



INSTITUTO FEDERAL
Santa Catarina

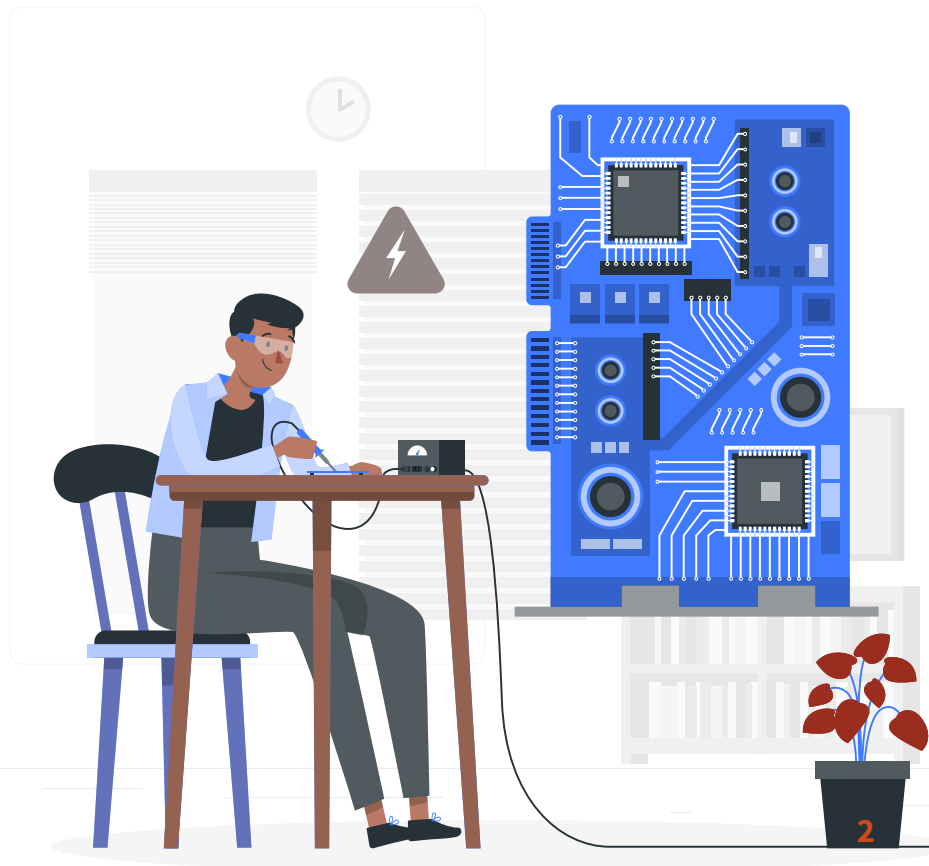
Eletrônica Digital I

- Aula 2 -

Professora: Ma. Luciana Menezes Xavier de Souza
e-mail: luciana.xavier@ifsc.edu.br

Conteúdo

- Revisão;
- Portas Lógicas;
- Álgebra;
- Exercícios.



Sistema de Numeração

- Um dígito binário é chamado de **bit** (*BI*nary *digi*T). O bit mais significativo é chamado de **MSB** (*most significant bit*) e o menos significativo de **LSB** (*least significant bit*).

Definições importantes:

Bit – 1 dígito binário

Nibble – 4 dígitos binários

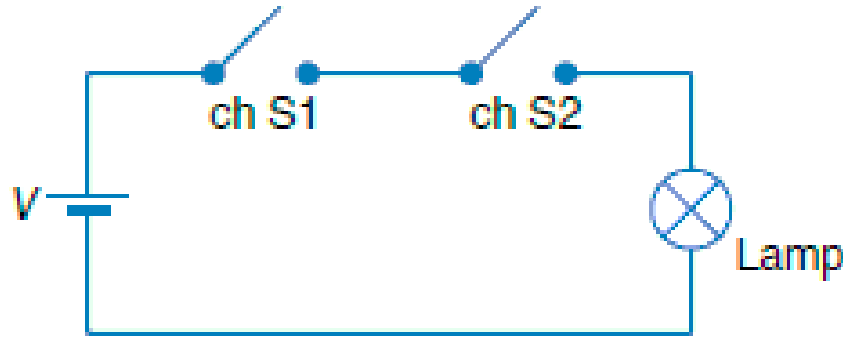
Byte – 8 dígitos binários

Exemplo. Contagem no sistema binário com 4 bits, números na faixa de 0 a 15 ($2^4 - 1$).

