

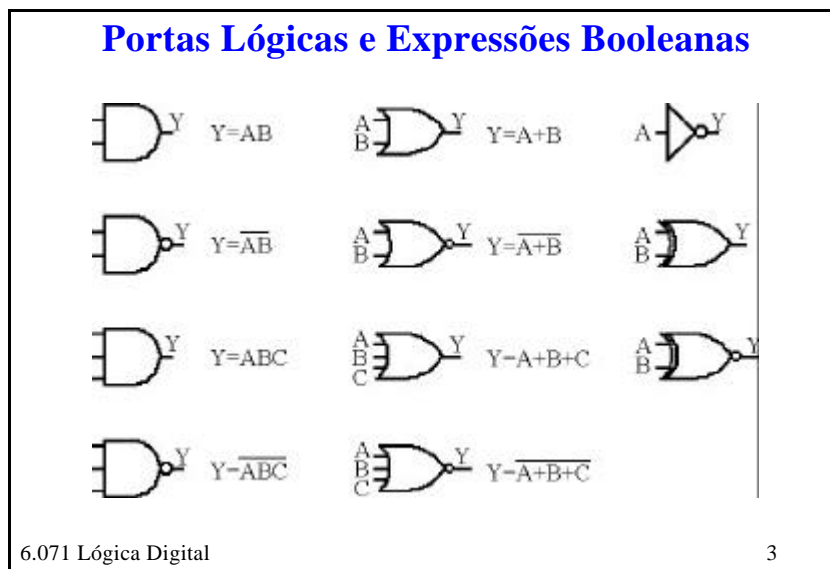
A lógica digital pode ser descrita em termos de símbolos lógicos padrão e suas tabelas-verdade correspondentes. As empresas de eletrônicos desenvolveram chips baseados em transistores que realizam a função de cada um destes. As linhas horizontais representam entradas ou saídas (nos exemplos acima, leia da esquerda para a direita). Os círculos pequenos nas saídas à direita correspondem a um inversor (realizam uma operação lógica NOT para a saída).

Álgebra Booleana	
<p><b>E</b></p> $0 \cdot 0 = 0$ $0 \cdot 1 = 1 \cdot 0 = 0$ $1 \cdot 1 = 1$	<p><b>OU</b></p> $0 + 0 = 0$ $0 + 1 = 1 + 0 = 1$ $1 + 1 = 1$
<p><b>XOR</b></p> $0 \oplus 0 = 0$ $0 \oplus 1 = 1 \oplus 0 = 1$ $1 \oplus 1 = 0$	<p><b>NÃO</b></p> <p>Se <math>A = 0</math>, então <math>\bar{A} = 1</math></p> <p>Se <math>A = 1</math>, então <math>\bar{A} = 0</math></p>

6.071 Lógica Digital
2

A ação de circuitos lógicos pode ser entendida em termos de lógica Booleana. Normalmente, iremos usar três elementos. Primeiro, você deve se lembrar que, nas nossas breves anotações, 0 é FALSO e 1 é VERDADEIRO. A operação AND é indicada por um ponto (que normalmente é omitido), e a tabela lógica acima parece familiar. A operação OR é indicada por um sinal +, e o conjunto de resultados é bastante familiar, mas observe que “VERDADEIRO ou VERDADEIRO” é VERDADEIRO. A operação NOT é simplesmente uma inversão, e é indicada por uma barra sobre o estado.

Também teremos, ocasionalmente, necessidade da porta “EXCLUSIVE OR”, que é semelhante ao OR, mas é indicado por um + com um círculo ao redor, sendo “VERDADEIRO EXCLUSIVE OR VERDADEIRO” é FALSO.



Podemos reescrever a porta lógica em termos de álgebra booleana. Observe que as portas AND e OR podem ser ampliadas para além de duas entradas; na verdade, elas podem ter qualquer número.

