) v	~~2,10,3,11 (8,1)~~	
	10 v	2,18 (16) v	2,10,18,26 (8,16) v	
	18 v	3,11 (8) v	~~2,18,3,19 (16,1)~~	
	20 v	3,19 (16) v	~~2,18,10,26 (16,8)~~	
3	11 v	5,21 (16)	3,11,19,27 (8,16) v	
	19 v	9,11 (2) v	~~3,19,11,27 (16,8)~~	
	21 v	9,25 (16) v	9,11,25,27 (2,16)	
	25 v	10,11 (1) v	~~9,25,11,27 (16,2)~~	
	26 v	10,26 (16) v	10,11,26,27 (1,16) v	
4	23 v	18,19 (1) v	~~10,26,11,27 (16,1)~~	
	27 v	18,26 (8) v	18,19,26,27 (1,8) v	
		20,21 (1)	~~18,26,19,27 (8,1)~~	
		11,27 (16) v		
		19,23 (4)		
		19,27 (8) v		
		21,23 (2)		
		25,27 (2) v		
		26,27 (1) v		

2.6. Analisar a tabela resultante

Após o trabalho de simplificação, analisamos a tabela resultado. Se todos os mintermos, pares, quadras e demais grupos intermediários estiverem completamente marcados, o último grupo representará a função definitiva.

Se existirem mintermos, pares, quadras ou qualquer item de grupos intermediários sem marcação (na tabela anterior, pares e quadras marcados em azul), será necessário montar uma nova tabela. Neste caso o objetivo é realizar a última simplificação, se esta for possível.

2.7. Tabela de implicativos diretos

Considerando a existência de mintermos, pares, quadras ou demais itens não marcados, é montada uma nova tabela visando simplificar, se ainda for possível, os itens não agrupados.

A tabela relaciona em linhas os itens não marcados (inclusive os do último grupo) e todos os mintermos originais em colunas. Os itens não marcados (em azul na última tabela) são relacionados em ordem decrescente de número de mintermos envolvidos. São marcados com um “X” as interseções entre cada expressão e os mintermos que a compõe, de onde partimos para o processo de simplificação.

Cada coluna de mintermo que possui apenas uma marcação tem o seu “X” identificado com um círculo. Após isto, todas os “X” das linhas que contém pelo menos um “X” circulado tem os demais “X” marcados com um quadrado. As expressões correspondentes obrigatoriamente farão parte da expressão final.

Após isto, de cima para baixo, são eliminadas as expressões que têm todos os seus mintermos representados em outras expressões que não tenham sido marcadas anteriormente.

Implicativos Diretos	Mintermos														
	1	2	3	5	9	10	11	18	19	20	21	23	25	26	27
2,3,10,11,18,19,26,27 (1,8,16)		X	$\boxed{X}$			X	$\boxed{X}$	X	$\boxed{X}$					X	$\boxed{X}$
9,11,25,27 (2,16)					$\boxed{X}$		$\boxed{X}$						X		$\boxed{X}$
~~1,3,9,11 (2,8)~~	~~X~~		~~X~~		~~X~~		~~X~~								
~~21,23 (2)~~											~~X~~	~~X~~			
19,23 (4)									X			X			
20,21 (1)										X	$\boxed{X}$				
~~5,21 (16)~~				~~X~~							~~X~~				
1,5 (4)	X			X											

2.8. Determinação da expressão final

Os componentes da expressão final são as expressões não eliminadas na última etapa. Todos os números entre parênteses representam as variáveis de entrada que são eliminadas, tornando simples a montagem da expressão final.

No exemplo, temos :

2.8.1. Expressão 2,3,10,11,18,19,26,27 (1,8,16) :

Os números entre parênteses demonstram a eliminação das variáveis de entrada **e4**, **e3** e **e0**. Sendo assim, a expressão correspondente é :

#	e4	e3	e2	e1	e0
2	0	0	0	1	0
3	0	0	0	1	1
10	0	1	0	1	0
11	0	1	0	1	1
18	1	0	0	1	0
19	1	0	0	1	1
26	1	1	0	1	0
27	1	1	0	1	1

As colunas eliminadas estão marcadas em marron. Como a variável **e2** está negada, a expressão final é **$\overline{e2}$. e1**.