Pesos				Número decimal	
$2^3 = 8$	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$		
0	0	0	0	→	0
0	0	0	1	→	1
0	0	1	0		2
0	0	1	1		3
0	1	0	0		4
0	1	0	1		5
0	1	1	0		6
0	1	1	1		7
1	0	0	0		8
1	0	0	1		9
1	0	1	0		10
1	0	1	1		11
1	1	0	0		12
1	1	0	1		13
1	1	1	0	→	14
1	1	1	1	→	15

↑
LSB

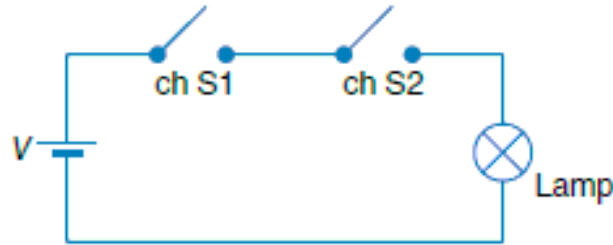
- Circuitos lógicos digitais realizam operações lógicas, pois usam faixas de tensões para representar os bits **0 (chaves aberta)** e **1 (chaves fechada)**.



0 Aberta

1 fechada

Em que condições a lâmpada acenderá?



$S1=S2=0$



CHAVE
ABERTA



$L=0$
LÂMPADA
APAGADA

$S1=S2=1$



CHAVE
FECHADA



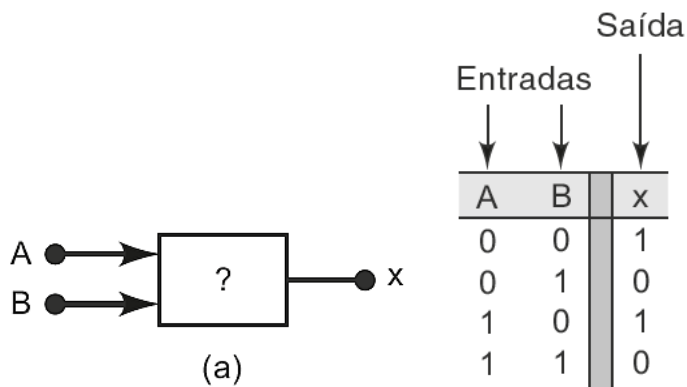
$L=1$
LÂMPADA ACESA

$S1 = 1$ (chave S1 fechada) e $S2 = 0$ (chave S2 aberta) $\rightarrow L = 0$ (lâmp. Apagada)

$S1 = 0$ (chave S1 aberta) e $S2 = 1$ (chave S2 fechada) $\rightarrow L = 0$ (lâmp. Apagada)

Tabela verdade

- Ferramenta para descrever as relações entre as **saídas dos circuitos e suas entradas**, relacionando todas as combinações possíveis de entradas e saídas.
- A tabela possui 2^N linhas para N entradas. **Exemplos:**



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

(b)

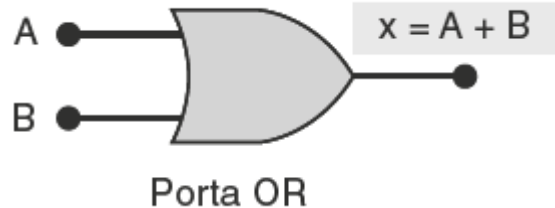
A	B	C	D	x
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

(c)

Exemplo de Tabela Verdade para circuitos (a) Duas; (b) Três e (c) Quatro entradas.

Portas Lógicas

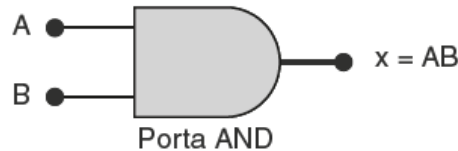
- ⊙ **Operação OR** (símbolo “+”): resulta em nível lógico 1 (verdadeiro) se uma ou mais entradas estiverem em nível lógico 1. A **porta OR** é um circuito de duas ou mais entradas cuja saída é a obtida com a operação OR.



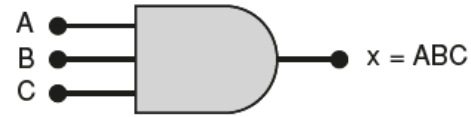
OR		
A	B	$x = A + B$
0	0	0
0	1	1
1	0	1
1	1	1

Porta AND

- © **Operação AND** (símbolo “.”): resulta em nível lógico 1 se todas as entradas estiverem em nível lógico 1. A **porta AND** é um circuito de duas ou mais entradas cuja saída é a combinação das entradas pela operação AND.