### Tabela de Identidades Lógicas

- |                                  |  |
|----------------------------------|--|
| 1) $A+B = B+A$                   | 13) $A+A = A$  |
| 2) $AB = BA$                     | 14) $\overline{\overline{A}} = A$                                    |
| 3) $A+(B+C) = (A+B)+C$           | 15) $\overline{\overline{A}} = A$                                    |
| 4) $A(BC) = (AB)C$               | 16) $A+\overline{A} = 1$   |
| 5) $A(B+C) = AB+AC$              | 17) $A\overline{A} = 0$  |
| 6) $(A+B)(C+D) = AC+AD+BC+BD$    | 18) $\overline{A+B} = \overline{A}\overline{B}$                      |
| 7) $\overline{\overline{1}} = 0$ | 19) $\overline{\overline{A+B}} = \overline{A+B}$                     |
| 8) $\overline{0} = 1$            | 20) $A+\overline{A}B = A+B$  |
| 9) $A \cdot 0 = 0$               | 21) $\overline{A}+AB = \overline{A}+B$                               |
| 10) $A \cdot 1 = A$              | 22) $A\oplus B = \overline{A}B + A\overline{B} = (A+B)\overline{AB}$ |
| 11) $A+0 = A$                    | 23) $A\oplus B = AB + \overline{A}\overline{B}$                      |
| 12) $A+1 = 1$                    |  |

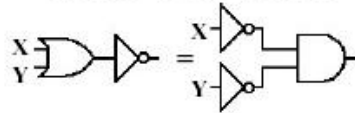
Álgebra booleana é simples depois que você se acostuma, mas isso leva um tempo. Os conjuntos de identidades acima são diretos. A primeira coluna você provavelmente conhece (se for para dizer em termos de VERDADEIRO ou FALSO) e as entradas da segunda coluna podem ser deduzidas, e não precisam ser memorizadas.

### Teorema de DeMorgan

A negação de quantidade, X OR Y são iguais a  
NOT X AND NOT Y

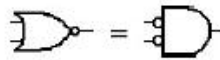
$$\overline{(X+Y)} = \overline{X} \cdot \overline{Y}$$

Not of the quantity, X OR Y is equal to  
NOT X AND NOT Y



X	Y	X+Y	$\overline{X+Y}$	X	Y	$\overline{X}$	$\overline{Y}$	$\overline{X} \cdot \overline{Y}$
0	0	0	1	0	0	1	1	1
0	1	1	0	0	1	1	0	0
1	0	1	0	1	0	0	1	0
1	1	1	0	1	1	0	0	0

Simplificamente



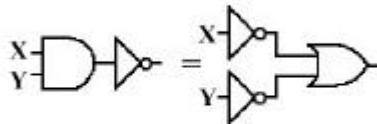
O teorema de DeMorgan é provavelmente a mais importante das identidades que não são imediatamente conhecidas. Aqui mostramos que ele é verdadeiro.

Observe que o teorema de DeMorgan torna concreto o conceito de que existem várias formas de atingir a mesma tabela verdade. Na verdade, demonstraremos mais adiante que toda lógica pode ser criada somente com portas NAND (embora esse normalmente não seja o método mais conveniente). Observe também que o círculo que inverte a entrada ou saída de um dispositivo pode tomar o lugar de um inversor.

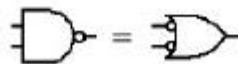
## Teorema de DeMorgan 2

Uma segunda versão é

$$\overline{X \cdot Y} = \overline{X} + \overline{Y}$$



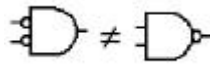
ou



Outras versões do teorema de DeMorgan, dessa vez levando um circuito tipo AND para OR. Já que toda lógica pode ser criada com NANDs e NANDs podem ser mapeados para NORs, então toda lógica também deve poder ser construída apenas a partir de NORs.

