Logic synthesis is the task of designing circuits at the level of gates and wires to meet a specification. As a research area, it is at once mature and wide-open. It is mature in the sense that progress in the area is slow and incremental. It is wide-open in the sense that even the best available tools are based on heuristics; these produce results that everyone admits are probably far from optimal.

In this problem you will explore some of the basic tasks of logic synthesis. Consider the following target functions:

$$\begin{aligned} f_1 &= \bar{x}_1 x_2 \bar{x}_3 + x_1 \bar{x}_2 \bar{x}_3 + x_1 \bar{x}_2 x_3 + \bar{x}_1 \bar{x}_2 x_3, \\ f_2 &= \bar{x}_1 \bar{x}_2 \bar{x}_3 + x_1 x_2 x_3 + x_1 x_2 \bar{x}_3 + x_1 \bar{x}_2 x_3, \\ f_3 &= \bar{x}_1 x_2 \bar{x}_3 + \bar{x}_1 \bar{x}_2 \bar{x}_3 + x_1 \bar{x}_2 \bar{x}_3 + \bar{x}_1 \bar{x}_2 x_3. \end{aligned}$$

Our cost measure for area will be the number of *literals* in the algebraic expressions – that is to say, the number of appearances of variables, without regard to negations. In the expressions above, each function has a cost of 12, for a total cost of 36.

## Two-Level Logic Minimization

The first step in logic design is to simplify the Boolean expressions individually, if possible. In the *sum-of-products* (SOP) form, a Boolean expression is formulated as the OR (disjunction) of AND (conjunctive) terms. Minimal in this context means with the fewest conjunctive terms, and the fewest literals per conjunctive term.

**Problem** [0.5 points]

Obtain minimal SOP forms for the target functions. (Solution has total area cost of 20.)

**Solution**

$$\begin{aligned} f_1 &= x_1 \bar{x}_2 + \bar{x}_1 x_2 \bar{x}_3 + \bar{x}_2 x_3, \\ f_2 &= x_1 x_2 + x_1 x_3 + \bar{x}_1 \bar{x}_2 \bar{x}_3, \\ f_3 &= \bar{x}_1 \bar{x}_3 + \bar{x}_1 \bar{x}_2 + \bar{x}_2 \bar{x}_3, \end{aligned}$$

## Two-Level Logic Minimization with Shared Terms

When minimizing several functions jointly, one can often save in area by sharing terms. For instance, the term

$$c_1 = x_1 \bar{x}_2 x_3$$

can be used in both the expressions for $f_1$ and $f_2$.

**Problem** [1.0 points]

Obtain minimal SOP expressions, sharing terms among them. (Solution has total area cost 17.)

**Solution**

$$\begin{aligned} c_1 &= x_1\bar{x}_2x_3, \\ c_2 &= x_1\bar{x}_2\bar{x}_3, \\ c_3 &= \bar{x}_1\bar{x}_2x_3, \\ c_4 &= \bar{x}_1x_2\bar{x}_3, \\ c_5 &= \bar{x}_1\bar{x}_2\bar{x}_3, \\ c_6 &= x_1x_2, \end{aligned}$$

we obtain the expressions

$$\begin{aligned} f_1 &= c_1 + c_2 + c_3 + c_4, \\ f_2 &= c_1 + c_5 + c_6, \\ f_3 &= c_2 + c_3 + c_4 + c_5, \end{aligned}$$

with a total cost of 17.