AND		
A	B	$x = A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1



A	B	C	$x = ABC$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

RESUMO DAS OPERAÇÕES BOOLEANAS

© As regras para as operações OR, AND e NOT podem ser resumidas como:

<i>OR</i>	<i>AND</i>	<i>NOT</i>
$0 + 0 = 0$	$0 \cdot 0 = 0$	$\overline{0} = 1$
$0 + 1 = 1$	$0 \cdot 1 = 0$	$\overline{1} = 0$
$1 + 0 = 1$	$1 \cdot 0 = 0$	
$1 + 1 = 1$	$1 \cdot 1 = 1$	

Porta NOR

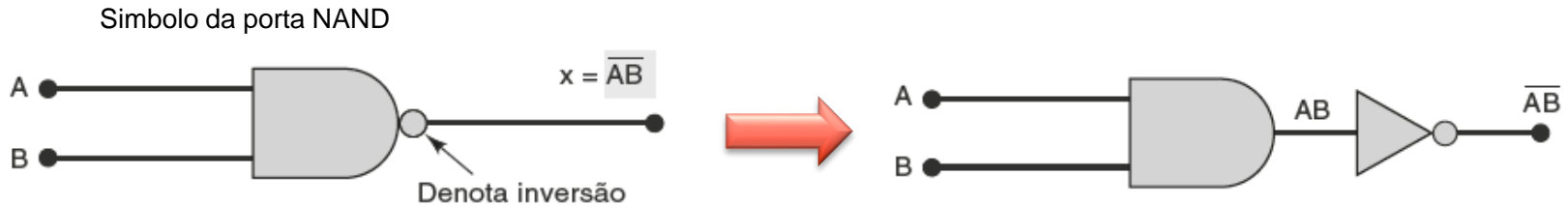
- ⊙ **Operação NOR:** composta por uma porta OR seguida de uma porta NOT.



		OR	NOR
A	B	$A + B$	$\overline{A + B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

Porta NAND

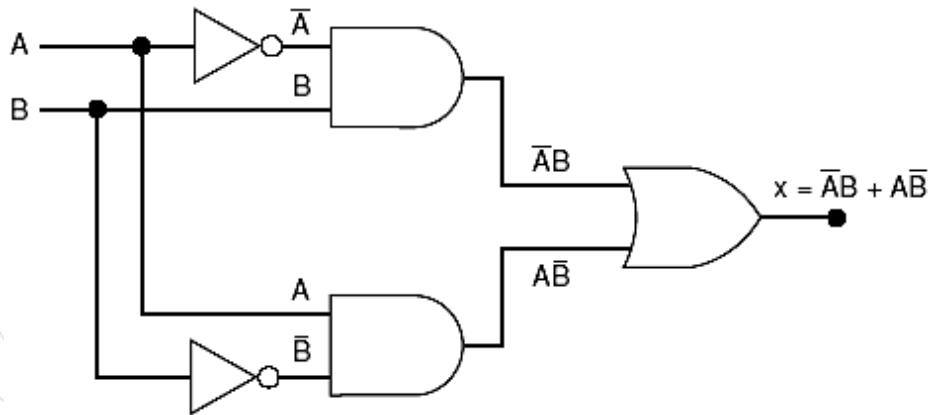
- © **Operação NAND:** composta por uma porta AND seguida de uma porta NOT.



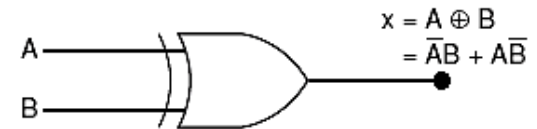
		AND	NAND
A	B	AB	\overline{AB}
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

Porta EX-OR ou XOR

- ⊙ **Porta EX-OR** (ou exclusivo ou XOR): circuito lógico que implementa a expressão algébrica booleana $x = \bar{A}B + A\bar{B}$
- ⊙ Esse circuito produz uma **saída em nível alto** sempre que as **entradas** estiverem em **níveis opostos**.

