

Função (Function)

É um componente muito útil, pois permite executar operações lógicas sem a utilização de portas lógicas primárias, ou seja, através de programação.

Resumindo, serve para definir o comportamento de pinos de saída através de expressões matemáticas ou lógicas.

Em vez de montar um circuito complexo com várias portas lógicas ou amplificadores operacionais, você pode usar este bloco para calcular valores diretamente.

Funções que podem ser configuradas:

Logic operations (Operações lógicas):

Not: ! ➔ ! i0 ou ! i1

And: &

Or: |

Xor: ^

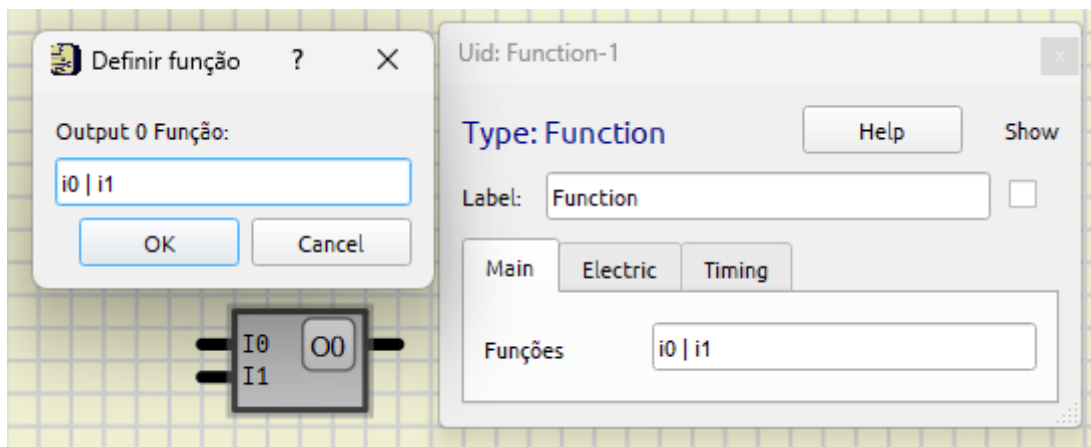
Logic states (Estados lógicos):

- Input (in): "0 / 1"
- Output (out): "0 / 1"

A figura abaixo ilustra uma função OR, com 2 entradas (I0 e I1), através da programação na opção "Funções".

A inserção do componente com duas entradas e função OR, é padrão do simulador, cuja configuração pode ser modificada, de acordo com as necessidades do projeto.

Dentro da área de trabalho para fins de simulação, podem ser criados vários blocos, com variadas operações lógicas, a exemplo do que é possível simular utilizando portas lógicas primárias.

