

## 6. FUNÇÕES LÓGICAS (portas lógicas)

### 6.1. FUNÇÕES: E, OU, NÃO, NE e NOU

Nas funções lógicas, teremos apenas dois estados:

- o estado 0 (zero) = DESLIGADO
- o estado 1 (um) = LIGADO

O estado “0” representará, por exemplo: DESLIGADO, ou seja, portão fechado, aparelho desligado, ausência de tensão, chave aberta, não, etc.;

O estado “1” representará, por exemplo: LIGADO, ou seja, portão aberto, aparelho ligado, presença de tensão, chave fechada, sim, etc.;

#### ESTADO “0”

Desligado  
Portão Fechado  
Aparelho Desligado  
Ausência de Tensão  
Chave Aberta  
Não

#### ESTADO “1”

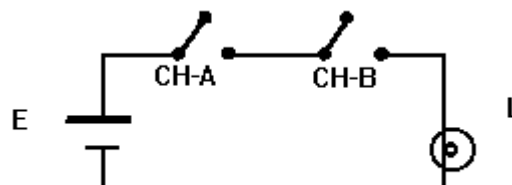
Ligado  
Portão Aberto  
Aparelho Ligado  
Presença de Tensão  
Chave Fechada  
Sim

Notação: se representarmos por zero (0) uma situação, representaremos por um (1) a situação contrária.

#### 6.1.1. Função “E” ou “AND” ( $\wedge$ )

A função “E” é aquela que executa a multiplicação de duas ou mais variáveis.

Representação Algébrica:  $S = A \bullet B$ , onde se lê:  $S = A$  e  $B$ .



Convenções: chave aberta = 0  
lâmpada apagada = 0

chave fechada = 1  
lâmpada acesa = 1

## Situações Possíveis:

- 1ª) Se CH-A = 0 e CH-B = 0, lâmpada apagada.  $A=0, B=0, S= A \bullet B = 0$
- 2ª) Se CH-A = 0 e CH-B = 1, lâmpada apagada.  $A=0, B=1, S= A \bullet B = 0$
- 3ª) Se CH-A = 1 e CH-B = 0, lâmpada apagada.  $A=1, B=0, S= A \bullet B = 0$
- 4ª) Se CH-A = 1 e CH-B = 1, lâmpada acesa.  $A=1, B=1, S= A \bullet B = 1$

Portanto, só teremos a lâmpada acesa quando as chaves A e B estiverem fechadas (1 e 1).

## Tabela da Verdade - “E” ou “AND”

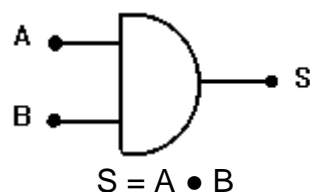
A	B	S
0	0	0
0	1	0
1	0	0
1	1	1

$$N = 2 \rightarrow 2^N \text{ Linhas} = 2^2 = 4 \text{ linhas}$$

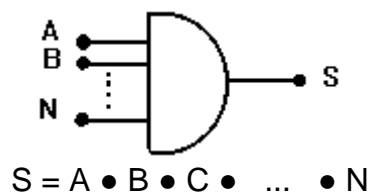
A tabela da verdade é um “mapa” onde colocamos todas as possíveis situações com seus respectivos resultados.

## Porta “E” ou “AND”

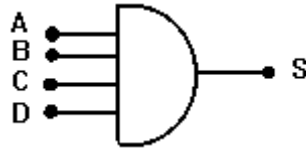
A porta lógica é um circuito que executa a função “E”.



Uma porta **AND** de N entradas terá saída “1”, se e somente se, todas as entradas forem iguais a “1”, e terá saída “0” nos demais casos.



Exemplos: Vamos mostrar uma porta **AND** de quatro entradas e sua tabela da verdade.



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

#### 6.1.2. Função “OU” ou “OR” ( $\vee$ )

A função “**OU**” é aquela que executa a adição de duas ou mais variáveis.

Representação Algébrica:  $S = A + B$ , onde se lê  $\Rightarrow S = A$  **ou**  $B$ .

