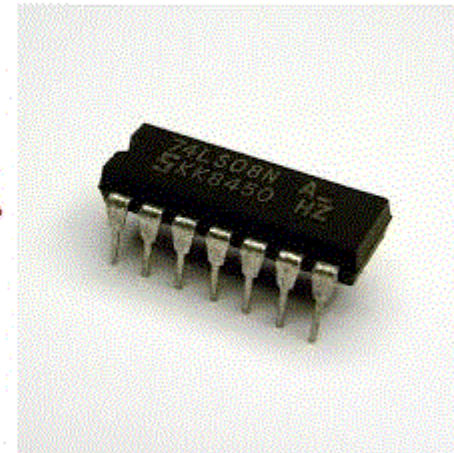
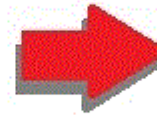
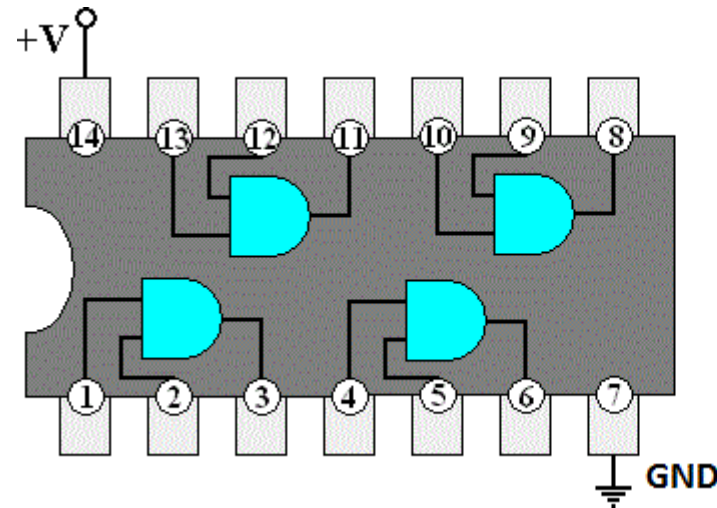


# Curso Técnico em Informática

## Portas e circuitos Lógicos.



# PORTAS E CIRCUITOS LÓGICOS

- **As portas lógicas:** são elementos básicos para a produção de circuitos lógicos.
- **Portas e circuitos lógicos:** permitem a operação de instruções lógicas de uma ULA de um processador moderno.
- Toda esta teoria lógico-matemática surgiu com o trabalho de George Boole quando desenvolveu um sistema matemático de **análise lógica denominada álgebra booleana**.
- Um engenheiro pode projetar uma ULA para desempenhar praticamente qualquer cálculo matemático e lógico (lógica booleana), e que quando mais complexo for um projeto de uma ULA, mais cara ela fica.



# PORTAS E CIRCUITOS LÓGICOS

- Como é composta uma ULA?
- Quais os elementos básicos dela?
- Como funciona?



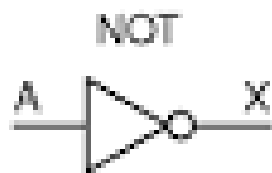
# PORTAS E CIRCUITOS LÓGICOS

- São as portas e os circuitos lógicos corretamente projetados que permitem o bom funcionamento de um processador de uma ULA.
- Em um circuito lógico, sempre entra pelo menos uma **corrente elétrica** (por uma ou mais portas) em dois tipos diferentes de tensão que representa o **número binário (corrente alta e corrente baixa)** e sai uma corrente elétrica trabalhada de acordo com o tipo da porta.
- Cada porta lógica trata a(s) corrente(s) que entra(m) de uma forma diferente.



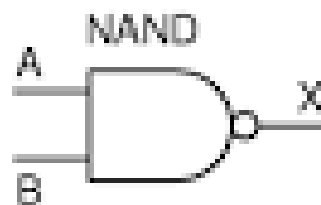
# PORTAS E CIRCUITOS LÓGICOS

- Exemplificando:** “A” e “B” são as portas de entrada de corrente, e “X” representa a saída da corrente, note o efeito que cada porta faz com as correntes que entram.



