EE 1315 DIGITAL LOGIC LAB

EE Dept, UMD

EXPERIMENT # 4: Code Converter Design

Combinational logic can be designed by following a standard design procedure as shown below. Design the given problem using a minimum number of gates.

1. State the problem
2. Determine the required number of input and output variables
3. Assign the input and output variables to letter symbols
4. Derive the truth table
5. Simplify the function to use a minimum number of gates
6. Draw the circuit using gates
7. Implement and test the circuit

Given problem: Design a code converter that converts from a BCD (Binary Coded Decimal) to a 7536' code using a minimum number of gates. Both BCD and 7536' codes are used to express decimal digits 0-9, and thus they only use 10 patterns out of the 16 available patterns from the four bit combination. The unused bit patterns should be treated as don't care minterms. The code conversion table is given below:

Decimal digit	BCD Code	7536' Code
0	0000	0000
1	0001	1001
2	0010	0111
3	0011	0010
4	0100	1011
5	0101	0100
6	0110	1101
7	0111	1000
8	1000	0110
9	1001	1111

Notice that the 7536' code has a complemented symmetric relation, i.e., 5 to 9 codes are obtained by complementing 4 to 0. This type of code is called a self-complementing code and helps error detection.

	Inputs				Outputs			
Scalar Port	X0	X1	X2	X3	Y0	Y1	Y2	Y3
Nyxys I/O	SW0	SW1	SW2	SW3	LD0	LD1	LD2	LD3
Site	G18	H18	K18	K17	J14	J15	K15	K14

Scalar Port Assignment Table: X0 and Y0 are the LSB

EXPERIMENT #4 RESULTS

(Print out following pages and bring with to the lab session)

Your Name: _____

Witnessed by

Instructor or TA: _____

Date _____

Step 1: State the problem

Step 2: Determine the number of required inputs and outputs

No of inputs: _____

No of outputs: _____

Step 3: Assign the input and output variables to letter symbols

Step 4: Derive the truth table

Input				**Output**			
0	0	0	0				
0	0	0	1				
0	0	1	0				
0							
0							
0							
0							
0							
1							
1							
1							
1							
1							
1							
1							
1							

Step 5: Simplify the output functions using K-maps.

Step 6: Draw the circuit. K-map gives you only minimization at the level of the minimum number of literals. For the actual implementation, you should minimize the circuit at the gate level, especially for the circuits with multiple outputs like this lab. For this lab, design the circuit to use a minimum number of gates.

Step 7: Implement and test your circuit. Fill in the truth table below according to the input and output signal patterns of your circuit.

Input				**Output**			
0							
0							
0							
0							
0							
0							
0							
0							
1							
1							
1							
1							
1							
1							
1							
1							