



portas lógicas em sistemas digitais
[álgebra de Boole]

sistemas digitais

“Um sistema digital é um sistema matemático que define informações como valores numéricos. Dessa forma, é possível definir operações digitais como cálculos matemáticos. Comumente trabalhamos com valores numéricos na base decimal, mas um sistema digital trabalha no sistema binário, onde cada dígito possui apenas 2 valores possíveis. Esses dois valores são definidos como “níveis lógicos” e adota-se o valor de 0 (zero) ou 1 (um) apenas.

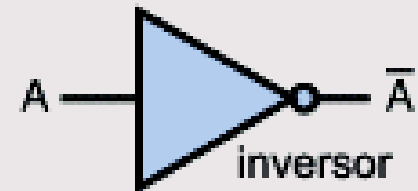
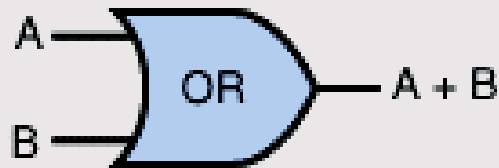
Transportando esse sistema para um sistema eletrônico, é necessário apresentar esses dois valores como sinais elétricos. Para tanto, podemos entendê-los como:

- Ligado ou desligado;
- Nível alto ou nível baixo;
- Alimentado ou em zero;
- VCC ou Terra.

sistemas digitais

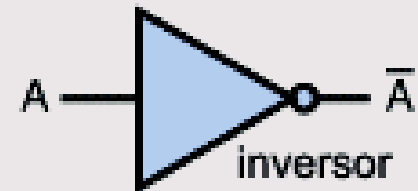
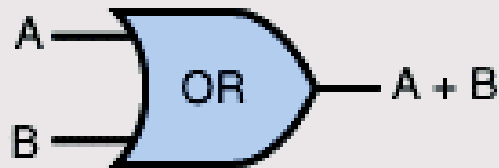
As operações observáveis para esses níveis lógicos são definidas como **operações lógicas**. Todas as possíveis operações lógicas são baseadas em apenas 3 **OPERAÇÕES PRIMÁRIAS**, que são:

- Produto lógico
- Soma lógica
- Inversão



Álgebra de Boole(?)

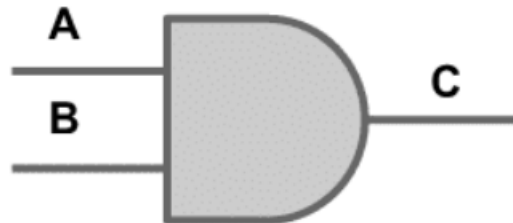
“É um ramo da matemática e da lógica que trata das operações sobre conjuntos finitos. Foi desenvolvida por George Boole no século XIX como uma formalização matemática da lógica. A álgebra de Boole é fundamental para a teoria dos circuitos digitais e para a computação, pois fornece uma estrutura para representar e manipular sistemas lógicos. Ela opera em variáveis que assumem apenas dois valores, tipicamente "verdadeiro" (1) e "falso" (0), e define operações como AND, OR e NOT, que correspondem às operações lógicas fundamentais. **Essas operações podem ser combinadas para criar expressões lógicas complexas** e são a base para o projeto e análise de circuitos digitais, bem como para a lógica de programação em computação”.



Porta lógica AND (E)

As portas lógicas E (AND) utilizam-se do operador de produto lógico. A saída é igual a 1 se todas as entradas for 1. A saída é igual a zero se ao menos uma entrada for 0, se todas entradas não forem 1.

PORTA E (AND)

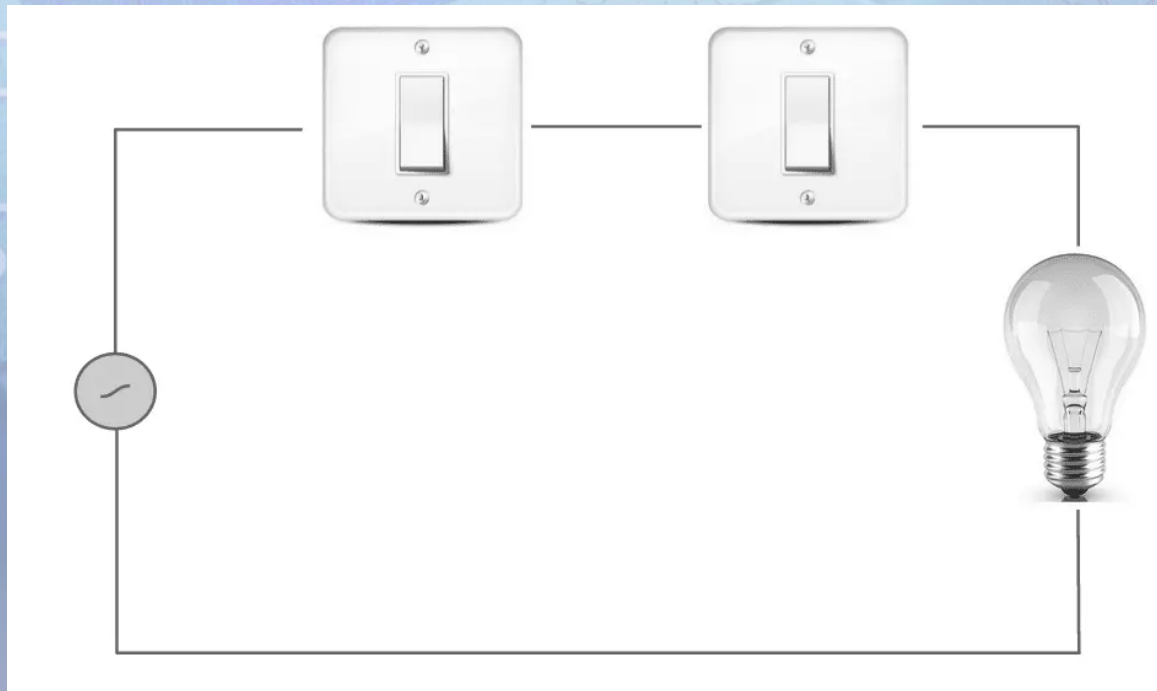


$$C = A \cdot B$$

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

Porta lógica AND (E)

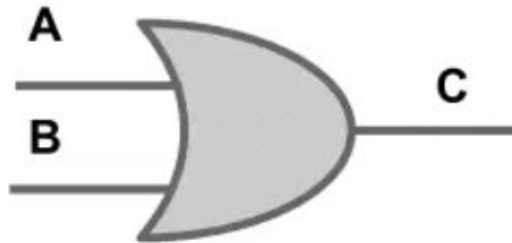
Para se explicar o funcionamento da porta E (AND) pode-se fazer um paralelo com um circuito com interruptores, como na figura abaixo. Para que a lâmpada acenda é preciso que os dois interruptores estejam ligados.



Porta lógica OR (OU)

A porta lógica OU (OR) utiliza-se do operador de soma lógica. A saída é igual a 1 se pelo menos uma das entradas for 1. A saída é igual a zero se nenhuma entrada for 1, todas forem zero.

