Logic Circuits

In this section we do three things:

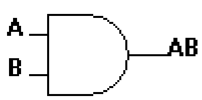
1. **Given a logic circuit, fill in the truth table for the circuit.**
2. **Given a logic circuit, find the Boolean expression for the circuit.**
3. **Given a Boolean expression, draw the logic circuit.**

When Boole developed his system of logic, he indicated that any complex statement can be written in terms of the three basic Boolean operators AND, OR, and NOT.

Truth Tables and Boolean Expressions for Logic Circuits

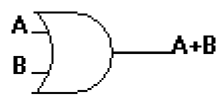
The truth tables and Boolean expressions for outputs of the three basic gates are indicated below.

AND gate



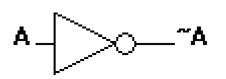
Inputs		Output
0	0	0
0	1	0
1	0	0
1	1	1

OR gate



Inputs		Output
0	0	0
0	1	1
1	0	1
1	1	1

NOT gate



Input	Output
0	1
1	0

They also can be written in the following form:

A AND B = A*B

A OR B = A+B

NOT A = $\overline{A}$

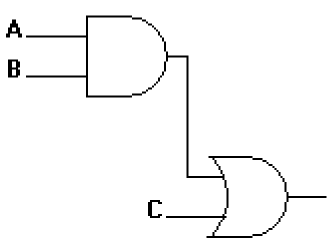
In the following example, we begin to build circuits that include more than one gate.

Example 1

An AND gate and an OR have been combined as indicated below.

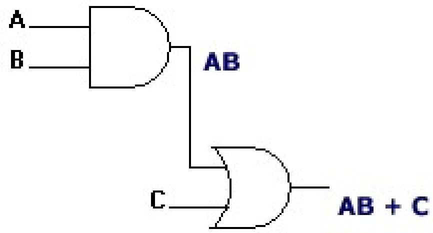
a) Fill in the truth table for this circuit.

b) Indicate the Boolean expression for the circuit.



Solution:

This circuit has three inputs A, B and C and just one output. Since the values of the inputs are either 0 or 1, there are eight possible combinations of inputs. The easiest way to list them is to count in binary from 0 0 0 to 1 1 1.
You can split the question into several steps, and use this steps fill the truth table.



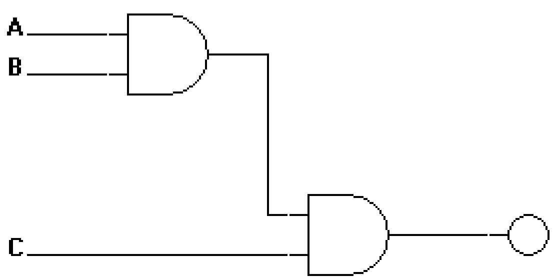
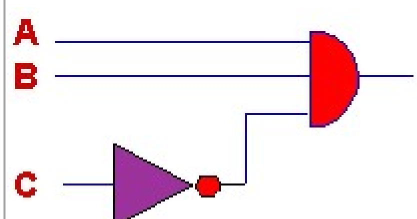
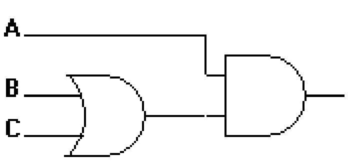
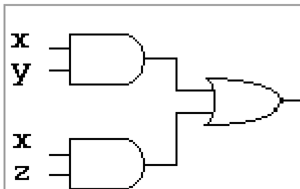
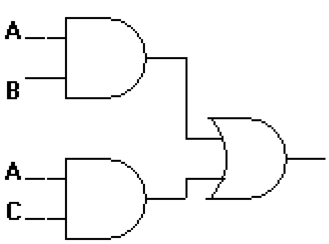
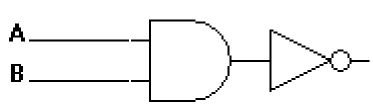
The output of the AND gate is the Boolean expression AB [we no longer include the . sign] and the Boolean expression for the output of the OR gate is AB + C.
We can use this information in the truth table:

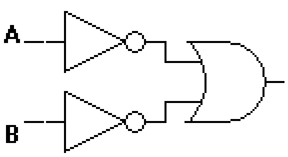
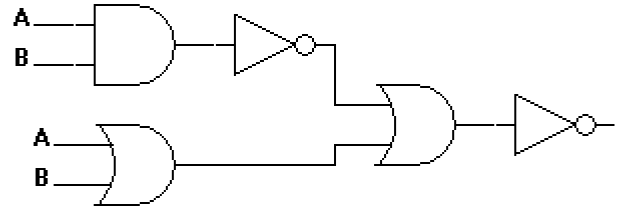
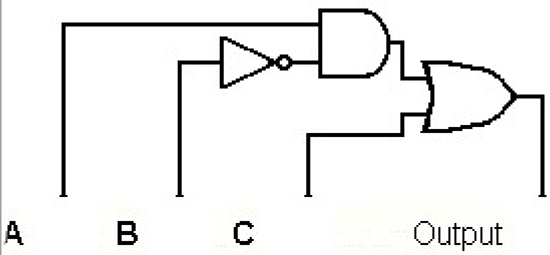
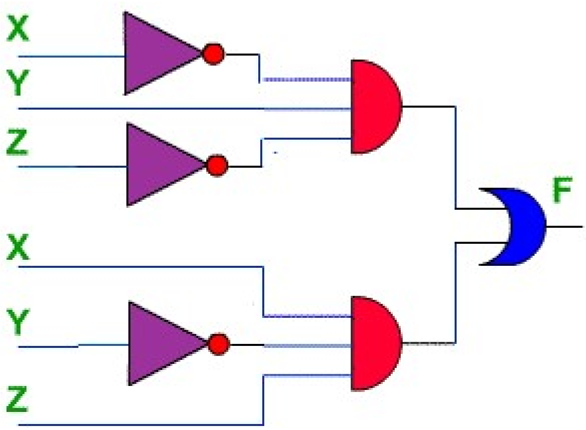
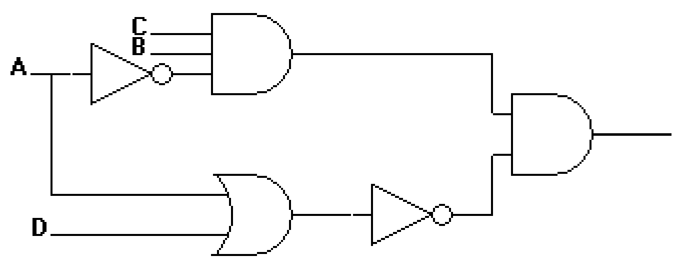
Inputs				Output
A	B	C	AB	AB+C
0	0	0	0	0
0	0	1	0	1
0	1	0	0	0
0	1	1	0	1
1	0	0	0	0
1	0	1	0	1
1	1	0	1	1
1	1	1	1	1

Boolean Expressions for Outputs of Logic Circuits

Problems 2

Give the Boolean expression for the output of each circuit:

a)		b)	
c)		d)	
e)		f)	

g)		h)	
i)			
j)		k)	

Drawing a Circuit Diagram Given A Boolean Expression for a Circuit

As illustrated above, given a circuit, it is possible to construct a truth table and a Boolean expression for the circuit.

It is also possible, given a Boolean expression such as AB + C, to construct the corresponding circuit. This concept is illustrated in Problems 3.

It is pointed out that there are a number of notations for NOT A.
NOT A, ~A, $\overline{A}$, and A' are four of the notations found in the literature. We will use $\overline{A}$.

