EE5607: ADVANCED COMPUTER ARCHITECTURE

Effective Term
Semester A 2026/27

Part I Course Overview

Course Title
Advanced Computer Architecture

Subject Code
EE - Electrical Engineering
Course Number
5607

Academic Unit
Electrical Engineering (EE)

College/School
College of Engineering (EG)

Course Duration
One Semester

Credit Units
3

Level
P5, P6 - Postgraduate Degree

Medium of Instruction
English

Medium of Assessment
English

Prerequisites
Nil

Precursors
EE2004 Microcomputer Systems or knowledge in microprocessor and logic circuits

Equivalent Courses
Nil

Exclusive Courses
Nil

Part II Course Details

Abstract

This course provides a comprehensive exploration of advanced computer architecture, with a focus on contemporary hardware and software challenges. Students will gain insights into the design principles and performance evaluation of modern computing systems. Key topics include processor and memory hierarchies, advanced pipelining, and cache mechanisms, emphasizing scalability and efficiency. The curriculum includes comparative analyses of high-performance vector processors, multi-computers, and massively parallel processors. Additionally, the course covers innovative architectures such as Very Long Instruction Word (VLIW) systems, fault-tolerant designs, and data flow machines. Emerging topics such as RISC-V architecture and Tensor Processing Units (TPUs) are also included, providing students with a broad understanding of the latest advancements and their applications in the field of computer architecture.

Course Intended Learning Outcomes (CILOs)

	CILOs	Weighting (if app.)	DEC-A1	DEC-A2	DEC-A3
1	Analyse the abstraction of various computer architectures involves understanding how complex computer systems are structured.		x	x	
2	Analyse various multi-processor architectures & connection mechanism		x	x	
3	Describe various memory technologies in multi-processor architectures		x	x	

A1: Attitude
Develop an attitude of discovery/innovation/creativity, as demonstrated by students possessing a strong sense of curiosity, asking questions actively, challenging assumptions or engaging in inquiry together with teachers.

A2: Ability
Develop the ability/skill needed to discover/innovate/create, as demonstrated by students possessing critical thinking skills to assess ideas, acquiring research skills, synthesizing knowledge across disciplines or applying academic knowledge to real-life problems.

A3: Accomplishments
Demonstrate accomplishment of discovery/innovation/creativity through producing /constructing creative works/new artefacts, effective solutions to real-life problems or new processes.

Learning and Teaching Activities (LTAs)

	LTAs	Brief Description	CILO No.	Hours/week (if applicable)
1	Lecture	Student will gain key concepts in the modern computer architecture. Afterwards, students can explain how some problems are solved and the techniques used.	1, 2, 3	
2	Test/quiz	Student will participate in practice problems in computer architecture.	1, 2, 3	
3	Assignments/Project	Student will participate in practice problems in computer architecture.	1, 2, 3	

Assessment Tasks / Activities (ATs)

	ATs	CILO No.	Weighting (%)	Remarks ("-" for nil entry)	Allow Use of GenAI?
1	Test/quiz	1, 2, 3	30	-	No
2	Assignments/ Project	1, 2, 3	30	-	Yes

Continuous Assessment (%)
60

Examination (%)
40

Examination Duration (Hours)
2

Minimum Continuous Assessment Passing Requirement (%)
30

Minimum Examination Passing Requirement (%)
30

Additional Information for ATs
To pass the course, students are required to achieve at least 30% in course work and 30% in the examination.

Assessment Rubrics (AR)

Assessment Task
Examination (for students admitted before Semester A 2022/23 and in Semester A 2024/25 & thereafter)

Criterion
Achievements in CILOs

Excellent
(A+, A, A-) High

Good
(B+, B, B-) Significant

Fair
(C+, C, C-) Moderate

Marginal
(D) Basic

Failure
(F) Not even reaching marginal level

Assessment Task
Coursework (for students admitted before Semester A 2022/23 and in Semester A 2024/25 & thereafter)

Criterion
Achievements in CILOs

Excellent

(A+, A, A-) High

Good
(B+, B, B-) Significant

Fair
(C+, C, C-) Moderate

Marginal
(D) Basic

Failure
(F) Not even reaching marginal level

Assessment Task
Examination (for students admitted from Semester A 2022/23 to Summer Term 2024)