PORTA OU (OR)

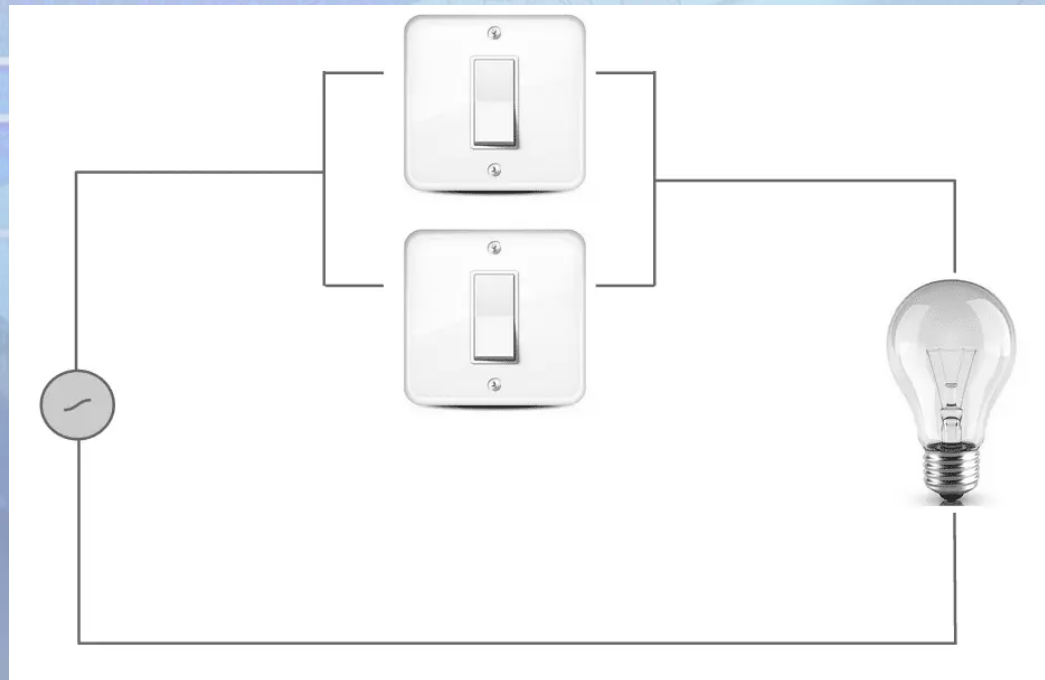


$C=A+B$

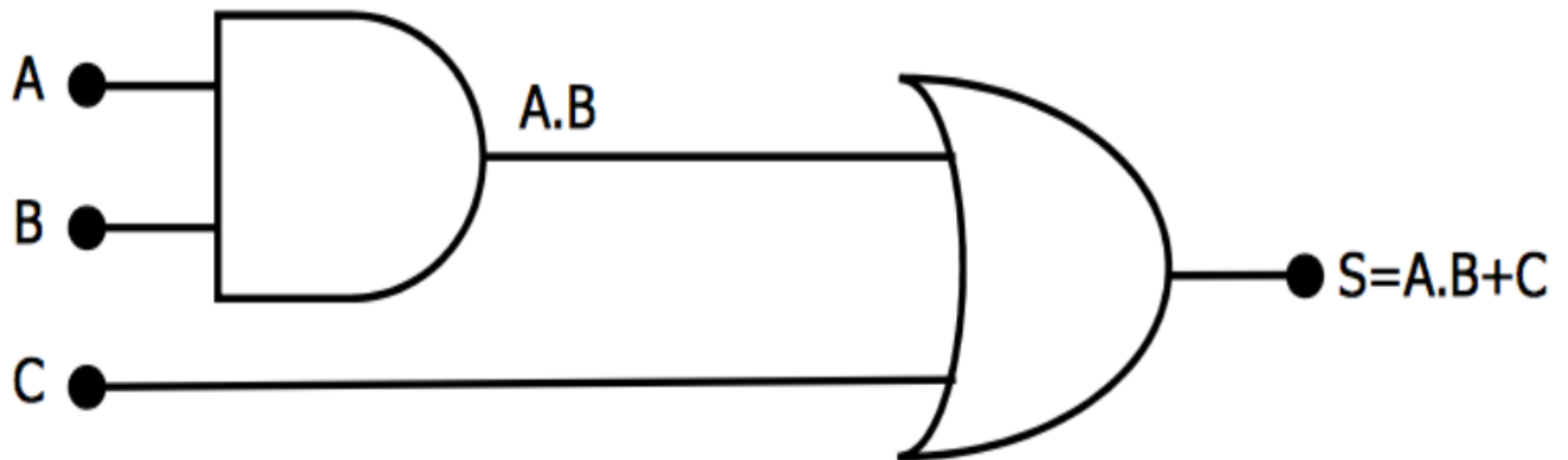
A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

Porta lógica OR (OU)

Uma forma simples de se entender o funcionamento da porta OU (OR) é pensar em um circuito com interruptores em paralelo, como na figura abaixo. Para que a lâmpada acenda é preciso que um dos dois interruptores esteja ligado.