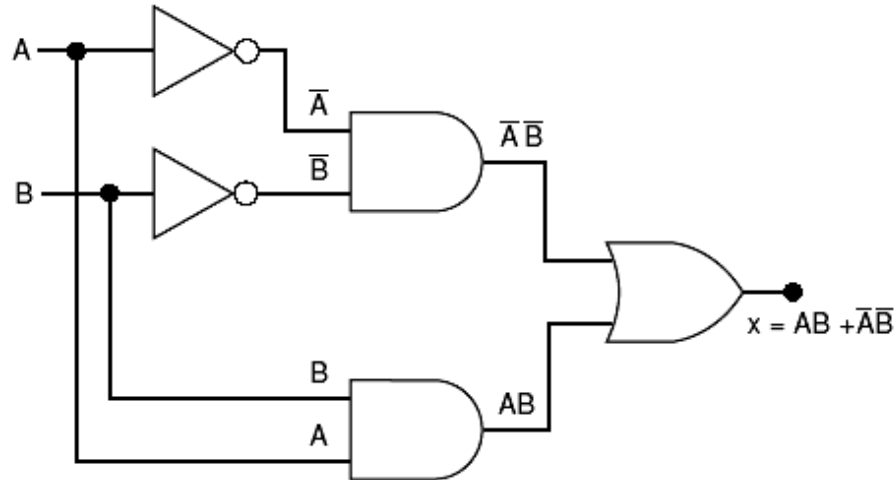


A	B	x
0	0	0
0	1	1
1	0	1
1	1	0

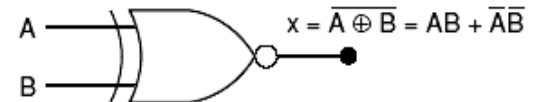


Porta EX-NOR ou XNOR

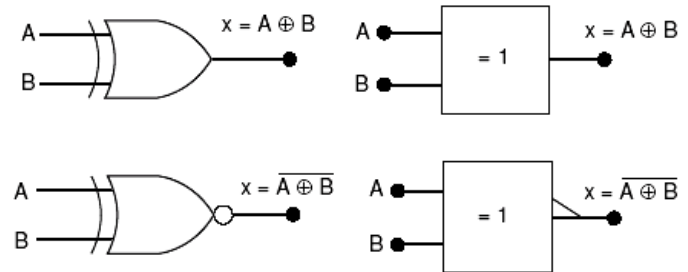
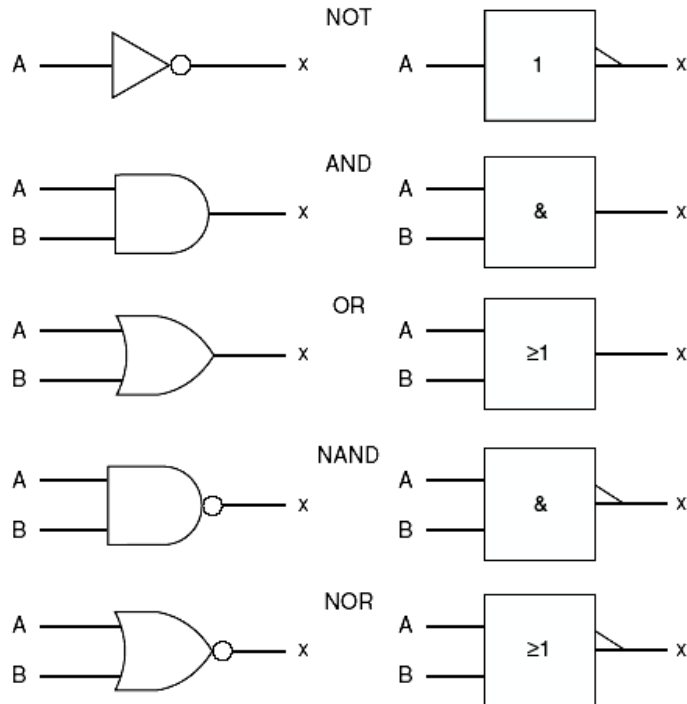
- ⊙ **Operação EX-NOR** (NOR exclusivo (XNOR) ou coincidência): circuito lógico que implementa a expressão algébrica booleana $x = AB + \overline{A}\overline{B}$
- ⊙ Esse circuito produz uma **saída em nível alto** sempre que as **entradas** estiverem em **níveis iguais**.



A	B	x
0	0	1
0	1	0
1	0	0
1	1	1



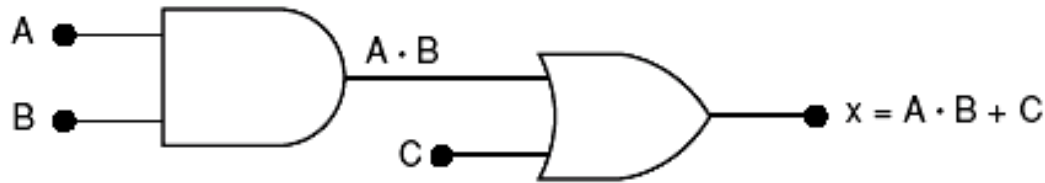
- ◎ **Símbolos Padrão IEEE/ANSI:** Desenvolvidos em 1984, utiliza formatos retangulares e indicações das dependências entre entradas e saídas.



Descrição algébrica de circuitos lógicos

- ⊙ Qualquer circuito digital lógico pode ser descrito usando as três operações lógicas básicas.

