CSCI 162 Lab 3

Boolean logic and gates

Purpose: To build interactive Boolean circuits to study their behaviour.

Resources: Circuit builder

http://www.jhu.edu/~virtlab/logic/logic.htm

Objectives: After this lab you should be able to

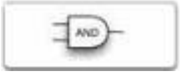
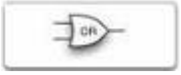
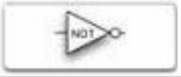
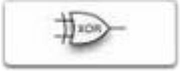
- Identify the gate symbols for the Boolean operations AND, OR, NOT and XOR.
- Construct a table with the input/output behaviour of each individual gate.
- Build and run a simple Boolean circuit simulation to match a diagram of a circuit.
- Build and run a simple Boolean circuit simulation to match an input/output table.
- Build and run a simple Boolean circuit simulation to match a Boolean equation.
- Create an XOR gate with only 4 other gates.

Background:

Since it can be expensive to build an actual computer circuit with physical components, computer scientists use software-based Boolean logic simulators to construct and test computer circuit diagrams before the physical version. In this lab you are going to use a Java applet Boolean logic simulator to construct and test computer circuits that illustrate each basic Boolean gate, simple computer circuits and memory flip-flops.

At the lowest level, information flows through computers as electrical signals. All components are connected by wires, and at any instant a single wire is on (conducting electricity) or off (not conducting). We can represent the on/off nature of an electric signal using one bit which is either 0 or 1. One bit gives us enough information to answer a True/False question on a quiz.

The four Boolean gates are diagrammed using the symbols in this table. All gates and connectors in the simulator take input on the left and produce output on the right.

Gate	Symbol	Inputs		Output
AND		0	0	0
		0	1	0
		1	0	0
		1	1	1
OR		0	0	0
		0	1	1
		1	0	1
		1	1	1
NOT		0		1
		1		0
XOR		0	0	0
		0	1	1
		1	0	1
		1	1	0

Example k-Map question:

Use K-Maps to design a Sum of Products (SOP) circuit for 4 inputs on M_0, M_1, M_4, M_5, M_B, M_E, M_F

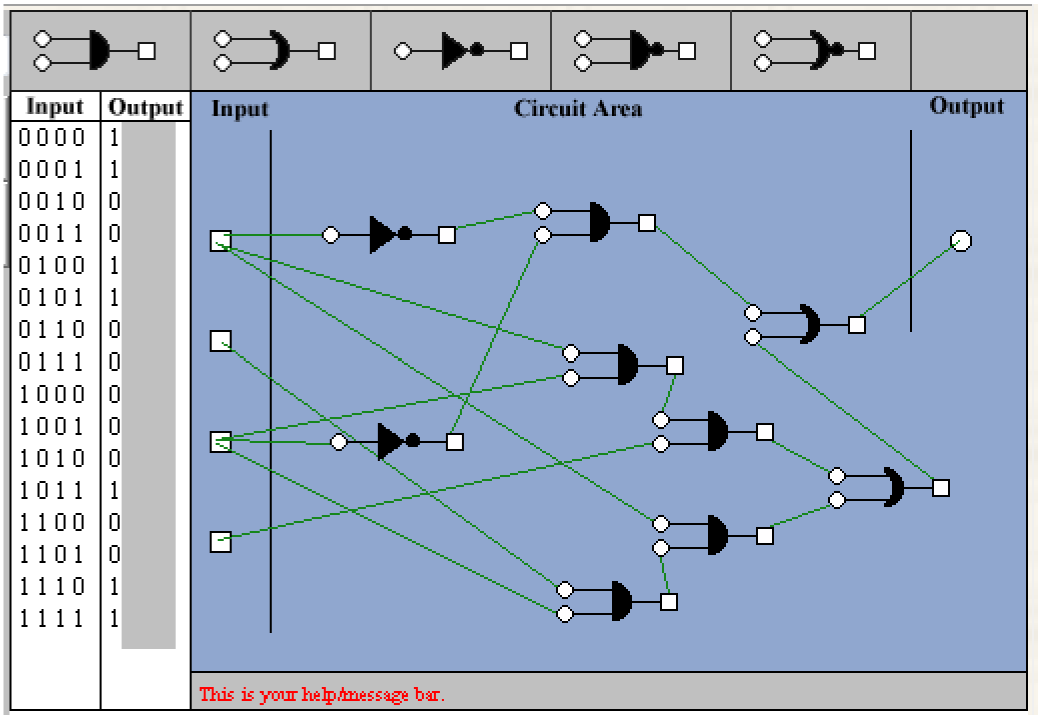
First, Create the K-Map:

WX\YZ	00	01	11	10
00	1	1		
01	1	1		
11			1	1
10			1	

Next give the SOPs:

!W!Y+WYZ+WXY

Create the circuit:



Input	Output
0000	1
0001	1
0010	0
0011	0
0100	1
0101	1
0110	0
0111	0
1000	0
1001	0
1010	0
1011	1
1100	0
1101	0
1110	1
1111	1

Compare your output with the original question.

Instructions:

Open a web browser to circuit builder at:

http://www.jhu.edu/~virtlab/logic/logic.htm

1. Create a 3-input AND gate from two 2-input AND gates.
2. Create a 4-input OR gate from three 2-input OR gates.
3. Create an XOR gate using only 4 of the available gates.
4. Use K-Maps to design a Sum of Products (SOP) circuit for 4 inputs on M_0, M_1, M_4, M_5, M_A, M_B, M_F

What to hand in:

Take snap shots of each of the 4 circuits, tar the files into a single file and hand in via Moodle.

Demo each of the circuits for the instructor.