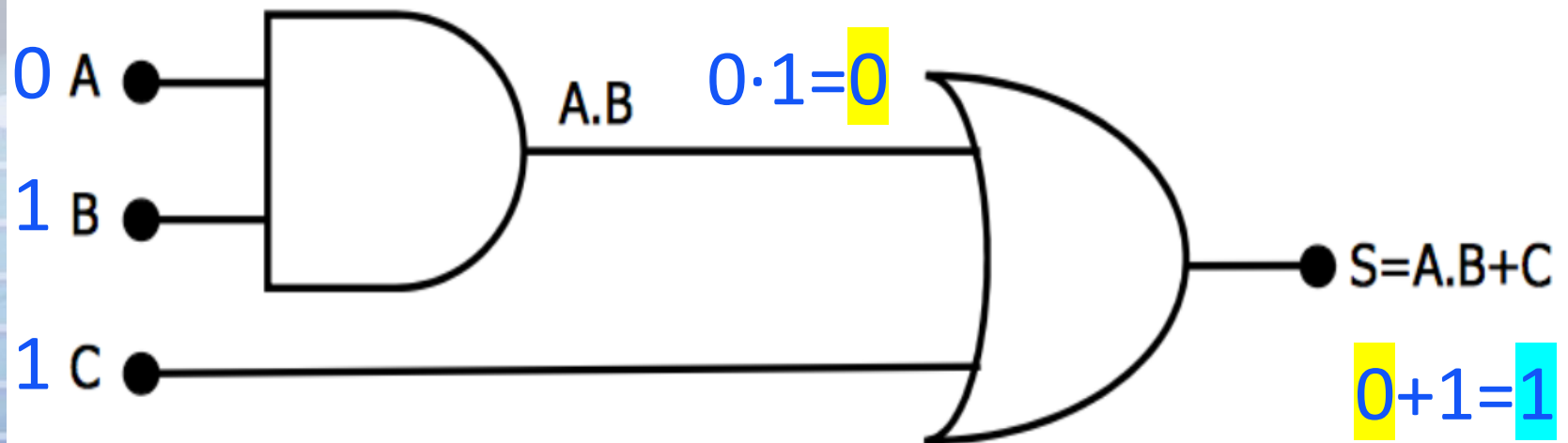


EXEMPLO 1



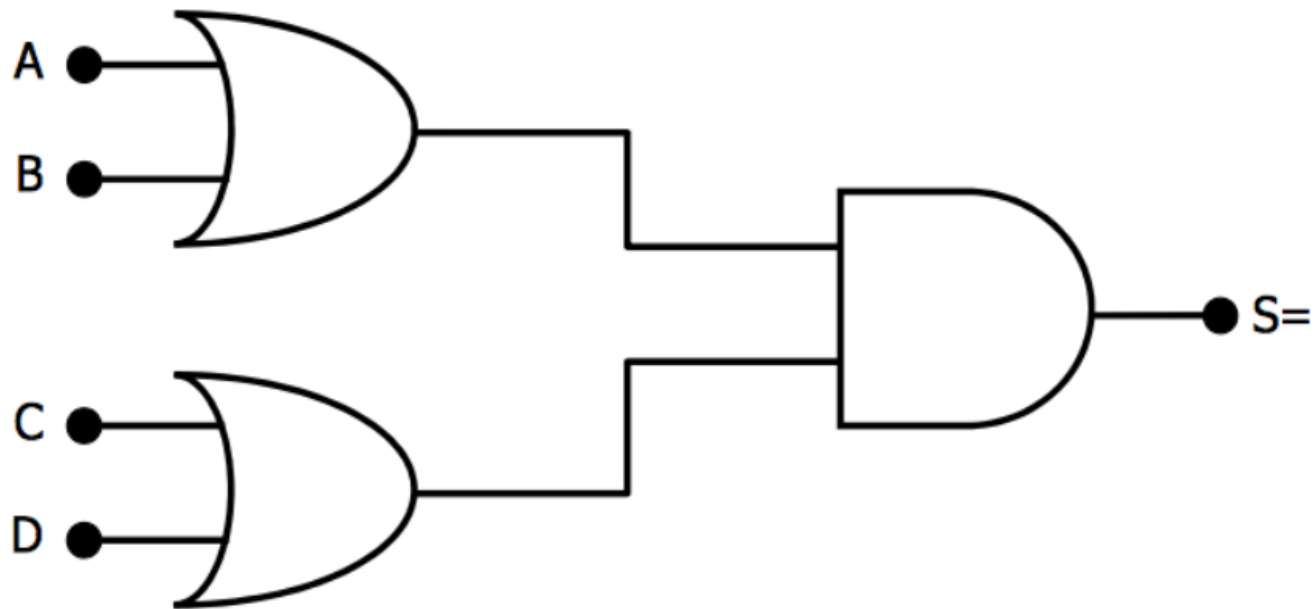
$$A=0 \quad B=1 \quad C=1$$

EXEMPLO 1



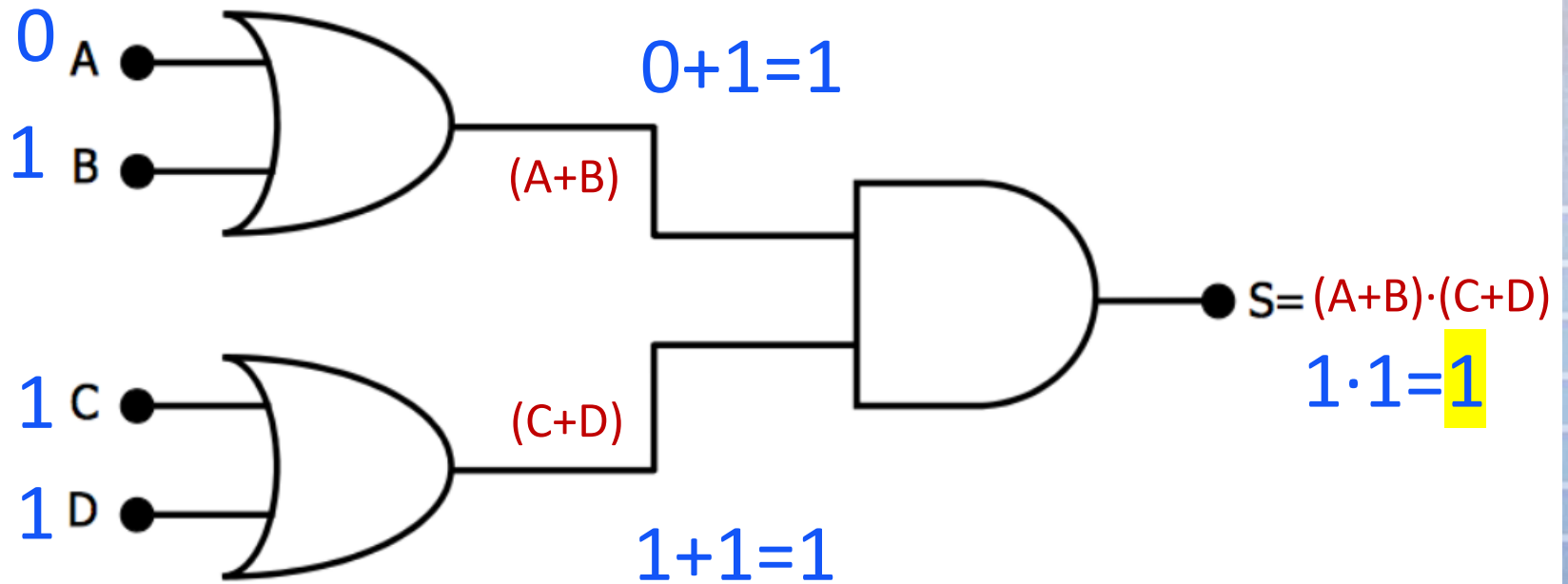
$$A=0 \quad B=1 \quad C=1$$

EXEMPLO 2



$A = 0$ $B = 1$ $C = 1$ $D = 1$

EXEMPLO 2



$$A = 0 \quad B = 1 \quad C = 1 \quad D = 1$$

Porta lógica NOT (NÃO)

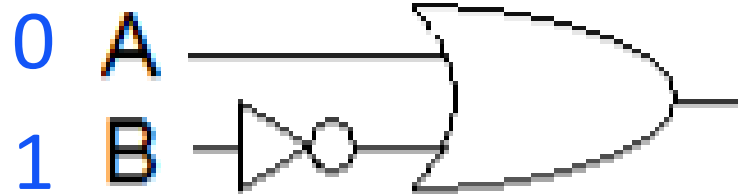
A porta NÃO ou inversora (NOT) utiliza o operador de inversão. Para um determinado valor na entrada, a saída possui um valor contrário ao da entrada. Se a entrada for 1, a saída será 0. Se a entrada for 0, a saída será 1. Ou seja, para um valor na entrada a saída será seu complemento, ou o inverso do valor na entrada.

PORTA NÃO (NOT)

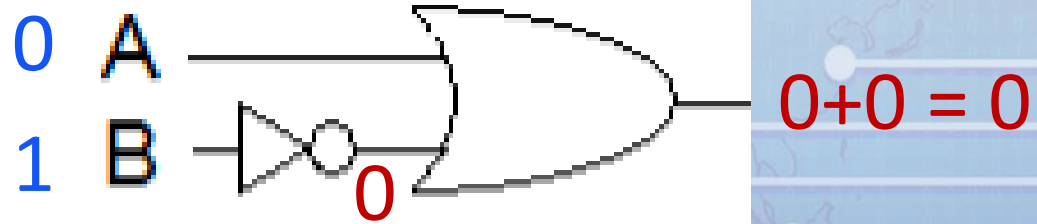


A	\bar{A}
0	1
1	0



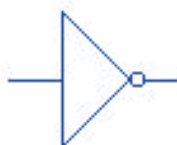
EXEMPLO 3



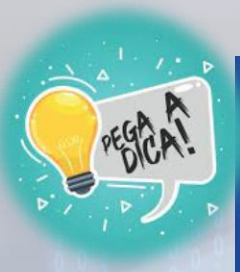
EXEMPLO 3



Portas lógicas em sistemas digitais



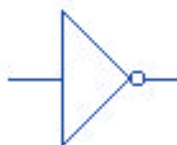
BLOCOS LÓGICOS BÁSICOS																			
Porta	Símbolo usual	Tabela Verdade	Função Lógica	Expressão															
E AND		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	S	0	0	0	0	1	0	1	0	0	1	1	1	Função E: assume 1 quando todas as variáveis forem 1 e 0 nos outros casos.	$S = A \cdot B$
A	B	S																	
0	0	0																	
0	1	0																	
1	0	0																	
1	1	1																	
OU OR		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	S	0	0	0	0	1	1	1	0	1	1	1	1	Função OU: assume 0 quando todas as variáveis forem 0 e 1 nos outros casos.	$S = A + B$
A	B	S																	
0	0	0																	
0	1	1																	
1	0	1																	
1	1	1																	
NÃO NOT		<table><tr><th>A</th><th>S</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	A	S	0	1	1	0	Função NOT: inverte a variável aplicada a sua entrada.	$S = \bar{A}$									
A	S																		
0	1																		
1	0																		

