

# FACULDADE PITÁGORAS

## PRONATEC

DISCIPLINA: ARQUITETURA DE COMPUTADORES

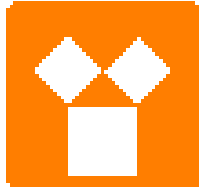
Prof. Ms. Carlos José Giudice dos Santos

[carlos@oficinadapesquisa.com.br](mailto:carlos@oficinadapesquisa.com.br)

[www.oficinadapesquisa.com.br](http://www.oficinadapesquisa.com.br)

**APOSTILA I**

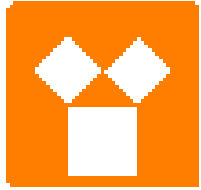
**INTRODUÇÃO À ARQUITETURA DE COMPUTADORES**



# Objetivos

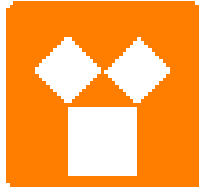
Ao final desta apostila, o aluno deverá ser capaz de:

1. Definir computador, bit, Byte, registrador, memória, processador, CPU, hardware, software e periféricos.
2. Saber o que são sistemas de numeração.
3. Saber fazer conversão de bases numéricas.
4. Definir porta lógica.
5. Reconhecer as portas lógicas.
6. Saber fazer a tabela verdade para as portas lógicas estudadas.



# Qual é a importância de se conhecer arquitetura de computadores?

1. Conhecer o funcionamento interno dos computadores;
2. Dar base para que o aluno consiga acompanhar a constante evolução tecnológica na informática;
3. Permitir ao aluno dimensionar sistemas computacionais de acordo com as necessidades de clientes (pessoas, empresas ou grandes organizações) - editoração, cálculos, jogos, comunicação, desenvolvimento de sistemas, servidores de redes, automação de processos, etc.

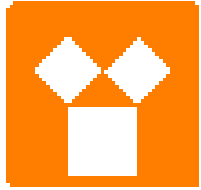


# Conceito de Computador

Um computador digital é uma máquina que pode resolver problemas para as pessoas executando instruções que lhe são dadas. Uma seqüência de instruções que descreve como realizar certa tarefa é chamada de programa.

Um computador doméstico típico de hoje (2014) possui as características seguintes

1. Alta velocidade de processamento de dados (superior a dois bilhões de operações por segundo);
2. Grande capacidade de armazenamento de informações (500 GB ou mais)
3. Suporte para diversas mídias
4. Possibilidade de comunicação com a Internet



# Alguns conceitos básicos

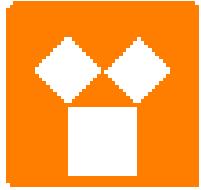
**Bit** → Acrônimo de Binary Digit, é a menor quantidade de informação possível de ser armazenada e interpretada por um computador. Um bit pode assumir apenas dois valores: 0 ou 1.

**Byte** → Acrônimo de Binary Term, é um conjunto de oito bits. Cada byte pode representar um número, letra ou instrução.

**Palavra** → Em computação, é o número máximo de bits que cabe em um registrador. É também o número máximo de bits que um computador pode manipular em uma única instrução.

**Registrador** → É um mecanismo que registra (armazena um dado ou instrução). Geralmente é uma memória de pequena capacidade localizada dentro de um processador.

**Sistema Operacional** → É um "programa" que permite ao usuário controlar e se comunicar com o computador.

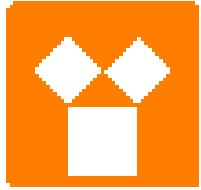


# Mais alguns conceitos básicos

**Memória** → Local onde os dados são armazenados em um computador. A memória é classificada em duas categorias: principal e secundária

**Memória Principal** → É a memória de trabalho de um computador, também conhecida como memória RAM (*Random Access Memory*). Os dados armazenados neste tipo de memória desaparecem assim que o computador é desligado. Quando precisarmos de guardar dados para serem acessados depois, temos que usar a Memória Secundária.

**Memória Secundária** → Também conhecida como memória de massa, é o local onde se armazenam dados e/ou informações de forma permanente.



# Mais alguns conceitos básicos

**Processador** → É um conjunto de circuitos eletrônicos que executam instruções. Dentro de um computador existem vários processadores. O processador principal é chamado de CPU.

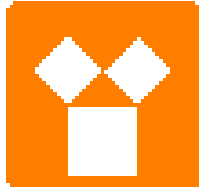
**CPU** → Sigla para *Central Processing Unit*, ou Unidade de Processamento Central, que corresponde ao "cérebro" do computador. É o local onde todas as operações lógicas são realizadas. Não confundir com o gabinete do computador.

