Course Name :	Computer Organization	**Approval: 9th senate meeting**
Course Number:	CS 201	
Credits:	3-0-0-3	
Prerequisites:	None	
Intended for:	UG	
Distribution:	Institute Core	
Semester:	V	

Preamble:

A professional in any field of computing should not regard the computer as just a black box that executes programs by magic. All students of computing should acquire some understanding and appreciation of a computer system's functional components, their characteristics, their performance, and their interactions. Students need to understand the computer architecture in order to make best use of the software tools and computer languages they use to create programs. In this introduction, the term architecture is taken to include instruction set architecture (the programmer's abstraction of a computer), organization or micro architecture (the internal implementation of a computer at the register and functional unit level), and system architecture (the organization of the computer at the cache, and the bus level). Computer architecture also underpins other areas of the computing curriculum such as operating systems (input/output, memory technology) and high level languages (pointers, parameter passing).

Course Outline:

Digital Logic and Data Representation

Introduction to digital logic (logic gates, flip-flops, circuits); Logic expressions and Boolean functions; adder, subtractor, Design of arithmetic and logic unit (ALU)., Representation of numeric data, signed and unsigned arithmetic; Range, precision and errors in floating-point arithmetic; Representation of text, audio and images.

Computer Architecture and Organization

History of the digital computer; Introduction to instruction set architecture, micro architecture and system architecture; Instruction sequencing, flow-of-control , subroutine call and return mechanisms; Structure of machine-level programs; Low-level architectural support for high-level languages.

Interfacing and I/O strategies

I/O fundamentals: handshaking and buffering; Interrupt mechanisms; vector and prioritized, interrupt acknowledgement; Buses: protocols, arbitration, direct memory access(DMA); Examples of modern buses: e.g. PCIe, USB, Hyper transport.

Memory Architecture

Storage systems and their technology, Storage standards (CD-ROM,DVD); Memory hierarchy, latency and throughput; Cache memories-operating principles, replacement policies, multilevel cache, virtual memory system: page table and TLB, cache coherency

Functional Organization

Review of register transfer language to describe internal operations in a computer; CISC vs RISC Designs simple implementation schemes datapath design, control unit Micro architectures-hardwired and micro programmed realizations; Instruction pipelining and instruction-level parallelism (ILP); Overview of superscalar architectures; Processor and system performance; Performance their measures and their limitations.

CO Lab:

1. The assignment will be designed to assist the theory covered in the class:
2. Some examples are given as follows:
3. Design adder, subtractor in breadboard
4. Designing ALU (Arithmetic and Logical Unit) in breadboard
5. Programming in assembly language and perform some basic operations addition, subtraction, sorting

Text books

1. C.Hamacher, Z.Vranesic and S.Zaky, Computer Organization, 5th Ed., McGraw-Hill, 2002
2. J.P.Hayes, Computer Architecture and Organization, 3rd Ed
3. D. A. Patterson and J.L.Hennessy, Computer Organization and Design – The Hardware/Software Interface
4. William Stallings, "Computer Organization & Architecture designing for performance, 7th Ed.