Portas lógicas em sistemas digitais



Falso

Verdadeiro

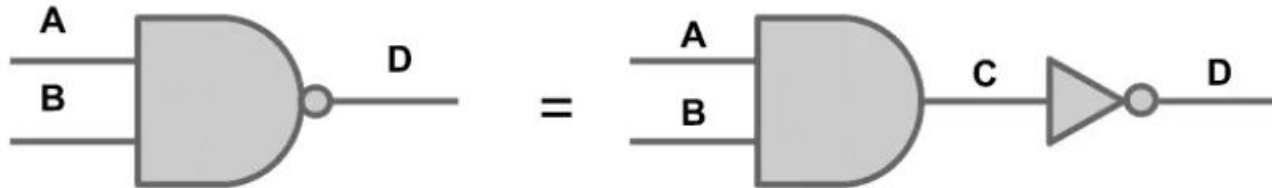
BLOCOS LÓGICOS BÁSICOS																			
Porta	Símbolo usual	Tabela Verdade	Função Lógica	Expressão															
E AND		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	S	0	0	0	0	1	0	1	0	0	1	1	1	Função E: assume 1 quando todas as variáveis forem 1 e 0 nos outros casos.	$S = A \cdot B$
A	B	S																	
0	0	0																	
0	1	0																	
1	0	0																	
1	1	1																	
OU OR		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	S	0	0	0	0	1	1	1	0	1	1	1	1	Função OU: assume 0 quando todas as variáveis forem 0 e 1 nos outros casos.	$S = A + B$
A	B	S																	
0	0	0																	
0	1	1																	
1	0	1																	
1	1	1																	
NÃO NOT		<table><tr><th>A</th><th>S</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	A	S	0	1	1	0	Função NOT: inverte a variável aplicada a sua entrada.	$S = \bar{A}$									
A	S																		
0	1																		
1	0																		

Porta lógica NAND (NÃO E)

A porta lógica NÃO E (NAND) utiliza-se do operador de produto lógico e o de inversão. A saída é igual a 0 se todas as entradas for 1. A saída é igual a 1 se ao menos uma entrada for 0, se todas entradas não forem 1.

PORTA NÃO E (NAND)

$$D = \overline{A \cdot B}$$



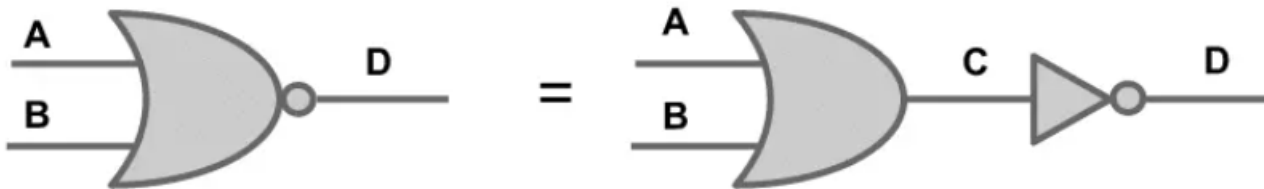
A	B	C	D
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

Porta lógica NOR (NÃO OU)

A porta lógica NÃO OU (NOR) utiliza-se do operador de soma lógica e o de inversão. A saída é igual a 0 se pelo menos uma das entradas for 1. A saída é igual a 1 se nenhuma entrada for 1, todas forem zero.

PORTA NÃO OU (NOR)

$$D = \overline{A+B}$$



A	B	C	D
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

Porta lógica XOR (OU EXCLUSIVO)

As portas lógicas OU EXCLUSIVO (XOR) utilizam-se do operador de soma lógica, com um círculo. A saída é igual a 0 se as entradas forem iguais. A saída é igual a 1 se as entradas não forem iguais, se uma delas diferirem das outras.

PORTA OU EXCLUSIVO (XOR) $C=A\oplus B$

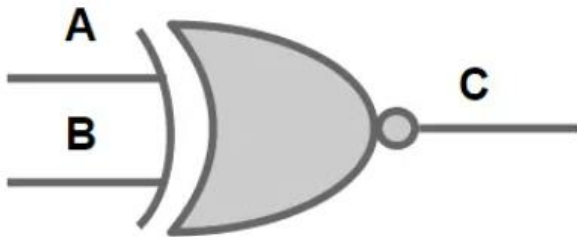


A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

Porta lógica XNOR (NÃO OU EXCLUSIVO)

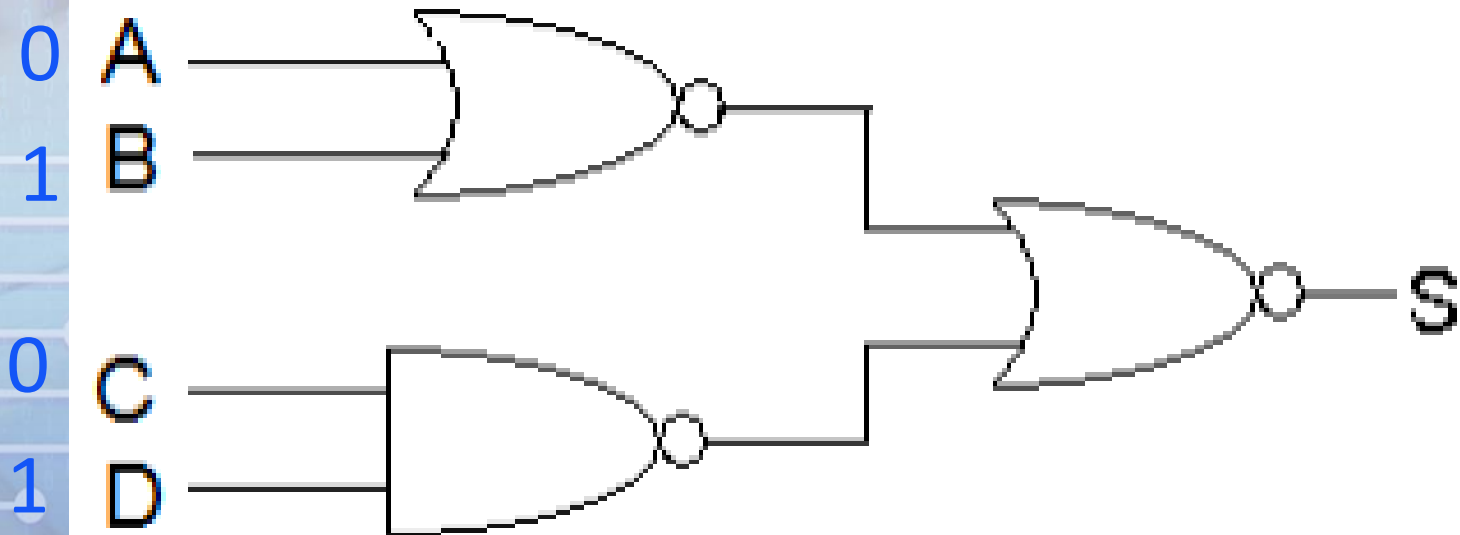
As portas lógicas NÃO OU EXCLUSIVO (XNOR) utilizam-se do operador de soma lógica, com um círculo e o de inversão. Tem as saídas inversas da operação XOR. A saída é igual a 1 se as entradas forem iguais. A saída é igual a 0 se se as entradas não forem iguais, se uma delas diferirem das outras.