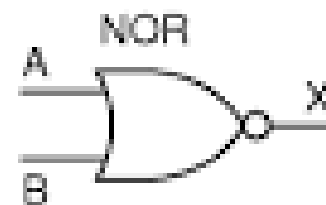
A	X
0	1
1	0

(a)



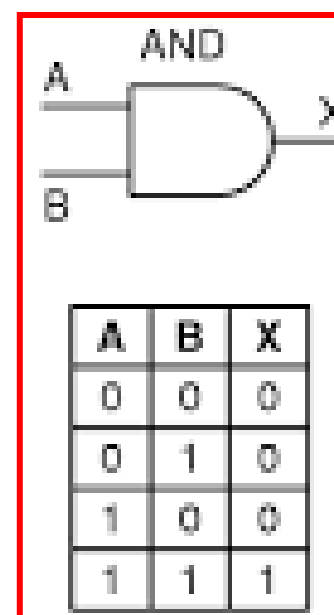
A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

(b)



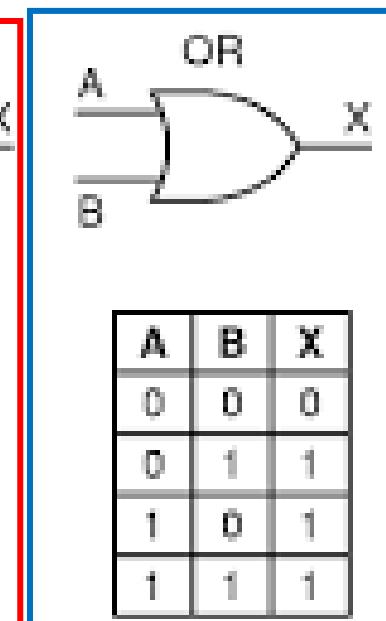
A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

(c)



A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

(d)



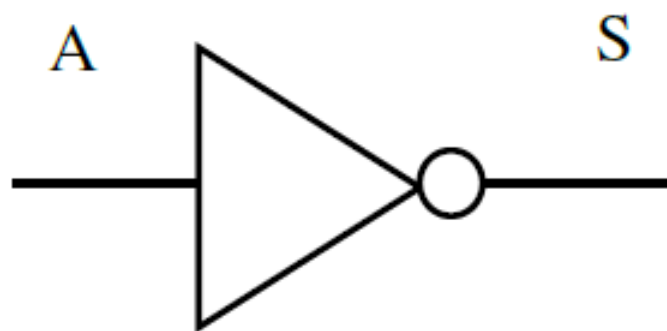
A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

(e)



# PORTAS E CIRCUITOS LÓGICOS

- **NOT:** É a ultima porta considerada básica, normalmente uma porta *not* tem apenas uma entrada. Em uma função booleana ela é considerada a porta que irá inverter qualquer sinal e seu símbolo é a **aspa simples**.



Representação gráfica

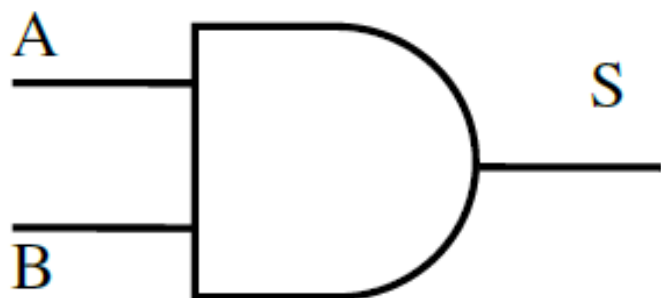
ENTRADA	SAÍDA	FUNÇÃO BOOLEANA
A	A'	A'
0	1	
1	0	

Tabela verdade



# PORTAS E CIRCUITOS LÓGICOS

- **AND (E):** É uma das portas básicas, em uma função booleana ela é representada pela **multiplicação**, ou **sinal de ponto**.



