17BCADLLP27: Programming Lab- Digital Logic

Practical Hours: 4 Hrs/week **Marks: Main exam: 80**
IA: 20

1. For the following functions, construct a truth table and draw a circuit diagram.
   1. y(A,B) = (AB)' + B'
   2. y(A,B,C) = (A + B)' C
   3. y(A,B,C) = (AC)' + BC
   4. y(A,B,C) = (A ⊕ B)C'
   5. y(A,B) = A' + B
   6. y(A,B,C) = ((A+B)'(B+C)')'
2. Study and verify the truth table of various logic gates
   - NOT, AND, OR, NAND, NOR, EX-OR, and EX-NOR
3. Simplify Boolean expressions and realize it.
4. Verification of Boolean Theorems using basic gates.
5. Design a 4-input NAND gate using two 2-input NAND gates and one 2-input NOR gate. Hint: Use DeMorgan's law
6. Construct the K-map for each of the following functions
   (a) f(A,B,C) = AB + A'BC' + AB'C
   (b) g(A,B,C) = A'C + ABC + AB'
   (c) h(A,B,C,D) = A'BC' + (A ⊕ B)C + A'B'CD' + ABC
7. For g(A,B,C) = A'C + ABC + AB', design the circuit for the minimal SOP expression found in problem 4 using just NAND gates and inverters. Label the pinouts on the circuit diagram. Build the circuit and demonstrate the working circuit.
8. For the functions listed below, construct a K-map and determine the minimal SOP expression.
   a. f(a,b,c) = a'b'c' + a'bc' + abc' + abc
   b. g(a,b,c) = ab'c' + abc' + abc + don't cares(a'bc + ab'c)
   Build the circuit required for (b).
9. Design and verify a half/full adder
10. Design and verify half/full subtractor
11. Design a 4 bit magnitude comparator using combinational circuits.
12. Design and verify the operation of flip-flops using logic gates.
13. A two bit counter is to be built that will count forward, 00 → 01 → 10 → 11 → 00, when a logical input is set high and counts in reverse order when it is low.
    (a) Draw the state transition diagram for this state machine.
    (b) Assuming a state machine were to be built using D flip-flops, determine the value of the next state for each of the flip-flops.
    Build and demonstrate the state machines
14. Verify the operation of a counter.
15. Verify the operation of a 4 bit shift register

- **Any open source simulator likeLogisimhttps://sourceforge.net/projects/circuit/ can be used.**
- **Breadboards/trainer kit may be used to realize logic gates**

Additional resource Verilog Online simulator: www.iverilog.com/ *may be used to* model digital circuits and systems.