**Hardware** → Corresponde à parte física (circuitos eletrônicos) de um computador, juntamente com seus periféricos.

**Software** → Conjunto de instruções e dados que são responsáveis pelo funcionamento do hardware. É sinônimo de **programa** ou **aplicativo**.

**Periféricos** → Todo e qualquer dispositivo que pode se conectar à CPU de um computador (mouses, teclados, monitores de vídeo, impressoras, scanners, etc).





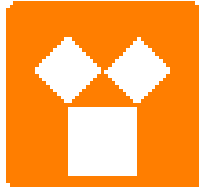
# SISTEMAS DE NUMERAÇÃO

Vimos anteriormente que computadores trabalham com bits, que podem assumir apenas dois valores: 0 ou 1. Os cálculos que envolvem bits facilitam o projeto de circuitos eletrônicos, pois estes circuitos devem trabalhar apenas com dois níveis de tensão: zero (muito pouca ou nenhuma voltagem) para um bit de valor 0, e qualquer tensão acima de 2,5 Volts para um bit de valor 1. Agora, veja as operações aritméticas abaixo:

Qual das contas abaixo está certa?

1	1	1	1
+ 1	+ 7	+ 9	+ F
-----	-----	-----	-----
10	10	10	10

Acertou quem disse: Todas!



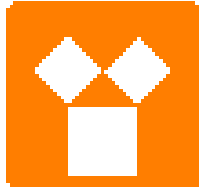
# Sistemas de Numeração

**Sistema Binário:** importante sistema de numeração, utilizado na tecnologia dos computadores. Sua base é "dois", tendo somente dois algarismos: { 0, 1 };

**Sistema Decimal:** sistema de números em que uma unidade de ordem vale dez vezes a unidade de ordem imediatamente anterior. Sua base numérica é de dez algarismos: { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 }.

**Sistema Octal:** Sistema de numeração cuja base é oito, adotado na tecnologia de computadores. Não é mais utilizado na prática. Sua base numérica é de oito algarismos: { 0, 1, 2, 3, 4, 5, 6, 7 };

**Sistema Hexadecimal:** Sistema de numeração cuja base é dezesseis. Esse sistema trabalha com dez algarismos numéricos baseados no decimal e com a utilização de mais seis letras. Os algarismos deste sistema são: { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F }.

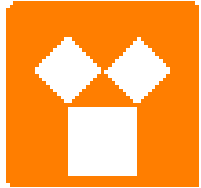


# Sistemas de Numeração

Em nosso curso vamos trabalhar com o sistema binário, o decimal e o hexadecimal. O sistema decimal é o mais utilizado para representar valores no mundo real. O fato de termos dez dedos nas mãos foi a base para o sistema decimal. Se em vez de 10 dedos, nós tivéssemos 12 dedos, provavelmente nosso sistema seria baseado em 12 símbolos.

O sistema binário é utilizado em computadores porque facilita o projeto de circuitos digitais chamados de portas lógicas, que veremos mais tarde.

O sistema hexadecimal passou a ser utilizado para transformar a representação binária em uma outra base numérica que facilite a leitura e o entendimento das pessoas que trabalham com informática. Veremos isso a seguir em conversão de bases numéricas.

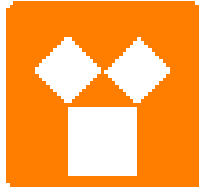


# Conversão de Bases Numéricas

Em nosso curso vamos trabalhar com o sistema binário, o decimal e o hexadecimal. O sistema decimal é o mais utilizado para representar valores no mundo real. O fato de termos dez dedos nas mãos foi a base para o sistema decimal. Se em vez de 10 dedos, nós tivéssemos 12 dedos, provavelmente nosso sistema seria baseado em 12 símbolos.

O sistema binário é utilizado em computadores porque facilita o projeto de circuitos digitais chamados de portas lógicas, que veremos mais tarde.

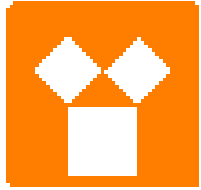
O sistema hexadecimal passou a ser utilizado para transformar a representação binária em uma outra base numérica que facilite a leitura e o entendimento das pessoas que trabalham com informática. Veremos isso a seguir em conversão de bases numéricas.



# Conversão de Bases Numéricas

**REGRA:** Qualquer número pode ser convertido de uma base numérica para outra;

- Para isto precisamos entender:
  1. **Valor Absoluto:** também chamado de valor intrínseco, é o número propriamente dito;
  2. **Valor Posicional:** é o valor que um número ocupa em uma determinada posição.