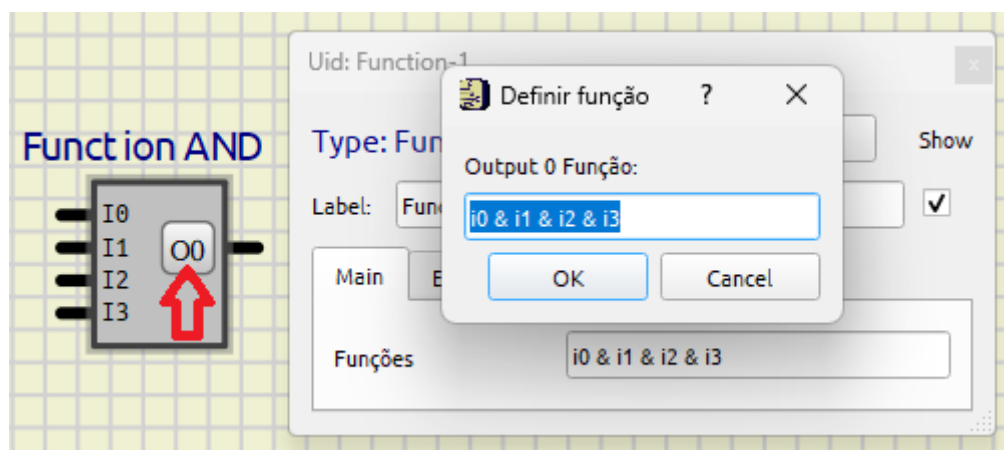
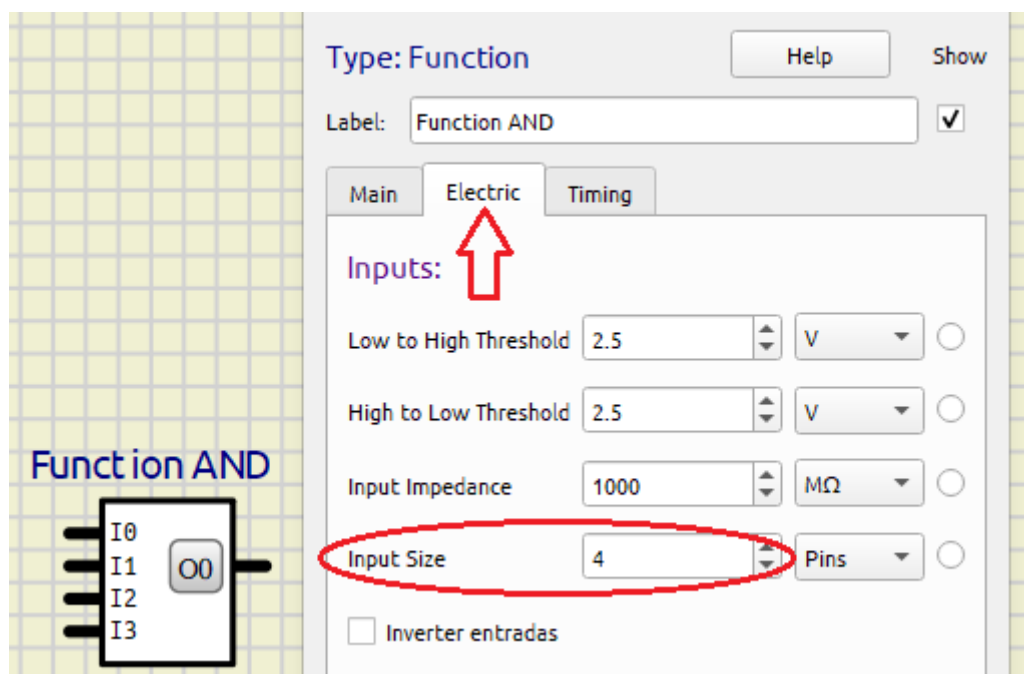


Criando uma função do tipo AND, com 4 entradas:

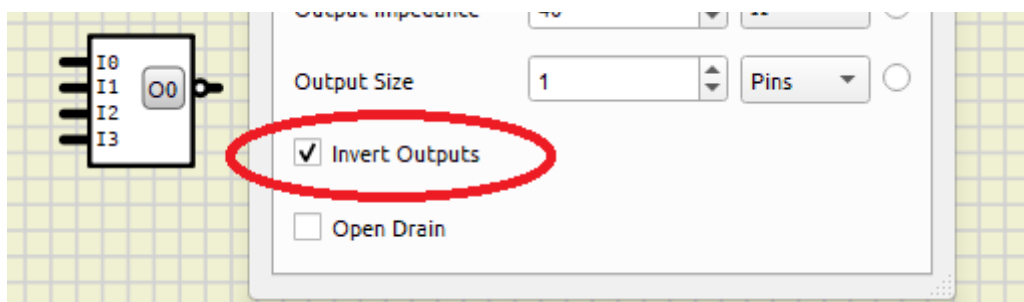
- a) Primeiro passo: na opção "Electric" ajustar "Input Size" para 4. Caso o projeto exija, as entradas podem ser invertidas marcando a opção "Inverter entradas".
- b) Segundo passo: escrever a equação correspondente a operação lógica AND, na opção "Main" no campo "Funções".

