F4 - Logic Circuit Design Exercises

2nd Semester, 2008

Lab 5

Ben A. Abderazek
The University of Aizu
E-mail:benab@u-aizu.ac.jp

Release Date:	Oct. 29, 2008
Due Date	**:** Nov. 5, 2008

1 Today's Laboratory Objective

Practice how to design multi-level logic functions [1].

2 Background

For information about how to use Cadence tool, you can jump to the following tutorials:
`http://www.u-aizu.ac.jp/~benab/classes/cs/doc/cadence/`
You may also refer to the following document for LC clossary:
`http://www.u-aizu.ac.jp/~benab/classes/ld/doc/clossary.html`

2.1 Conversion

The canonical forms you have studied so far are expressed in terms of AND and OR gates, but you will rarely encounter these in digital systems. The underlying technologies are more efficient at implementing NAND and NOR gates.
Frequently, you will be confronted with the task of mapping a network with an arbitrary number of levels of AND and OR gates into one that consists only of NAND or NOR gates.

- The conversion process depends critically on DeMorgan's theorem.
- A NAND function can be implemented just as well by an OR gate with its inputs inverted.
- A NOR function can be implemented by an AND gate with its inputs complemented.
- Two-level: At most two gates between a primary input and a primary output.
- Multi-level: Any number of gates between a primary input and a primary output.

3 Exercise 5

Design one of the logic functions shown in Table 1 according to the last digit of your ID.
Note: More designs more scores!

Table 1: Truth Table.

Last Digit	3,9	4,5	0,6	1,7	2,8
a3a2a1a0	F1	F2	F3	F4	F5
0000	0	1	0	1	0
0001	0	0	1	0	1
0010	1	1	1	0	1
0011	0	1	0	0	0
0100	0	0	1	1	1
0101	1	0	1	1	0
0110	1	0	0	1	0
0111	0	1	1	0	1
1000	1	0	0	1	0
1001	0	1	1	0	1
1010	1	1	1	1	1
1011	1	0	1	0	0
1100	1	0	0	0	1
1101	0	0	0	1	0
1110	*	*	*	*	*
1111	*	*	*	*	*

3.1 Exercise 4.1 Tasks

1. Design of the minimized $sum - of - products$ (SOP) form using Karnaugh map.
2. Derivation of multi-level logic
   Transform the SOP fom using factoring or decomposition as much as possible so that the total number of literals is optimized. Show the number of literals for each function.
3. INV-NAND network
   Represent the obtained multi-level logic in the form of INV-NAND network. Here, NAND corresponds to 2-input NAND.
4. Technology mapping
   Using the tree covering algorithm, map library gates to INV-NAND network. Library gates area are shown in Table 2.
   Show the number of used gates for each type.
5. Draw the schematic for obtained logic circuit.
6. Prepare "testfixture.new" so that all input combinations are simulated.
7. Simulate the circuit and verify the correctness of the circuit using waveform.

4 Report submission

4.1 Contents

Your report should be prepared in English and should contain the following:

1. Your name, your ID and the Lab #.

Table 2: Gates Area.

TYPE	Function	AREA
INV	Inverter	2
NAND2	2 input NAND	4
NOR2	2 input NOR	4
AND2	2 input AND	6
OR2	2 input OR	6
NAND3	3 input NAND	6
NOR3	3 input NOR	6
AND3	3 input AND	8
OR3	3 input OR	8

2. **Exercise 5**

   - Truth table
   - Minimized SOP forms with the number of literals.
   - Show the process of transformation. Show the number of literals after transformations.
   - Draw the INV-NAND network.
   - Show the process of technology mapping. Show the number of used logic gates.
   - Schematics
   - Testfixure files
   - Waveform

4.2 Submission Format

Hard Copy.

References

[1] Hiroshi Saito, Logic Circuit Design Course Web page, Second Semester, 2008.