**Problema:**

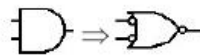
Convença-se de que



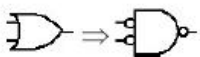
Escreva a tabela verdade disso e convença-se de que você não pode simplesmente inverter todas as entradas e saídas para ter a mesma ação.

## Avanço de Inversor

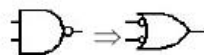
O teorema de DeMorgan afirma



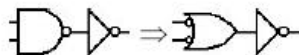
ou



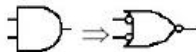
Obs.: quando citamos o teorema de DeMorgan, ele afirmava  $\overline{A \cdot B} = \overline{A} + \overline{B}$



Agora adicionamos um NOT para cada saída:



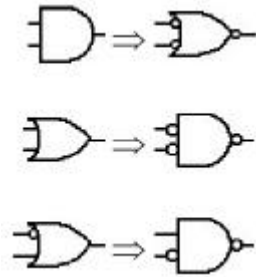
ou



Agora vemos uma afirmação muito mais ampla do teorema de DeMorgan.

## Avanço de Inversor 2

- 1.) troque AND para OR  
ou OR para AND.
- 2.) inverta todas as entradas e saídas.




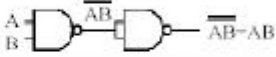



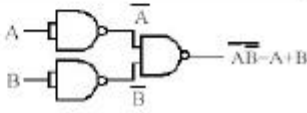


Isso não parece tão elegante ou matemático quanto o teorema de DeMorgan, mas cobre uma gama muito maior de exemplos.

## Teorema de DeMorgan Generalizado


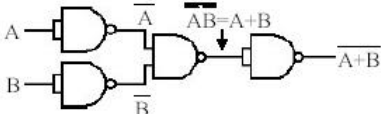

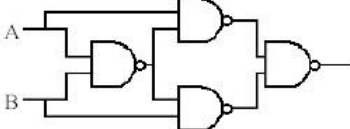

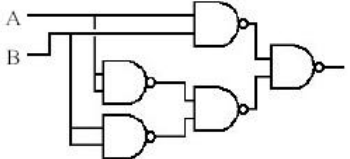
$$\begin{array}{l} \overline{A \cdot B \cdot C} = \overline{A} + \overline{B} + \overline{C} \\ \overline{A + B + C} = \overline{A} \cdot \overline{B} \cdot \overline{C} \end{array} \quad \Bigg|$$

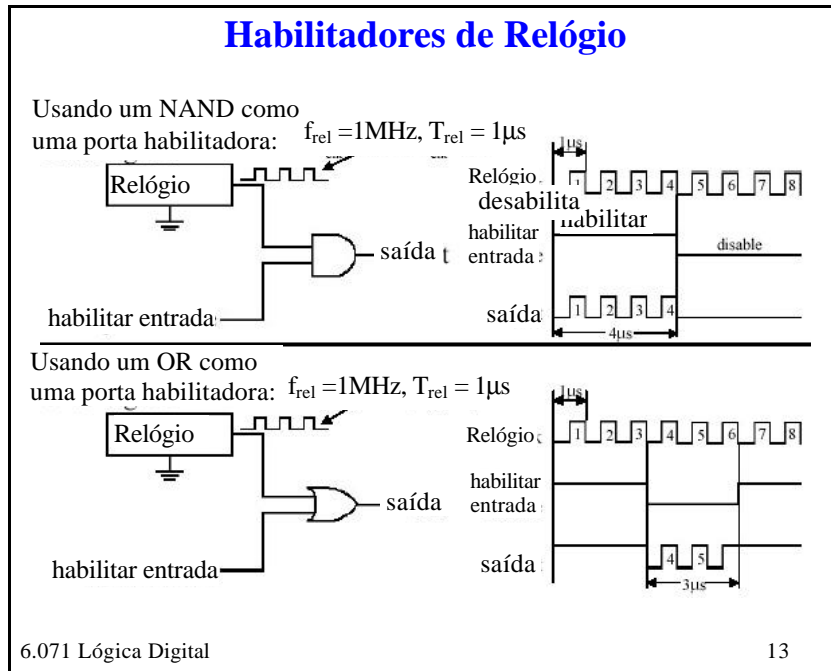
A coisa toda pode ser generalizada para qualquer número de entradas e sempre mantém a mesma estrutura.