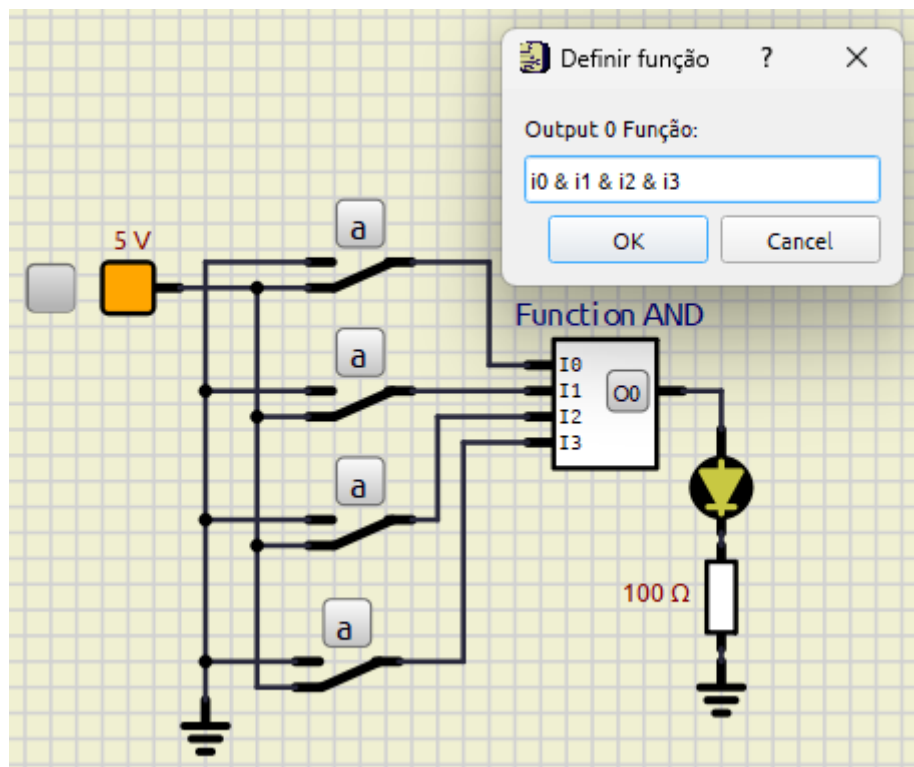
i1 & i2 & i3 & i4

Veja a orientação nas figuras a seguir:



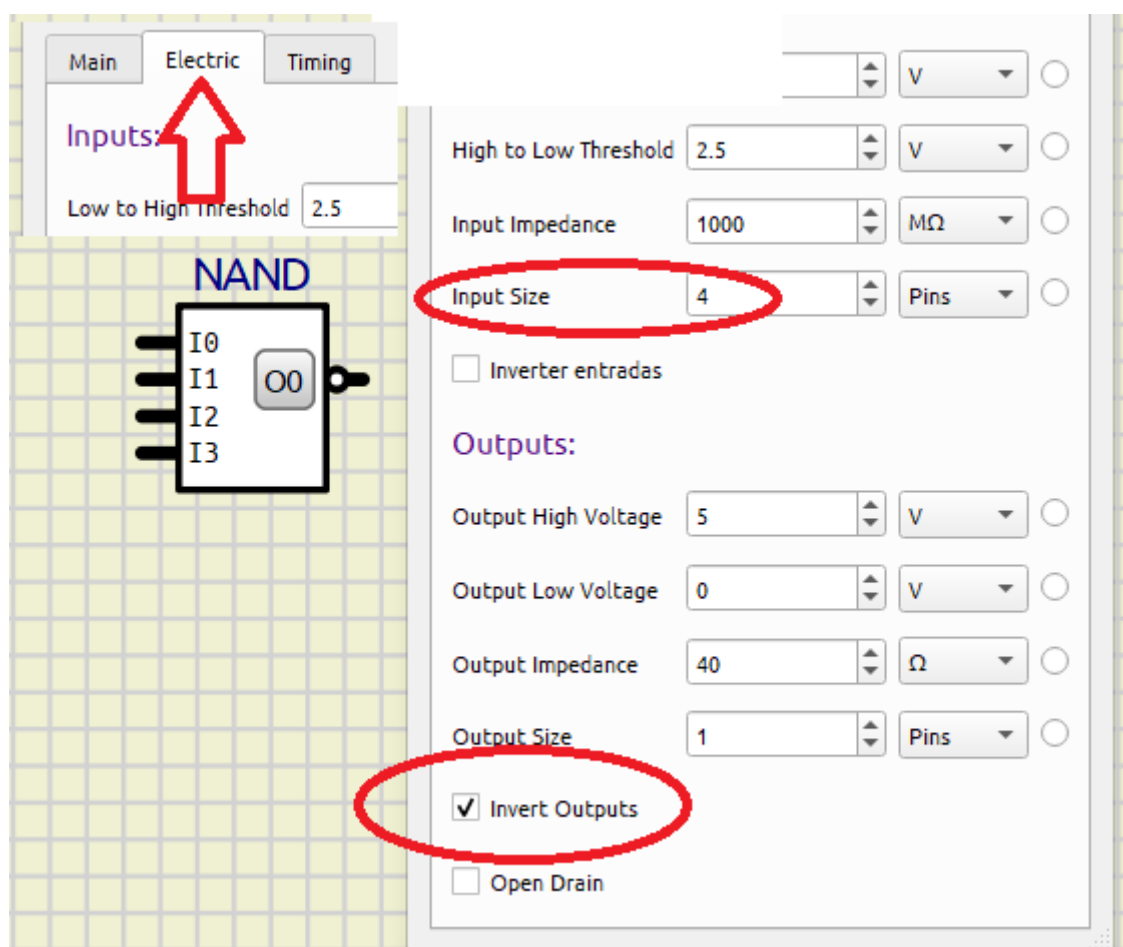
Para fazer a conversão para uma função NAND, não há necessidade de alterar a programação, apenas alterar o parâmetro "Invert Outputs" no guia Electric.





