Course Title	**Digital Logic Design (Lab)**
Course Code	**CC-110-L**
Credit Hours	1 (0,1)
Category	Computer Science Core
Prerequisite	None
Co-Requisite	None
Follow-up	CC-210 Computer Organization and Architecture
Course Introduction	The course introduces the concept of digital logic, gates and the digital circuits. Further, it focuses on the design and analysis combinational and sequential circuits. It also serves to familiarize the student with the logic design of basic computer hardware components.
Course Description	***Implementation and Practice of the concepts studied in "CC-110 Digital Logic Design"*** **Topics:** Introduction to Digital Systems, Number Systems, Introduction to Boolean Algebra, Basic theorems and properties of Boolean Algebra, Boolean Functions, Logic Gates, NAND and NOR Implementation, Representation of Function in Sum of Minterms or Product of Maxterms, Simplification of Boolean function using Karnaugh Map, Don't care Conditions, The Tabulation Method, Introduction to Combinational Logic, Design of Adders, Design of Subtractors, Code Convertors, Analysis Procedure of Combinational Circuits, Binary Parallel Adders, Decimal Adders, Magnitude Comparator, Decoders and its applications, Multiplexers, Demultiplexers, Encoders, ROM, Programmable Logic Array (PLA), Introduction to Sequential Circuits, Basic Flip Flop, Clocked RS Flip Flop, Clocked D Flip Flop, Clocked JK Flip Flop, Clocked T Flip Flop, Analysis of Clocked Sequential Circuits, State Reduction and Assignment, Flip Flop Excitation tables, Design Procedure, Design of Counters, Design with State Equations, Introduction to Registers, Shift Registers, Ripple Counters, Synchronous Counters , Timing Sequences, Memory Unit, Random Access Memory. Introduction Programmable Logic Devices (CPLD, FPGA), Lab Assignments using tools such as Verilog HDL/VHDL, MultiSim. Familiarization with Digital Electronic Trainer, Logic gates operations, Half Adder Operation, Full Adder Operation, Half Subtractor Operation, Full Adder Operation, 7-Segment Display Operation, Decoder Operation, BCD To 7-Segment Display, Multiplexer Operation, Using Multiplexer and Demultiplexer / Decoder, Multiplexing 7-Segment Displays, Comparator Operations, D Latch and Flip-Flop Operation, Latching BCD Data for Displaying On 7- Segment Display, JK Flip-Flop Operation, Random Access Memories
Text Book(s)	M. Morris Mano, Digital Logic and Computer Design, 1st Edition, Pearson, 1979, ISBN: 0132145103.
Reference Material	4. Thomas L. Floyd, Digital Fundamentals, 10th Edition, Prentice Hall, 2008, ISBN: 0132359235. 5. Fundamental of Digital Logic with Verilog Design, Stephen Brown, 2/e

<table>
<tr><td colspan="4">Data Structures</td></tr>
<tr><td>CLOs</td><td colspan="3">1. Implement and analyze various data structures such as arrays, linked lists, stacks, queues, trees, and graphs.
2. Select appropriate data structures to optimize algorithm performance.
3. Demonstrate proficiency in traversing and manipulating data structures.
4. Evaluate the efficiency of algorithms based on data structure choices.
5. Solve complex problems using suitable data structures and algorithms.</td></tr>
<tr><td rowspan="6">CLO–PLO Mapping</td><td>CLO</td><td>Exact Statement (Short)</td><td>PLO Mapping</td></tr>
<tr><td>CLO1</td><td>Implement & analyze arrays, linked lists, stacks, queues, trees, graphs</td><td>PLO2, PLO3</td></tr>
<tr><td>CLO2</td><td>Select appropriate data structures to optimize algorithms</td><td>PLO2, PLO3, PLO4</td></tr>
<tr><td>CLO3</td><td>Traverse & manipulate data structures efficiently</td><td>PLO2, PLO5</td></tr>
<tr><td>CLO4</td><td>Evaluate efficiency of algorithms based on data structures</td><td>PLO2, PLO3, PLO4</td></tr>
<tr><td>CLO5</td><td>Solve complex problems using suitable data structures & algorithms</td><td>PLO2, PLO4</td></tr>
<tr><td rowspan="6">Bloom’s Taxonomy Mapping</td><td>CLO</td><td>Bloom Level</td><td>Reason</td></tr>
<tr><td>CLO1</td><td>Apply → Analyze (C3–C4)</td><td>Implementing and analyzing arrays, linked lists, stacks, queues, trees, graphs</td></tr>
<tr><td>CLO2</td><td>Analyze → Evaluate (C4–C5)</td><td>Selecting appropriate data structures to optimize algorithm performance</td></tr>
<tr><td>CLO3</td><td>Apply → Create (C3–C6)</td><td>Traversing and manipulating data structures efficiently</td></tr>
<tr><td>CLO4</td><td>Analyze → Evaluate (C4–C5)</td><td>Evaluating efficiency of algorithms based on data structure choices</td></tr>
<tr><td>CLO5</td><td>Apply → Create (C3–C6)</td><td>Solving complex problems using suitable data structures and algorithms</td></tr>
</table>