Representação gráfica

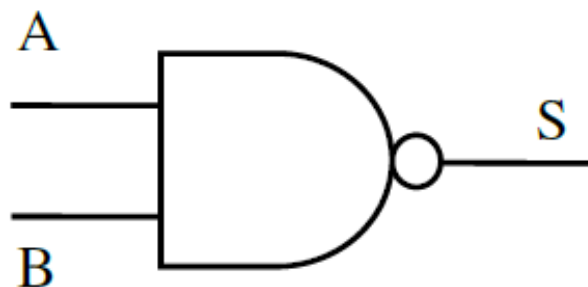
ENTRADA		SAÍDA	FUNÇÃO BOOLEANA
A	B	A and B	(A.B)
0	0	0	
0	1	0	
1	0	0	
1	1	1	

Tabela verdade



# PORTAS E CIRCUITOS LÓGICOS

- **NAND:** É uma porta que é a junção de uma porta *and* (*E*) mais a inversão (*not*), logo seu resultado será sempre o inverso da *and*.
- **Note** que a representação gráfica do *not* é apenas a “**bolinha**” antes da saída.



Representação gráfica

ENTRADA		SAÍDA	FUNÇÃO BOOLEANA
A	B	A and B – I*	(A.B)'
0	0	1	
0	1	0	
1	0	0	
1	1	0	

\* Invertido

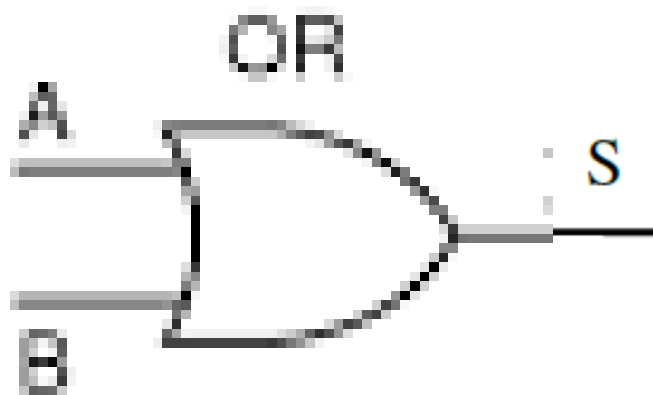
Tabela verdade





# PORTAS E CIRCUITOS LÓGICOS

- **OR (OU):** É a outra portas básicas, em uma função booleana ela é representada pela soma, ou sinal de +.



Correção gráfica

Representação gráfica

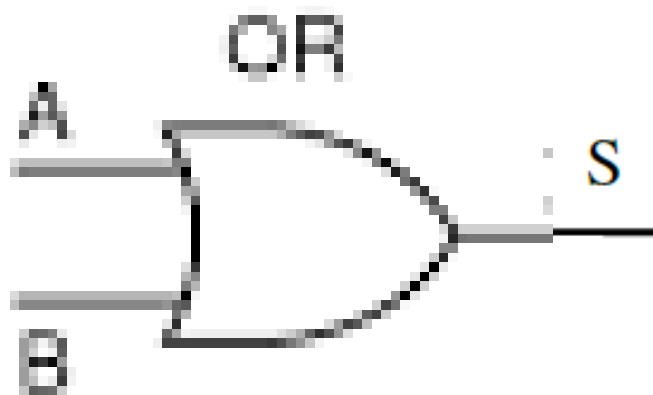
ENTRADA		SAÍDA	FUNÇÃO BOOLEANA
A	B	A or B	(A+B)
0	0	0	
0	1	1	
1	0	1	
1	1	1	

Tabela verdade



# PORTAS E CIRCUITOS LÓGICOS

- **OR (OU):** É a outra portas básicas, em uma função booleana ela é representada pela soma, ou sinal de +.



