COURSE PLAN AND EVALUATION PLAN

1. Course Code: EC792
2. Course Title: **HIGH PERRFORMANCE COMPUTING ARCHITECTURES**
3. L – T – P**:** **3-0-2**
4. Credits**:** **4**
5. Pre-requisite**:** Digital System Design
6. Teaching Department**:** **Electronics & Communication Engg.**
7. Course Instructor**:** **Dr SUMAM DAVID S.**
8. Objectives of the Course:
   - Understand the role of performance in designing computer systems and compare alternate design choices
   - Understand how to design an instruction set and its impact on processor design.
   - To design processor data path and control path for scalar and pipelined systems and Implement a minimal pipelined RISC ISA
   - Understand memory hierarchy design and its impact on overall processor performance
   - Awareness of current design issues in high performance computing systems
9. Course Outcomes:

At the end of the course the student must be able to

CO1: Understand basic aspects of high-performance computer architectures
CO2: Perform quantitative analysis of modern computing systems
CO3: Evaluate the performance of available choices for exploiting parallelism in Instruction, data and memory
CO4: Design a representative high performance computing subsystem/system within the given constraints

Course Coverage**:**

Module	Contents	Objectives	Lecture	Evaluation
Introduction	Introduction to COA – Objectives of the course, motivation, course plan, evaluation method, references, combinational, sequential, pipelined architectures	• Appreciate the relevance of the course	**L1**	
	Components of a computer system. Evolution of Technology. Factors affecting computer systems design Performance measures - Benchmark suites, CPI, IPC, speedup, Amdhal's law	• Understand architecture of a basic computer system and its components, role of performance in designing computer systems	**L2 – L4**	**Analysis**
Instruction Set Architecture	The role of an instruction set. interface between hardware and software; issues to consider when designing an instruction set; addressing modes.	• Understand how to design an instruction set and its impact on processor design	**L5-L7**	**Comprehension**
Computer Arithmetic	Number system, addition and subtract, adders; multiplication and multipliers; division and dividers; floating point numbers and floating point units	• To Implement arithmetic algorithms	**L8-L9**	**Design**

Processor Design	Data path and control; single cycle design and implementation; simplifying control design; multicycle implementation of data path and control	• To implement datapath and control path of processors	**L10-L14**	**Design**
Single cycle RISC processor implementation	Design a simple RISC processor for a small subset of instructions and implement on FPGA board	• Understanding processor architecture by implementing a small subset of instructions (single cycle)	**P1-2**	**Design**
Assembly language programming	Write simple programs in MIPS assembly language and test these on an instruction set simulator	• Understanding architecture and instructions through assembly programming	**P3**	**Comprehension**
Pipelining	Basic concepts in pipelining; data path for pipeline processor implementation, data hazard and forwarding, data hazard and stalling; control design for pipelines, Scheduling (static and dynamic) and forwarding to reduce/ minimise pipeline stalls	• Implement a minimal RISC ISA (MIPS) pipeline • Examine techniques to improve pipeline efficiency	**L15-L20**	**Design**
Instruction pipeline visualisation	Using an instruction pipeline visualization tool understand the performance issues related to pipelining	• Understand the performance issues related to pipelining using a simulator	**P4**	**Analysis**
Pipelined RISC processor implementation	Design a pipelined RISC processor for a small subset of instructions and implement on FPGA board Enhance the design to include more instructions	• Implement architectural and performance enhancements	**P5-7**	**Design**
Instruction level parallelism	Dynamic scheduling, hardware speculation, Exploiting instruction level using software approaches – static branch prediction, hardware support, Limits of Instruction level parallelism	• Examine hardware and software techniques for ILP • Limits of ILP	**L21-L27**	**Analysis**
Memory Hierarchy	Cache memories - Mapping and replacement policies, reducing misses, reducing penalties, main memory, virtual memory	• Examine techniques to improve memory management	**L28-L34**	**Design**
Architectural simulation	Analysing processor and memory performance using architectural simulator	• Understanding memory access patterns and changing cache configuration to analyse the impact on performance using architectural simulator	**P8-10**	**Design and Analysis**
Data Level and Thread Level Parallelism	Data Level Parallelism – Vector, SIMD, VLIW, GPU architectures, multiple instruction issue, Thread level parallelism, centralized shared memory architectures, memory consistency issues, Introduction to Domain specific architectures (DSA)	• Examine ways to improve processor performance by exploiting Data and thread level parallelism.	**L35-L39**	**Analysis**

Current trends in microprocessor architectures	Issues in applications (optimizing the hardware – software interface), ASIP, reconfigurable computing, asynchronous architectures, future architectures	• Appreciate current trends in VLSI architecture	**L40-L42**	**Comprehension**

9. Course web page : Moodle on iris

10. Reference Books

a) David Patterson and John Hennessy, Computer Architecture - A Quantitative Approach, Elsevier, 6th Edition, 2019 b) David Patterson and John Hennessy, Computer Organization and Design, The hardware software interface, Elsevier, 6th edition, 2021	c) NPTEL/MOOC courses on Computer Architecture d) Recent processor architecture literature

EVALUATION PLAN :

Mid semester exam - 20% Quiz, Lab assessment, Project - 40% End semester exam - 40%

Prepared by: Approved by

Dr. Sumam David S.
Course Instructor

Prof. N. Shekar V. Shet
Head, Dept of E&C and DPGC Chairperson