

PORTAS LÓGICAS OR, AND e NOT

O que é uma porta lógica?

Uma porta lógica poderá ajudá-lo a tomar uma decisão lógica. Em eletrônica digital, uma porta lógica pode ser definida como um circuito com somente uma saída, mas, duas ou mais entradas.

Um sinal aparecerá na saída da porta somente para certa combinação de sinais aplicados nas entradas.

Estes sinais são denominados níveis lógicos, conforme descrito abaixo:

Nível lógico 0 – ou condição FALSO
Nível lógico 1 – ou condição VERDADEIRO

O nível lógico 0 pode ser representado por uma entrada aterrada, ou seja, ligada diretamente a um referencial normalmente denominado terra ou GND (ausência de tensão), enquanto que o nível lógico 1 é representado por uma tensão aplicada à entrada (normalmente 5 volts para circuitos integrados da família TTL).

A partir das portas OR, AND e NOT são desenvolvidas outras portas lógicas. Vejamos então o que significam essas funções:

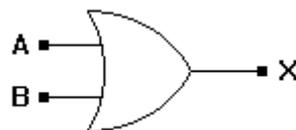
OR (OU) – é uma função representada pelo símbolo matemático "+". A porta OR também é chamada de disjunção lógica.

AND (E) – é uma função representada matematicamente como sendo um produto ".", também chamada de conjunção lógica.

NOT (NÃO) – é uma função de negação ou inversão de uma variável na entrada

PORTA OR

A figura abaixo mostra a simbologia de uma porta OR de duas entradas.



As entradas que denominamos A e B, combinam as variáveis e apresentam um resultado na saída X, conforme mostra a tabela da verdade abaixo:

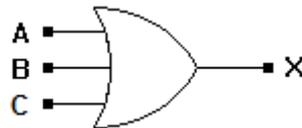
ENTRADAS		SAÍDA
A	B	
0	0	0
0	1	1
1	0	1
1	1	1

Como a porta OR desempenha uma função representada matematicamente pela soma "+", a expressão na saída ficará:

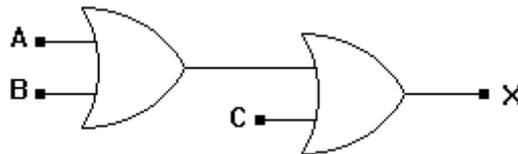
$$X = A + B$$

A figura abaixo mostra uma porta OR com 3 entradas (A, B e C), onde a expressão na saída será:

$$X = A + B + C$$



Podemos implementar uma porta OR com 3 entradas a partir de duas portas de 2 entradas, entradas essas que definimos como A, B e C:



$$X = (A + B) + C, \text{ que é a mesma coisa que: } X = A + B + C$$

A tabela da verdade abaixo satisfaz as duas condições, ou seja, por OR com 3 entradas ou porta OR implementada com duas portas OR de 2 entradas.

ENTRADAS			SAÍDA
A	B	C	
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

Para a tabela da verdade acima, teremos a expressão:

$$X = A + B + C$$

Analisando as tabelas com 2 e 3 variáveis na entrada, podemos definir, levando-se em consideração que o nível lógico 0 representa uma condição FALSA e o nível lógico 1 uma condição VERDADEIRA:

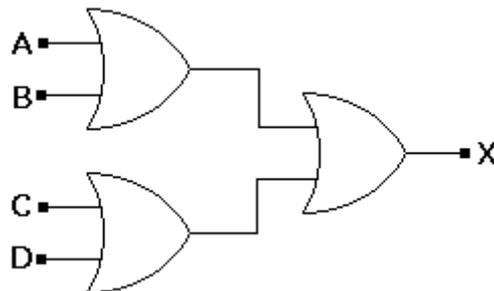
“Na saída teremos um valor lógico FALSO se, e somente se, todas as entradas tiverem o valor lógico FALSO”

Assim, se apenas uma das entradas apresentar a condição VERDADEIRA a saída apresentará uma condição VERDADEIRA.

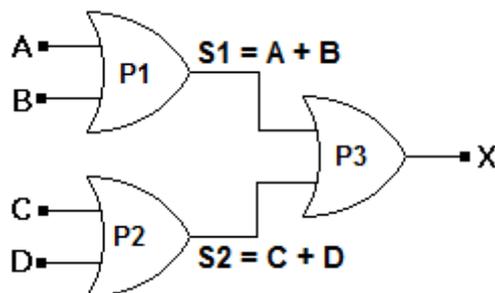
Da mesma forma, se todas as entradas apresentarem uma condição verdadeira, a saída apresentará uma condição verdadeira.

Resumindo: Em uma porta OR com duas ou mais entradas, a saída será 0 somente se todas as entradas forem igual a 0. Se apenas uma das entradas for submetida a nível lógico 1, teremos na saída nível lógico 1.

Vejamos então como fica a expressão de um circuito com portas OR um pouco mais complicado:



A expressão na saída será: $X = (A+B) + (C+D)$



As portas 1, 2 e 3 estão identificadas como P1, P2 e P3.