Correção gráfica

Representação gráfica

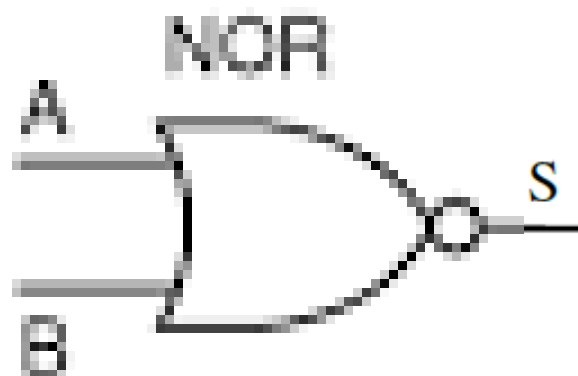
ENTRADA		SAÍDA	FUNÇÃO BOOLEANA
A	B	A or B	(A+B)
0	0	0	
0	1	1	
1	0	1	
1	1	1	

Tabela verdade



# PORTAS E CIRCUITOS LÓGICOS

- **NOR:** Esta porta é a junção de uma **porta or (OU)** mais a **inversão (not)**, logo seu resultado será sempre o inverso da **or**. Observe sua função booleana, é apenas a porta **or** mais a porta **not**.



Correção gráfica

Representação gráfica

ENTRADA		SAÍDA	FUNÇÃO BOOLEANA
A	B	A or B – I*	$(A+B)'$
0	0	1	
0	1	0	
1	0	0	
1	1	0	

\* Invertido

Tabela verdade



# PORTAS E CIRCUITOS LÓGICOS

- **XOR** A porta XOR traz o conceito de **exclusividade** às portas lógicas. A função da exclusividade nas portas lógicas é permitir **que a saída do circuito seja sempre 0 para as entradas iguais.**



Correção gráfica

Representação gráfica

ENTRADA		SAÍDA	FUNÇÃO BOOLEANA
A	B	$A \oplus B$	$(A+B).(A.B)'$
0	0	0	
0	1	1	
1	0	1	
1	1	0	

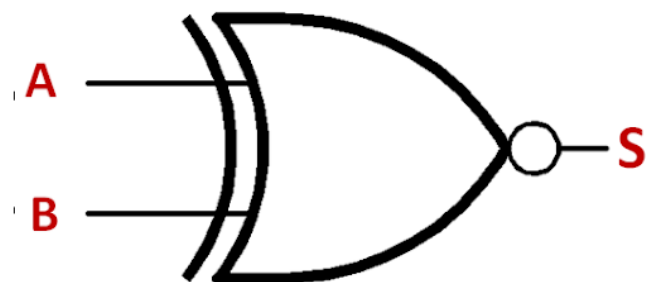
Tabela verdade



# PORTAS E CIRCUITOS LÓGICOS

**XNOR:** Esta porta é a inversão da porta XOR, portanto, os sinais iguais que entrarem, a saída sempre será 1, diferente da porta XOR que é 0.

