

2. Algebraic Techniques (20 points)

Write algebraic expressions for f_A and f_B . Show, algebraically, that the two functions are equivalent.

3. Implicit Techniques (10 points)

Explain how, conceptually, you can create a new circuit that will help you with the task of proving that Circuits A and B are equivalent. Specifically, how can you tie the outputs of the circuits together to create a new circuit that computes an output that is identically 0 if and only if the two circuits compute the same Boolean function? Indicate what logical function should replace the question mark in Figure 2. Call the function implemented by the new circuit g .

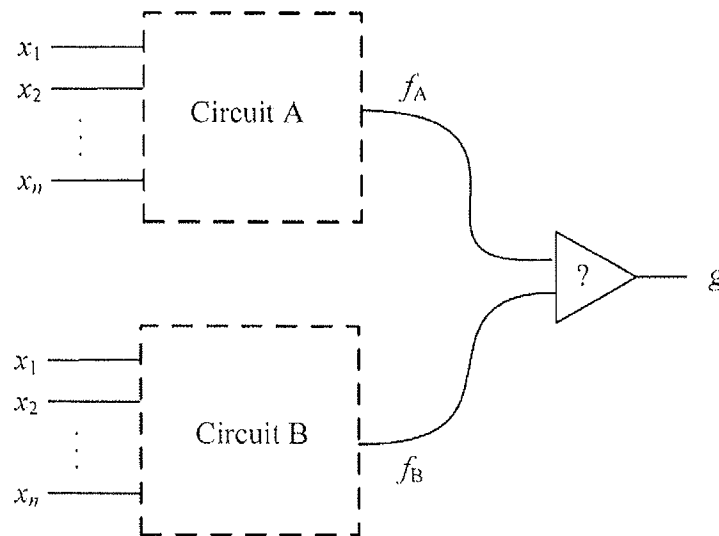


Figure 2: A conceptual circuit for verifying whether Circuits A and B are equivalent.

4. Binary Decision Diagrams (25 points)

First proposed in 1959 by Lee, binary decision diagrams (BDDs) were popularized in 1986 through a seminal paper by Bryant. A BDD consists of a directed graph in which nodes either have associated input variables or else are designated as a constant nodes (“0” or “1”). To evaluate a function one begins at a designated source node and follows a path dictated by the values of the variables until one arrives at one of the two constant nodes. The value of this constant node specifies the value of the function. An example is shown in Figure 3. The BDD in the figure represents the function

$$f = x_1(x_2 + x_3).$$

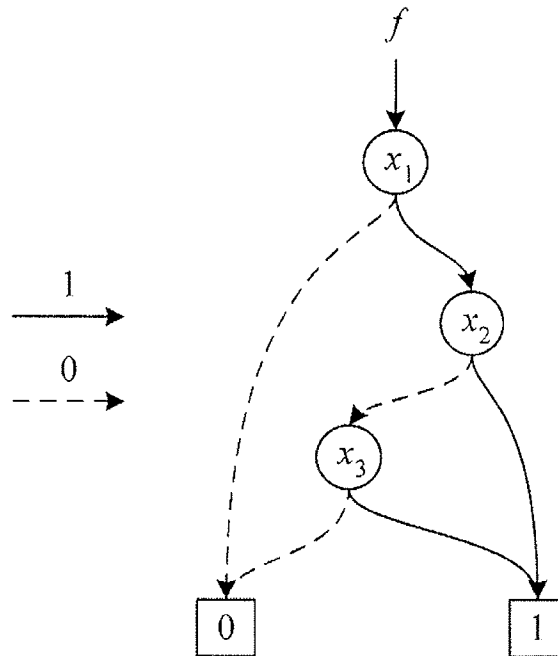


Figure 3: A binary decision diagram (BDD).

Although comparable in size to a truth table in the worst case, BDDs are surprisingly compact for many of the Boolean functions encountered in practice. This is due to the fact that BDDs can often be reduced in size by *collapsing* redundant nodes and *merging* equivalent nodes.

Draw reduced BDDs for the functions f_A , f_B and g in Part 3.

5. **Boolean Satisfiability** (25 points)

In our verification problem, the question that we are trying to answer – namely, whether the circuits are equivalent – has an affirmative answer if equivalence holds for *all* possible input assignments. It has a negative answer if equivalence does not hold for *any* input assignment. So-called SAT-based techniques, based on heuristic solutions to the Boolean satisfiability problem, can be used to answer questions that fit this mold. In theory, such algorithms can take time that is exponential in the number of variables to complete. In practice, they have shown themselves to be remarkably efficient.

SAT-based analysis begins with a circuit structure and proceeds by packaging the Boolean function that it computes in conjunctive normal form (CNF). This is passed to heuristic algorithms known as SAT solvers. If the solver returns “UNSAT,” this means that there is no satisfying assignment to the formula. Otherwise, the solver returns “SAT” along with a satisfying assignment. For instance, consider the circuit in Figure 4. The corresponding CNF formula is:

$$(\bar{x}_2 + y)(\bar{x}_3 + y)(x_2 + x_3 + \bar{y})(x_1 + \bar{h})(y + \bar{h})(\bar{x}_1 + \bar{y} + h)(h).$$

The solver would return SAT. A satisfying assignment for this formula is $x_1 = x_2 = x_3 = y = h = 1$.

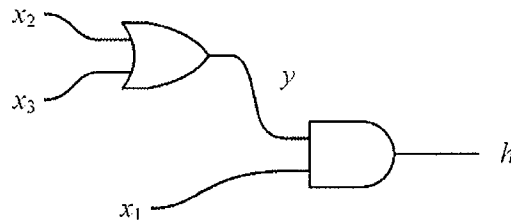


Figure 4: A circuit illustrating Boolean satisfiability.

Based on the construct in Figure 2, write a CNF formula for the question: are Circuits A and B equivalent?