PORTA NÃO OU EXCLUSIVO (XNOR) $C = \overline{A \oplus B}$

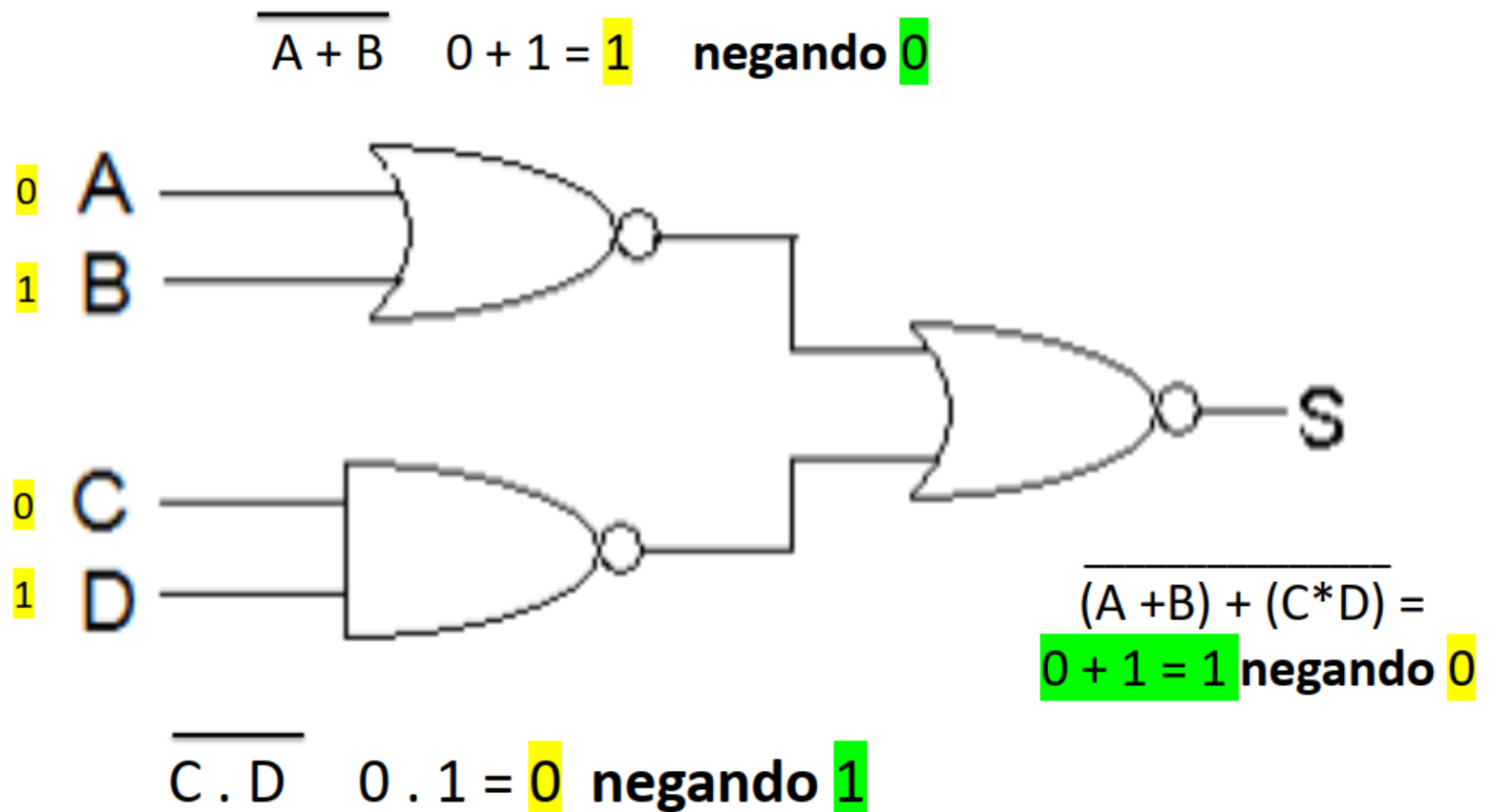


A	B	C
0	0	1
0	1	0
1	0	0
1	1	1

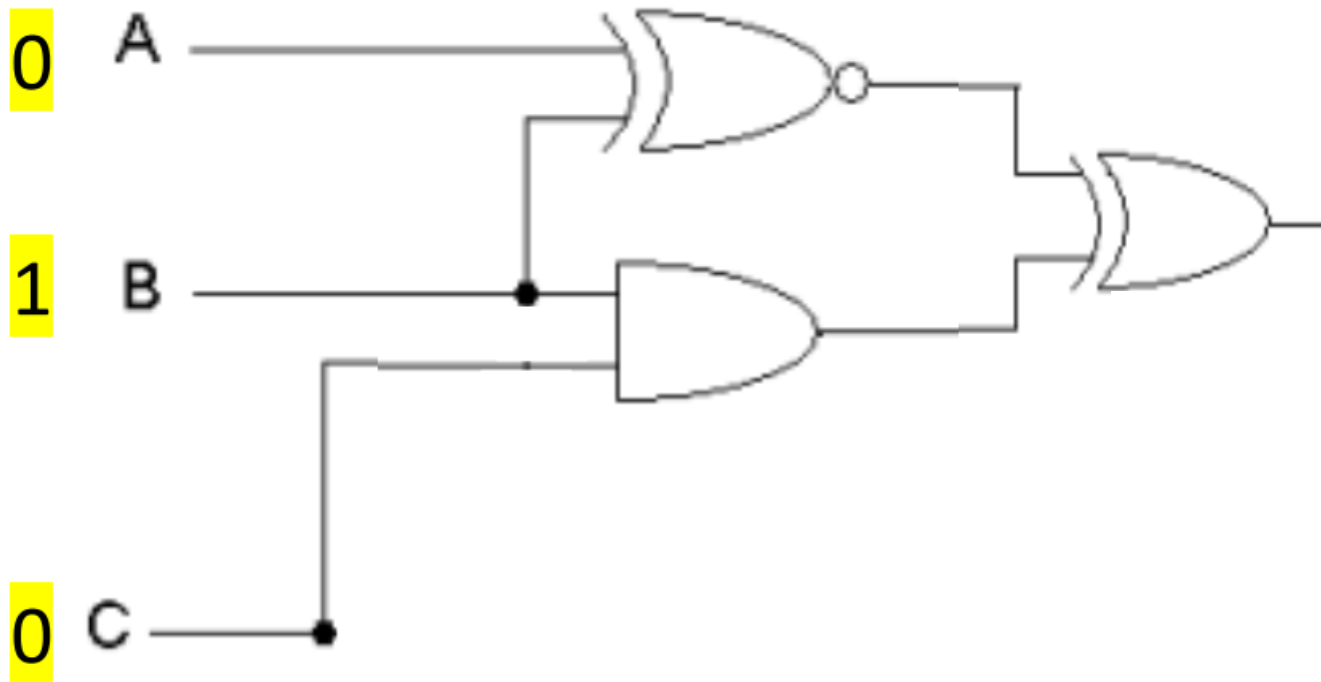
EXEMPLO 4



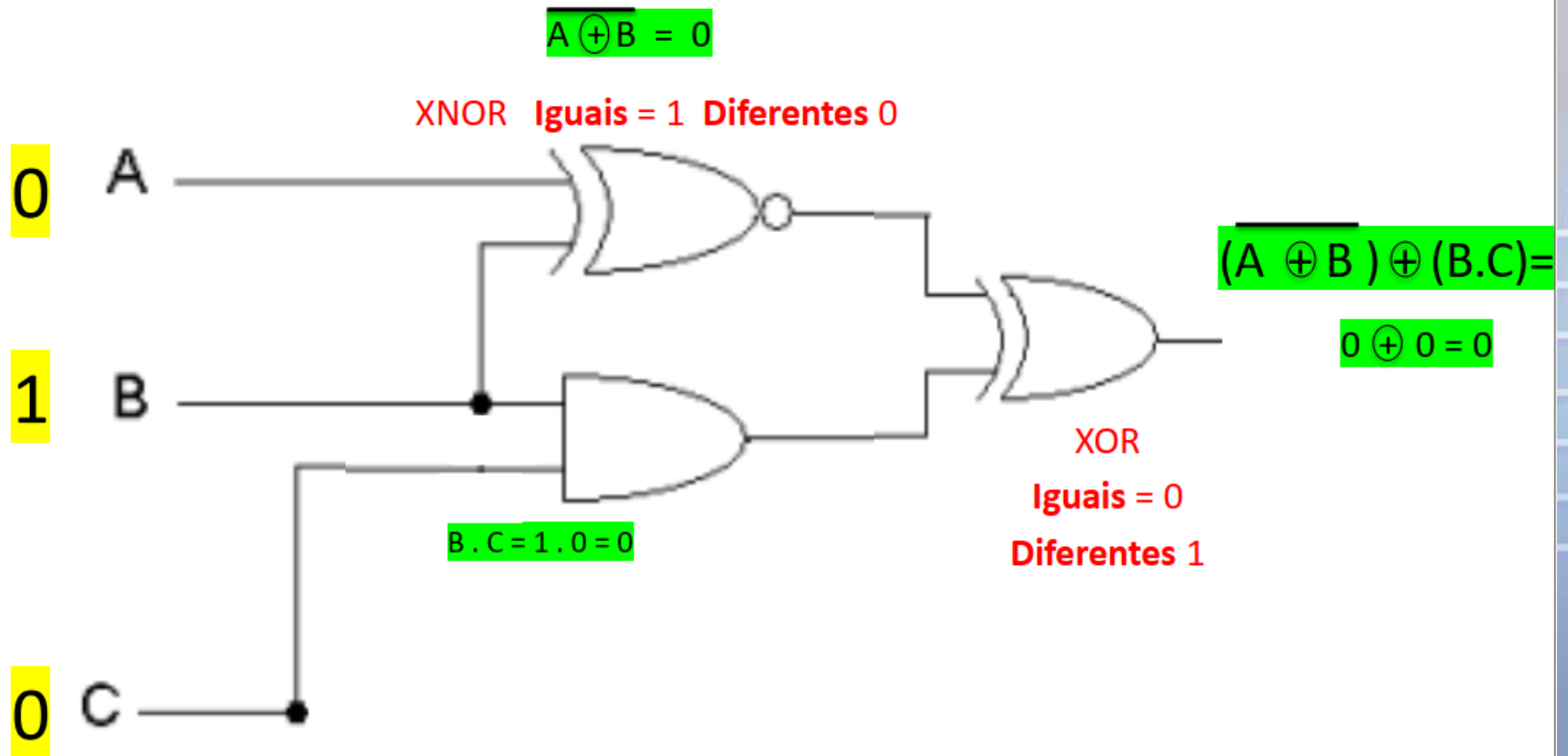
EXEMPLO 4










EXEMPLO 5



EXEMPLO 5






BLOCOS LÓGICOS BÁSICOS

PORTA	Símbolo Usual	Tabela da Verdade	Função Lógica	Expressão															
E AND		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	S	0	0	0	0	1	0	1	0	0	1	1	1	Função E: Assume 1 quando todas as variáveis forem 1 e 0 nos outros casos.	$S=A.B$
A	B	S																	
0	0	0																	
0	1	0																	
1	0	0																	
1	1	1																	
OU OR		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	S	0	0	0	0	1	1	1	0	1	1	1	1	Função E: Assume 0 quando todas as variáveis forem 0 e 1 nos outros casos.	$S=A+B$
A	B	S																	
0	0	0																	
0	1	1																	
1	0	1																	
1	1	1																	
NÃO NOT		<table><tr><th>A</th><th>S</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	A	S	0	1	1	0	Função NÃO: Inverte a variável aplicada à sua entrada.	$S=\bar{A}$									
A	S																		
0	1																		
1	0																		
NE NAND		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	S	0	0	1	0	1	1	1	0	1	1	1	0	Função NE: Inverso da função E.	$S=(\overline{A.B})$
A	B	S																	
0	0	1																	
0	1	1																	
1	0	1																	
1	1	0																	
NOU NOR		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	S	0	0	1	0	1	0	1	0	0	1	1	0	Função NOU: Inverso da função OU.	$S=(\overline{A+B})$
A	B	S																	
0	0	1																	
0	1	0																	
1	0	0																	
1	1	0																	
OU Exclusivo XOR		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	S	0	0	0	0	1	1	1	0	1	1	1	0	Função OU Exclusivo: Assume 1 quando as variáveis assumirem valores diferentes entre si.	$S=A\oplus B$ $S=\bar{A}.B+A.\bar{B}$
A	B	S																	
0	0	0																	
0	1	1																	
1	0	1																	
1	1	0																	
Coincidência XNOR		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	S	0	0	1	0	1	0	1	0	0	1	1	1	Função Coincidência: Assume 1 quando houver coincidência entre os valores das variáveis.	$S=A\odot B$ $S=\bar{A}.\bar{B}+A.B$
A	B	S																	
0	0	1																	
0	1	0																	
1	0	0																	
1	1	1																	







BLOCOS LÓGICOS BÁSICOS

PORTA	Simbolo Usual	Tabela da Verdade	Função Lógica	Expressão															