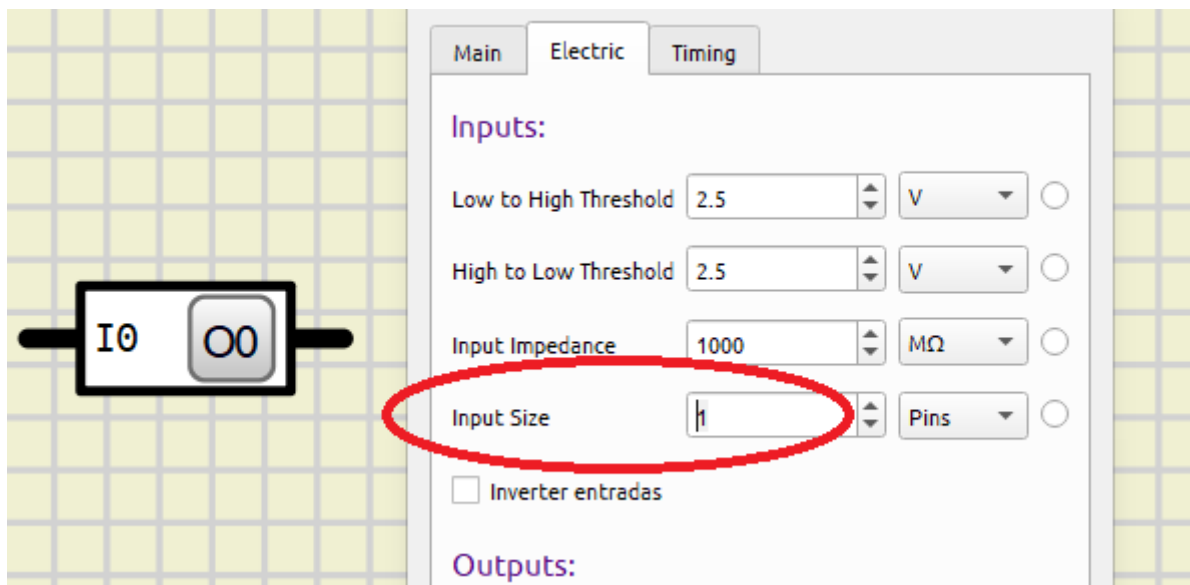
Criando uma função do tipo NAND, com 4 entradas:

Para criar uma função NAND de 4 entradas, apenas uma simples modificação no parâmetro elétrico é necessária ("Invert Outputs"), não havendo a necessidade de alterar a programação, conforme ilustra a figura abaixo:

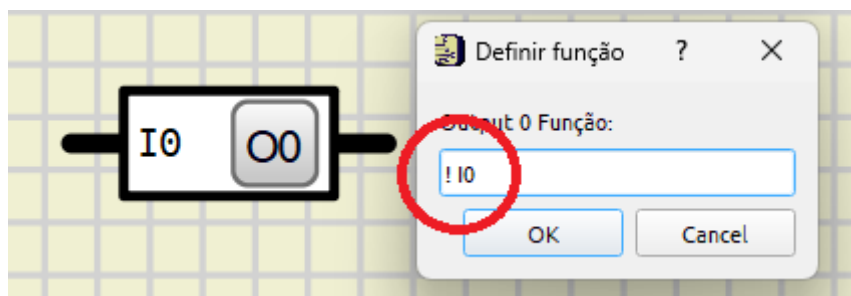


Criando uma função NOT (inversor):