**Note** o símbolo da porta *not* “bolinha”.



Correção gráfica

Representação gráfica

ENTRADA		SAÍDA	FUNÇÃO BOOLEANA
A	B	$A \oplus B$	$((A+B).(A.B))'$
0	0	1	
0	1	0	
1	0	0	
1	1	1	

Tabela verdade



# PORTAS E CIRCUITOS LÓGICOS

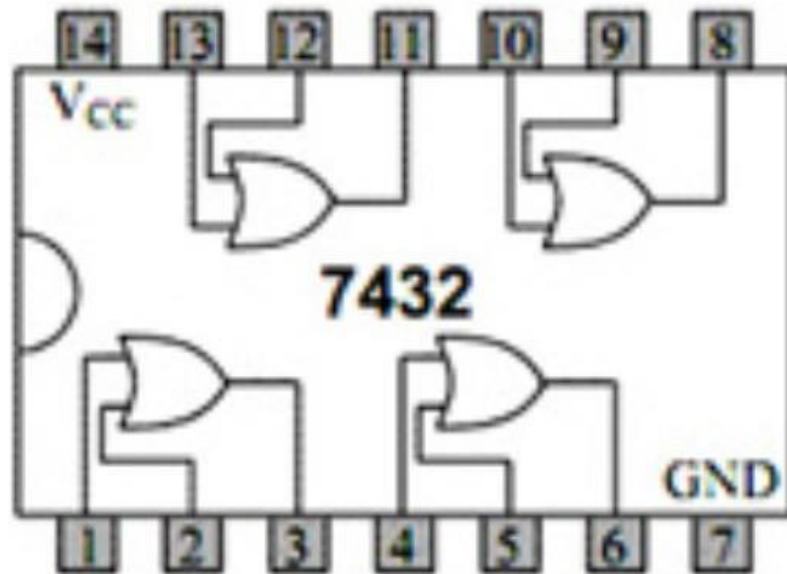
- São com estas portas lógicas, que projeta-se a maioria dos circuitos digitais atualmente. São várias as literaturas sobre este assunto na Internet, você poderá pesquisar e se aprofundar neste assunto de acordo com seu interesse ou a necessidade.
- **Mas agora me deu uma duvida como isso funciona na pratica?**



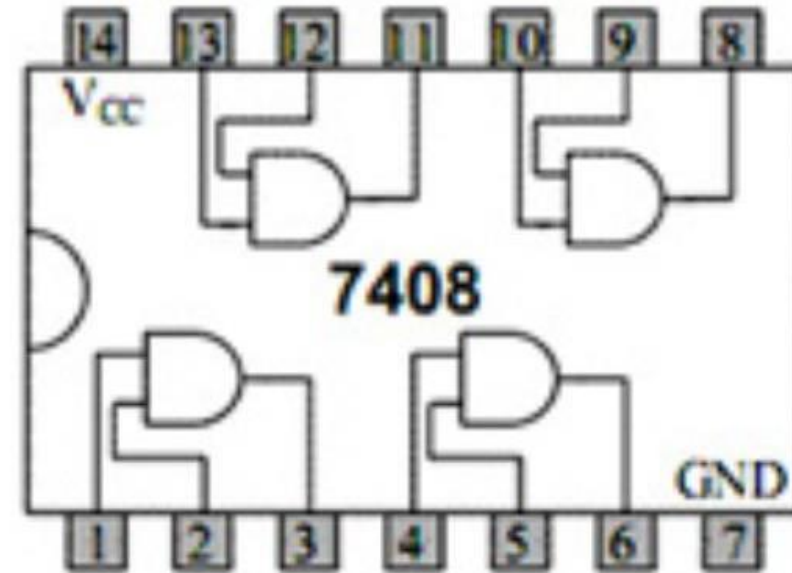
# PORTAS E CIRCUITOS LÓGICOS

- Porta lógica 7432 e 7408.

- 4 Portas OR:

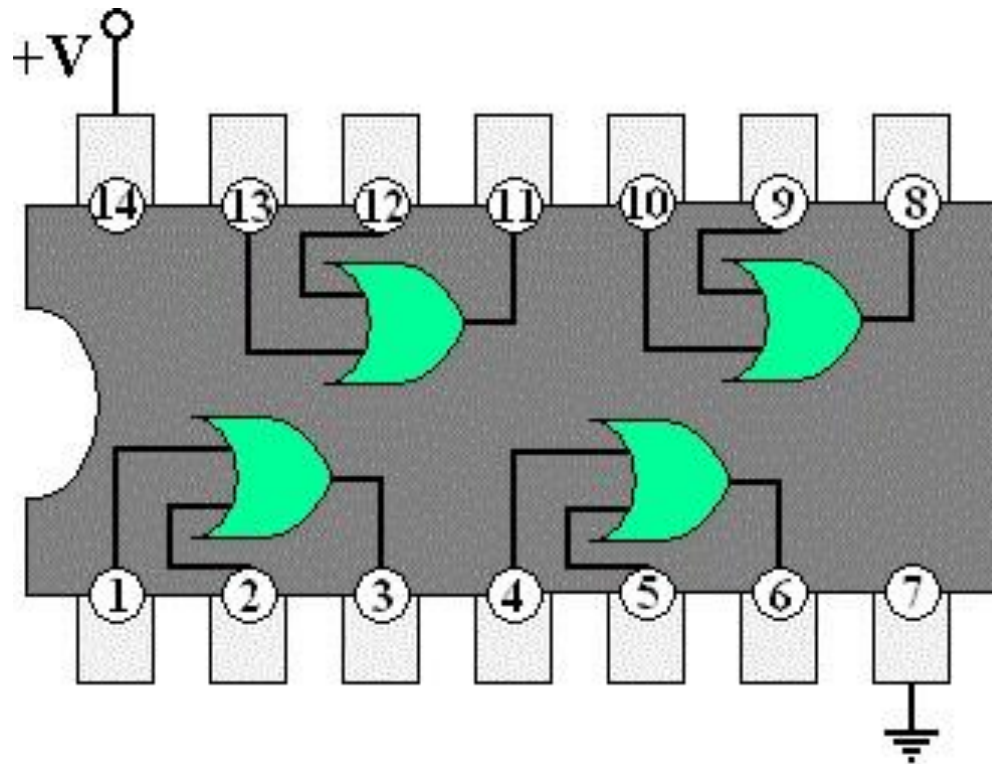


- 4 Portas AND



# PORTAS E CIRCUITOS LÓGICOS

- E no computador como podemos ver isso?
- Porta lógica 7432.





# PORTAS E CIRCUITOS LÓGICOS

- E na pratica como isso poderia funciona?

Aplicação prática : Controle da porta de um elevador

- Em um prédio de três andares deseja-se projetar um circuito lógico para controlar a abertura da porta de um elevador. As variáveis de entrada são  $A$ ,  $B$ ,  $C$ ,  $D$  em que :
  - $A$  indica que o elevador está em movimento quando igual a 1.
  - $B$ ,  $C$ ,  $D$  indicam que o elevador está posicionado nos andares 1, 2, 3 quando iguais a 1, respectivamente.
- Projete a saída  $Ab$  que indica, quando em nível alto, que o elevador deve abrir a porta. Para isto :
  - Determine a tabela verdade do problema.
  - Faça simplificações utilizando um dos métodos estudados.
  - Desenhe o circuito lógico correspondente.



# PORTAS E CIRCUITOS LÓGICOS

- E na pratica como isso poderia funciona?

Aplicação prática : Controle da porta de um elevador

Tabela verdade :

A	B	C	D	Ab
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	X
0	1	0	0	1
0	1	0	1	X
0	1	1	0	X
0	1	1	1	X
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	X
1	1	0	0	0
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

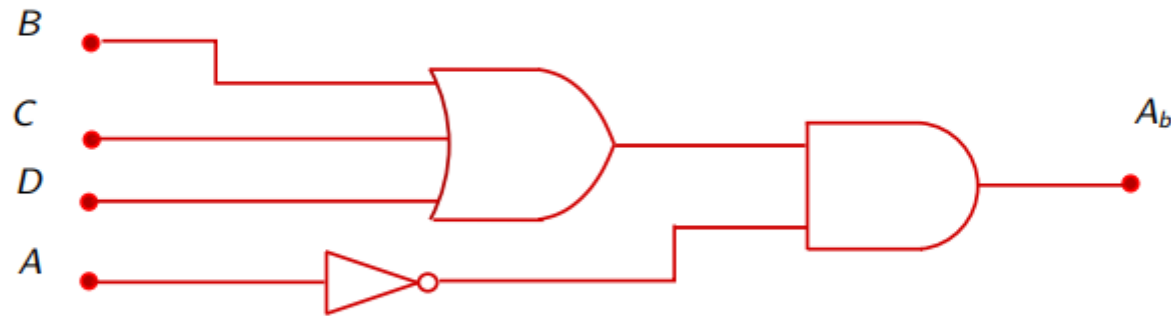


# PORTAS E CIRCUITOS LÓGICOS

- E na pratica como isso poderia funciona?

Aplicação prática : Controle da porta de um elevador

Representação do circuito :



# Processadores

- Duvidas???



# Processadores

Fim.

Ass. FEGS