2.8.2. Expressão 9,11,25,27 (2,16) :

Os números entre parênteses demonstram a eliminação das variáveis de entrada **e1** e **e4**. Sendo assim, a expressão resultante é :

#	e4	e3	e2	e1	e0
9	0	1	0	0	1
11	0	1	0	1	1
25	1	1	0	0	1
27	1	1	0	1	1

As colunas eliminadas estão marcadas em marron. Novamente a variável **e2** está negada. Sendo assim, a expressão final é $\mathbf{e3 \,.\, \overline{e2} \,.\, e0}$.

2.8.3. Expressão 19,23 (4) :

O número entre parênteses demonstra a eliminação da variável de entrada **e2**. Sendo assim, a expressão resultante é :

#	e4	e3	e2	e1	e0
19	1	0	0	1	1
23	1	0	1	1	1

A coluna eliminada está marcada em marron. Agora é a variável **e3** que está negada. Sendo assim, a expressão final é $\mathbf{e4 \,.\, \overline{e3} \,.\, e1 \,.\, e0}$.

2.8.4. Expressão 20,21 (1) :

O número entre parênteses demonstra a eliminação da variável de entrada **e0**. Sendo assim, a expressão resultante é :

#	e4	e3	e2	e1	e0
20	1	0	1	0	0
21	1	0	1	0	1

A coluna eliminada está marcada em marron. As variáveis **e3** e **e1** estão negadas. Sendo assim, a expressão final é $\mathbf{e4 \,.\, \overline{e3} \,.\, e2 \,.\, \overline{e1}}$.

2.8.5. Expressão 1,5 (4) :

O número entre parênteses demonstra a eliminação da variável de entrada **e2**. Sendo assim, a expressão resultante é :

#	e4	e3	e2	e1	e0
1	0	0	0	0	1
5	0	0	1	0	1

A coluna eliminada está marcada em marron. Apenas a variável **e0** não está negada. Sendo assim, a expressão final é $\mathbf{\overline{e4} \,.\, \overline{e3} \,.\, \overline{e1} \,.\, e0}$.

Agrupando os resultados parciais, chegamos à expressão final já simplificada :

$$\mathbf{S = \overline{e2} \,.\, e1 \;+\; e3 \,.\, \overline{e2} \,.\, e0 \;+\; e4 \,.\, \overline{e3} \,.\, e1 \,.\, e0 \;+\; e4 \,.\, \overline{e3} \,.\, e2 \,.\, \overline{e1} \;+\; \overline{e4} \,.\, \overline{e3} \,.\, \overline{e1} \,.\, e0}$$

2.9. Uso do método Quine-McCluskey com condições irrelevantes

Ante a existência de condições irrelevantes, aplica-se o método da mesma forma, considerando-se as condições irrelevantes como válidas na tabela principal.

Já na tabela de implicativos diretos, os mintermos irrelevantes devem ser apresentados em colunas separadas, sendo que estes só devem ser utilizados com o objetivo de simplificar ainda mais a expressão final.

Exercícios :

1) Simplifique as expressões abaixo com base nos Diagramas de Veitch-Karnaugh :

... a preparar ...

2) Utilizando-se do método tabular de Quine-McCluskey, obtenha a expressão mais simples da seguinte função de mintermos :

$$f(a,b,c,d,e) = \sum m(0,6,8,10,12,14,17,19,20,22,25,27,28,30)$$

3) Em um calendário digital, dois bits (**b0** e **b1**) determinam para o circuito seqüencial qual a data final que deve ser considerada na contagem da data de cada mês. Para tanto, o sistema usa o número do mês (variável entre 1 e 12) e o número do ano (os dois últimos identificam se o mesmo é bissexto).

Data Final p/ contagem	B1	B0
28	0	0
29	0	1
30	1	0
31	1	1

Com base nestas características, utilize o método de simplificação mais adequado para obter as expressões finais de B1 e B0.