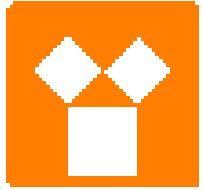
## EXEMPLO: 2014

### Valor Absoluto:

Milhar	Centena	Dezena	Unidade
2	0	1	4

### Valor Posicional:

3	2	1	0
2	0	1	4

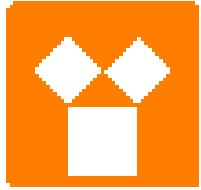


## FÓRMULA PARA CONVERSÃO DE BASES

$$vp = va \times base^{np}$$

Onde:

- Vp = Valor da Posição;
- Va = Valor Absoluto ;
- Np = Número da Posição.



# EXEMPLOS DE APLICAÇÃO DA FÓRMULA

**Exemplo:** Converter  $01100001_{(2)}$  para a base 10

$$1 \times 2^0 = 1 \times 1 = 1$$

$$0 \times 2^1 = 0 \times 2 = 0$$

$$0 \times 2^2 = 0 \times 4 = 0$$

$$0 \times 2^3 = 0 \times 8 = 0$$

$$0 \times 2^4 = 0 \times 16 = 0$$

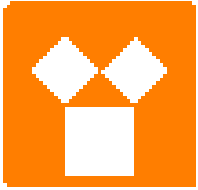
$$1 \times 2^5 = 1 \times 32 = 32$$

$$1 \times 2^6 = 1 \times 64 = 64$$

$$0 \times 2^7 = 0 \times 128 = 0$$

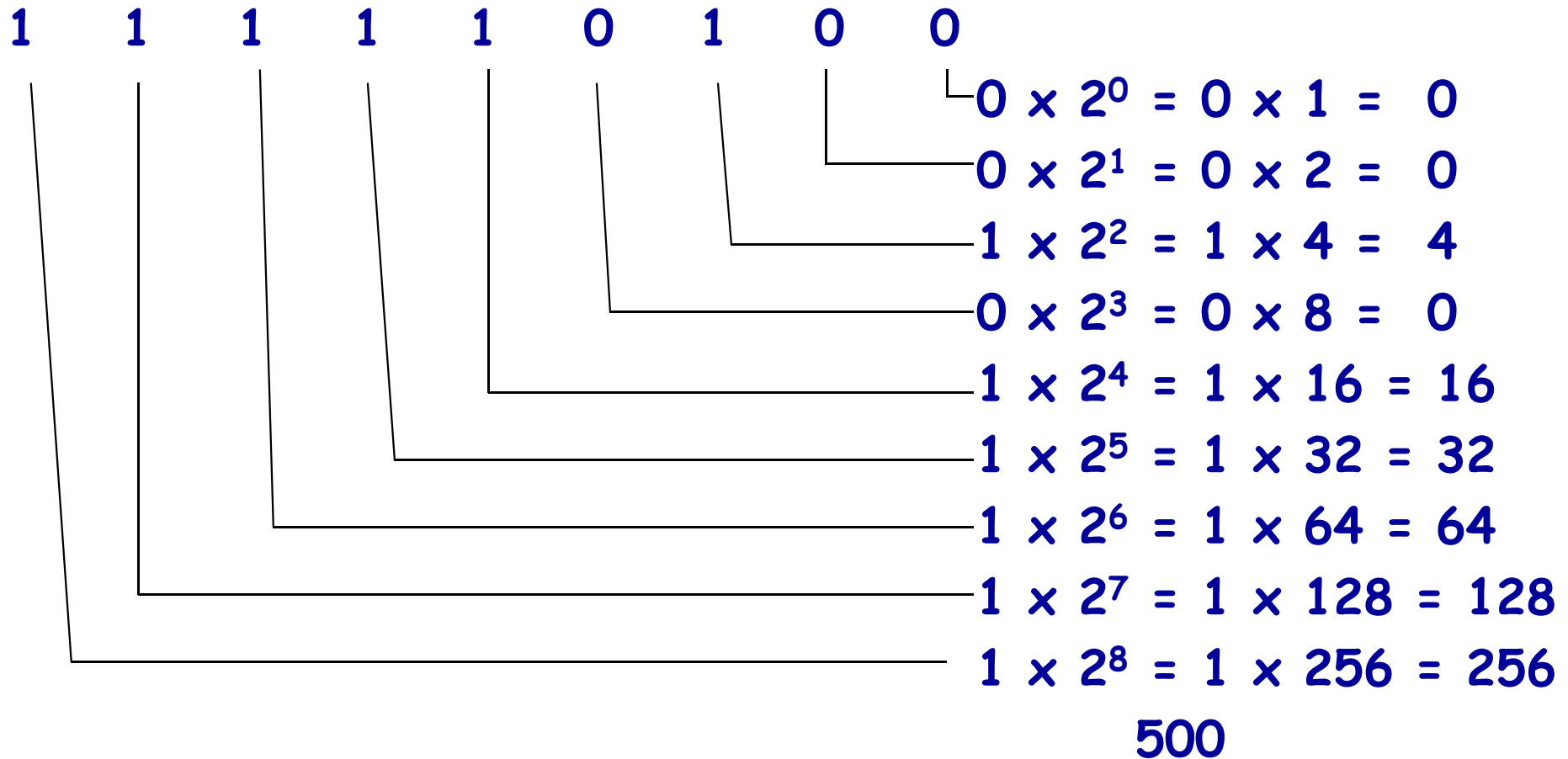
Em que:  $1 + 32 + 64 = 97_{(10)}$ .