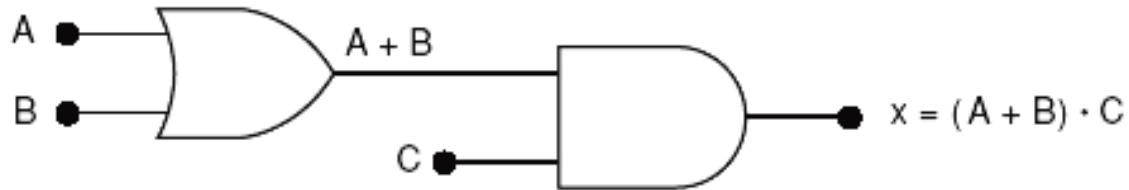
Exemplo:



Descrição algébrica de circuitos lógicos

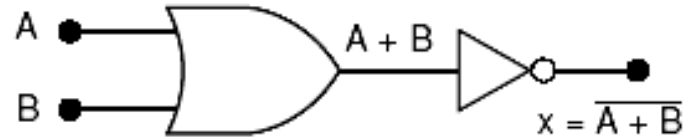
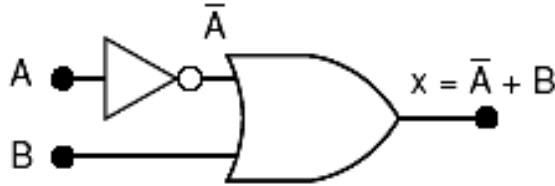
- ⦿ A precedência é sempre da operação AND, caso se deseje que a OR ocorra primeiro, deve-se usar parêntesis.

Exemplo:



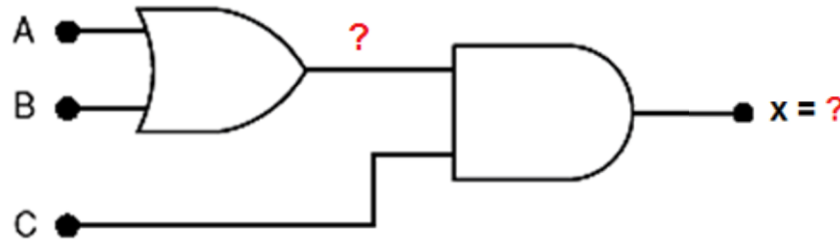
- ⊙ Quando o circuito possui inversores, usam-se barras sobre as variáveis para indicar a operação de inversão.

Exemplos:

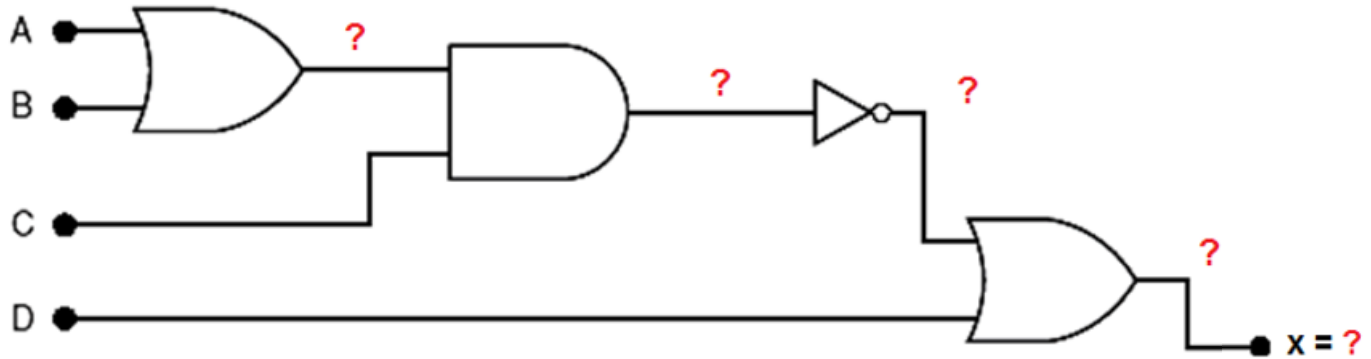


- No primeiro circuito, a **inversão tem precedência sobre a OR**.
- No segundo, a **OR tem precedência**, pois a inversão está cobrindo toda a operação.