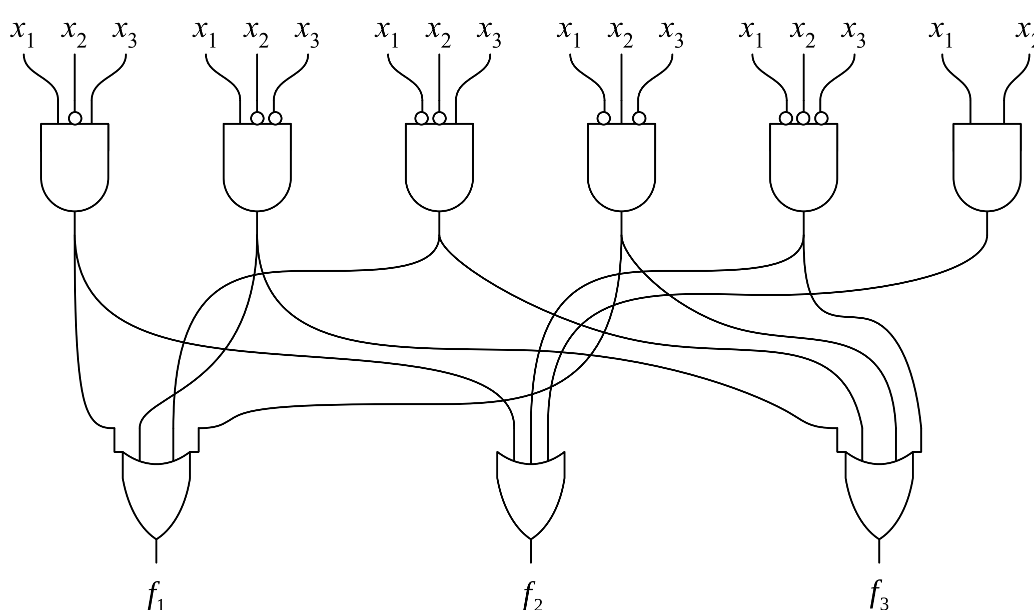


Figure 1: A two-level implementation for the target functions.

## Multi-Level Logic

In multi-level designs, an arbitrary structure is permitted. Algebraically, a factored form is a parenthesized expression of OR and AND operations. For instance, the expression

$$x_1 + x_2x_3 + x_2x_4$$

can be written as

$$x_1 + x_2(x_3 + x_4).$$

**Problem** [1.0 points]

Find minimal factored forms for the target functions. (Solution has total area cost 17.)

## Solution

$$\begin{aligned} f_1 &= \bar{x}_1x_2\bar{x}_3 + \bar{x}_2(x_1 + x_3), \\ f_2 &= \bar{x}_1\bar{x}_2\bar{x}_3 + x_1(x_2 + x_3), \\ f_3 &= \bar{x}_3(\bar{x}_1 + \bar{x}_2) + \bar{x}_1\bar{x}_2 \end{aligned}$$

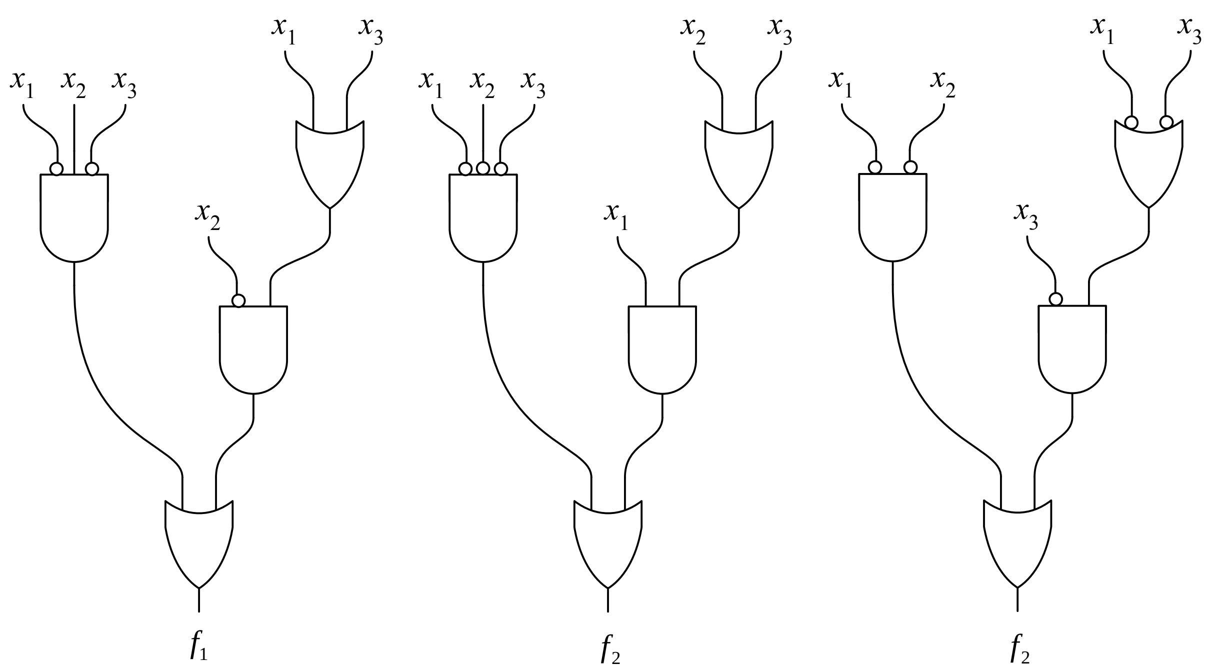


Figure 2: A factored form for the target functions.