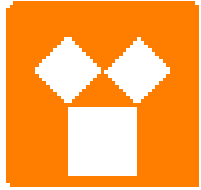


## OUTRO EXEMPLO: Binário para Decimal

Converter o binário  $111110100_{(2)}$  para decimal.



$$111110100_{(2)} = 500_{(10)}$$



## Conversão de Decimal para Binário

**Exemplo:** Converter  $71_{(10)}$  para a base 2

Processo → Dividir o número decimal sucessivamente pela base até que o resultado chegue a 0 ou 1.

$$71 / 2 = 35 \quad \rightarrow \text{Sobra } 1$$

$$35 / 2 = 17 \quad \rightarrow \text{Sobra } 1$$

$$17 / 2 = 8 \quad \rightarrow \text{Sobra } 1$$

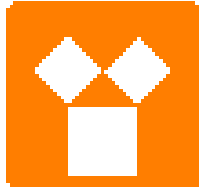
$$8 / 2 = 4 \quad \rightarrow \text{Sobra } 0$$

$$4 / 2 = 2 \quad \rightarrow \text{Sobra } 0$$

$$2 / 2 = 1 \quad \rightarrow \text{Sobra } 0$$

Depois é só pegar o resultado de baixo para cima:

Logo,  $71_{(10)}$  equivale a  $1000111_{(2)}$

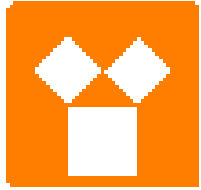


## Conversão de Decimal para Hexadecimal

**Exemplo:** Converter  $71_{(10)}$  para a base 16

**Processo** → Dividir o número decimal sucessivamente pela base até que o resultado chegue a um número compreendido entre 0 e 15. Depois basta utilizar a seguinte correspondência:

0 → 0	1 → 1	2 → 2	3 → 3
4 → 4	5 → 5	6 → 6	7 → 7
8 → 8	9 → 9	10 → A	11 → B
12 → C	13 → D	14 → E	15 → F



## Conversão de Decimal para Hexadecimal

Fazendo o exemplo: Converter  $71_{(10)}$  para a base 16

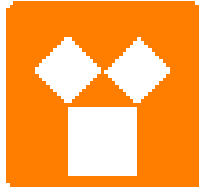
$$71 / 16 = 4 \quad \rightarrow \text{Sobra } 7 \rightarrow \text{Logo, } 71_{(10)} = 47_{(16)}$$

Outro exemplo: Converter  $94_{(10)}$  para a base 16

$$94 / 16 = 5 \quad \rightarrow \text{Sobra } 14 \rightarrow \text{Logo, } 94_{(10)} = 5E_{(16)}$$

Tente converter a sua idade de decimal para hexadecimal.

Tente converter o ano de seu nascimento de decimal para hexadecimal.



## Conversão de Hexadecimal para Decimal

**Exemplo:** Converter  $23_{(16)}$  para a base 10

$$23_{(16)} = (2 \times 16^1) + (3 \times 16^0) = 32 + 3 = 35$$

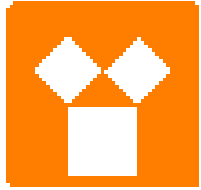
**Outro exemplo:** Converter  $3C_{(16)}$  para a base 10

$$3C_{(16)} = (3 \times 16^1) + (12 \times 16^0) = 48 + 12 = 60$$

**Outro exemplo:** Converter  $CAFE_{(16)}$  para a base 10

$$CAFE_{(16)} = (12 \times 16^3) + (10 \times 16^2) + (15 \times 16^1) + (14 \times 16^0)$$

$$CAFE_{(16)} = 49152 + 2560 + 240 + 14 = 51966$$



## Conversão de Hexadecimal para Binário

## Conversão de Binário para Hexadecimal

São as conversões mais fáceis de fazer. Basta usar a tabela de conversão que será dada em sala de aula.

