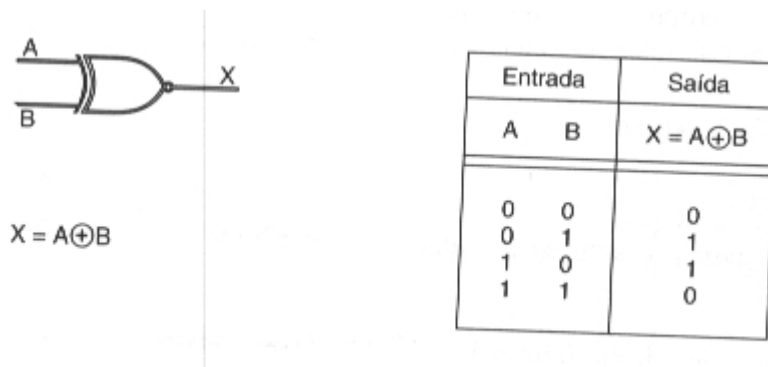


1.5 - Operação Lógica XOR - EXCLUSIVE OR

A operação lógica XOR, abreviação do termo EXCLUSIVE OR, pode ser considerada um caso particular da função OR, ou seja, sua definição: “a saída será verdade se exclusivamente uma ou outra entrada for verdade”. Isto só se aplica, é claro, se houver apenas 2 entradas e o termo exclusivamente é crucial. Não podem ambas as entradas ser verdade e é esta a diferença para os resultados da porta OR (ver tabela).

A figura abaixo mostra os símbolos e a tabela verdade para a operação lógica XOR, e realmente podemos verificar que os resultados mostrados na tabela são iguais aos da operação OR, exceto quanto ao resultado de $0 \oplus 1$. Isto por causa do termo *exclusivo*. No caso da operação lógica OR, não há exclusividade de que uma ou outra entrai sejam verdadeiras, mas no que se refere à operação XOR, sim.

O XOR é um elemento lógico bastante versátil, aparecendo em diversos locais e com diferentes utilidades em circuitos digitais. A saída (resultado) de urna operação XOR é *verdade se os valores de suas entradas forem diferentes e será falsa se os valores das entradas forem iguais*. Esta definição se aplica melhor do que a que mencionamos anteriormente, porque é abrangente para qualquer quantidade de linhas de entrada.



Porta lógica XOR (Exclusive or = Ou exclusivo).

Além disso, esta particularidade dos circuitos lógicos XOR permite que se fabrique um testador de igualdade entre valores, por exemplo, para testar, de modo rápido, se duas palavras são iguais. Se as duas palavras forem iguais, as saídas do circuito XOR serão todas FALSAS. A figura mostra o exemplo completo do teste, que inclui um circuito lógico NOR para receber, como entrada, a saída de todos os circuitos XOR (um para o teste de cada bit das duas palavras) e produzir uma única saída.

Se as palavras forem iguais, isto significa que o bit 0 da palavra 1 e o bit 0

da palavra 2 têm valores iguais e que cada par de bits até o bit (m-1) das duas palavras também são iguais. Isto garante que, se as entradas de cada circuito lógico XOR forem iguais, a sua saída será FALSA. Se todas as entradas do circuito lógico NOR forem FALSAS, então sua saída será VERDADE e este valor (bit 1) é armazenado em um registrador especial de estado da UCP (denominado Flag em algumas UCP ou Código de Condição em outras).

Vamos, em seguida, apresentar alguns exemplos de operações lógicas XOR.

Exemplo

Seja $A = 0$ e $B = 1$. Calcular: $X = A \text{ xor } B$

Solução:

Utilizando os resultados da tabela verdade da figura 4.13, teremos:

$A = 0$ e $B = 1$, então: $0 \text{ e } 1 = 1$

Resultado: $X = A \text{ xor } B = 1$

Exemplo

Seja $A = 11001$ e $B = 11110$. Calcular: $X = A \text{ xor } B$

Solução:

Adotando a tabela verdade da figura acima, teremos:

$11001 \leftarrow A$

$\text{xor}11110 \leftarrow B$

$00111 \leftarrow X$

Resultado: $X = A \text{ xor } B = 00111$

Exemplo

Calcular:

$$X = A \text{ xor } B \text{ xor } C$$

Seja $A = 11001$, $B = 10011$ e $C = 11100$.

Solução:

Aplicando os resultados mostrados na tabela verdade da figura acima, realizaremos o cálculo em 2 etapas.

Primeiro, $T = A \text{ xor } B$ e, em seguida, $X = T \text{ xor } C$.

$$\begin{array}{rcl} 11001 & \leftarrow A & 01010 \leftarrow T \\ \text{xor } 10011 & \leftarrow B & \text{xor } 11100 \leftarrow C \\ \hline 01010 & \leftarrow T & 10110 \leftarrow X \end{array}$$

Resultado: $X = A \text{ xor } B \text{ xor } C = 10110$

