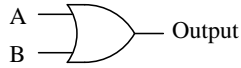


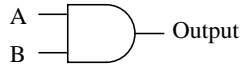
Questions

Question 1

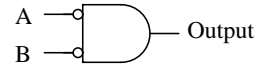
Identify each of these logic gates by name, and complete their respective truth tables:



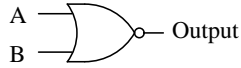
A	B	Output
0	0	
0	1	
1	0	
1	1	



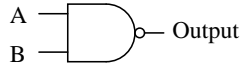
A	B	Output
0	0	
0	1	
1	0	
1	1	



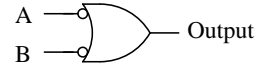
A	B	Output
0	0	
0	1	
1	0	
1	1	



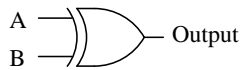
A	B	Output
0	0	
0	1	
1	0	
1	1	



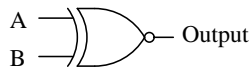
A	B	Output
0	0	
0	1	
1	0	
1	1	



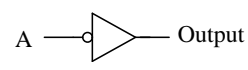
A	B	Output
0	0	
0	1	
1	0	
1	1	



A	B	Output
0	0	
0	1	
1	0	
1	1	



A	B	Output
0	0	
0	1	
1	0	
1	1	

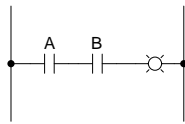


A	Output
0	
1	

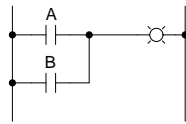
file 01249

Question 2

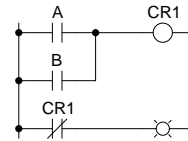
Identify each of these relay logic functions by name (AND, OR, NOR, etc.) and complete their respective truth tables:



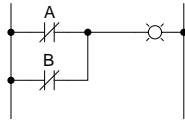
A	B	Output
0	0	
0	1	
1	0	
1	1	



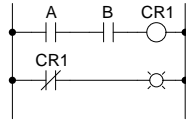
A	B	Output
0	0	
0	1	
1	0	
1	1	



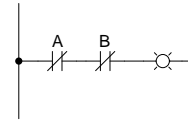
A	B	Output
0	0	
0	1	
1	0	
1	1	



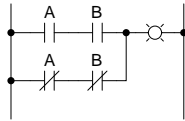
A	B	Output
0	0	
0	1	
1	0	
1	1	



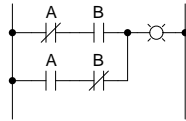
A	B	Output
0	0	
0	1	
1	0	
1	1	



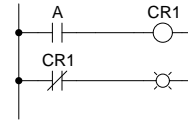
A	B	Output
0	0	
0	1	
1	0	
1	1	



A	B	Output
0	0	
0	1	
1	0	
1	1	



A	B	Output
0	0	
0	1	
1	0	
1	1	



A	Output
0	
1	

file 01335

Question 3

A *Karnaugh map* is nothing more than a special form of truth table, useful for reducing logic functions into minimal Boolean expressions.

Here is a truth table for a specific three-input logic circuit:

A	B	C	Out
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

Complete the following Karnaugh map, according to the values found in the above truth table:

	C	
	0	1
AB		
00		
01		
11		
10		

file 02834

Question 4

A *Karnaugh map* is nothing more than a special form of truth table, useful for reducing logic functions into minimal Boolean expressions.

Here is a truth table for a specific four-input logic circuit:

A	B	C	D	Out
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

Complete the following Karnaugh map, according to the values found in the above truth table:

	CD			
	00	01	11	10
AB	00			
	01			
	11			
	10			

file 01310

Question 5

Here is a truth table for a four-input logic circuit:

A	B	C	D	Out
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

If we translate this truth table into a Karnaugh map, we obtain the following result:

		CD			
		00	01	11	10
AB	00	0	0	0	0
	01	0	1	1	0
	11	0	1	1	0
	10	0	0	0	0

Note how the only 1's in the map are clustered together in a group of four:

		CD			
		00	01	11	10
AB	00	0	0	0	0
	01	0	1	1	0
	11	0	1	1	0
	10	0	0	0	0

If you look at the input variables (A, B, C, and D), you should notice that only two of them actually change within this cluster of four 1's. The other two variables hold the same value for each of these conditions where the output is a "1". Identify which variables change, and which stay the same, for this cluster.

file 01311

Question 6

Here is a truth table for a four-input logic circuit:

A	B	C	D	Out
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

If we translate this truth table into a Karnaugh map, we obtain the following result:

		CD			
		00	01	11	10
AB	00	0	0	0	0
	01	0	0	0	0
	11	1	1	1	1
	10	0	0	0	0

Note how the only 1's in the map all exist on the same row:

		CD			
		00	01	11	10
AB	00	0	0	0	0
	01	0	0	0	0
	11	1	1	1	1
	10	0	0	0	0

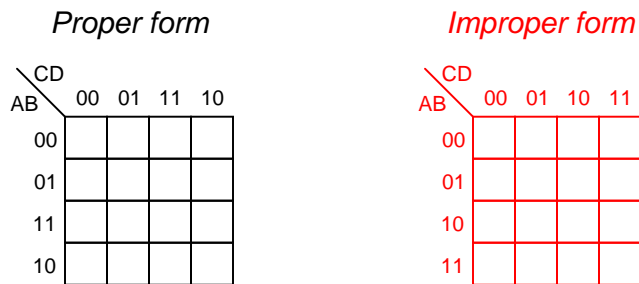
If you look at the input variables (A, B, C, and D), you should notice that only two of them are constant for each of the "1" conditions on the Karnaugh map. Identify these variables, and remember them.

Now, write an SOP (Sum-of-Products) expression for the truth table, and use Boolean algebra to reduce that raw expression to its simplest form. What do you notice about the simplified SOP expression, in relation to the common variables noted on the Karnaugh map?

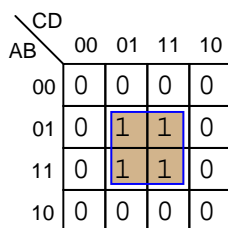
file 02835

Question 7

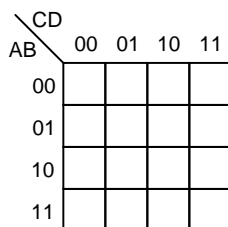
One of the essential characteristics of Karnaugh maps is that the input variable sequences are always arranged in Gray code sequence. That is, you never see a Karnaugh map with the input combinations arranged in binary order:



The reason for this is apparent when we consider the use of Karnaugh maps to detect common variables in output sets. For instance, here we have a Karnaugh map with a cluster of four 1's at the center:



Arranged in this order, it is apparent that two of the input variables have the same values for each of the four "high" output conditions. Re-draw this Karnaugh map with the input variables sequenced in binary order, and comment on what happens. Can you still tell which input variables remain the same for all four output conditions?



file 01312

Question 8

Examine this truth table and corresponding Karnaugh map:

A	B	C	D	Out
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

CD \ AB	00	01	11	10
00	1	0	0	1
01	0	0	0	0
11	0	0	0	0
10	1	0	0	1

Though it may not be obvious from first appearances, the four "high" conditions in the Karnaugh map actually belong to the same group. To make this more apparent, I will draw a new (oversized) Karnaugh map template, with the Gray code sequences repeated twice along each axis:

CD \ AB	00	01	11	10	00	01	11	10
00								
01								
11								
10								
00								
01								
11								
10								

Fill in this map with the 0 and 1 values from the truth table, and then see if a grouping of four "high" conditions becomes apparent.

[file 01342](#)