Depois que a tabela for dada, tente fazer as seguintes conversões:

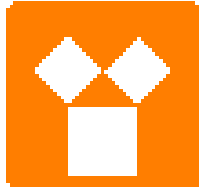
$23_{(16)}$  para binário

$CC_{(16)}$  para binário

$CAFE_{(16)}$  para binário

$FE_{(16)}$  para binário

$100100100100100100011_{(2)}$  para hexadecimal



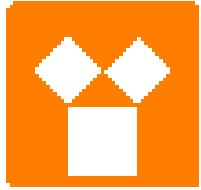
# Princípios Básicos de Eletrônica Digital

Os computadores fazem cálculos com bits utilizando circuitos digitais chamados de **PORTAS LÓGICAS**. Uma porta lógica é um circuito eletrônico que produz um sinal de saída que é resultado de uma operação booleana (lógica) sobre os sinais de entrada.

Existem vários tipos de portas lógicas. As portas lógicas básicas são:

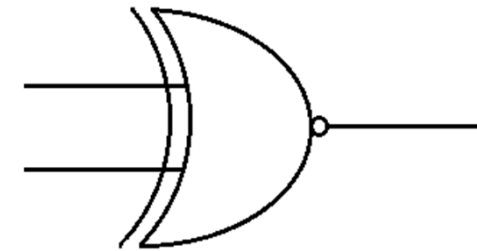
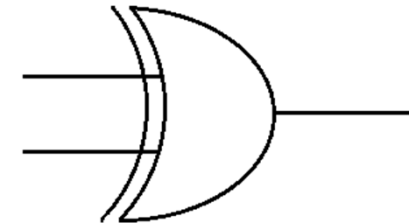
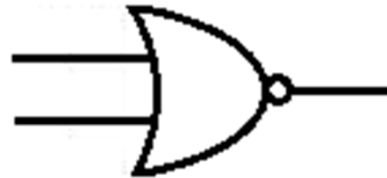
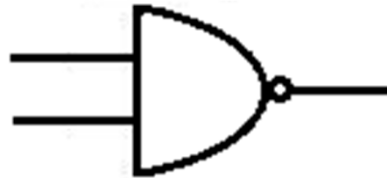
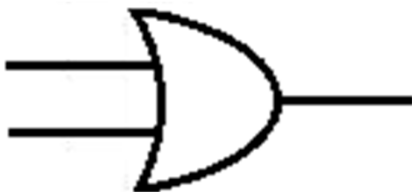
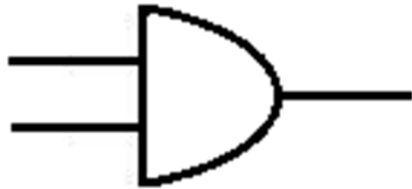
- Porta **NOT** ou Porta **NÃO** ou Porta Inversora.
- Porta **AND** ou Porta **E**;
- Porta **OR** ou Porta **OU**;

Todas as outras portas são construídas a partir de combinações destas três portas básicas.

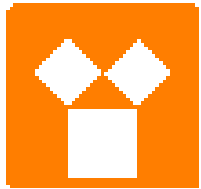


# PRINCIPAIS PORTAS LÓGICAS

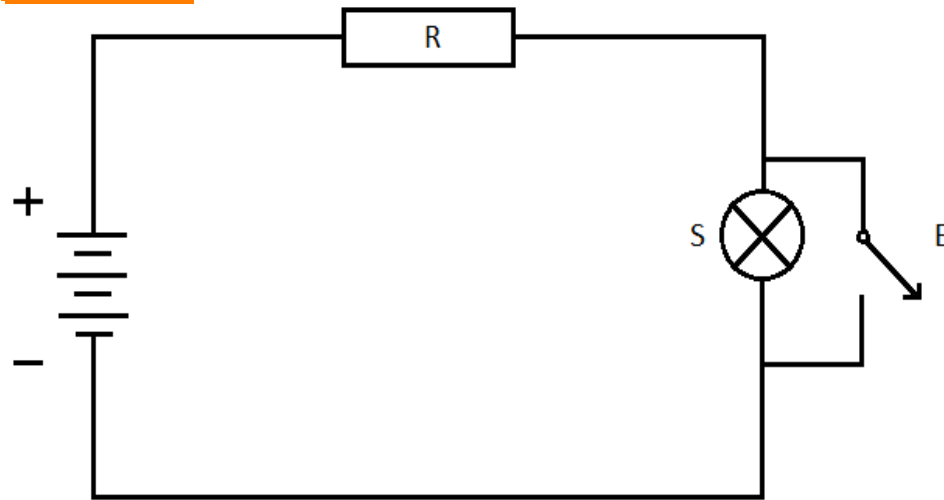
Para os propósitos de nosso curso, é necessário saber o que é uma porta lógica e saber reconhecer as principais portas lógicas usadas em circuitos digitais.