Veja a tabela da verdade para esse circuito:

Entrada P1		S1	Entrada P2		S2	Entrada P3		X
A	B		C	D		S1	S2	
0	0	0	1	1	1	0	1	1
0	1	1	0	1	1	1	1	1
1	0	1	0	0	0	1	0	1
1	1	1	1	1	1	1	1	1

Modificando os níveis lógicos nas entradas, teremos outros valores para a saída “X”.

Veja como fica a tabela da verdade com essas alterações. É importante salientar que o que interessa no circuito em ambos os casos, é o que mostra a saída "X".

Entrada P1		S1	Entrada P2		S2	Entrada P3		X
A	B		C	D		S1	S2	
0	0	0	1	1	1	0	1	1
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
1	0	1	1	1	1	1	1	1

As tabelas da verdade mostradas sugerem que o circuito se comporta de forma idêntica a uma porta OR de 4 entradas.

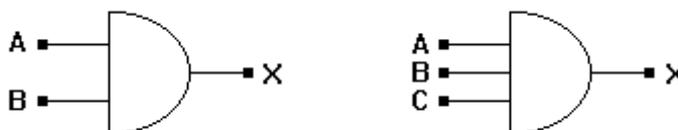
PORTA AND

Uma porta AND é projetada para que exista sinal de saída se existir sinais em todas as entradas.

A porta AND pode ser considerada como uma porta de *tudo ou nada*, pois é necessário que exista 1 em todas as entradas para que a saída seja 1.

Em contrapartida em uma porta OR, qualquer nível lógico 1 na entrada leva a saída para 1.

Veja a simbologia para uma porta AND de 2 e 3 entradas:



ENTRADAS		SAÍDA
A	B	
0	0	0
0	1	0
1	0	0
1	1	1

Para duas entradas a expressão na saída será: $X = A \cdot B$

ENTRADAS			SAÍDA
A	B	C	
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

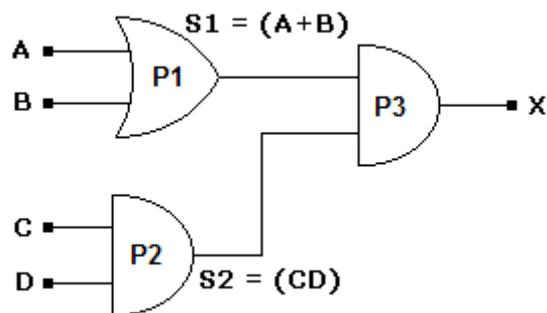
Teremos então para 3 entradas: $X = A \cdot B \cdot C$

Na prática, para indicar a conjunção "E" ou multiplicação, não há necessidade de escrever o ponto, então:

$$X = AB$$

$$X = ABC$$

A figura abaixo mostra um exemplo de uma porta AND com 2 entradas e respectiva tabela da verdade:



Entrada P1		S1	Entrada P2		S2	Entrada P3		X
A	B		C	D		S1	S2	
0	0	0	1	0	0	0	0	
0	1	1	0	1	0	1	0	
1	0	1	0	0	0	1	0	
1	1	1	1	1	1	1	1	

Modificando os níveis lógicos nas entradas, teremos outros valores de níveis lógicos na saída "X".

Entrada P1		S1	Entrada P2		S2	Entrada P3		X
A	B		C	D		S1	S2	
0	1	1	1	1	1	1	1	
0	0	0	1	0	0	0	0	
1	0	1	0	1	0	1	0	
1	1	1	0	0	0	1	0	

Temos então, para fins de análise dos níveis lógicos apresentados na saída "X":

Porta 1 (P1) = função OR

Porta 2 (P2) = função AND

Porta 3 (P3) = função AND

A análise final, para determinar os valores lógicos presentes na saída "X" fica restrita então a Porta 3 (P3).

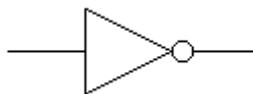
PORTA NOT

A Porta NOT implementa a negação lógica, de acordo com a tabela verdade abaixo.

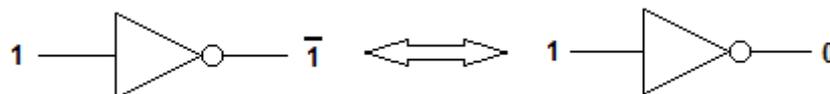
ENTRADA	SAÍDA
0	1
1	0

Uma entrada ALTA (nível lógico 1) resulta em uma saída BAIXA (nível lógico 0) e analogamente uma entrada BAIXA (0) resulta em uma saída ALTA (1). Ou seja, a porta NOT sempre produzirá como saída o inverso de sua entrada.

A simbologia para uma porta NOT é mostrada abaixo:



A saída complementada é identificada com uma barra, conforme ilustram as figuras abaixo:



A barra indica que a variável aplicada na entrada foi complementada ou invertida.