Porta Lógica	Equivalente NAND
NÃO 	
E 	
NAND 	
OU 	

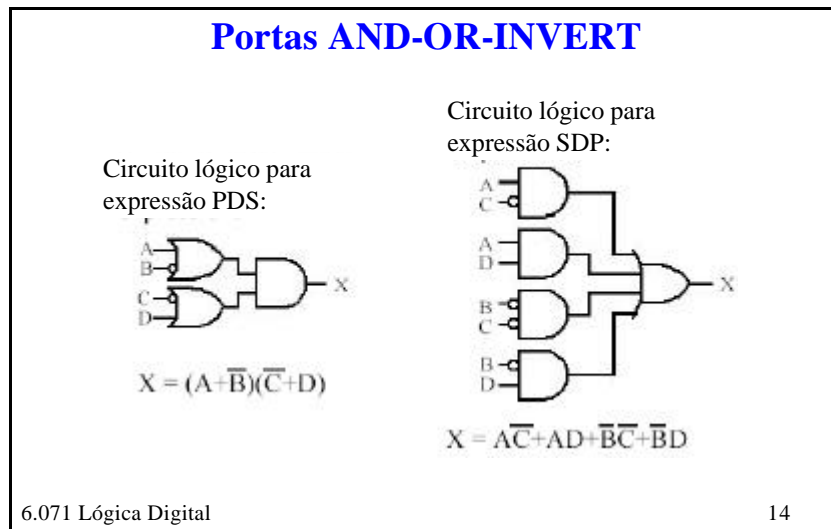
Como dissemos, toda lógica pode ser escrita em termos de NANDs, e aqui estão alguns exemplos. Observe que, em alguns casos, as duas entradas do NAND estão ligadas para formar um inversor.

### Circuitos equivalentes NANDs 2

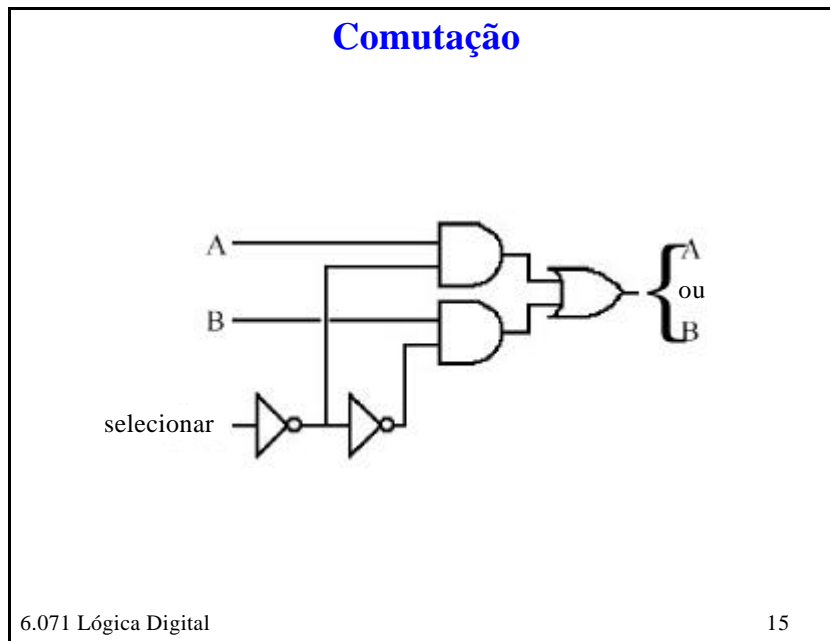
Porta Lógica	Equivalente NAND
 <p>NOR</p>	 <p><math>\overline{AB} = A+B</math></p>
 <p>XOR</p>	
 <p>XNOR</p>	



Um dos vários usos da lógica digital é permitir que um sinal seja transmitido. No caso, o relógio é o sinal e AND ou OR atuam para controlar se ele é transmitido. Observe as diferentes ações e os estados de saída quando o dispositivo é desabilitado.