# PORTA LÓGICA NOT

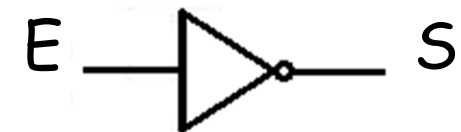


Circuito elétrico equivalente

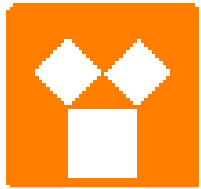
TABELA VERDADE	
E	S
0	1
1	0

$$S = \overline{E}$$

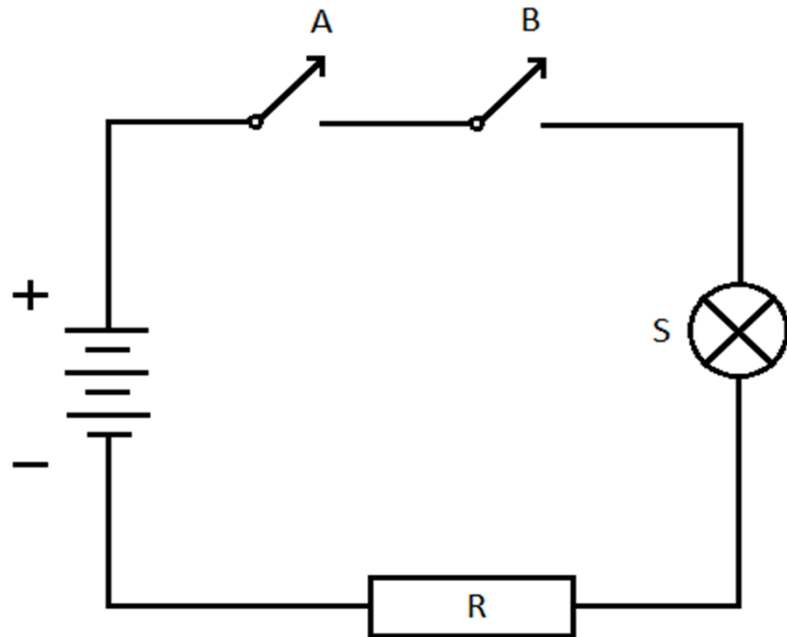
Esta porta produz um sinal de saída que é o inverso do sinal de entrada. Se entrou 0, sai 1. Se entrou 1, sai 0. A saída é sempre o inverso da entrada, razão pela qual esta porta é chamada também de porta inversora. O que interessa para o nosso curso é reconhecer o símbolo gráfico e a tabela verdade.



Símbolo gráfico



# PORTA LÓGICA AND

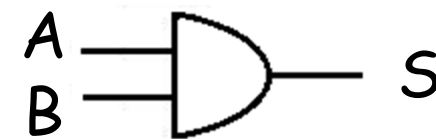


Circuito elétrico equivalente

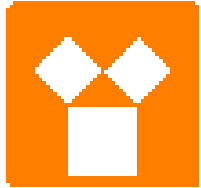
TABELA VERDADE		
A	B	S
0	0	0
0	1	0
1	0	0
1	1	1

$$S = A.B$$

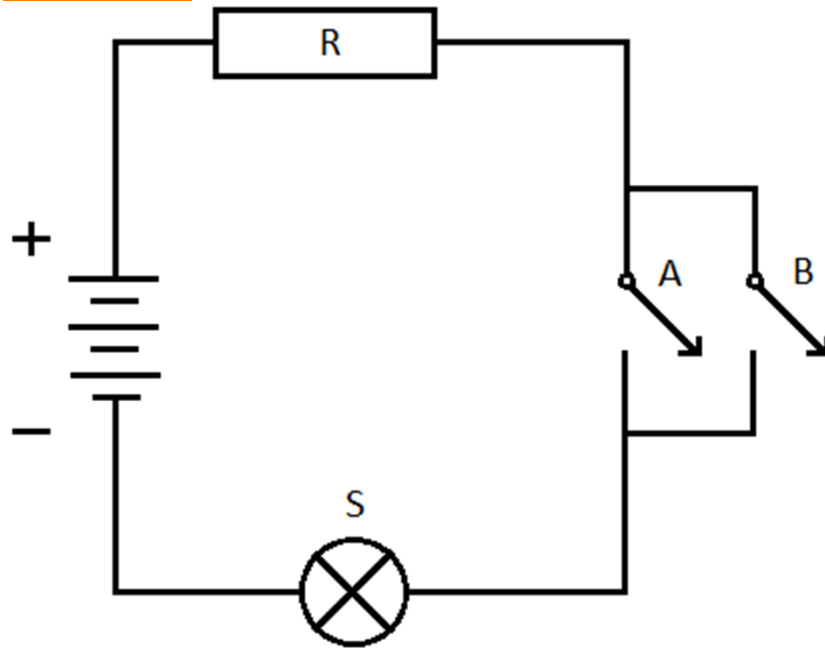
Esta porta produz um sinal de saída que depende do sinal de entrada. Como podemos ver na tabela verdade, a saída só será 1 se as duas entradas forem 1. Se pelo menos uma entrada for zero, a saída será zero. Estas portas são utilizadas em um circuito chamado Deslocador, usado para fazer multiplicações aritméticas.



