Give me the truth table of the follow boolean function f = not((a and b and c) xor d)

Sources

bitwise operators - XOR from only OR and AND - Stack Overflow
stackoverflow · 1

Boolean logical operators - AND, OR, NOT, XOR - Microsoft Learn
learn.microsoft · 2

Why there is no logical XOR
reddit · 3

Answer

To create the truth table for the given Boolean function f = not((a and b and c) xor d), we need to evaluate the function for all possible combinations of input variables a, b, c, and d. Let's break this down step by step:

a	b	c	d	(a and b and c)	(a and b and c) xor d	f = not((a and b and c) xor d)
0	0	0	0	0	0	1
0	0	0	1	0	1	0
0	0	1	0	0	0	1
0	0	1	1	0	1	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	1	0
1	0	0	0	0	0	1
1	0	0	1	0	1	0
1	0	1	0	0	0	1
1	0	1	1	0	1	0
1	1	0	0	0	0	1
1	1	0	1	0	1	0
1	1	1	0	1	1	0
1	1	1	1	1	0	1

Explanation

1. The first three columns (a, b, c) represent all possible combinations of binary inputs for these variables.
2. The fourth column (d) represents the two possible values for d.
3. The fifth column calculates (a and b and c). This is only true (1) when a, b, and c are all 1.

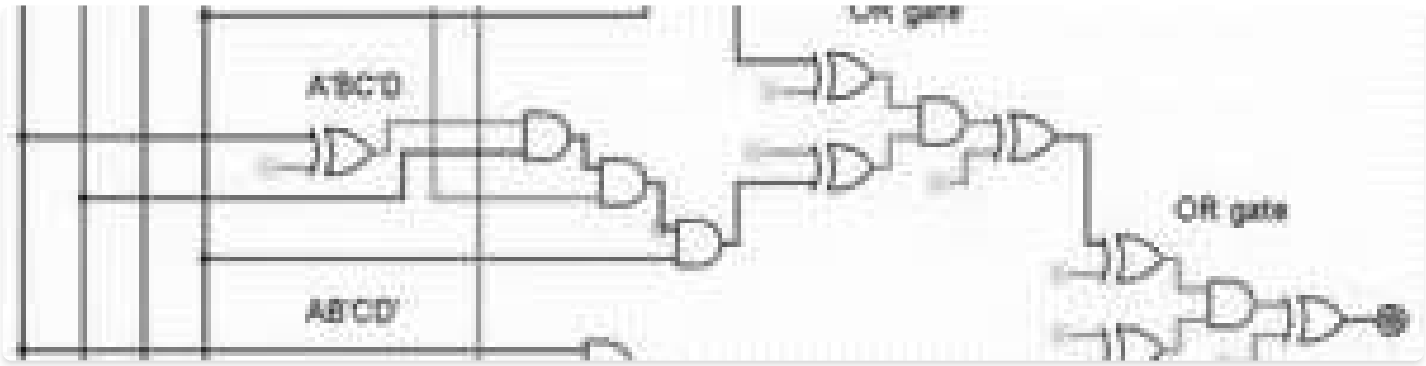
4. The sixth column calculates (a and b and c) xor d. XOR is true when the inputs are different.
5. The last column is the final result f, which is the negation (not) of the previous column.

