LOGICAL COMPLETENESS: IMPLEMENTING ANY TRUTH TABLE

7:21 / 7:21 1.0x

Download transcript .txt

Show Discussion New Post

A function F is a 1 when **only one of the three** inputs (A, B, and C) is a 0 and is a 0 otherwise.

1 A. CHECK YOUR UNDERSTANDING (1/2 points)

Fill in the truth table values for F.

A B C	F	
0 0 0	1	**Answer:** 0
0 0 1	1	**Answer:** 0

Help

0	**1**	**0**	1	**Answer:** 0
0	**1**	**1**	1	**Answer:** 1
1	**0**	**0**	1	**Answer:** 0
1	**0**	**1**	1	**Answer:** 1
1	**1**	**0**	1	**Answer:** 1
1	**1**	**1**	0	**Answer:** 0

SOLUTION OR EXPLANATION HEADING

F is a 1 for every row of the truth table that has a single 0 for the inputs.

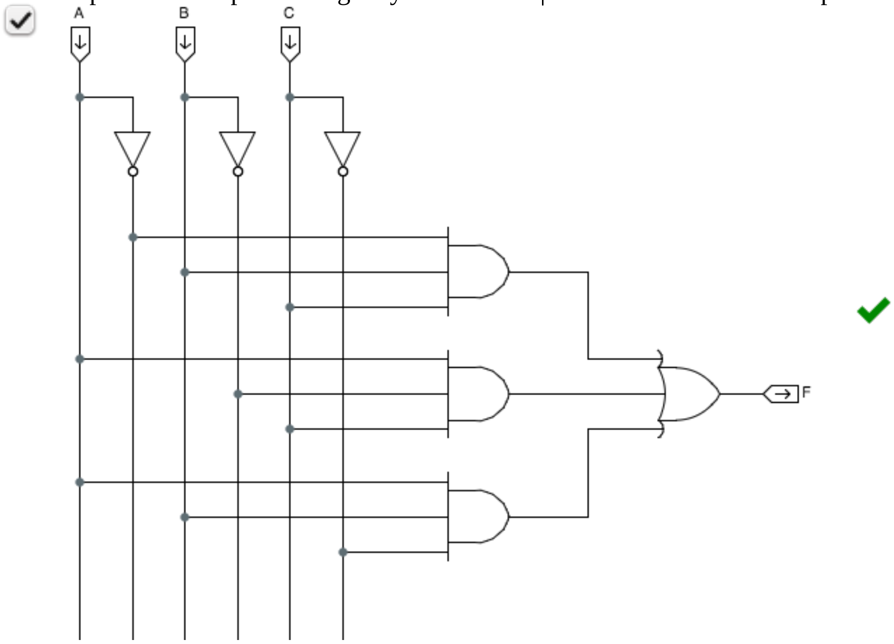
Hide Answer *You have used 1 of 1 submissions*

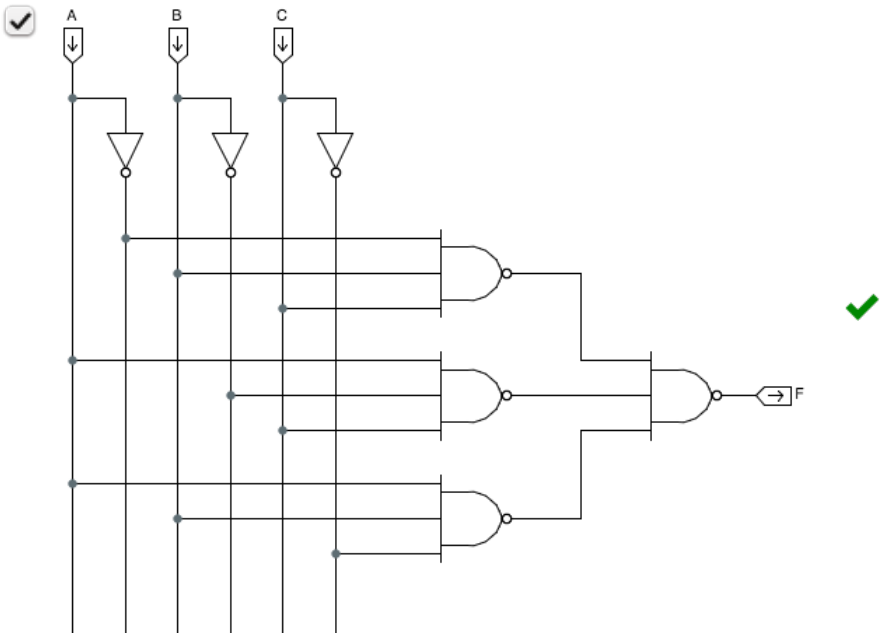
Show Discussion

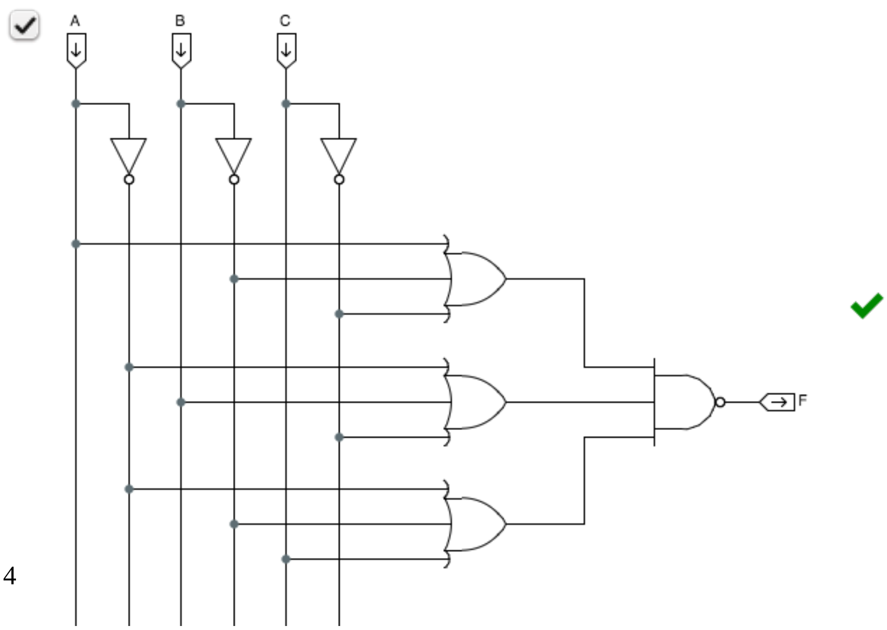
New Post

1 B. CHECK YOUR UNDERSTANDING (1/1 point)

Which of the following circuits implements F? [Check all that apply]


A
B
C
F


A
B
C
F


A
B
C
F

Help

EXPLANATION

All the circuits shown implement F.

The first circuit uses a 3-input AND gate to generate each of the 3 cases F is true (011,101,110), and ORs the cases together

The second circuit is the same as the first; use DeMorgan's Theorem to see that two nested levels of NAND gates here are functionally equivalent to a layer of AND gates which are then combined with an OR gate.

The third circuit uses 3-input OR gates to generate the combinations when F is true. Note that the inputs to the 3-input OR gates are the inverted versions of A, B, and C, that the other two circuits use. If you apply DeMorgan's theorem, you will see that the OR gates can be transformed into AND gates with inverters at the inputs, which explains why you would connect the OR gates to the inverted senses of A, B, and C that are used by the other two circuits.

Hide Answer *You have used 2 of 2 submissions*

Show Discussion New Post