E AND		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	S	0	0	0	0	1	0	1	0	0	1	1	1	Função E: Assume 1 quando todas as variáveis forem 1 e 0 nos outros casos.	$S=A.B$
A	B	S																	
0	0	0																	
0	1	0																	
1	0	0																	
1	1	1																	
OU OR		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	S	0	0	0	0	1	1	1	0	1	1	1	1	Função E: Assume 0 quando todas as variáveis forem 0 e 1 nos outros casos.	$S=A+B$
A	B	S																	
0	0	0																	
0	1	1																	
1	0	1																	
1	1	1																	
NÃO NOT		<table><tr><th>A</th><th>S</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	A	S	0	1	1	0	Função NÃO: Inverte a variável aplicada à sua entrada.	$S=\overline{A}$									
A	S																		
0	1																		
1	0																		

Falso

Verdadeiro

PORTA	Símbolo Usual	Tabela da Verdade	Função Lógica	Expressão															
NE NAND		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	S	0	0	1	0	1	1	1	0	1	1	1	0	Função NE: Inverso da função E.	$S=(\overline{A.B})$
A	B	S																	
0	0	1																	
0	1	1																	
1	0	1																	
1	1	0																	
NOU NOR		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	S	0	0	1	0	1	0	1	0	0	1	1	0	Função NOU: Inverso da função OU.	$S=(\overline{A+B})$
A	B	S																	
0	0	1																	
0	1	0																	
1	0	0																	
1	1	0																	
OU Exclusivo XOR		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	S	0	0	0	0	1	1	1	0	1	1	1	0	Função OU Exclusivo: Assume 1 quando as variáveis assumirem valores diferentes entre si.	$S=A\oplus B$ $S=\bar{A}.B+A.\bar{B}$
A	B	S																	
0	0	0																	
0	1	1																	
1	0	1																	
1	1	0																	
Coincidência XNOR		<table><tr><th>A</th><th>B</th><th>S</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	S	0	0	1	0	1	0	1	0	0	1	1	1	Função Coincidência: Assume 1 quando houver coincidência entre os valores das variáveis.	$S=\overline{A\oplus B}$ $S=\bar{A}.\bar{B}+A.B$
A	B	S																	
0	0	1																	
0	1	0																	
1	0	0																	
1	1	1																	

Iguals = 0
Diferentes = 1

Iguals = 1
Diferentes = 0

Portas lógicas: Teorema de De Morgan

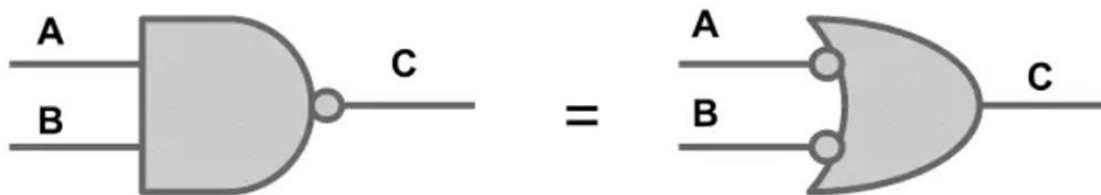
- Uma operação NAND é igual a uma operação OR com todas as entradas invertidas;
- Uma operação NOR é igual a uma operação AND com todas as entradas invertidas.

Com essas duas afirmações podemos fazer diversas simplificações em expressões lógicas, referentes a circuitos digitais.