Você pode construir diretamente circuitos digitais a partir da lógica booleana. Os dois circuitos são a mesma coisa, o esquerdo escrito com um produto de somas (PDS) e o direito como uma soma de produtos (SDP). Existem também abordagens para simplificar uma rede (incluindo pacotes de software).

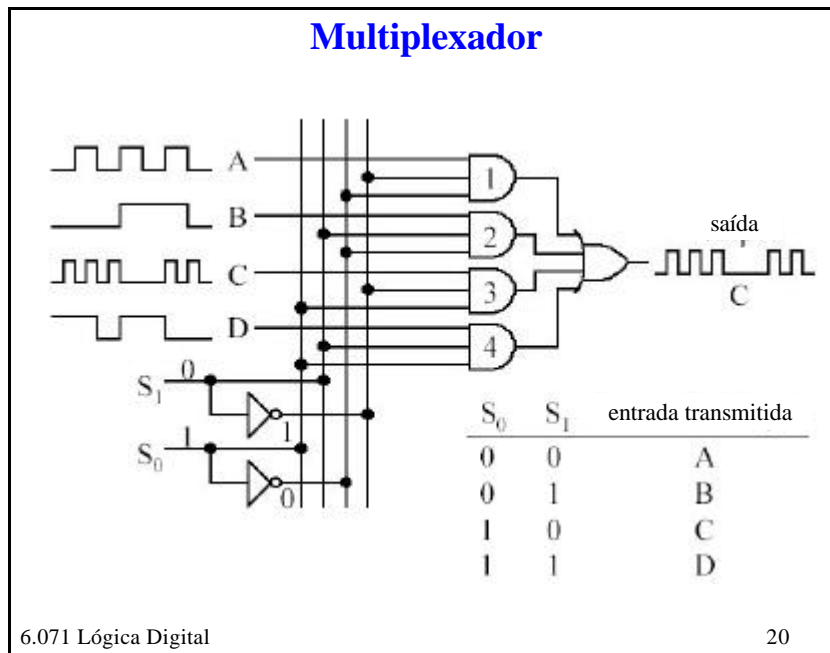


Isso mostra a ação simples de um multiplexador, ele toma duas entradas e comuta a saída entre elas.