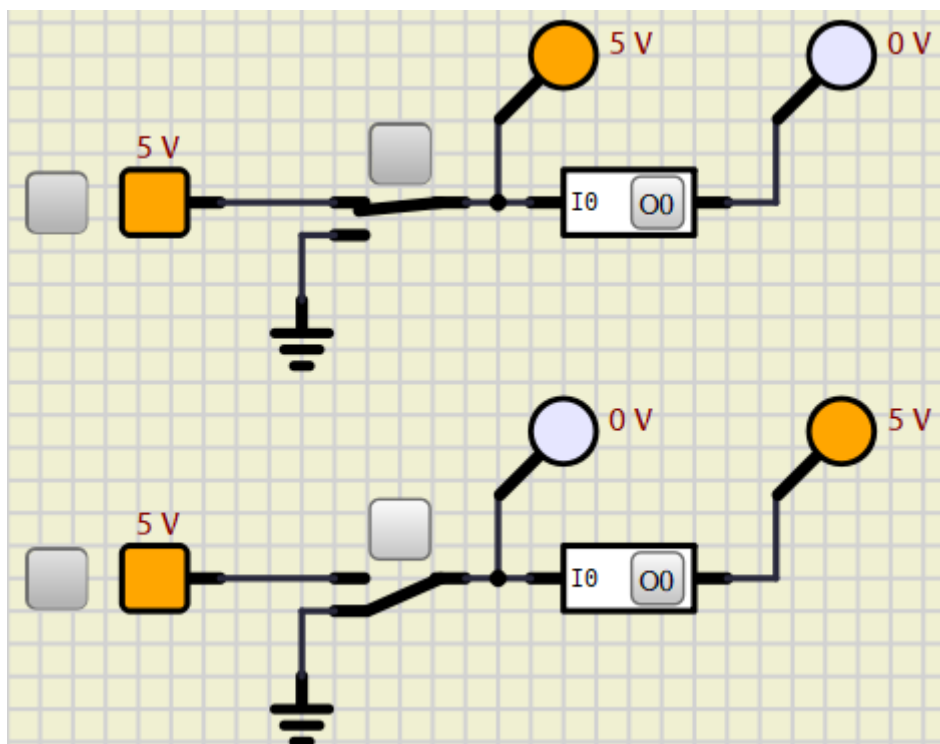
Uma porta NOT ou inversor pode ser criada a partir da configuração do parâmetro elétrico, definindo o tamanho da entrada "Input Size".



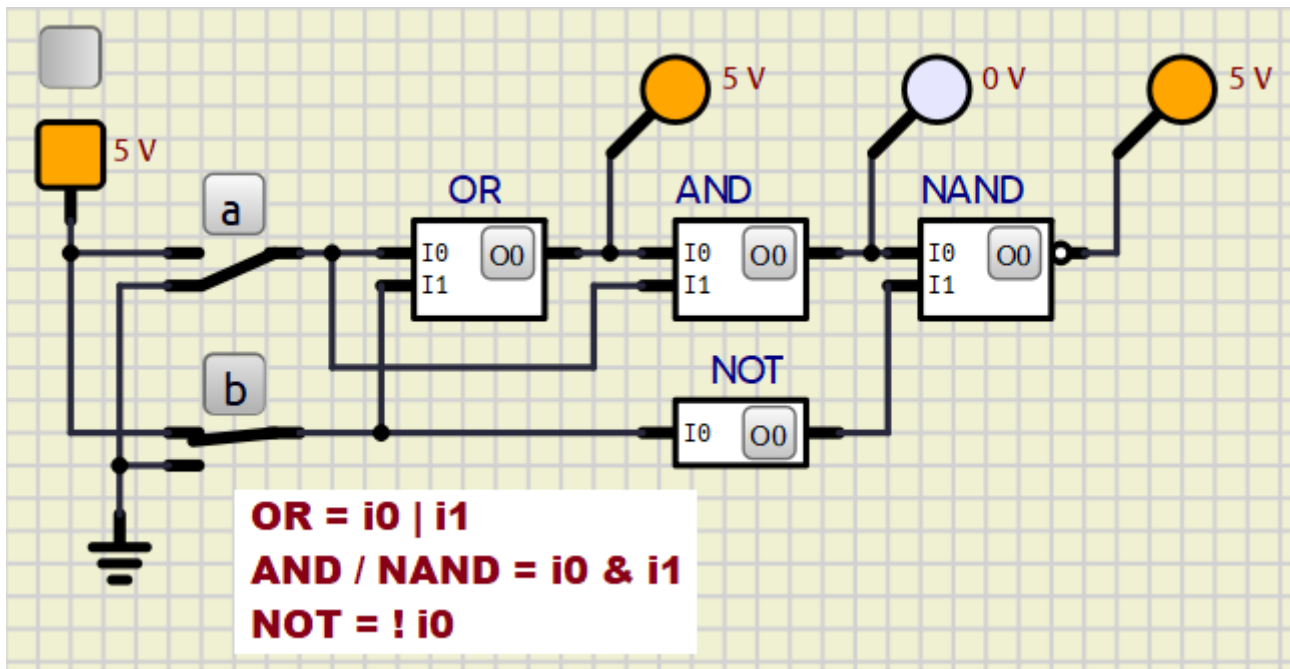
A programação deve ser modificada, com a expressão ! I0



A figura abaixo ilustra uma operação com o uso de uma função NOT:

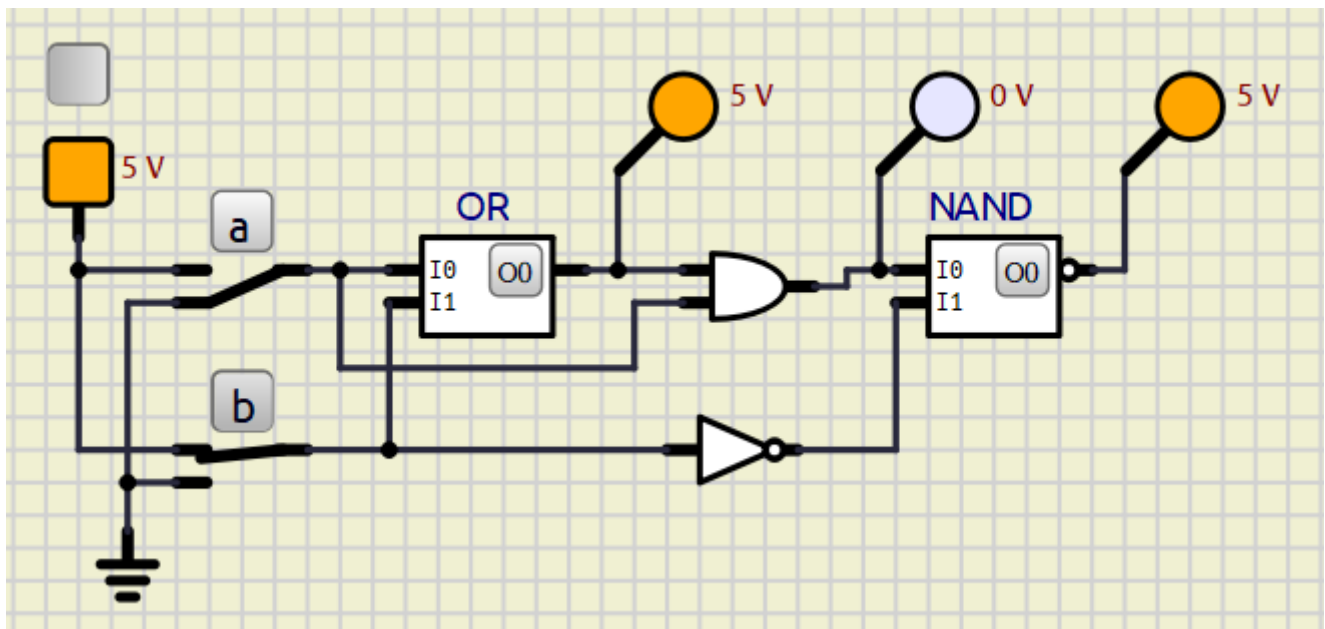


A figura abaixo ilustra um exemplo de simulação com as funções (operações lógicas) OR, AND, NOT e NAND:



É possível também interagir com uma ou mais funções simultaneamente, com portas lógicas primárias, sem alterar o resultado final da simulação.

No exemplo a seguir, as funções AND e NOT foram substituídas por portas lógicas primárias AND e NOT respectivamente, sem que ocorra qualquer alteração no resultado final da simulação.

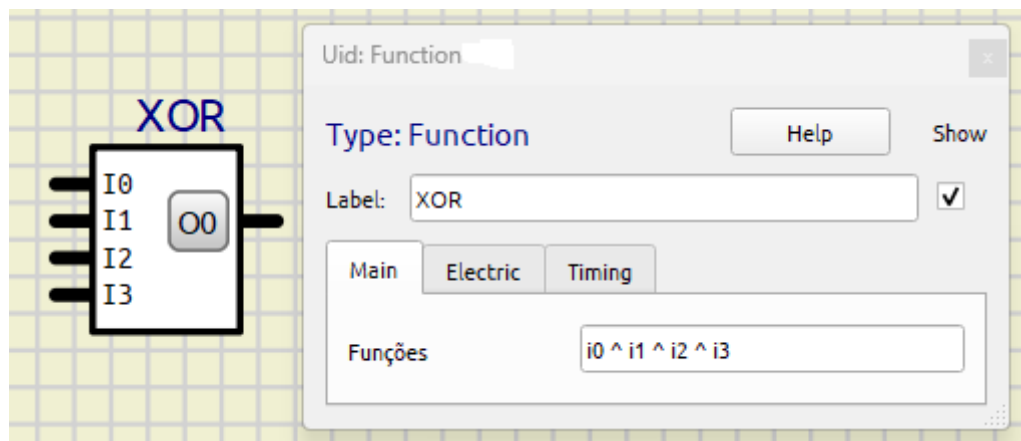
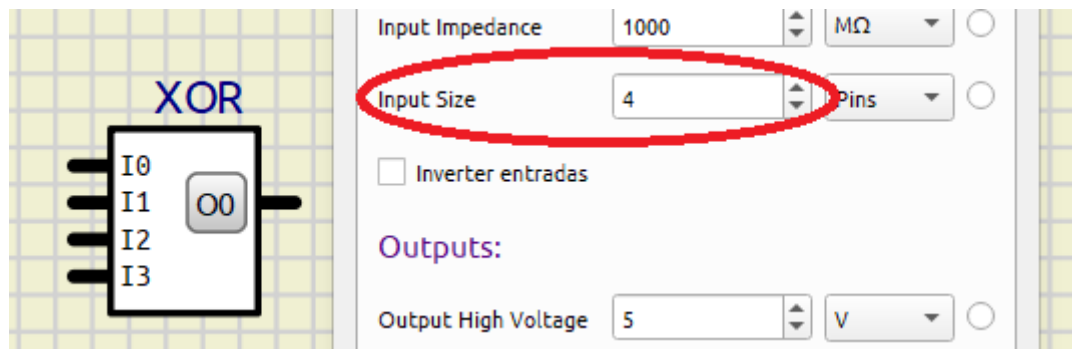