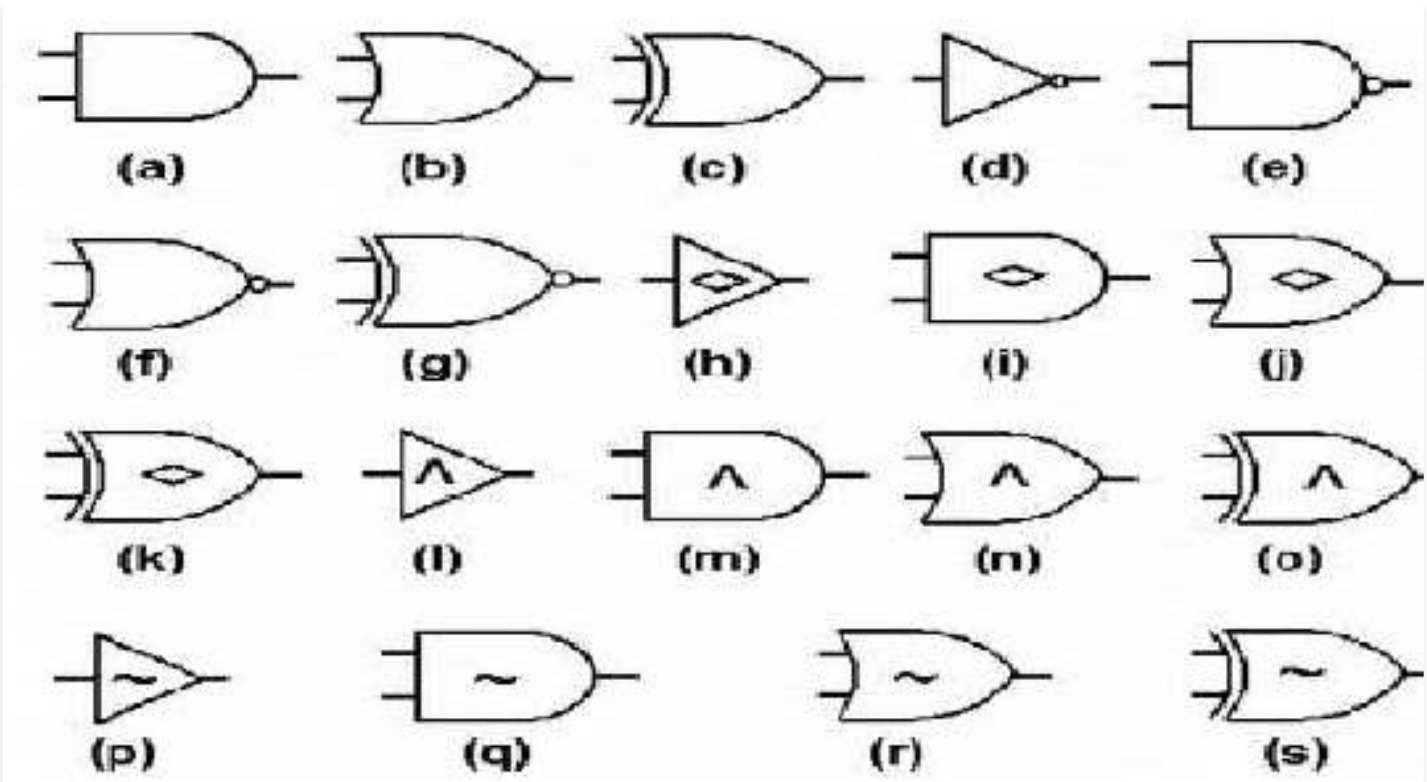
This truth table shows all possible outcomes for the given Boolean function based on the input variables [1] [2].

Share Rewrite

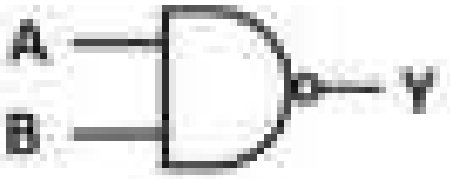
Related

- How does the XOR operator differ from the AND and OR operators
- Can you explain the logic behind the XOR operation in simple terms
- What are some practical applications of the XOR operator in real-world scenarios
- How can I implement the XOR operation without using the XOR symbol
- What is the significance of the NOT operator in boolean functions



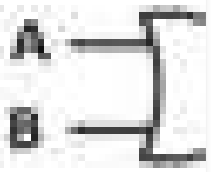


Y
T
F



NAND

A	B	Y
F	F	T
F	T	T



N

A
F
T