Problems 3

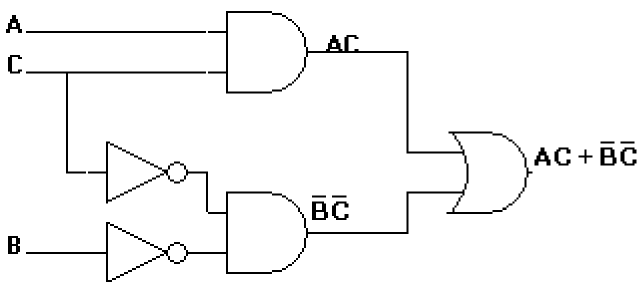
Construct the circuit diagram for these Boolean expressions. Fill in the truth tables.

Examine the truth tables for b) and c). What rule is illustrated?

Examine the truth tables for d) and e). What must be true about these Boolean expressions?

a) $\overline{A}$ **+ B** **b) A(B + C)** **c) AB + AC** **d)** $\overline{AB}$ **e)** $\overline{A}$ **+** $\overline{B}$ **f) AC +** $\overline{BC}$

Answer to part f)



Inputs							**Output**
A	**B**	**C**	**AC**	$\overline{B}$	$\overline{C}$	$\overline{BC}$	**AC +** $\overline{BC}$
0	0	0	0	1	1	1	1
0	0	1	0	1	0	0	0
0	1	0	0	0	1	0	0
0	1	1	0	0	0	0	0
1	0	0	0	1	1	1	1
1	0	1	1	1	0	0	1
1	1	0	0	0	1	0	0
1	1	1	1	0	0	0	1

Problems 2

Draw the circuit diagrams and truth tables for these Boolean expressions:

1. **B + C**

2. **AB**

3. **AB + C**

4. **A(B + c)**

5. $\overline{B + \bar{C} + D}$

6. **(X + Z)(X + Y)**

7. **AB + AC**

Problems 2.5.1 - Answers

a) ABC b) ABC c) X(Y+Z) d) XY + XZ e) AB f) $\overline{A}\,\overline{B}$

g) AB + (A + B) h) $\overline{\overline{AB}+(A+B)}$ i) AB + C j) XYZ + XYZ

k) (ABC)(A+D)

Problems 2.5.2

Construct the circuit diagram for these Boolean expressions. Fill in the truth tables.

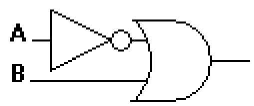
Examine the truth tables for b) and c). What rule is illustrated?

Examine the truth tables for d) and e). What must be true about these Boolean expressions?

a) A + B b) A(B + C) c) AB + AC d) AB e) A + B

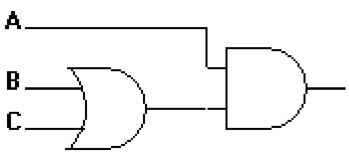
f) AC + B C

a)



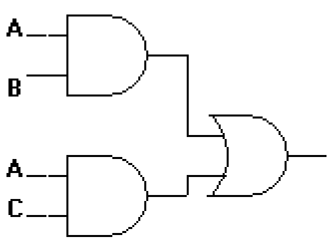
A	B	A	A+B
0	0	1	1
0	1	1	1
1	0	0	0
1	1	0	1

b)



A	B	C	B+C	A(B+C)
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

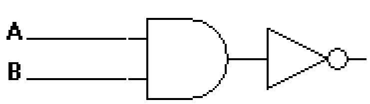
c)



A	B	C	AB	AC	AB+AC
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	1	1

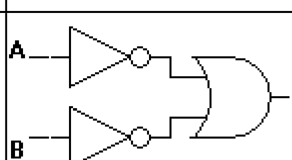
Because the truth table outputs for b) and c) are identical for corresponding inputs, we may conclude that A(B+C) = AB + AC -- distributivity of AND over OR holds!

d)



A	B	AB	AB
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

e)



A	B	A	B	A+B
0	0	1	1	1
0	1	1	0	1
1	0	0	1	1
1	1	0	0	0

Since d) and e) have the same truth table outputs for corresponding inputs, we may conclude that AB

	= A+B. This is a well known law in Boolean algebra called DeMorgan's Theorem.	
f)	see solution above	