Criando uma função do tipo XOR, com 4 entradas:

Os padrões para a criação de uma função qualquer que seja a quantidade de entradas, obedecem às configurações do parâmetro elétrico "Electric" para a definição da quantidade de entradas.

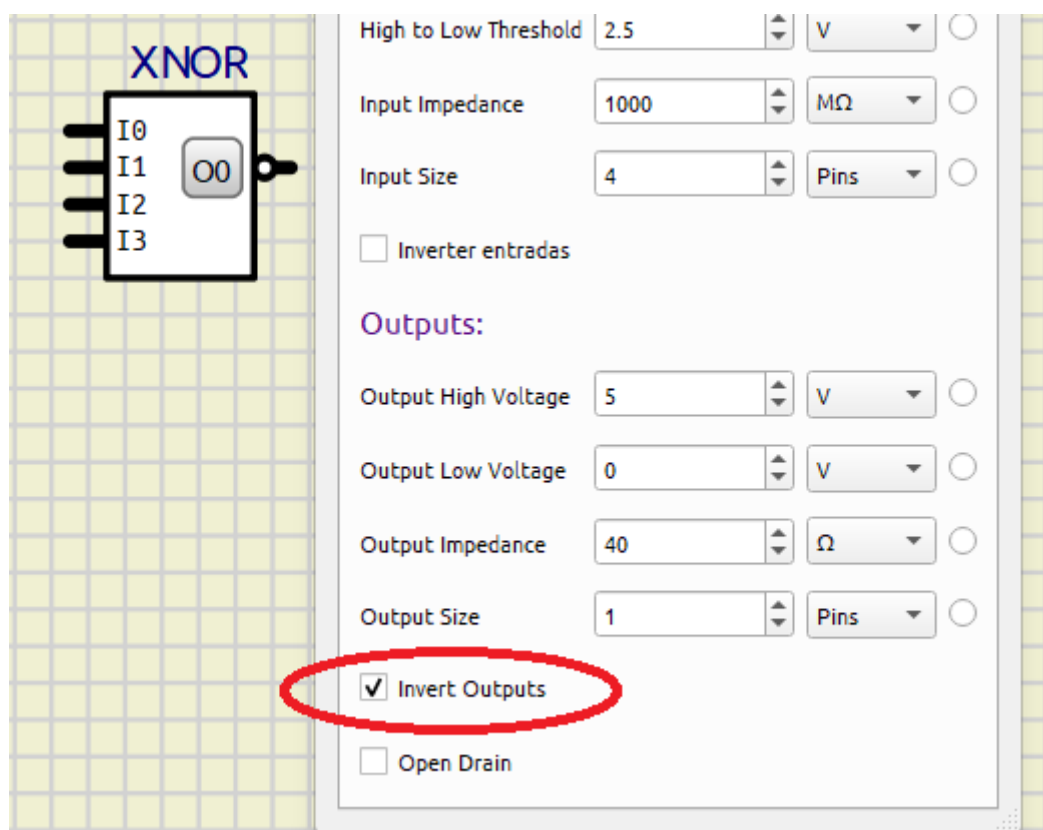
A programação do comportamento da função é essencial.