Situações Possíveis:

- 1ª) Se CH-A = 0 e CH-B = 0, lâmpada apagada.  $A=0, B=0, S= A + B = 0$
- 2ª) Se CH-A = 0 e CH-B = 1, lâmpada acesa.  $A=0, B=1, S= A + B = 1$
- 3ª) Se CH-A = 1 e CH-B = 0, lâmpada acesa.  $A=1, B=0, S= A + B = 1$
- 4ª) Se CH-A = 1 e CH-B = 1, lâmpada acesa.  $A=1, B=1, S= A + B = 1$

Portanto, só teremos a lâmpada apagada quando as chaves A e B estiverem abertas (0 e 0).

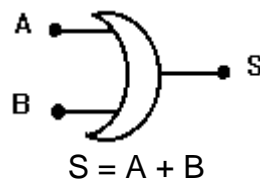
## Tabela da Verdade - “OU” ou “OR”

A	B	S
0	0	0
0	1	1
1	0	1
1	1	1

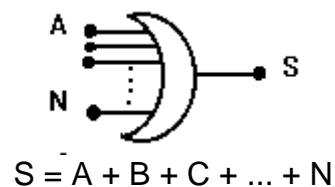
$$N = 2 \rightarrow 2^N \text{ Linhas} = 2^2 = 4 \text{ linhas}$$

## Porta “OU” ou “OR”

A porta lógica é um circuito que executa a função “OU”.



Uma porta **OR** de N entradas terá saída “0”, se e somente se, todas as entradas forem iguais a “0”, e terá saída “1” nos demais casos.



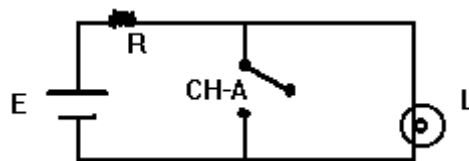
Exemplos: Vamos mostrar uma porta **OR** de três entradas e sua tabela da verdade.

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

### 6.1.3. Função “NÃO” ou “NOT” ( $\neg$ )

A função **NÃO** ou função complemento é aquela que inverte o estado da variável.

Representação:  $S = A$  ou  $S = A'$ , onde se lê ( $A$  barra ou apóstrofo) ou (NÃO  $A$ ).



Situações Possíveis:

1ª) Se  $CH-A = 0$ , então  $A = 0$ ,  $A = 1$ , lâmpada acesa.

2ª) Se  $CH-A = 1$ , haverá um curto circuito  $A = 1$ ,  $A = 0$ , lâmpada apagada.

### Tabela da Verdade - “NÃO” ou “NOT”

A	A
0	1
1	0

### Inversor

É o bloco lógico que executa a função **NOT (NÃO)** ( $'$ ) ( $\neg$ ) ( $\neg$ )

Representação:  $A \text{ -----} \rightarrow O \text{-----} A'$  ou  $\bar{A}$

 ----- após um bloco lógico    -----  antes de um bloco lógico

No caso do inversor, pode-se ter somente uma entrada e uma saída.

### 6.1.4. Função “NÃO E” ou “NE” ou “NAND”

É uma composição da função **E** com a função **NÃO**, ou seja, teremos a função **E** invertida.

Representação Algébrica:  $S = (A \bullet B)'$

## Situações Possíveis:

- 1ª) Se CH-A = 0 e CH-B = 0, lâmpada acesa.  $A=0, B=0, S = (A \bullet B)' = 1$   
 2ª) Se CH-A = 0 e CH-B = 1, lâmpada acesa.  $A=0, B=1, S = (A \bullet B)' = 1$   
 3ª) Se CH-A = 1 e CH-B = 0, lâmpada acesa.  $A=1, B=0, S = (A \bullet B)' = 1$   
 4ª) Se CH-A = 1 e CH-B = 1, lâmpada apagada.  $A=1, B=1, S = (A \bullet B)' = 0$

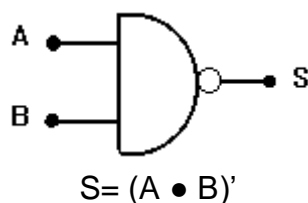
Portanto, só teremos a lâmpada apagada quando as chaves A e B estiverem abertas (1 e 1).