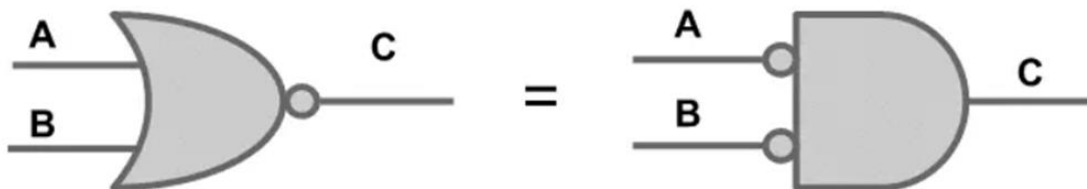
Portas lógicas: Teorema de De Morgan

- Uma operação NAND é igual a uma operação OR com todas as entradas invertidas;
- Uma operação NOR é igual a uma operação AND com todas as entradas invertidas.

$$C = \overline{A \cdot B} = \overline{A} + \overline{B}$$

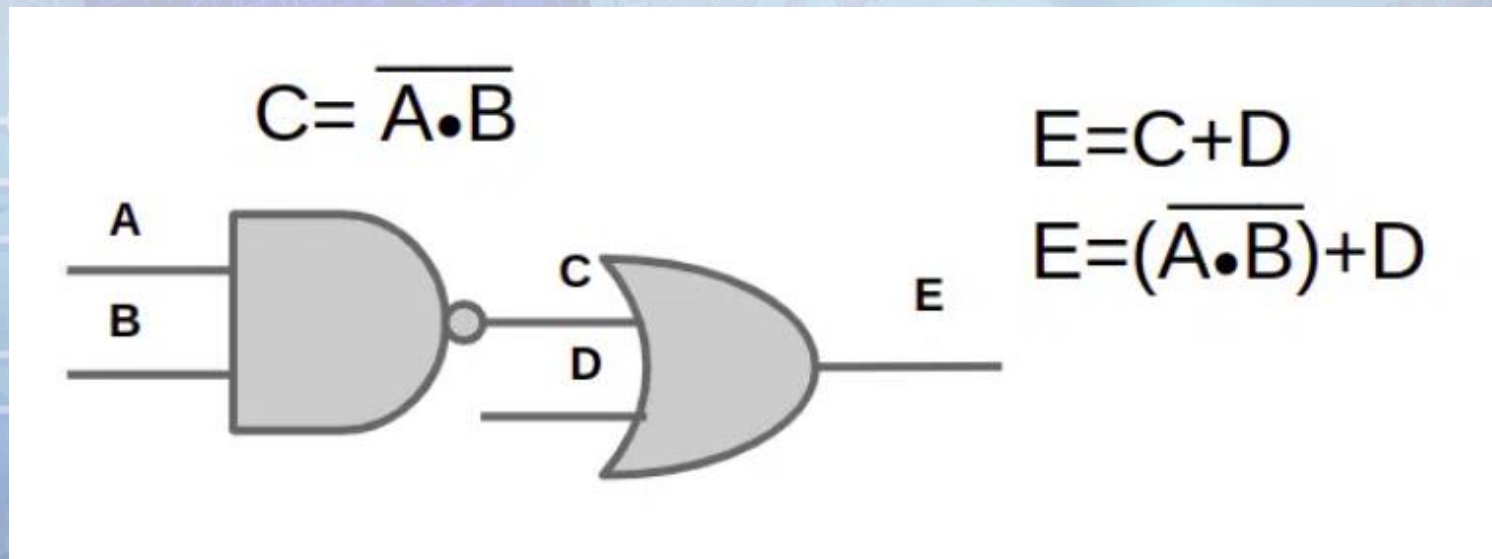


$$C = \overline{\overline{A} + \overline{B}} = \overline{\overline{A}} \cdot \overline{\overline{B}}$$



Associação de portas lógicas

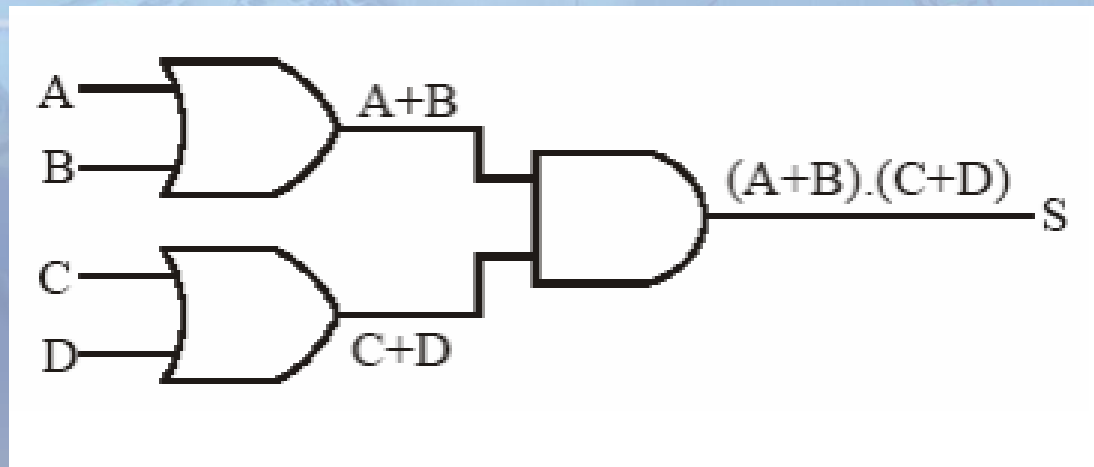
Se utilizarmos expressões booleanas, podemos determinar diretamente a expressão lógica de saída de um circuito.



Associação de portas lógicas

Todo circuito lógico, por mais complexo que seja, é formado pela combinação de portas lógicas básicas.

Todo o circuito lógico executa uma função booleana e, por mais complexo que seja, é formado pela interligação das portas lógicas básicas. Assim, pode-se obter a expressão booleana que é executada por um circuito lógico qualquer.



Associação de portas lógicas

é possível desenhar um circuito lógico que executa uma função booleana qualquer, ou seja, pode-se desenhar um circuito a partir de sua expressão característica

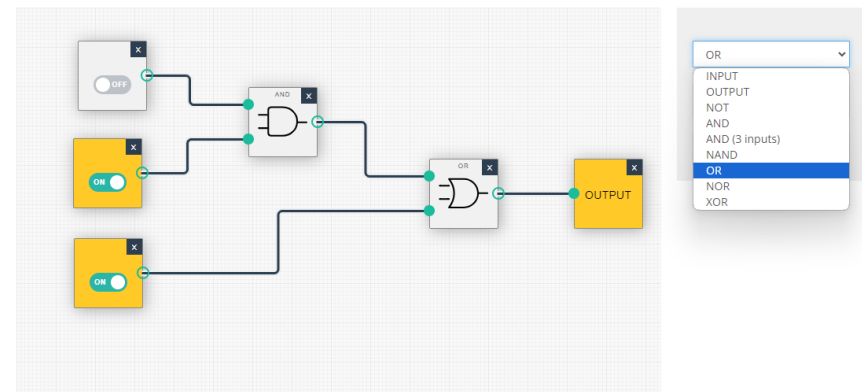
Exemplo
expressão $S = (A+B).C.(B+D)$

Logic Gate Simulator

A free, simple, online logic gate simulator. Investigate the behaviour of AND, OR, NOT, NAND, NOR and XOR gates. Select gates from the dropdown list and click "add node" to add more gates. Drag from the hollow circles to the solid circles to make connections. Right click connections to delete them. See below for more detailed instructions.