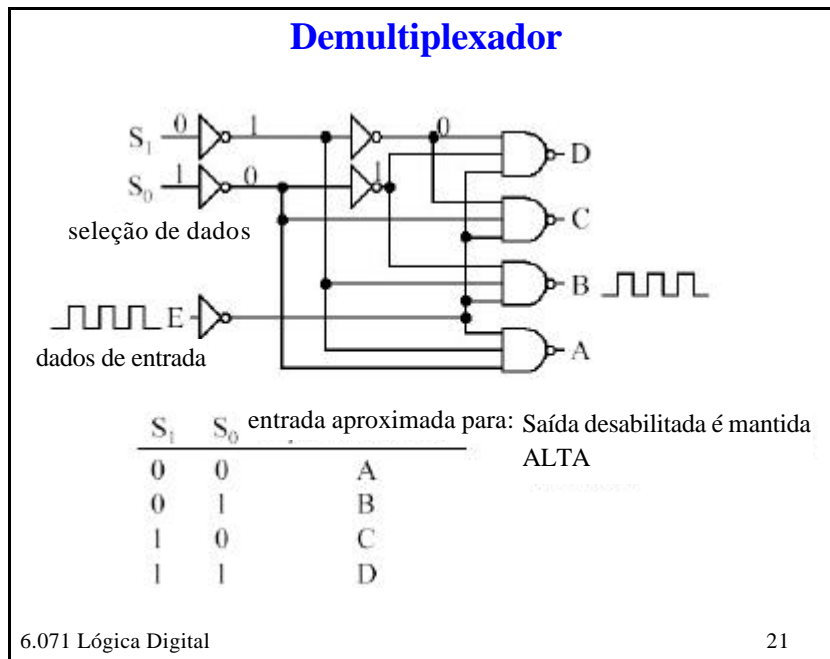
22a 16-19

Entre no site do fabricante para obter um manual dos produtos. Favor seguir estas etapas:

1. Vá para o site de Fairchild Semiconductor: <http://www.fairchildsemi.com/>
2. Veja as condições de uso do site, visitando o link “Site Terms and Conditions” da página inicial, ou entrando neste link: <http://www.fairchildsemi.com/legal/index.html>
3. Volte à página inicial.
4. Na caixa de busca, digite o número do produto DM74LS157 ou DM74LS158, selecione “Product Folders and Datasheets” e clique em “go”. Você procura o manual para Multiplexadores/Seletores de Dados Quad. 2 Linhas para 1 Linha (*Quad 2-Line to 1-Line Data Selectors/Multiplexers*).
5. Várias opções serão apresentadas a você (por exemplo, download PDF ou email). Selecione o meio pelo qual você gostaria de receber o manual.

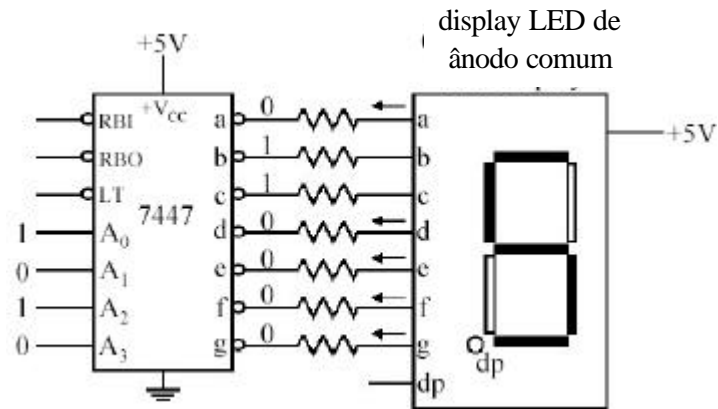


O multiplexador pode ser expandido para muito mais linhas. Observe que, nesse caso, cada porta AND foi expandida para três entradas, de forma que a codificação completa pode aparecer em cada uma. Nós precisaríamos acrescentar mais uma entrada para cada AND para cada aumento à segunda potência no número de entradas. Como construir o mesmo circuito usando os ANDs apenas como habilitadores?



Evidentemente, a ação oposta também pode ser implementada. Um demultiplexador envia o sinal para uma de várias linhas.

### Decodificador BCD para 7 segmentos (LED)



Outro chip complexo, nesse caso projetado para controlar um display numérico LED.

22a 23-26

Entre no site do fabricante para obter o manual dos produtos. Favor seguir estas etapas:

1. Vá para o site de Fairchild Semiconductor: <http://www.fairchildsemi.com/>
2. Veja as condições de uso do site, visitando o link “Site Terms and Conditions” da página inicial, ou entrando neste link: <http://www.fairchildsemi.com/legal/index.html>
3. Volte à página inicial.
4. Na caixa de busca, digite o número do produto DM7446A ou DM7447A, selecione “Product Folders and Datasheets” e clique em “go”. Você procura o manual para Drivers/Decodificadores BCD para 7 Segmentos (*BCD to 7-Segment Decoders/Drivers*).
5. Várias opções serão apresentadas a você (por exemplo, download PDF ou email). Selecione o meio pelo qual você gostaria de receber o manual.