Problem Sets for Module 3: Basic Digital Gates and Combinational Logic

PROBLEM 3.1: Basic Gates
For each of the applications described below, tell which type of gate or combination of gates will work:

A. A stream of pulses are high when they should be low, and low when they should be high. Which gate can fix the problem?

B. To launch a missile, two men at different locations must press a launch button at the same time. Each launch button puts out a high signal. Which gate is used to launch the missile when both men press their buttons?

C. An incinerator starts up to burn garbage any time garbage is thrown down one of four different shoots. A photoelectric system detects garbage coming down each of the shoots, and puts out a logical "1", or high signal, when the garbage is coming. Which gate takes these four signals and turns on the fire when any one of the shoots has garbage coming?

PROBLEM 3.2: Basic Gates
In the system shown, for the given pulse stream going into the inputs, draw in the expected outputs at the points indicated.

PROBLEM 3.3: Basic Gates
In the system shown, for the given pulse stream going into the inputs, draw in the expected outputs at the points indicated.
PROBLEM 3.4: Basic Gates
In the system shown, for the given pulse stream going into the inputs, draw in the expected outputs at the points indicated.

PROBLEM 3.5: Gates Equivalence
For the two types of gates shown, plot the outputs for the common input stream. Remember, a small circle indicates that the signal is inverted. What can we say about the two gates? What type of gate is it?

PROBLEM 3.6: Gates Equivalence
For the two types of gates shown, plot the outputs for the common input stream. Remember, a small circle indicates that the signal is inverted. What can we say about the two gates? What type of gate is it?
**PROBLEM 3.7: Gates Rules**
Each gate has a general rule that governs how it processes input signals. For each of the gates listed, write the rule.

**A.** AND Gate._____________________________________________________

**B.** OR Gate._____________________________________________________

**C.** NOT or INVERT Gate.___________________________________________

**D.** NAND Gate.___________________________________________________

**E.** NOR Gate._____________________________________________________

**F.** XOR Gate._____________________________________________________

**PROBLEM 3.7: Basic Logic Design**
How can you use two AND gates, an INVERTER, and an OR gate to make a circuit that will have two signal inputs, A and B, and a control input S. When S is in a high, or logical "1" state, input A is selected, goes to the OR gate and is output to C. When S is low, or in a logical "0" state, input B is selected, goes to the OR gate and is output to C. In other words you are to design a circuit that will select between input signals A and B. This circuit is called a multiplexer.

**PROBLEM 3.8: Basic Logic Design**
Design a circuit using XOR gates that will look at four input lines (A, B, C, and D), and output a logical "1" at an output (X) whenever the four input lines have an odd number of high, or logical "1", inputs.

**PROBLEM 3.9: Basic Logic Design**
The figure below is a simplified version of a programmable logic array. To build a circuit, the computer makes connections at the correct intersections of the grid lines. Put a dot at each intersection to show where the computer would need to make connections to build the circuit you designed in Problem 3.7 for a two bit multiplexer.

**Note:** It is not necessary to use all input and output lines or gates. Extra devices in a programmable logic array do nothing when they have not been incorporated into the design.
**PROBLEM 3.10: Basic Logic Design**

The text shows the combination of AND, OR, and INVERT gates that go together to make an XOR gate. Use the simplified programmable logic array to design an XOR gate. Refer to the text for the gate design.