A figura abaixo ilustra a configuração para a criação da função XOR:

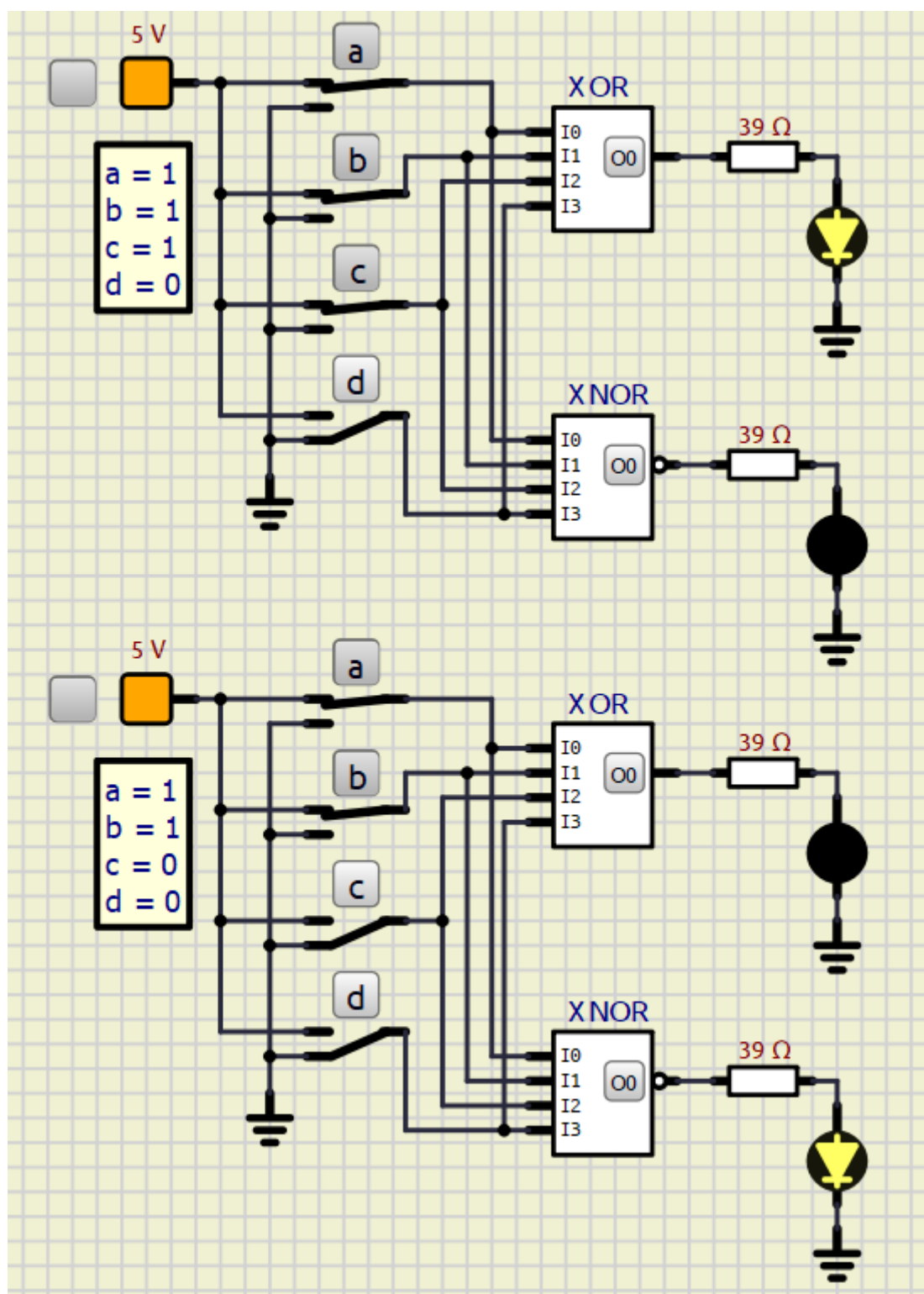


Criando uma função do tipo XNOR, com 4 entradas:

Um função XNOR nada mais é do que o complemento de uma função XOR, por isso, a única modificação a ser feita é na inversão da saída ("Invert Outputs"), sem a necessidade de alterar a programação:



A figura abaixo ilustra a simulação de uma função XOR de 4 entradas, sendo comparada com uma função XNOR também com 4 entradas, onde observa-se a comprovação teórica dessas funções:



Podemos então observar o que ocorre com as funções XOR ("Ou Exclusivo") e XNOR ("Circuito de Coincidência"), comprovando a complementação entre ambas.

XOR - entradas "1" em quantidade ímpar, saída igual a 1

XNOR - entradas "1" em quantidade par, saída igual a 1