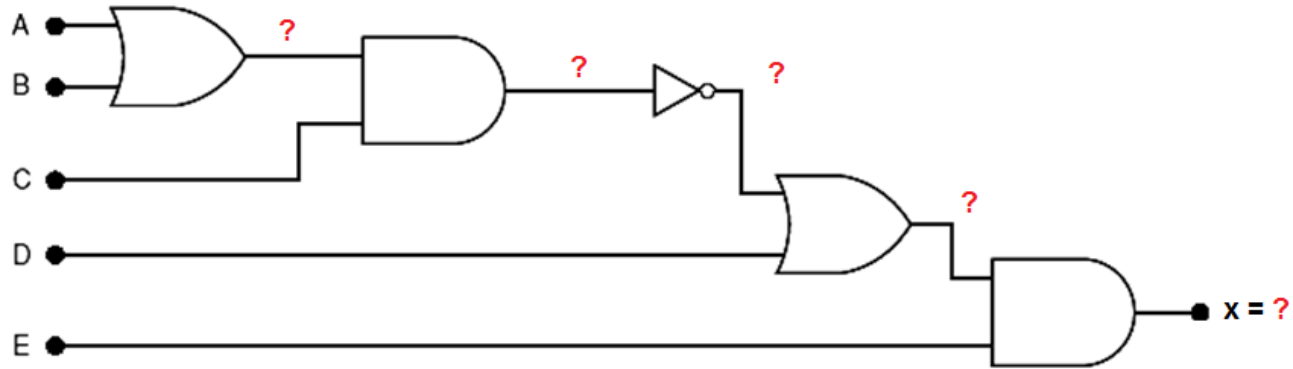
- Para se **obter a saída de um circuito** para uma dada condição das entradas, utiliza-se a **expressão lógica**.
- **Encontre a expressão lógica de saída** da figura abaixo:



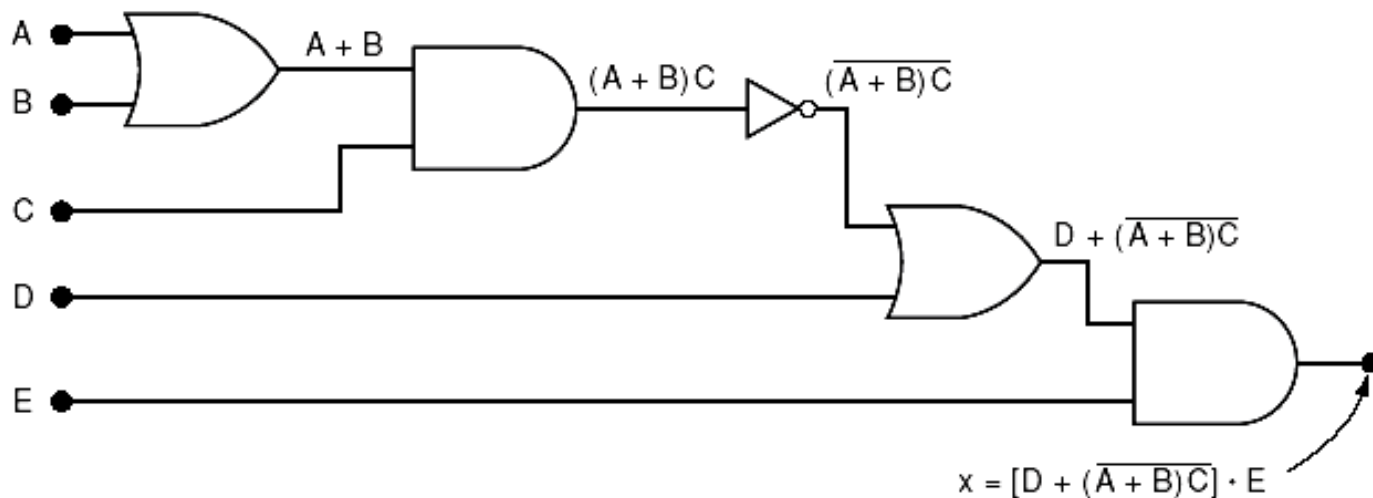
- **Encontre a expressão lógica de saída** da figura abaixo:



- **Encontre a expressão lógica de saída** da figura abaixo:

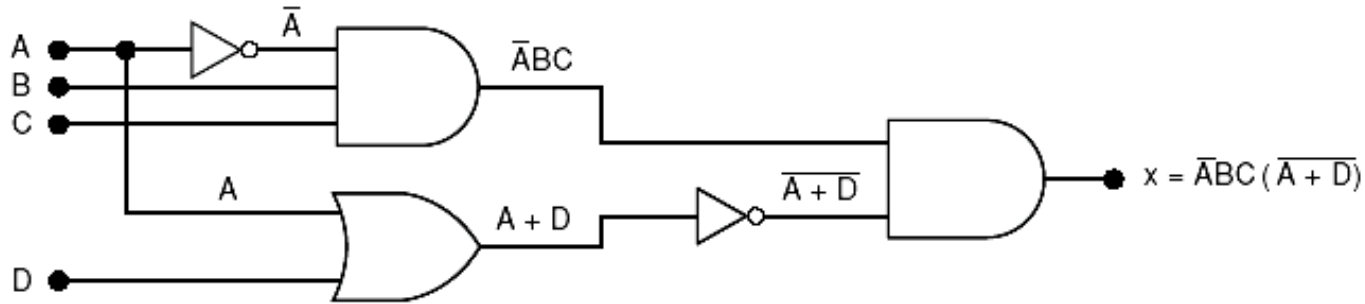


⊙ Expressão lógica de saída da figura abaixo:



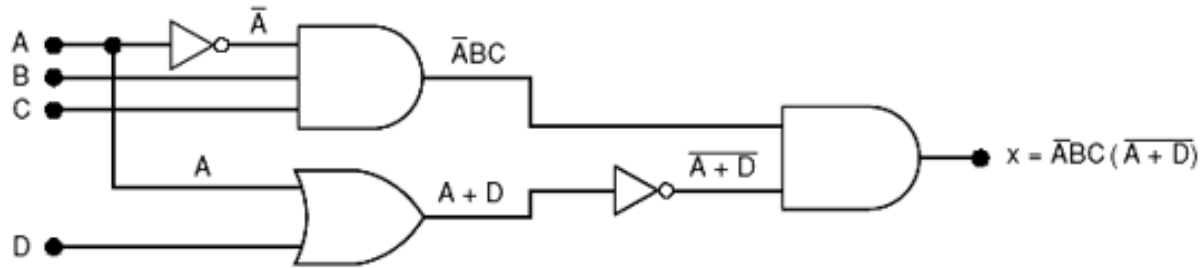


⊙ **Exercício 1:** Encontre a expressão lógica de saída da figura abaixo:



⊙ Encontre x quando $A=0$, $B=1$, $C=1$ e $D=1$.

⊙ **Exercício 2:** Encontre x quando A=0, B=1, C=1 e D=1.



$$\begin{aligned}x &= \overline{A}BC\overline{(A+D)} \\&= \overline{0} \cdot 1 \cdot 1 \cdot \overline{(0+1)} = 1 \cdot 1 \cdot 1 \cdot \overline{(1)} \\&= 1 \cdot 1 \cdot 1 \cdot 0 = 0\end{aligned}$$

- ⊙ Quando se possui a **expressão algébrica booleana** de um circuito, pode-se **desenhar seu diagrama** pela **decomposição nas operações básicas**.
- ⊙ **Exercício 3:** Implemente o circuito cuja saída seja dada pela expressão abaixo.

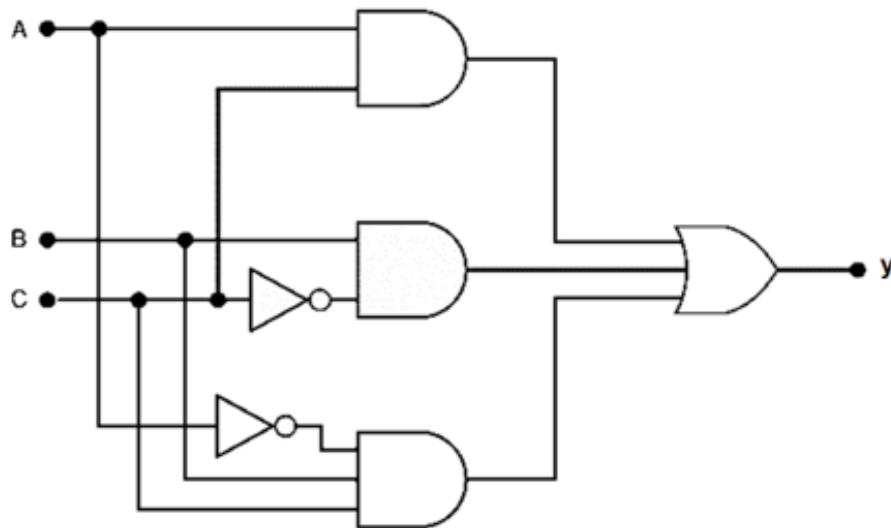
$$y = AC + B\overline{C}$$

- ⊙ Quando se possui a **expressão algébrica booleana** de um circuito, pode-se **desenhar seu diagrama** pela **decomposição nas operações básicas**.
- ⊙ **Exercício 4:** Implemente o circuito cuja saída seja dada pela expressão abaixo.