Símbolo gráfico



# PORTA LÓGICA OR



Circuito elétrico equivalente

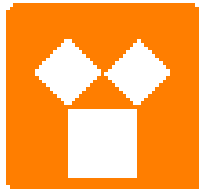
TABELA VERDADE		
A	B	S
0	0	0
0	1	1
1	0	1
1	1	1

$$S = A + B$$

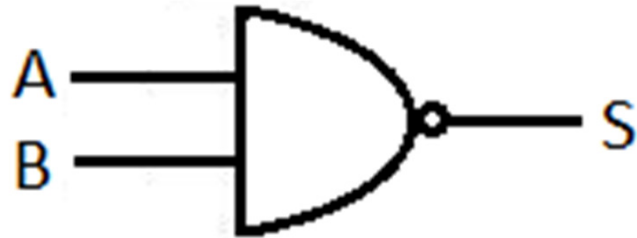
Esta porta produz um sinal de saída que depende do sinal de entrada. Como podemos ver na tabela verdade, a saída será 1 se pelo menos uma das entradas for 1. A saída só será zero se as duas entradas forem zero. Estas portas são utilizadas em circuito chamado de Somador, utilizado para fazer operações aritméticas de soma.



Símbolo gráfico



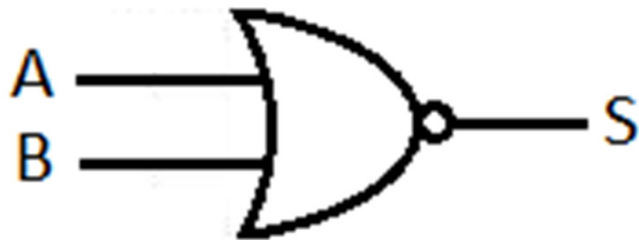
# OUTRAS PORTAS LÓGICAS



Porta NAND

TABELA VERDADE		
A	B	S
0	0	1
0	1	1
1	0	1
1	1	0

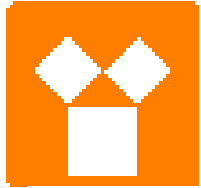
$$S = \overline{A \cdot B}$$



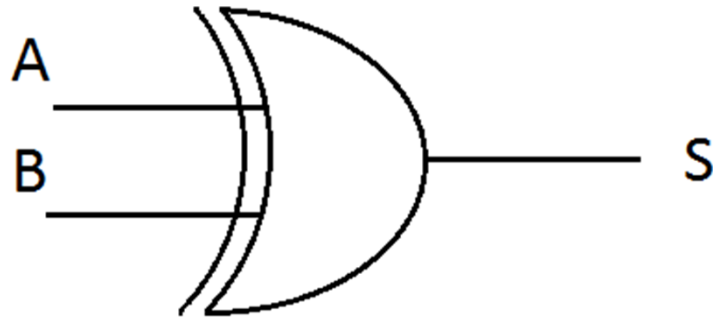
Porta NOR

TABELA VERDADE		
A	B	S
0	0	1
0	1	0
1	0	0
1	1	0

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



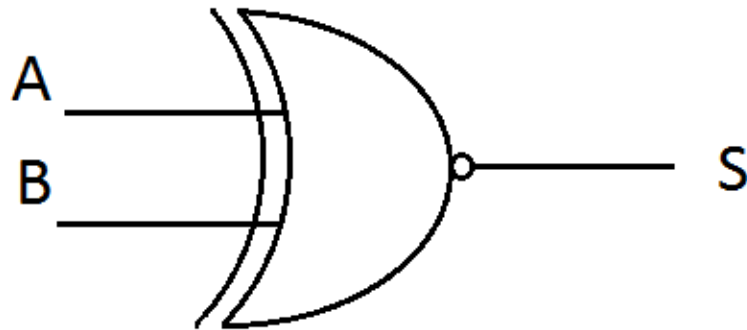
# OUTRAS PORTAS LÓGICAS



Porta XOR  
(OU exclusivo)