**Substitution Orderings**

Beyond sharing of terms in SOP expressions and factoring, arbitrary substitutions can be made in multi-level logic synthesis. For instance, for the target functions, we can express $f_1$ in terms of $f_2$ and $f_3$,

$$f_1 = \bar{x}_2 x_3 + \bar{f}_2 f_3.$$

**Problem** [1.5 points]

Exploit such substitutions to obtain the least costly expressions for the target functions. (Solution has a total area cost of 14.)

**Solution**

$$\begin{aligned} f_1 &= \bar{x}_2 x_3 + \bar{f}_2 f_3 \\ f_2 &= \bar{x}_1 \bar{x}_2 \bar{x}_3 + x_1 \bar{f}_3 \\ f_3 &= \bar{x}_3(\bar{x}_1 + \bar{x}_2) + \bar{x}_1 \bar{x}_2. \end{aligned}$$

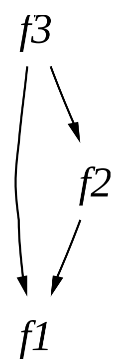


Figure 3: Acyclic substitution order.

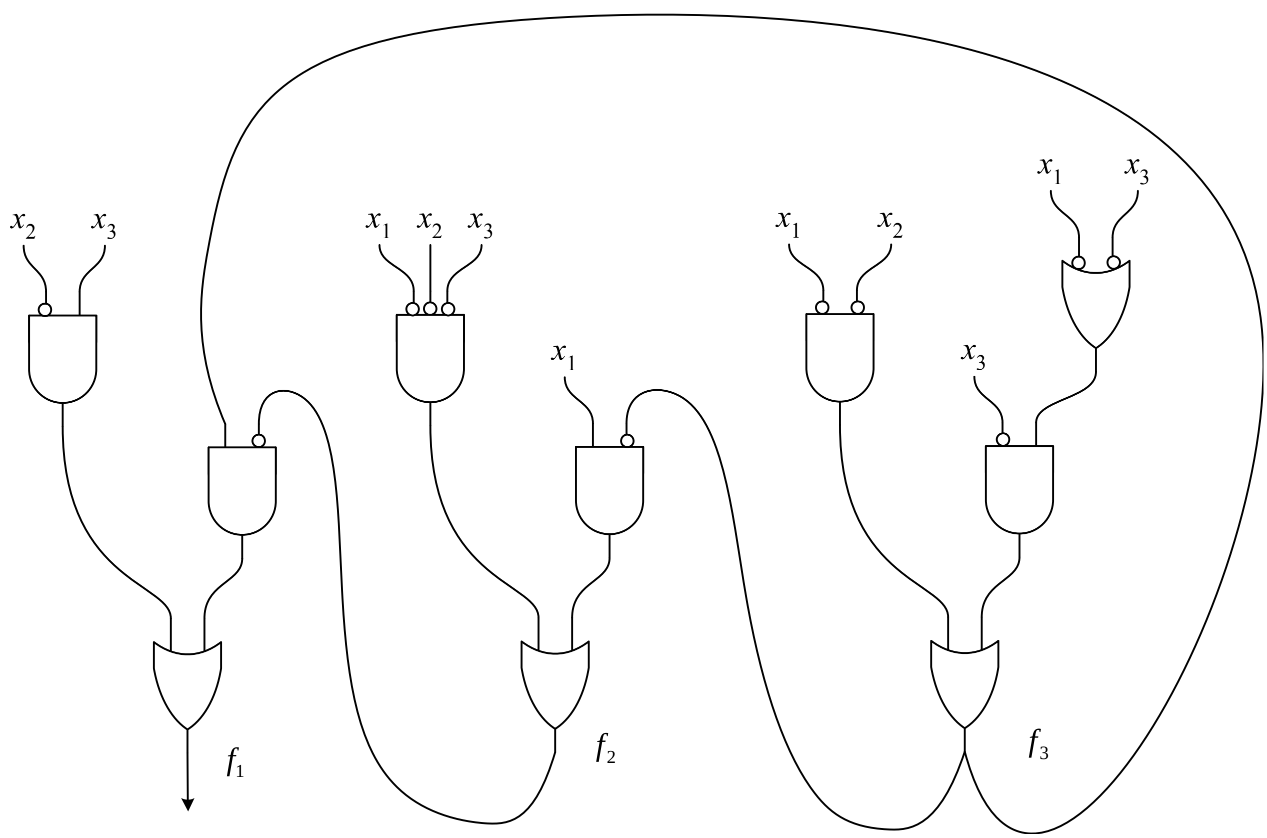


Figure 4: Implementation of the acyclic solution in Figure 3.