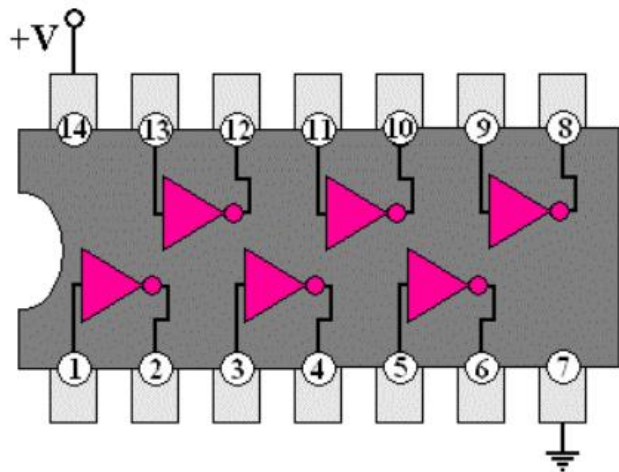
$$y = AC + B\overline{C} + \overline{A}BC$$

Solução exercício 4

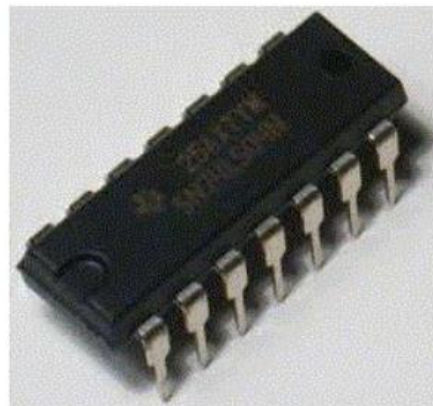
$$y = AC + B\bar{C} + \bar{A}BC$$



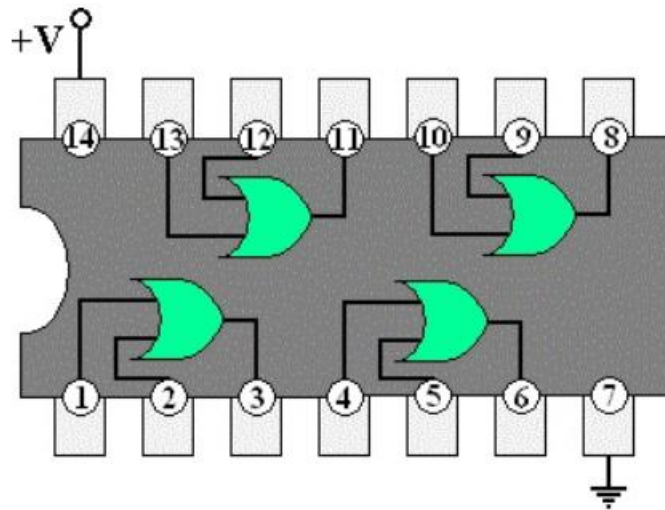
Curiosidades



CI7404



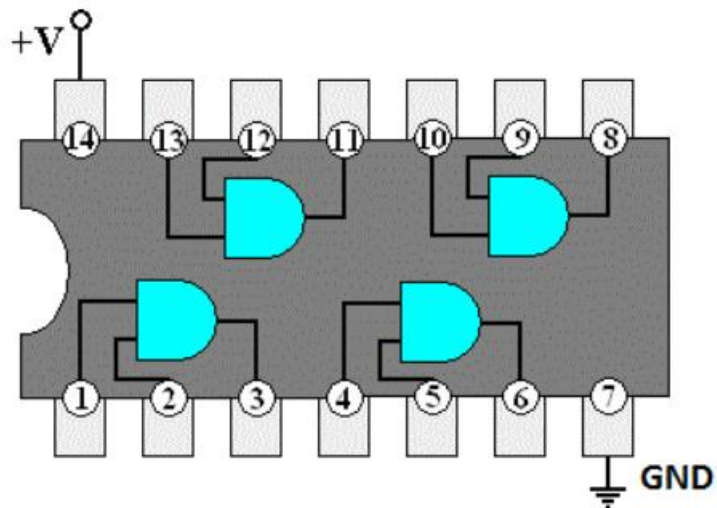
Curiosidades



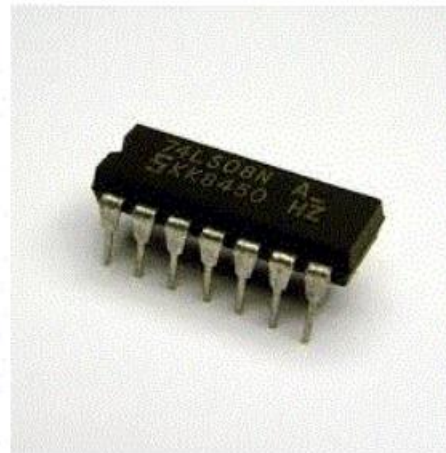
CI7432



Curiosidades



CI7408



LOGISIM

- ⊙ Utilizando o software implemente as portas lógicas AND, OR, NAND e NOR. Obtenha sua tabela verdade.
- ⊙ Implemente a solução do exercício 3 e 4 e verifique se ela está de acordo com a resposta obtida manualmente.