LESSON PLAN 2025-26(WINTER)

NAME OF THE TEACHER : DEEPAK KUMAR BARDA, LECT.(STAGE-II,CSE)

Subject: **DIGITAL ELECTRONICS AND COMPUTER ORGANISATION(Course Code: CSEPC 207/TH4)**
Program: Diploma in Computer Science and Engineering
Semester: 3rd
Total Contact Hours: 45
Total Marks: 100
Assessment: Internal Assessment – 30, End Term – 70

After completion of the course, the students will be able to:
CO1- Define key concepts of digital electronics, including number systems, Boolean algebra, and logic gates.
CO2- Explain the principles behind combinational and sequential circuits, such as multiplexers, flip-flops, and counters, and their applications.
CO3-Implement simplified logic circuits using Karnaugh Maps and Boolean algebra to solve real-world digital design problems.
CO4-Analyze the instruction cycle, memory organization, and processor architecture to evaluate system performance and identify bottlenecks.
CO5-Design to simulate a basic CPU operation or create a functional digital circuit using the concepts of digital electronics and computer organization.

Day	Unit	Topic/Sub-Topic	Learning Objective	Activities	Homework	COURSE OBJECTIVE
Unit-1-Introduction to Digital Electronics(8 Periods)						
1	I	Difference Between Analog and Digital Signals	Differentiate between analog and digital signals and their applications.	Compare analog vs. digital clocks and thermometers . Discuss the advantages of digital systems.	Find and list 3 real-world examples of both analog and digital signals.	CO1
2	I	Number Systems: Binary, Octal, Decimal, and Hexadecimal	Explain the concept of different number systems used in computing.	Convert numbers between the four systems on a whiteboard and with online tools.	Convert the decimal number 255 to binary, octal, and hexadecimal.	CO1
3	I	Conversion Between Number Systems (1/2)	Convert numbers from any base to decimal.	Practice converting binary, octal, and hexadecimal numbers to their decimal equivalents.	Convert 10110 from binary to decimal and 3A from hexadecimal to decimal.	CO1

4	I	Conversion Between Number Systems (2/2)	Convert numbers from decimal to other bases.	Practice converting decimal numbers to binary, octal, and hexadecimal.	Convert 150 from decimal to binary and 43 from decimal to octal.	CO1
5	I	Binary Arithmetic: Addition and Subtraction	Perform basic arithmetic operations in binary.	Solve binary addition and subtraction problems on the board. Emphasize the concept of borrowing and carrying.	Solve 3 binary addition and 3 binary subtraction problems.	CO1
6	I	Boolean Algebra: Basic Operations and Laws	Understand the fundamental operations and laws of Boolean algebra.	Walk through the laws of Boolean algebra (e.g., De Morgan's Law, Associative Law) with truth tables.	Create truth tables for A + B * C and (A + B) * (A + C).	CO1
7	I	Boolean Algebra: Simplification	Simplify Boolean expressions using the laws of Boolean algebra.	Work through examples of simplifying complex expressions on the board.	Simplify the Boolean expression A * B + A * (B + C).	CO1
8	I	Unit I Review & Quiz	Consolidate knowledge of number systems and Boolean algebra.	Solve a mix of conversion and simplification problems.	N/A	CO1
Unit-2-Logic Gates and Circuits: Logic Gates(7 Periods)						
9	II	Logic Gates: AND, OR, NOT, NAND, NOR, XOR, XNOR	Identify and explain the function of the basic and universal logic gates.	Draw the symbols for each gate. Use interactive simulators to demonstrate their functionality.	Draw the logic symbols and truth tables for all 7 logic gates.	CO1

