

# Projeto de Lógica Combinatória

S. W. Song

MAC 412 - Organização de Computadores

# Lógica combinatória

Esse assunto já foi visto em Álgebra Booleana.

Aqui vamos apenas tratar de alguns tópicos específicos.

- Projeto de lógica combinatória usando apenas um tipo de portas:
  - apenas portas NAND
  - apenas portas NOR
- Projeto de lógica combinatória usando multiplexador (ou seletor).  
Veremos também demultiplexador (ou decodificador).

- Projeto de lógica combinatória usando ROM.
- Projeto de lógica combinatória usando PLA.

# Lógica Combinatória usando NAND

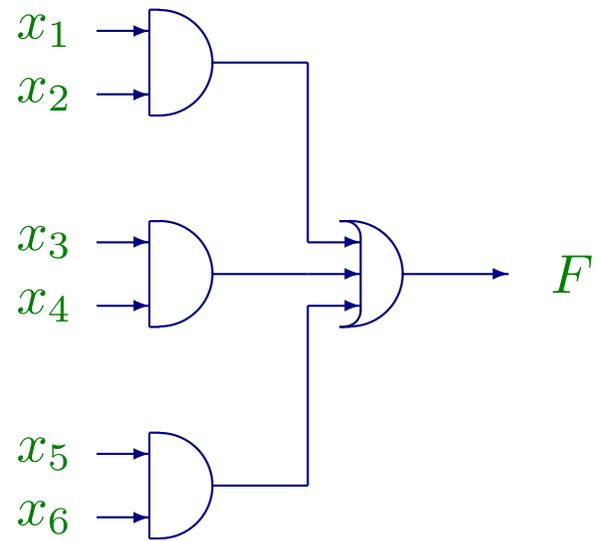
Dada uma função combinatória qualquer, expressa na forma canônica (soma de produtos), em lugar dos operadores AND e OR, podemos usar apenas NAND.

Usamos um exemplo. Seja

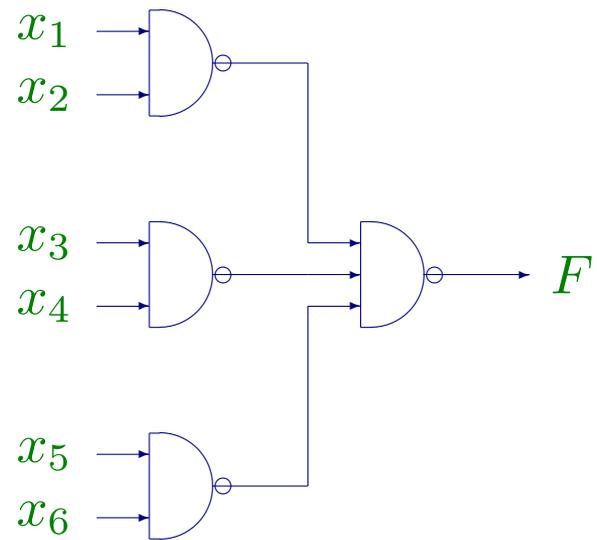
$$F = x_1x_2 + x_3x_4 + x_5x_6$$

Temos

$$F = [x_1x_2 + x_3x_4 + x_5x_6]'' = [(x_1x_2)'(x_3x_4)'(x_5x_6)']'$$



equivalente a:



# Lógica Combinatória usando NOR

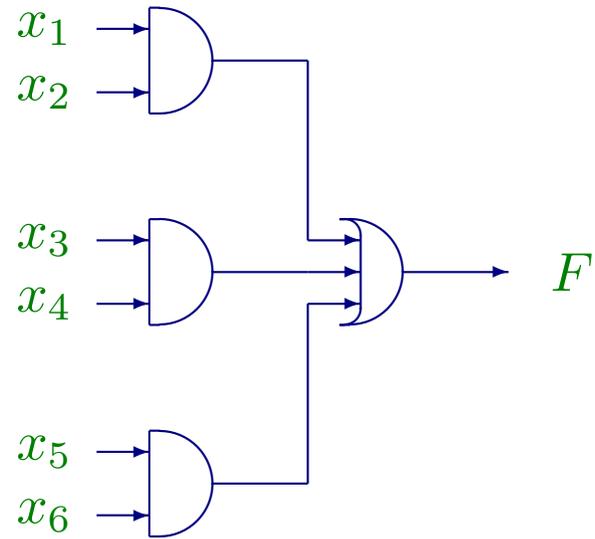
Dada uma função combinatória qualquer, expressa na forma canônica (soma de produtos), em lugar dos operadores AND e OR, podemos usar apenas NOR.

Usamos um exemplo. Seja

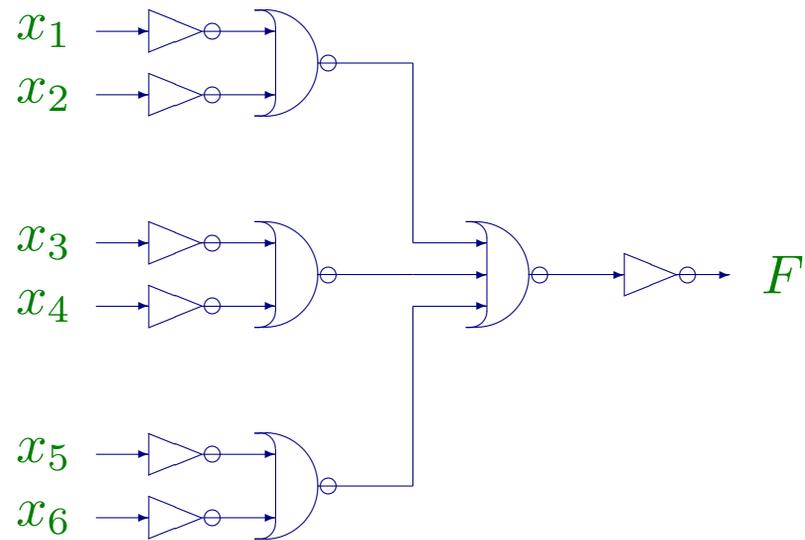
$$F = x_1x_2 + x_3x_4 + x_5x_6$$

Temos

$$F = [(x_1x_2)'' + (x_3x_4)'' + (x_5x_6)'']'' = [(x'_1 + x'_2)' + (x'_3 + x'_4)' + (x'_5 + x'_6)']''$$



equivalente a:

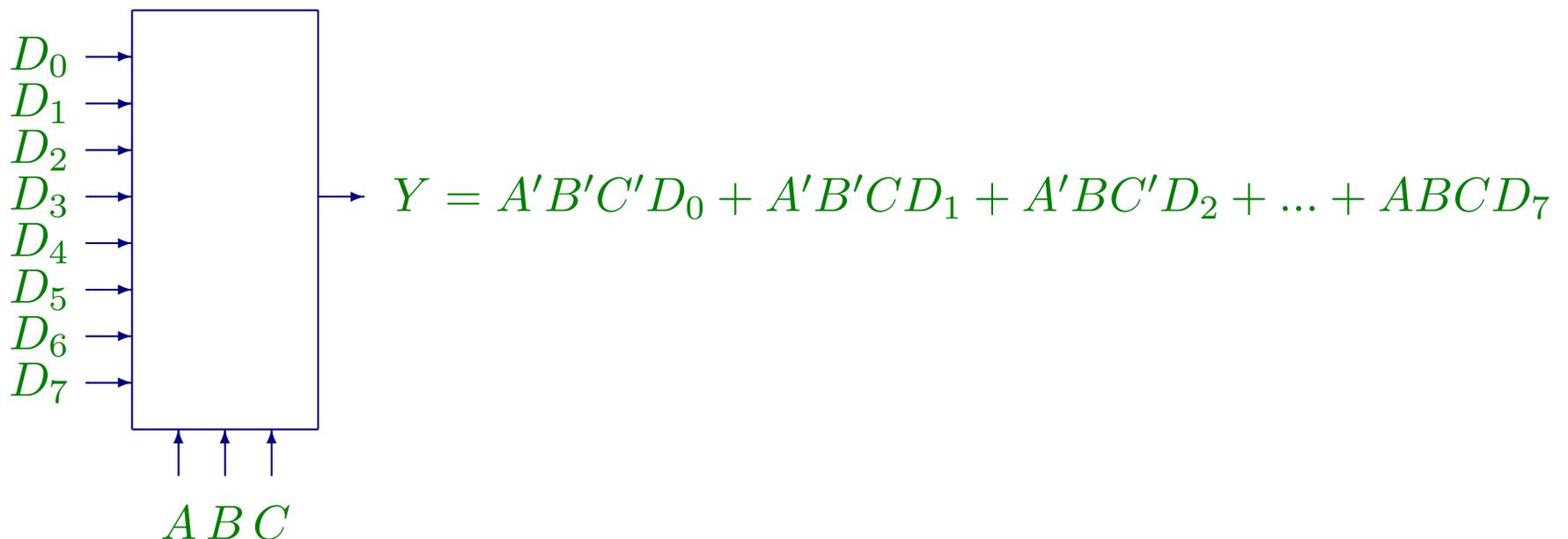


# Exercício em classe

1. Implemente a função  $F = AB + A'C$  em circuito NMOS, usando o diagrama de palito. (a) usando NAND (b) usando NOR.
2. Quantas portas NOT e NAND você usou na solução acima, item 1(a)? Quantas portas NOT e NOR você usou em 1(b)?
3. Você acredita que dá para implementar F do exercício 1 com apenas *uma* porta NOT e nenhuma porta a mais? (Parece incrível mas dá! Isso será mostrado na próxima aula).
4. Implemente a função OU EXCLUSIVO de 2 entradas em NMOS, usando diagrama de palito.

# Projeto de lógica combinatória com multiplexador

Mesmo com o avanço de VLSI e circuitos complexos podem ser implementados sob medida em um único *chip*, ainda é importante sabermos projetar lógica combinatória e sequencial, por exemplo, usando circuitos como multiplexador, PLA, *standard cells*, etc.



# Uso do multiplexador

O multiplexador funciona como um seletor. Mas pode ser usado também para projetar lógica combinatória.

Um multiplexador de 8 entradas pode ser usado para implementar qualquer lógica combinatória de 4 entradas. (Em geral, um multiplexador de  $n$  entradas pode ser usado para projetar qualquer lógica combinatória de  $\log n + 1$  entradas.)

Mostramos por um exemplo para implementar a função lógica

$$F = A'B'C'D' + A'B'CD + A'BC'D + A'BC'D' + AB'C'D + AB'CD' + ABC'D' + ABC'D$$

# Exemplo

$$F = A'B'C'D' + A'B'CD + A'BC'D + A'BC'D' + AB'C'D + AB'CD' + ABC'D' + ABC'D$$

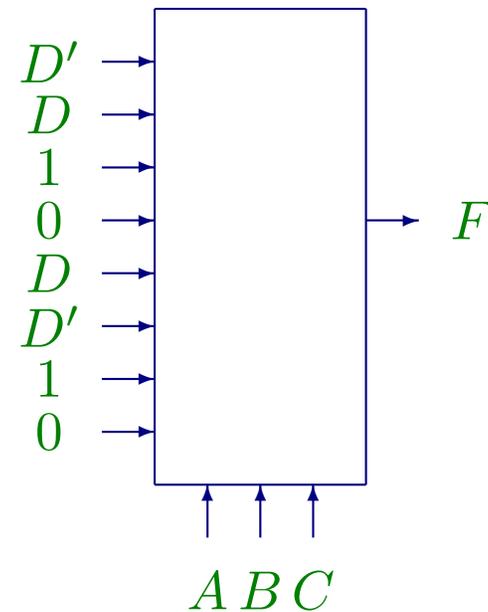
No multiplexador, os sinais de controle  $A, B, C$  correspondem às variáveis  $A, B, C$  da função  $F$ . A saída  $Y$  deve dar origem ao valor de  $F$ .

