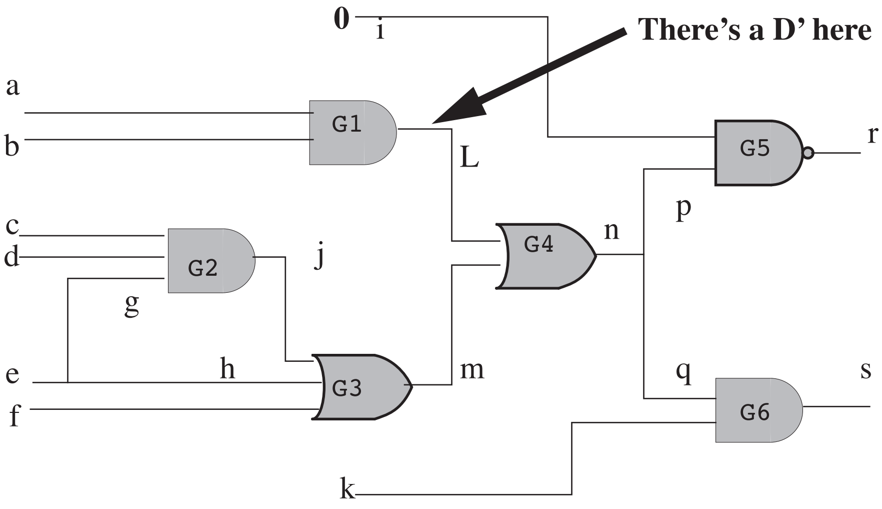
18-360
Intro to Computer-Aided Digital Design
ECE Department

Carnegie Mellon

Out: 3/16/04 Homework 5 Due: 3/25/04

We'll discuss the solutions to these problems at the beginning of class on the due date. Bring your solutions then. Solutions will be handed out, so we can't accept late assignments.

1. Do the implications — the non-PODEM approach 20%

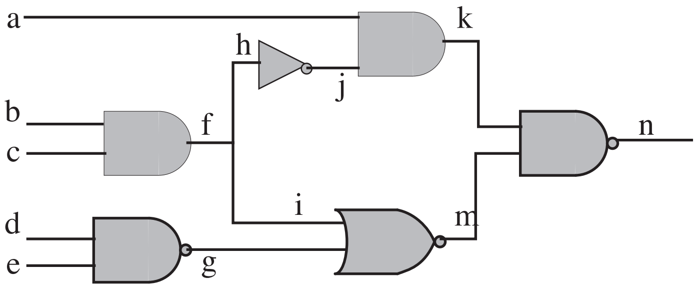


There's a D' here

Starting with the given values, list all of the possible implications, forward and backward. Label all nodes with their implied values. Assumptions to make: if there is only one gate on the D frontier, the D can be implied forward through it and a non-controlling value is assumed on the other input (this is called unique D drive). If a value is not given, we will assume that it is an x. Make a list of implications:

```
Implication          Why Implicated?
```

2. You-da-Code: The PODEM Experience 20%



a. Using PODEM, find a test for the fault i s-a-1. Whenever a choice has to be made, always choose the top path to backtrace, propagate or justify first. Describe each step taken (make a table) and show a decision tree.

b. Do this again for the fault i s-a-0.

Objective	PI assignment	Implications	D Frontier	comments

<fill in>

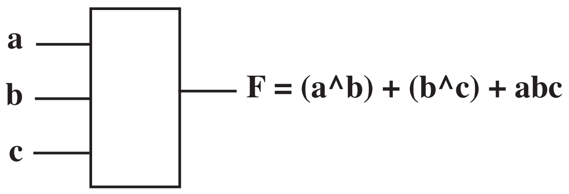
3. Inside PODEM 20%

Shown below is the PODEM algorithm taken straight from the text and lectures. Explain a situation where the algorithm will return FAILURE to point A. Draw a very simple gate diagram that illustrates the situation. Tie this diagram into your explanation.

```
PODEM()
begin
      if (error at PO) then return SUCCESS
      if (test not possible) then return FAILURE
      (k,$v_k$) = Objective()
      (j,$v_j$) = Backtrace(k,$v_k$) // j is a PI
      Imply(j,$v_j$)
      if PODEM() == SUCCESS then return SUCCESS
      // reverse direction
      Imply((j,$\bar{v}_j$)
      if PODEM() /* point A*/ == SUCCESS then return SUCCESS
      Imply(j,x)
      return FAILURE
      end
```

4. What dominates what? 20%

OK, here's a logic module with ports. Find the fault dominance relationship(s) among the single stuck faults of the ports (a, b, c, F) of this device. Using a Venn diagram might help. Briefly explain your answer.



Notes: we are not concerned with any of the internal connections in this circuit, only the ports. That " ^" is an XOR.

5. Partition It — K&L 20%

Here's a graph; think of each of these nodes as a gate. (Shaded and weighted lines have been used to make the interconnections more clear.) Initially, assume that A, B, C and D are in one partitoin and E, F, G, and H are in the other. Show your work when running one pass of the K&L algorithm on this. Show each step with the costs. What is the best partition?

Would an extra pass help?