**VE527 Computer-Aided Design of Integrated Circuits**

**Written Assignment Four Solution**

1. (12%) Given the Boolean function , obtain its cofactors and .

**Solution**:

2. (12%) In class, we have shown one form of Shannon expansion:

The above form can be thought of as a “sum of products” form. Actually, there is also a “product of sums” form of the Shannon expansion:

Prove the above “product of sums” expression.

**Solution**:

When , the right-hand side is ; when , the right-hand side is .

3. (16%) Consider the small logic network shown below. Obtain the Boolean differences and .

In the picture, there is a small logic network having four primary inputs, a, b, x and y; one primary output F; five internal gates from G1 to G5. Connectivity of the network is as below: 
G1 is a NOR gate having two inputs and connects to primary inputs a and b.
G2 is an inverter gate having one input and connects to primary input x.
G3 is a NAND gate having two inputs and connects to gate G1 and primary input x.
G4 is an AND gate having two inputs and connects to gate G2 and primary input y.
G5 is an OR gate having two inputs and connects to gates G3 and G4.
Primary output F connects to gate G5.

**Solution**:

From the circuit, we have , , ,   
and . Thus, we have

4. (10%) For the logic network from Problem 3, obtain the universal quantification and the existential quantification .

**Solution**:

5. (24%) Network repair.

The carry output of a 1-bit adder has the Boolean function , where and are the 1-bit numbers we want to add, and is the input carry bit. The figure below shows an implementation of the above function. However, the implementation is not correct. We suspect that the gate with the “??” label is incorrect. Use the logic network repair procedure discussed in the lecture to fix the suspicious gate. What could this gate be?

In the picture, there are a small logic network and a four-to-one multiplexor. 

For the network, it has three primary inputs, a, b, and Cin; one primary output Cout; four internal gates G1, G2, G5 and G_question_mark. Connectivity of the network is as below: 
G1 is an AND gate having two inputs and connects to primary inputs a and b.
G_question_mark is the gate labeled with question mark having two inputs and connects to primary inputs a and b.
G2 is an AND gate having two inputs and connects to gate G_question_mark and primary input Cin.
G3 is an OR gate having two inputs and connects to gates G1 and G2.
Primary output Cout connects to gate G3.

For the four-to-one multiplexor, it has select signals a and b; four inputs d0, d1, d2 and d3; one output q. Connectivity of the multiplexor is as below:
When signals a and b equal to 00, output q will connect to input d0. 
When signals a and b equal to 01, output q will connect to input d1. 
When signals a and b equal to 10, output q will connect to input d2. 
When signals a and b equal to 11, output q will connect to input d3. 

**Solution**:

Suppose that we replace the “??” gate with the 4-to-1 MUX shown below.

In the picture, there are a small logic network and a four-to-one multiplexor. 

For the network, it has three primary inputs, a, b, and Cin; one primary output Cout; four internal gates G1, G2, G5 and G_question_mark. Connectivity of the network is as below: 
G1 is an AND gate having two inputs and connects to primary inputs a and b.
G_question_mark is the gate labeled with question mark having two inputs and connects to primary inputs a and b.
G2 is an AND gate having two inputs and connects to gate G_question_mark and primary input Cin.
G3 is an OR gate having two inputs and connects to gates G1 and G2.
Primary output Cout connects to gate G3.

For the four-to-one multiplexor, it has select signals a and b; four inputs d0, d1, d2 and d3; one output q. Connectivity of the multiplexor is as below:
When signals a and b equal to 00, output q will connect to input d0. 
When signals a and b equal to 01, output q will connect to input d1. 
When signals a and b equal to 10, output q will connect to input d2. 
When signals a and b equal to 11, output q will connect to input d3. 

Then the function of the circuit is

The correct function is

We want to find a set of values for so that for any , . In other words, let . We want to find a set of values for so that

Notice that , where

Therefore, we have . There are two possible repairs: either an OR gate or an XOR gate.

6. (10%) Suppose that we have the following cube-list at one node of our URP tautology recursion, and we need to decide on the splitting variable to use to cofactor and recurse. Which variable will you pick, and why?

|  |  |  |  |
| --- | --- | --- | --- |
| *x* | *y* | *z* | *w* |
| 11 | 01 | 01 | 11 |
| 01 | 11 | 10 | 01 |
| 01 | 10 | 11 | 11 |
| 10 | 11 | 10 | 10 |
| 01 | 11 | 01 | 10 |

**Solution**:

: binate, 4 cubes depend on it, diff = 2

: binate, 2 cubes depend on it, diff = 0

: binate, 4 cubes depend on it, diff = 0

: binate, 3 cubes depend on it, diff = 1

Variable *z*. Both *x* and *z* are the most binate variables. For *x*, ; for *z*, . Since *z* has the smaller , we choose *z*.

7. (16%) For the cube-list from Problem 6, suppose that we choose the splitting variable as *w*. What are the resulting cube-lists for its positive and negative cofactors, respectively?

**Solution**:

Positive cofactor is

|  |  |  |  |
| --- | --- | --- | --- |
| *x* | *y* | *z* | *w* |
| 11 | 01 | 01 | 11 |
| 01 | 11 | 10 | 11 |
| 01 | 10 | 11 | 11 |
| -- | -- | -- | -- |
| -- | -- | -- | -- |

Negative cofactor is

|  |  |  |  |
| --- | --- | --- | --- |
| *x* | *y* | *z* | *w* |
| 11 | 01 | 01 | 11 |
| -- | -- | -- | -- |
| 01 | 10 | 11 | 11 |
| 10 | 11 | 10 | 11 |
| 01 | 11 | 01 | 11 |