Course Objectives
To familiarize with organizational aspects of memory, processor and I/O.
Course Outcomes

Upon successful completion of the course, the students will be able to
1. Identify different types of instructions
2. Differentiate between micro-programmed and hard-wired control units.
3. Analyse the performance of hierarchical organization of memory.
4. Summarize different data transfer techniques.
5. Demonstrate arithmetic operations on fixed- and floating-point numbers and illustrate concepts of parallel processing.

UNIT – I
Register Transfer Language and Micro Operations: Introduction- Functional units, computer registers, register transfer language, register transfer, bus and memory transfers, arithmetic, logic and shift micro-operations, arithmetic logic shift unit.
Basic Computer Organization and Design: Instruction codes, instruction cycle.
Register reference instructions, Memory – reference instructions, input – output and interrupt.

UNIT – II
CPU and Micro Programmed Control: Central Processing unit: Introduction, instruction formats, addressing modes.Control memory, address sequencing, design of control unit - hard wired control, micro programmed control.

UNIT – III
Memory Organization: Memory hierarchy, main memory, auxiliary memory, associative memory, cache Memory and mappings.

UNIT – IV
Input-Output Organization: Peripheral Devices, input-output interface, asynchronous datatransfer, modes of transfer- programmed I/O, priority interrupt, direct memory access, Input – Output Processor (IOP).

UNIT – V
Computer Arithmetic and Parallel Processing: Data representation- fixed point, floating point, addition and subtraction, multiplication and division algorithms.
Parallel Processing-Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline.

Text Books:
1. M. Moris Mano, "Computer Systems Architecture", 3rd edition, Pearson/ PHI.

Reference Books:

1. Carl Hamacher, ZvonksVranesic, SafeaZaky, "Computer Organization", 5th edition,McGraw Hill.
2. William Stallings, "Computer Organization and Architecture", 8th edition, Pearson/PHI.

SUGGESTED CO-CURRICULAR ACTIVITIES & EVALUATION METHODS:
Unit 1: Activity: Quiz competition on micro-operations.
   Evaluation Method: Accuracy and speed in answering quiz questions.
Unit 2: Activity: Instruction Format Puzzle: Solving a puzzle to decode and understand instruction formats.
   Evaluation Method: Accuracy and speed in completing the puzzle.
Unit 3: Activity: Memory Hierarchy Poster: Creating informative posters or infographics onmemory hierarchy.
   Evaluation Method: Clarity of information, presentation and creativity of visual design.
Unit 4: Activity: I/O Troubleshooting Challenge
   Evaluation Method: problem identification, feasibility of proposed solutions, and clarityof explanations.
Unit 5: Activity: Case Study on Parallel processing architecture.
 Evaluation Method: Understanding of parallel processing concepts and architectures.

<div align="center">

III Semester
Course 3: Computer Organization
Credits -1

</div>

Lab Experiments

1. Implement a C program to convert a Hexadecimal, octal, and binary number to decimal number vice versa.
2. Implement a C program to perform Binary Addition & Subtraction.
3. Implement a C program to perform Multiplication of two binary numbers.
4. Implement arithmetic micro-operations using logic gates.
5. Implement logic and shift micro-operations using logic gates.
6. Implement a C program to perform Multiplication of two binary numbers (signed) using Booth's Algorithms.
7. Implement a C program to perform division of two binary numbers (Unsigned) using restoring division algorithm.
8. Implement a C program to perform division of two binary numbers (Unsigned) using non-restoring division algorithm.
9. Write assembly language code for A+B*(C-D) using various instruction formats in MASMor any open-source assembler.
10. Write assembly language code for A+B*C using various addressing modes in MASM or any open-source assembler.