# Portas e Circuitos Lógicos

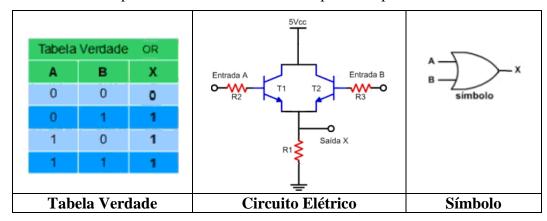
Para entender o funcionamento das Portas e Circuitos Lógicos, será usada uma Expressão Textual (afirmativa) que indica uma situação corrente e seguida da análise e a representação esquemática com um transistor que pode controlar os zeros e uns.

Assim pode-se entender o funcionamento das Portas Lógicas Básicas: E, OU e NÃO.

#### Porta OU

Considere a seguinte afirmativa:

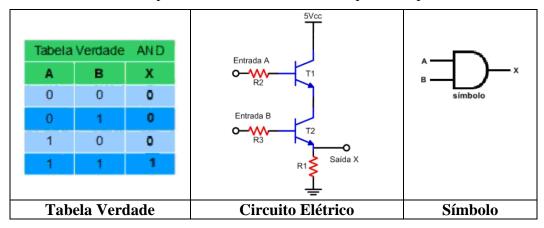
O circuito é o que executa a função OR, ou seja, a implementação eletrônica da porta lógica OR. Observe a imagem abaixo, sabendo que em caso Verdadeiro tem-se o sinal "1", situação que indica "foi" ao local indicado por A ou por B, e em caso Falso tem-se o sinal "0", neste caso indica que "não foi" ao local indicado por A ou por B:



#### Porta E

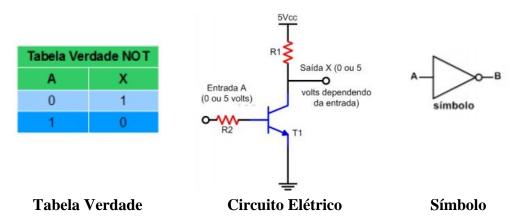
Considere a seguinte afirmativa:

O circuito simples é o que executa a função AND, ou seja, a implementação eletrônica da porta lógica AND. Observe a imagem abaixo, sabendo que em caso Verdadeiro tem-se o sinal "1", situação que indica "foi" ao local indicado por A ou por B, e em caso Falso tem-se o sinal "0", neste caso indica que "não foi" ao local indicado por A ou por B:



## Porta Não

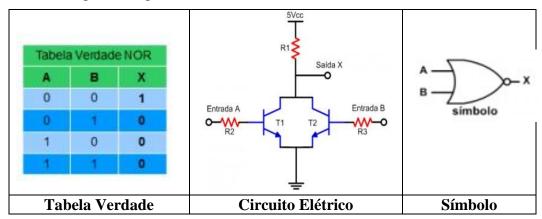
Um dos circuitos mais simples é o que executa a função NOT, ou seja, a implementação eletrônica da porta lógica NOT. Observe a imagem abaixo, sabendo que em caso Verdadeiro tem-se o sinal "1", situação que indica "foi" ao local indicado por A e em caso Falso tem-se o sinal "0", neste caso indica que "não foi" ao local indicado por A. O circuito Não, simplesmente gera uma Inversão de estados, "0" → "1" e "1" → "0":



#### Porta Não-OU

Considere a seguinte afirmativa:

Neste sentido têm-se outros circuitos, é aquele que executa a função NOR, ou seja, a implementação eletrônica da porta lógica NOT OR. Como saída da Porta tem-se a Negação de qualquer saída padrão após a aplicação do Circuito OU, ou seja, "0, 1, 1, 1" após a Negação fica "1, 0, 0, 0", segue a imagem:



#### Porta Não-E

Considere a seguinte afirmativa:

Ainda, tem-se o circuito que executa a função NAND, neste caso a implementação eletrônica da porta lógica NOT AND. Como saída da Porta tem-se a Negação de qualquer saída padrão após a aplicação do Circuito E mostrado acima, ou seja, "0, 0, 0, 1" após a Negação fica "1, 1, 1, 0", segue a imagem:



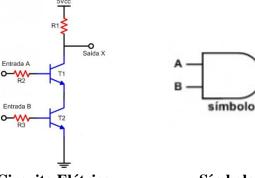


Tabela Verdade

Circuito Elétrico

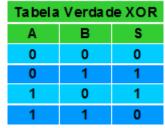
Símbolo

## Porta OU-Exclusivo

Considere a seguinte afirmativa:

Mamãe: Eu **OU vou a Escola OU vou ao Cinema B** 

Ainda, tem-se o circuito que executa a função XOR, neste caso a implementação eletrônica das portas lógicas AND, NOT e OR conforme a Circuito Lógico. Como saída da Porta tem-se a Validação apenas quando apenas A ou B são Verdade "1", o que acontece apenas no segundo e terceiros estados. A porta Lógica é representada como sendo  $S = A \oplus B$ , segue a imagem:



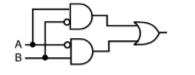




Tabela Verdade

Circuito Lógico

Símbolo

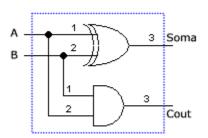
## Aritmética Binária

# Meio somador – apenas 1 Bit

A	В	S=A⊕B	Vai 1
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$S=A \oplus B = \overline{A} \cdot B + A \cdot \overline{B}$$
Vai 1 =  $A \cdot B$ 





Circuito Meio Somador

# Circuito Somador Completo de 1 Bit

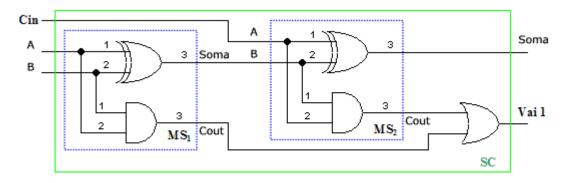
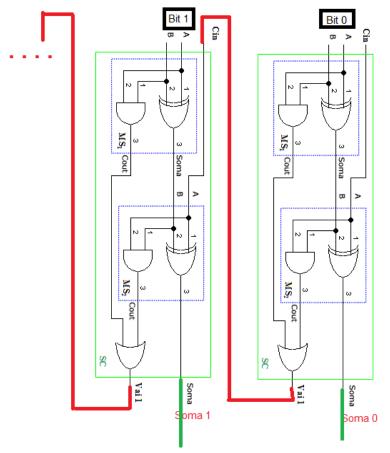


Tabela Verdade – Somador Completo

Cin	A	В	Soma MS <sub>1</sub>	Cout MS <sub>1</sub>	Cout MS <sub>2</sub>	Soma	Vai 1
0	0	0	0	0	0	0	0
0	0	1	1	0	0	1	0
0	1	0	1	0	0	1	0
0	1	1	0	1	0	0	1
1	0	0	0	0	0	1	0
1	0	1	1	0	1	0	1
1	1	0	1	0	1	0	1
1	1	1	0	1	0	1	1



Somador com 2 bits completos ..