## Tabela da Verdade - “NE” ou “NAND”

A	B	S
0	0	1
0	1	1
1	0	1
1	1	0

## Porta “NE” ou “NAND”

A porta lógica é um circuito que executa a função “NE”



A porta **NAND** é a composição da porta **AND** com um inversor ligado à sua saída. A porta **NAND** pode ter duas ou mais entrada, assim como os outros blocos lógicos.

### 6.1.5. Função “NÃO OU” ou “NOU” ou “NOR”

É uma composição da função **NÃO** com a função **OU**, ou seja, a função **NOU** será o inverso da função **OU**.

Representação Algébrica:  $S = (A + B)'$

## Situações Possíveis:

- 1ª) Se CH-A = 0 e CH-B = 0, lâmpada acesa.  $A=0, B=0, S = (A + B)' = 1$   
 2ª) Se CH-A = 0 e CH-B = 1, lâmpada apagada.  $A=0, B=1, S = (A + B)' = 0$   
 3ª) Se CH-A = 1 e CH-B = 0, lâmpada apagada.  $A=1, B=0, S = (A + B)' = 0$   
 4ª) Se CH-A = 1 e CH-B = 1, lâmpada apagada.  $A=1, B=1, S = (A + B)' = 0$

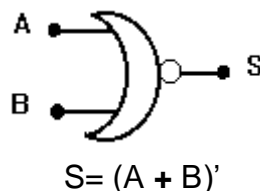
Portanto, só teremos a lâmpada acesa quando as chaves A e B estiverem abertas (0 e 0).

## Tabela da Verdade - “NOU” ou “NOR”

A	B	S
0	0	1
0	1	0
1	0	0
1	1	0

## Porta “NOU” ou “NOR”

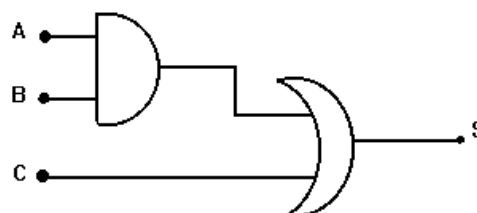
A porta lógica é um circuito que executa a função “NOR”



A porta **NOR** é a composição da porta **OR** com um inversor ligado à sua saída. A porta **NOR** pode ter duas ou mais entrada, assim como os outros blocos lógicos.

## 6.2. EXPRESSÕES BOOLEANAS GERADAS POR CIRCUITOS LÓGICOS:

Vejamos qual a expressão que o circuito abaixo executa:



## Circuitos Obtidos de Expressões Booleanas.

Vimos até agora que podemos obter uma expressão booleana que um circuito lógico executa. Vamos estudar que a partir de uma expressão booleana podemos desenhar um circuito.

Por exemplo, obter o circuito lógico da seguinte expressão:

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

## EXERCÍCIOS (valendo pontos para a avaliação/prova)

1) Desenhe os circuitos lógicos que executam as seguintes expressões booleanas:

a)  $S = [(A \bullet B) + (C \bullet D)]'$

b)  $S = (A \bullet B) + C + (C \bullet D)$

c)  $S = (A \bullet B \bullet C) + ((A+B) \bullet C)$

d)  $S = \{[(A \bullet B \bullet C) + B + C]' \bullet [A' + C + D + (B \bullet C' \bullet D)] \bullet [A + B + C + (B \bullet C' \bullet D)]'\}$

e)  $S = [(A + B' + C)' \bullet (A + D' + B)]' \bullet A' \bullet B \bullet C'$

f)  $S = [(A' + B) \bullet (C + D')] + D' + [B \bullet D + D']'$

g)  $S = [(A' \bullet B) \bullet (B \bullet C)' \bullet (B + D)]'$

h)  $S = [(A' \bullet B)' + (A \bullet B')' + C]' \bullet (C + D)$

i)  $S = A \bullet B \bullet C + A \bullet D + A \bullet B \bullet D$

j)  $S = (A' + B) + (A \bullet B \bullet C')$

l)  $S = A \bullet B \bullet C + A \bullet B' \bullet C + A' \bullet B' \bullet C + A' \bullet B' \bullet C'$

m)  $S = [(A + B) \bullet C]' + [D \bullet (C + B)]'$

n)  $S = [A \bullet C' + D + B]' + [C \bullet (A \bullet C \bullet D)]'$