Digital Logic (BIT 126) Assignment

Title: Digital Logic

Lecturer: Rohit Pandey, Pradip Bastola, Binod Adhikari

Academic Year: 2025 Fall Intake

Purpose To evaluate students' understanding of digital logic principles and their ability to design, analyze, and optimize digital circuits.

Learning Outcomes (LOs)

1. Understand number systems, binary codes, and Boolean algebra foundations.
2. Analyze combinational and sequential circuits (gates, adders, flip-flops, counters).
3. Design efficient digital systems using logic minimization techniques (K-Maps).

Objectives

- Assess theoretical knowledge of binary logic and IC families.
- Evaluate practical skills in converting specifications into optimized logic diagrams.
- Develop analytical skills for solving real-world digital design problems.

Task 1: Theoretical Analysis (50 Marks)

Answer ALL questions. Each question corresponds to a chapter of the syllabus.

1. **Binary Systems**

   a) What is the difference between Even Parity and Odd Parity? Explain how a parity bit detects errors during data transmission.

   b) Compare and contrast the 8421 BCD code with the Excess-3 code.

2. **Boolean Algebra and Logic Gates**

   a. Using the properties and theorems of Boolean Algebra, simplify the following Boolean expressions to their minimum number of literals:

   (a) A . B + A. (B + C) + B. (B + C)

   (b) (A + B) . (A + B')

   (c) A . B'. C + A. B. C + A. B. C'

   b. Given the Truth Table below, write the Boolean function F in Sum of Products (SOP) and POS form and simplify the SOP form it if possible.

A	**B**	**C**	**Output F**
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

3. **Simplification of Boolean Function**

   a) Simplify the following Boolean function using a 4-variable K-Map: F(A, B, C, D) = $\sum_m \quad (0, 1, 2, 4, 5, 6, 8, 9, 12, 13, 14)$, write the minimal SOP expression. Also write a POS expression for it.

   b) Simplify the following Boolean function using a 3-variable K-Map: F(A, B, C) = $\sum_m \quad (0, 1, 2, 4) + d(3,7)$.

c. Consider the following Boolean function given in Sum of Products form: F(A, B, C) = AB + BC + AC

Draw the standard AND-OR logic diagram for this function. Convert the circuit to use only NAND gates. Verify that the two-level NAND-NAND implementation is equivalent to the AND-OR implementation.

4. **Combinational Logic**

a) Design a 2421 code to Gray Code converter circuit.

b) What is parity? Design 3 bit parity generator and checker circuit.

5. **Combinational Logic with MSI and LSI**

a. Design a 3-to-8 Line Decoder. Show how to implement the following Boolean function using a 3-to-8 decoder and a single OR gate: F (A, B, C) = $\sum_m \quad (1, 3, 5, 7)$

b. Implement full adder using PLA.

6. **Sequential Logic**

a) Explain the fundamental difference between a Latch and a Flip-Flop. Which one is level-triggered and which one is edge-triggered? Draw the logic diagram of an SR Latch using NOR gates. Explain the "Forbidden State" (Indeterminate State) in an SR Latch. Why must the input condition S = 1, R = 1 be avoided?

b) The JK Flip-Flop is designed to eliminate the forbidden state of the SR Flip-Flop. What happens to the output Q when both inputs J = 1 and K = 1? Define the "Toggle" mode. The Race-Around Condition: Explain what the "Race-Around Condition" is in a level-triggered JK Flip-Flop. How does the Master-Slave configuration solve this problem?

7. **Registers, Counters and Memory Units**

a) Explain about Parallel – In Serial – Out (PISO) register.

b) Design 4 bit binary synchronous counter with any flip flop you like.

Task 2: Practical Components (50 Marks)

Perform ALL tasks and include screenshots in your submission.

Practical 1: Combinational Logic (15 Marks)

Design the combinational logic circuit (a BCD to 7-Segment Decoder) that takes a 4-bit Binary Coded Decimal input and illuminates the correct segments to display the decimal digit.

Practical 2: Combinational Logic with MSI and LSI (20 Marks)

Derive the Boolean Function Design a combinational circuit with 3 inputs (A, B, C) and 3 outputs (X, Y, Z). The circuit behaves according to the magnitude of the input:

i. When the binary input is less than 3 (0, 1, 2): The output is the Bitwise Inverse (NOT operation) of the input. *(Example: Input 010 becomes Output 101).*

ii. When the binary input is 3 or greater (3, 4, 5, 6, 7): The output is equal to the Input minus 3.

Tasks:

Draw the Truth Table listing all inputs and corresponding outputs.

Obtain the simplified Sum of Products (SOP) expression for outputs X, Y, and Z.

Make a logic diagram.

Implement the simplified expressions using NAND gates.

Practical 3: Tinkercad Observation (15 Marks)

Refer to the practical 2 question and make the circuit based on the obtained expression in Tinker Cad Website. Use following components:

a. Breadboard for connection base
b. For inputs use push buttons
c. Use leds for outputs
d.Use respective ICs for AND, OR, NOT, NAND gates
e. Check any 3 combinations based on the truth table of question 2 and take screen shot of the output you obtained and attach the picture as the answer. And briefly explain the steps you followed.

Your Partner for Excellence

Texas

College of Management & IT

LINCOLN

UNIVERSITY COLLEGE

Texas College of Management & IT

Bachelor of Information Technology (BIT)

Assignment Title: Digital Logic

Submitted By:

Name: ____________

LCID: ____________

Year/Semester: BIT 2nd Year / 3rd Semester

Submitted To:

Lecturer: Rohit Pandey/ Pradip Bastola/ Binod Adhikari

Course: Digital Logic (BIT 126)

Date of Submission: ____________

Texas College of Management & IT, Kathmandu

Assignment Structure

- Title Page
- Table of Contents
- Introduction
- Task 1 (Theory)
- Task 2 (Practical)
- Conclusion
- References (APA 7th edition)
- Appendix (Slides, Diagrams)