Precisamos saber quais os valores que devem ser usados como as entradas  $D_0, D_1, \dots, D_7$ . Isso é facilmente obtido como se segue.

# Exemplo

$$F = A'B'C'D' + A'B'CD + A'BC'D + A'BC'D' + AB'C'D + AB'CD' + ABC'D' + ABC'D$$

$A'B'C'$	$D_0 = D'$
$A'B'C$	$D_1 = D$
$A'BC'$	$D_2 = D + D' = 1$
$A'BC$	$D_3 = 0$
$AB'C'$	$D_4 = D$
$AB'C$	$D_5 = D'$
$ABC'$	$D_6 = D' + D = 1$
$ABC$	$D_7 = 0$



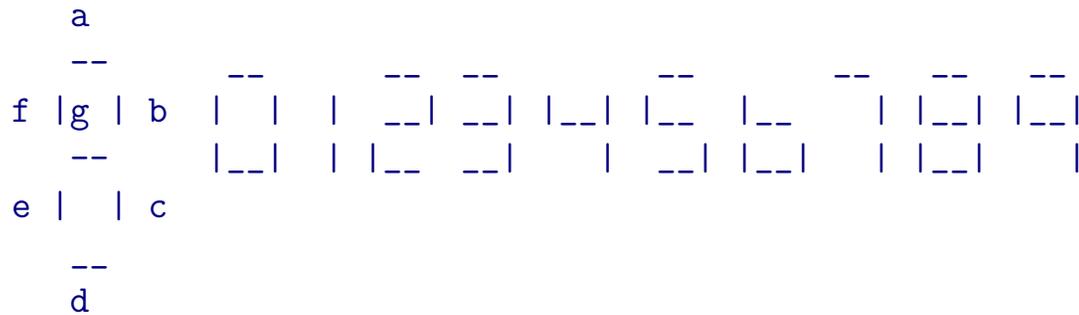
# Uso do ROM ou PROM

ROM = Read Only Memory

PROM = Programmable ROM

Idéia simples: Os valores das funções lógicas são armazenados na memória. As entradas da função lógica são decodificadas para dar um endereço para acessar a memória.

# Exemplo: display de algarismo



		a	b	c	d	e	f	g
	0	1	1	1	1	1	1	0
	1	0	1	1	0	0	0	0
A	2	1	1	0	1	1	0	1
B	3	1	1	1	1	0	0	1
C	4	0	1	1	0	0	1	1
D	5	1	0	1	1	0	1	1
	6	0	0	1	1	1	1	1
	7	1	1	1	0	0	0	0
	8	1	1	1	1	1	1	1
	9	1	1	1	0	0	1	1

Decodificador -----  
decimal

# Uso de PLA

PLA = Programmable Logic Array

A ROM implementa toda a tabela de verdade na memória. PLA é usada para evitar o desperdício de espaço da ROM, quando muitas combinações de variáveis de entrada nunca ocorrem.

A PLA consta de dois planos, chamados *plano AND* e *plano OR*. Mostramos um exemplo para implementar as funções lógicas  $f_1$ ,  $f_2$ ,  $f_3$  e  $f_4$ .

# Exemplo do uso da PLA

$$f_1 = AB'C + B'F + BCD'EF' + EF$$

$$f_2 = AB'CF + B'D' + B'EF$$

$$f_3 = CD'F + A'BC'DE'F + B'CF'$$

$$f_4 = ACF + B'EF + B'D' + ACF'$$