Engineering Electronics Logic

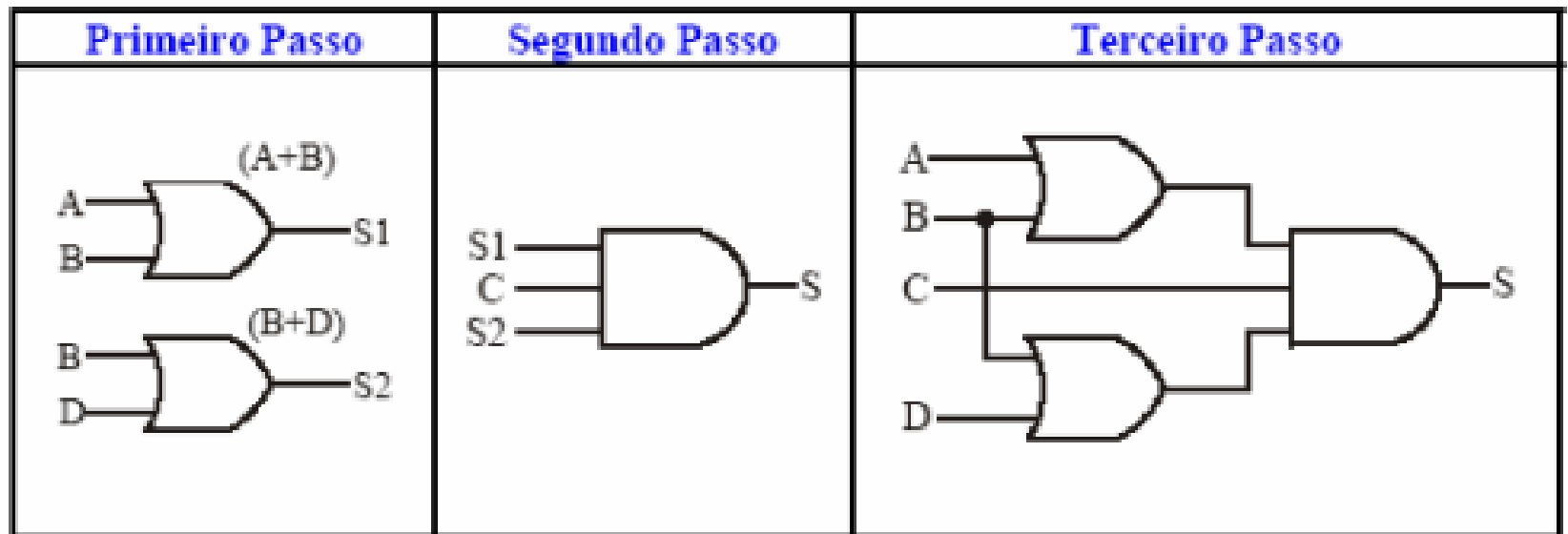
[Share](#) [Tweet](#) [BECOME A PATRON](#) [Support us on Ko-fi](#)



Exemplo

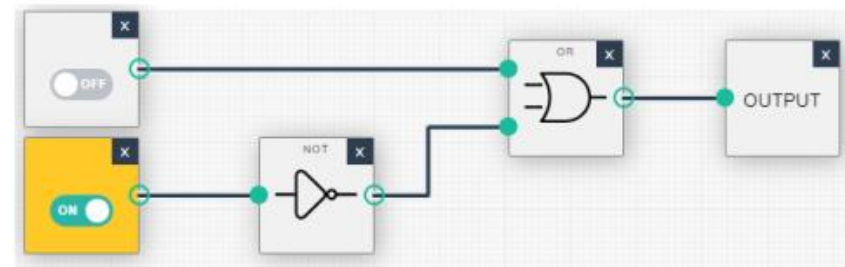
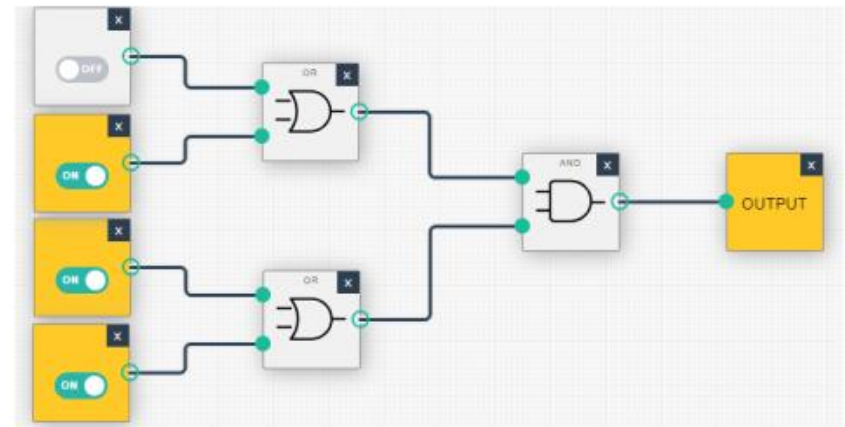
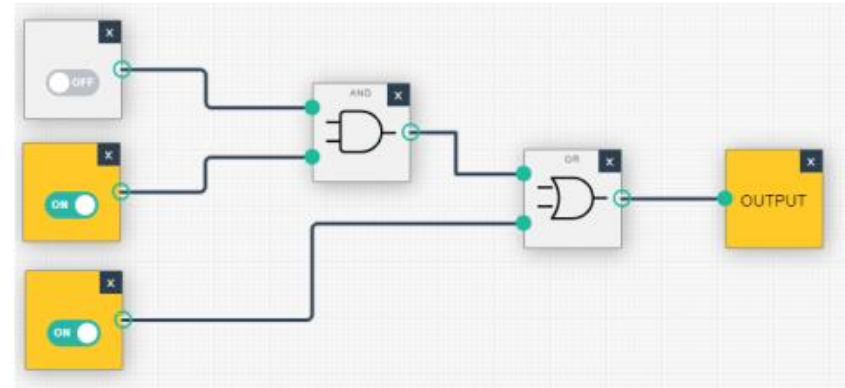
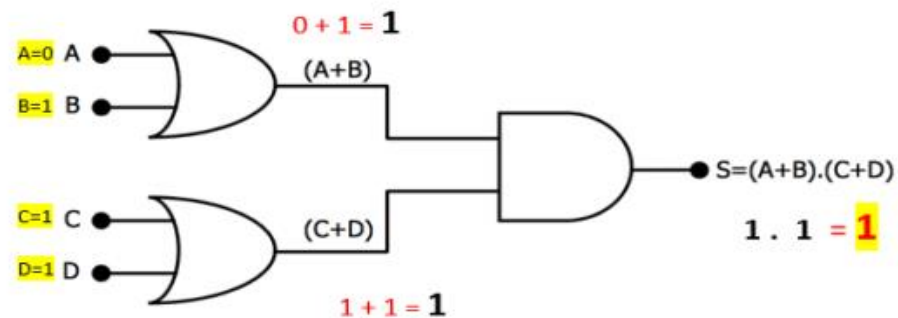
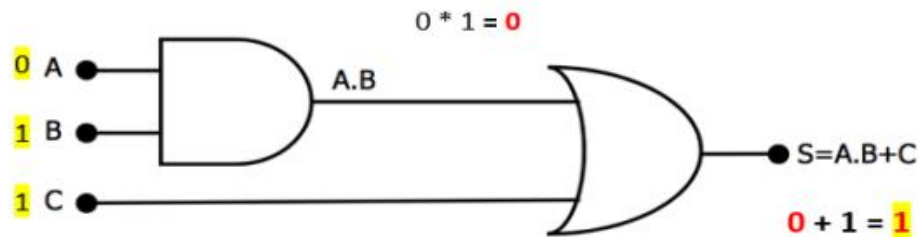
$$\text{expressão } S = (A+B) \cdot C \cdot (B+D)$$

Para o primeiro parêntese tem-se uma soma booleana $A+B$, logo o circuito que o executa será uma porta OU. Para o segundo, tem-se outra soma booleana $B+D$, logo o circuito será uma porta OU. Posteriormente tem-se a multiplicação booleana de dois parênteses juntamente com a variável C , sendo o circuito que executa esta multiplicação uma porta E. Para finalizar, unem-se as respectivas ligações obtendo o circuito completo.



Exemplos

<https://academo.org/demos/logic-gate-simulator/>



PRATICANDO...

1- Esboce (à mão) os circuitos obtidos a partir das seguintes expressões:

$$S = A \cdot B + \overline{C} + \overline{(C \cdot D)}$$

$$S = (A+B) \cdot (C+D)$$

2- Sendo $A=1$, $B=0$, $C=0$, $D=1$, escreva a expressão referente e represente no <https://academo.org/demos/logic-gate-simulator/> o seguinte circuito:

