Combinational Circuit Minimization

Section 4-11 of Floyd's *Digital Fundamentals* (handout) introduces a small practical example of the use of Boolean algebra in digital circuit design.

The segments of a familiar seven-segment display (labelled **a** through **g**) are activated to read-out as integers (0-9) by a network of logic gates actualized in silicon. The integers to be displayed are input into the logic circuit encoded as four binary bits (0000 to 1001, with 1010 through 1111 not used), in a code named BCD for binary-coded-decimal.

Thus the *BCD-to-7segment*. parsing problem is to convert four binary input signals into seven binary output signals in a given configuration. The parser/decoder itself is built onto a silicon chip.

There are many different configurations of logic gates which achieve bcd-to-7segment decoding. And there are many different criteria that a circuit manufacturer might wish to optimize when designing the decoder logic circuit. To get the best price at a particular silicon foundry, or for a particular substrate material, the logic network might need fewer wires but more gates, or it may need to consume very little power, or perhaps it might have fit a regular array of particular types of gates.

The parse logic can be represented as a set of seven equations in Boolean algebra, with four inputs (the four BCD bits) and seven outputs (the on-off bits for each segment). Fortunately, fundamental features of the mathematical representation map well onto important design features of the silicon circuit.

For example, here is one particular solution in which each parenthesis boundary is a logical NOR gate:

```
a = (( D B ((C)(A)) (C A) ))
b = (( D (C) (B A) ((B)(A)) ))
c = (( D C (B) A ))
d = (( (C A) (C (B)) ((B) A) ((C) B (A)) ))
e = (A ((C) B))
f = (( D (B A) ((C) A) ((C) B) ))
g = (( D ((B) A) ((C) B) (C (B)) ))
```

Inputs: The number of occurences of input labels (A-D) is the fanout of the input, which relates to wiring and to power consumption. It is customary to use literals (either positive or negative occurences of an input, as in A or (A)) as circuit inputs, since both signal and negated signal are usually available. This version has a low number of input references, 42, but 24 is a minimum.

Chip Area: The number of parentheses represents the number of NOR gates, which maps well to the surface area of silicon that the circuit will take. Again in counting gates, the (A) form is an input literal, and does not count in the gate count. This version uses 28 gates.

Wiring: Wiring is becoming the dominant design issue for sub-micron silicon layout technology. The number of wires in a circuit is indicated by the number of subterms of each above expression, viewing the parentheses as a representation of a tree structure. The above solution has few wires, 70 (equal to the number of gates plus the number of inputs), but it is easy to reduce this number.

Timing: Perhaps the most important design criteria for a combinational circuit is its *critical path*, the longest path from any input to an output. This determines the delay time of the circuit and thus the rate of the driving clock. Critical path is modeled by the deepest nesting of parentheses, since each parenthesis is a gate. This solution is well balanced for timing, several equations have the maximum depth of 3 gates.

Noise: The noise in a circuit refers to the

Power Consumption: When a signal passes through a logic gate

Finally, we must recognize that silicon layout introduces new geometrical issues which require the simple Boolean equation model to be extended. The primary example is *structure sharing*, when the output of a subtree is used more than once. This converts the Boolean tree model into a Boolean graph model. In the above example, the subcircuit "((C) B)" occurs in equations e, f, and g. These can be implemented as one circuit with three outputs, resulting in a savings of two gates and four literals. (The example is not well suited for structure sharing.) To indicate structure sharing, construct an new variable name for the shared structure:

```
n = ((C) B)
e = ((A n))
f = (( D (B A) ((C) A) n ))
g = (( D ((B) A ) n (C (B)) ))
```

EXERCISE

Find the (close to) minimal configurations of the BCD-to-7segment decoder for

number of simple NOR logic gates
number of wires between gates
length of critical path
number of literals

Early in the assignment, read the section from DeMicheli's *Synthesis and Optimization of Digital Circuits*. Try your understanding of optimization techniques on example 8.2.2.