10	II	Design of Logic Circuits Using Boolean Algebra (1/2)	Design a simple logic circuit from a given Boolean expression.	Work through an example of translating an expression like (A + B) * C into a circuit diagram.	Draw the logic circuit for the expression A * B + C.	CO1
11	II	Design of Logic Circuits Using Boolean Algebra (2/2)	Design a complex logic circuit from a word problem.	Translate a word problem into a truth table, then a Boolean expression, and finally a circuit.	Design a circuit that outputs a 1 only when two of its three inputs are 1.	CO1
12	II	Karnaugh Maps (K-Maps) for 2- and 3-Variable Simplification	Use K-Maps to simplify Boolean expressions with up to 3 variables.	Introduce K-Maps as a visual simplification tool. Practice grouping and finding simplified expressions.	Simplify the expression Σ(0, 1, 4, 5) using a K-Map.	CO1
13	II	K-Maps for 4-Variable Simplification	Simplify 4-variable expressions using K-Maps.	Practice grouping 4-variable K-Maps, including overlapping groups.	Simplify the expression Σ(0, 2, 5, 7, 8, 10, 13, 15) using a K-Map.	CO1
14	II	Practical Applications of Logic Gates	Discuss real-world applications of logic gates.	Watch videos on how logic gates are used in alarms, calculators, and computer circuits.	Find and explain how logic gates are used in a traffic light control system.	CO1
15	II	Unit II Review & Quiz	Consolidate knowledge of logic gates and K-Maps.	Solve a mix of design and simplification problems using both Boolean algebra and K-Maps.	N/A	CO1

Unit-3-Combinational and Sequential Circuits(7 periods)						
16	III	Combinational Circuits: Multiplexers and Demultiplexers	Explain the function and design of MUX and DEMUX circuits.	Draw block diagrams and truth tables for a 4-to-1 MUX and a 1-to-4 DEMUX.	Design a 8-to-1 MUX using 4-to-1 MUXes.	CO2
17	III	Combinational Circuits: Encoders and Decoders	Explain the function and design of encoders and decoders.	Draw and explain a priority encoder and a 3-to-8 line decoder.	Explain how a decoder is used to select a specific memory location.	CO2
18	III	Sequential Circuits: Flip-Flops (SR, JK)	Explain the operation of SR and JK flip-flops and their characteristics.	Use timing diagrams to demonstrate the behavior of SR and JK flip-flops.	Draw the logic diagram and truth table for a JK flip-flop.	CO2
19	III	Sequential Circuits: D and T Flip-Flops	Explain the operation of D and T flip-flops and their applications.	Discuss how D and T flip-flops are derived from the JK flip-flop.	Explain the difference between an SR flip-flop and a D flip-flop.	CO2
20	III	Counters: Synchronous and Asynchronous Counters	Differentiate between synchronous and asynchronous counters.	Draw and explain the operation of a 3-bit ripple counter and a 3-bit synchronous counter.	Design a 4-bit asynchronous counter using T flip-flops.	CO2
21	III	Registers and Shift Registers	Explain the function and types of registers.	Discuss the purpose of registers in CPUs. Demonstrate the operation of a simple shift register.	Draw a 4-bit SIPO (Serial-In, Parallel-Out) shift register.	CO2
22	III	Unit III Review & Quiz	Consolidate knowledge of combinational and sequential circuits.	Solve problems related to flip-flops, counters, and registers.	N/A	CO2

Unit-4-Fundamentals of Computer Organization(6 Periods)						
23	IV	Basic Structure of a Computer	Identify the main components of a computer system.	Draw a block diagram of a computer system, labeling the CPU, memory, and I/O devices.	Write a short summary of the role of the CPU, memory, and I/O in a computer.	CO3
24	IV	The Instruction Cycle: Fetch, Decode, Execute	Explain the steps involved in executing a single instruction.	Walk through a simple example of the instruction cycle using a mock instruction.	Explain each step of the instruction cycle in your own words.	CO3
25	IV	Memory Organization: Types of Memory (1/2)	Differentiate between RAM and ROM.	Discuss the characteristic s, volatility, and uses of RAM and ROM.	Create a table comparing RAM and ROM based on volatility, speed, and size.	CO3
26	IV	Memory Organization: Types of Memory (2/2)	Understand the purpose of Cache and Virtual Memory.	Explain the concept of the memory hierarchy. Discuss how cache memory improves performance.	Write a paragraph explaining what virtual memory is and why it's used.	CO3
27	IV	Introduction to Buses	Explain the function of the Address Bus, Data Bus, and Control Bus.	Use a diagram to illustrate how the three buses connect the CPU, memory, and I/O devices.	Describe the purpose of each bus in a computer system.	CO3
28	IV	Unit IV Review & Quiz	Consolidate knowledge of computer organization.	Answer conceptual questions about the instruction cycle, memory, and buses.	N/A	CO3

