Semester – II

BRANCH	Subject Title	Subject Code
B.Sc. (Computer Science – Minor/GE)	COMPUTER SYSTEM ARCHITECHTURE	CSM2/CS-121

Course Learning Outcomes(CLO)

On completion of this course, learners will be able to:

1. Understand the basic structure, operation and characteristics of digital computer.
2. Be able to design simple combinational digital circuits based on given parameters.
3. Familiarity with working of arithmetic and logic unit as well as the concept of pipelining
4. Know about hierarchical memory system including cache memories and virtual memory.
5. Understand concept and advantages of parallelism.threading, multiprocessors and multicore processors.
6. Know the contributions of Indians in the field of computer architecture and related technologies.

CONTENT

Unit - I

Fundamentals of Digital Electronics: Data Types, Complements, Fixed-Point Representation, Floating-Point Representation, Binary and other Codes, Error Detection Codes.

Logic Gates, Boolean Algebra, Map Simplification. Combinational Circuits, Sequential Circuits, simple combinational circuit design problems.

Circuits- Adder- Subtractor, Multiplexer, Demultiplexer. Decoders, Encoders Flip-Flops, Registers, Counters.

Unit - II

Basic Computer Organization: Instruction codes, Computer Registers, Computer Instructions, Timing & Control, Instruction Cycles, Memory Reference Instruction. Input-Output & Interrupts, Complete Computer Description & Design of Basic Computer.

Unit - III

Instructions - Instruction formats, Addressing modes, Instruction codes, Machine language, Assembly language.

Register Transfer and Micro operations - Register Transfer language, Register Transfer, Bus & Memory Transfer, Arithmetic Micro-operations, Logic Micro-operations, Shift Micro-operations.

Unit - IV

Processor and Control Unit - Hardwired vs. Micro programmed Control Unit, General Register Organization, Stack Organization, Instruction Format, Data Transfer & Manipulation, Program Control, Introductory Format, Data Transfer & Manipulation, Program Control, Introductory concept of RISC, CISC, advantages and disadvantages of both.

Pipelining - concept of pipelining, introduction to pipelined data path and control - Handling Data hazards & Control hazards.

Unit - V

Memory and I/O System - Peripheral Devices, I/O Interface.,

Data Transfer Schemes- Program control, Interrupt, DMA Transfer. I/O Processor.

Memory Hierarchy, Processor vs. Memory Speed, High-Speed Memories, Main memory, Auxiliary memory, Cache Memory, Associative Memory, Interleaving, Virtual Memory, Memory Management.

Unit - VI

Parallelism - meaning, types of parallelism, introduction to Instruction- level-parallelism, Parallel processing challenges, Applications.

Flynn's classification - Introduction to SISD, SIMD, MISD, MIMD

Hardware multithreading - Introduction, types, advantages and applications.

Multicore processors - Introduction, advantages, difference from Multiprocessor.

Unit - VII

Indian contribution to the field - Contributions of reputed scientists of Indian origin - like - Dr. Vinod Dham - Father of Intel Pentium Processor, Dr. Ajay Bhat - Co-Inventor of USB Technology, Dr. Vinod Khosla- co-founder of Sun Microsystems, Dr. Vijay P Bhatkar - architect of India's national initiative in supercomputing, and many others.

Parallel Computing projects of India - PARAM, ANUPAM, FLOSOLVER, CHIPPS etc. Other relevant contributors and contributions.

Practical

Suggestive list of Practicals

1. To study basic gates (AND, OR. NOT) and verify their truth tables.

2. To convert a given binary number to Gray code using IC 7486.

3.To study and verify NAND as Universal gate using IC 7400.

4. To study half adder using basic gates and verify its truth table.

5. To study Full Adder using basic gates and verify its truth table.

6. To realize basic gates (AND, OR, NOT) from Universal gates (NAND and NOR).

7. To verify truth table of 4-bit adder using IC 7483.

8. To design and construct RS flip Flop using gates and verify the truth

9. To design and construct JK flip Flop using gates and verify the truth

10. To verify DeMorgan's Theorem.