TABELA VERDADE		
A	B	S
0	0	0
0	1	1
1	0	1
1	1	0

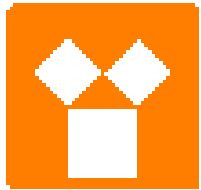
$$S = A \oplus B$$



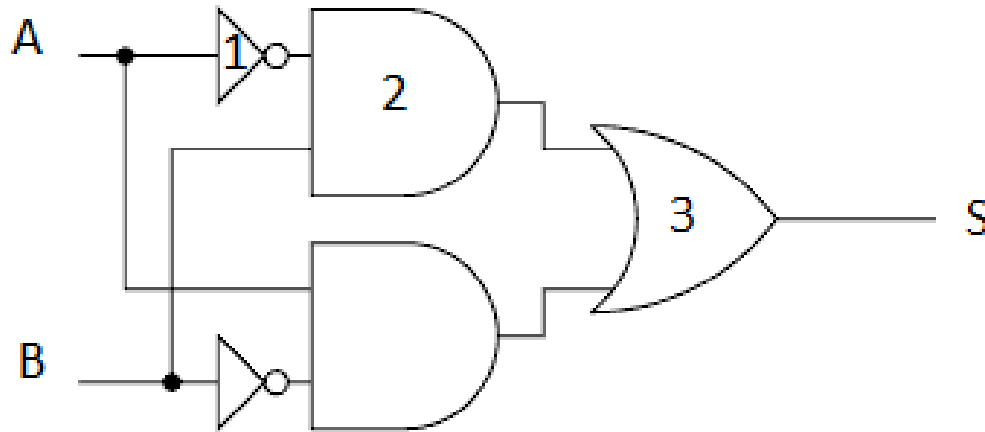
Porta XNOR  
(NÃO OU exclusivo)

TABELA VERDADE		
A	B	S
0	0	1
0	1	0
1	0	0
1	1	1

$$S = \overline{A \oplus B}$$



# EXERCÍCIO DE FIXAÇÃO 1

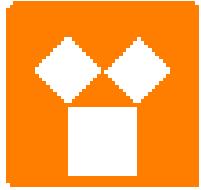


1. [Fácil]: Quais são os nomes das portas lógicas numeradas da figura ao lado?

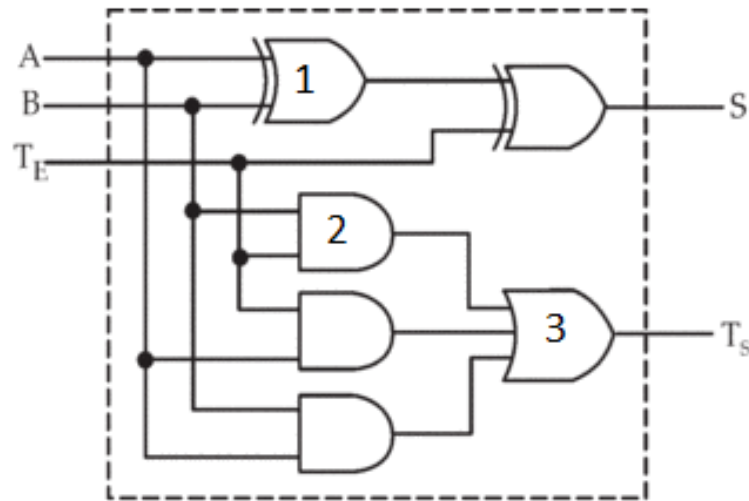
2. [Desafio]: Faça a tabela verdade para a saída S quando as entradas (A e B) forem [0,0], [0,1], [1,0] e [1,1].

3. [Fácil]: Qual das portas lógicas anteriores possui a mesma tabela verdade deste circuito?

4. [Muito, muito fácil]: Existe alguma coisa mais fácil de se fazer que reconhecer portas lógicas?



## EXERCÍCIO DE FIXAÇÃO 2



Somador completo (2 bits)

1. [Fácil]: Quais os nomes das portas lógicas numeradas da figura ao lado?

2. [Fácil]: Faça o desenho de uma porta OR com três entradas e uma saída.

3. [Médio]: Faça a tabela verdade para a porta lógica do exercício anterior.

4. [Fácil]: Se você tivesse que escolher apenas uma entre duas pessoas para ser seu padrinho, que porta lógica você teria que usar para ajudar a você tomar uma decisão?