Unit-5-Processor Architecture and Control(6 Periods)						
29	V	Microprocessors vs. Microcontrollers	Differentiate between microprocessors and microcontrollers.	Discuss the architecture and typical applications of both.	Research and list 3 devices that use a microprocessor and 3 that use a microcontroller.	CO4
30	V	Basics of Arithmetic Logic Unit (ALU) and Control Unit	Understand the function of the ALU and Control Unit.	Draw a block diagram of the CPU, highlighting the ALU and Control Unit. Discuss their roles.	Write a short paragraph explaining the relationship between the ALU and the Control Unit.	CO4
31	V	Instruction Set Architecture (ISA): RISC vs. CISC	Compare and contrast RISC and CISC architectures.	Discuss the key characteristics and trade-offs of RISC vs. CISC.	Research and list a microprocessor that uses a RISC architecture and one that uses a CISC architecture.	CO4
32	V	Pipelining and Performance Optimization (1/2)	Explain the concept of instruction pipelining.	Use a simple analogy (e.g., an assembly line) to explain pipelining and its benefits.	Identify a potential hazard that can occur in a simple 3-stage pipeline.	CO4
33	V	Pipelining and Performance Optimization (2/2)	Discuss other techniques for performance optimization.	Talk about techniques like branch prediction and superscalar execution.	Research and explain one performance optimization technique not discussed in class.	CO4
34	V	Unit V Review & Quiz	Consolidate knowledge of processor architecture.	Solve conceptual problems related to pipelining and ISA.	N/A	CO4

Unit-6-Input/Output Systems and Advanced Topics(11 Periods)						
35	VI	I/O Devices and Interfaces	Identify common I/O devices and their interfaces.	Discuss how devices like keyboards, mice, and printers communicate with the computer.	Research the interface used by a modern SSD (e.g., SATA, PCIe).	CO5
36	VI	Interrupts and DMA (Direct Memory Access)	Explain the concepts of interrupts and DMA.	Use a simple scenario to explain why interrupts are necessary. Discuss how DMA improves I/O performance.	Write a paragraph on the difference between programmed I/O and interrupt-driven I/O.	CO5
37	VI	Overview of Modern Trends: Multicore Processors	Understand the concept and benefits of multicore processors.	Discuss the difference between a single-core and a multi-core processor and the challenges of parallel programming.	Research and explain the concept of hyper-threading.	CO5
38	VI	Modern Trends: GPUs and Embedded Systems	Explain the purpose and applications of GPUs and embedded systems.	Discuss how GPUs are different from CPUs and their role in parallel processing and AI.	Find and explain an example of an embedded system in a household appliance.	CO5
39	VI	Mini-Project: Problem Selection & Proposal	Choose a final project to design or simulate.	Brainstorm project ideas from the syllabus (e.g., a simple counter, a basic CPU operation). Write a project proposal.	Write a 1-page proposal for your mini-project, including objectives and components.	CO5
40	VI	Mini-Project: Design Phase (1/2)	Create the high-level design for the mini-project.	Draw a block diagram and flowcharts for the chosen project.	Complete the logical design and flowcharts for your project.	CO5

41	VI	Mini-Project: Design Phase (2/2)	Refine the circuit design or simulation steps.	Draw the detailed logic circuit for the digital circuit project or the detailed steps for the CPU simulation.	Complete the detailed design and check it for logical errors.	CO5
42	VI	Mini-Project: Implementation & Testing (1/2)	Begin building the project.	Use a logic simulator or a programming language to build and test the first part of the project.	Implement the first major component of your project.	CO5
43	VI	Mini-Project: Implementation & Testing (2/2)	Complete the project and test for functionality.	Finish building and testing the project. Debug any issues that arise.	Complete the implementation and write a test plan.	CO5
44	VI	Mini-Project: Finalization and Documentation	Prepare the project for demonstration.	Add comments to the code, write a README.md file, and prepare a presentation.	Write a report summarizing your project, including the design, implementation, and results.	CO5
45	Final	Mini-Project Demonstration & Final Review	Present the project and review key course concepts.	Students demonstrate their final projects. A brief review of the entire course.	N/A	CO5

Deepak kumar Bal
11-07-2025
Signature of Teacher

Deepak kumar Bal
11-07-2025
Signature of HOD