Criterion
Achievements in CILOs

Excellent
(A+, A, A-) High

Good
(B+, B) Medium

Marginal
(B-, C+, C) Low

Failure
(F) Not even reaching marginal level

Assessment Task
Coursework (for students admitted from Semester A 2022/23 to Summer Term 2024)

Criterion
Achievements in CILOs

Excellent
(A+, A, A-) High

Good
(B+, B) Medium

Marginal
(B-, C+, C) Low

Failure
(F) Not even reaching marginal level

Additional Information for AR

Constructive Alignment with Programme Outcomes

PILO 2: With the obtained knowledge, students are able to evaluate and analyze the performance of new architecture technologies

PILO 3: Students are able to use obtained knowledge to deliver the solution of a specified requirement.

PILO 4: With the teaching activities, students are able to evaluate and formulate solutions to computer architecture problems and design.

Part III Other Information

Keyword Syllabus

Overview of Microprocessor and Microcontroller Microarchitectures

- Microprocessor microarchitecture: CPU core (datapath, ALU, control unit), hardwired vs microprogrammed control units, memory types, I/O buses (AMBA, Wishbone), Von Neumann vs Harvard architectures, general-purpose computing, and clock gating for power efficiency.
- Microcontroller microarchitecture: SoC integration (CPU + SRAM + timers/ADCs), interfaces including SPI, I2C, UART, and CAN, embedded systems, embedded RTOS support, low-power modes (sleep/idle), simple ISA subsets, and real-time constraints.

RISC and CISC Architectures

- RISC Architecture Principles: Instruction Set Architecture (ISA), reduced instruction set computing, fixed-length instructions, load/store architecture, and extensive register files.
- ARM processor architecture overview: Fixed 32-bit instructions, trust zone security, AMBA interconnects, Cortex-M/R/A profiles, low-power design features, AArch32 versus AArch64, thumb-2, NEON, SVE/SVE2, and big LITTLE heterogeneous multicore.
- CISC Architectures: Complex instruction set with variable-length encoding, multi-cycle operations, direct memory access operands, microcode control units for instruction emulation, and CISC decoding hardware.

RISC-V Open Instruction Set Architecture (ISA) Design

- Modularity: Base integer ISAs (RV32I, RV64I) and standard extensions (M, A, F, D, C).
- Privileged architecture specification: Machine, supervisor, and user-level modes.
- Vector processing with the RISC-V 'V' extension.
- Designing custom ISA extensions for application acceleration

RISC-V Processor Design and Implementation

- Quantitative Principles of Computer Design: Performance, Power, and Cost Analysis
- Datapath and Control Unit Implementation: Single-cycle and multi-cycle.
- Hardware implementation on FPGAs for prototyping and performance tuning.

Instruction Pipelining and Performance Engineering

- Pipeline Architecture: Stages including instruction fetch, decode, execute, memory access, write-back; overlapping execution for throughput enhancement, clock cycle latency analysis.
- Hazard Mitigation: Data hazards (RAW/WAR/WAW), control hazards (branches/jumps), structural hazards (resource contention); pipeline stalls, forwarding techniques, branch prediction algorithms (static/dynamic), performance impact on instruction-level parallelism.
- Superscalar and out-of-order execution, register renaming, reorder buffer, speculative execution, and precise exception handling

Parallelism and Multiprocessing

- Flynn's Taxonomy: SIMD (single instruction multiple data), MIMD (multiple instruction multiple data), SPMD (single program multiple data); data-level vs. task-level parallelism in engineering applications.
- Multiprocessing Systems: Symmetric multiprocessing (SMP), asymmetric multiprocessing (AMP), multithreading models (fine-grained/coarse-grained), concurrency primitives (locks, semaphores), synchronization challenges (deadlocks, race conditions), scalability metrics (Amdahl's Law), performance/latency trade-offs in multi-core RISC-V clusters.

Multiprocessor Architectures

- Distributed Memory Systems: Private L1/L2 caches per core, message-passing interfaces (MPI), interconnect topologies (e.g., NoC - Network-on-Chip), scalability in large-scale multiprocessing.
- Array and Vector Processors: Systolic arrays with simple processing elements, SIMD data parallelism; vector registers for array operations, high-throughput engineering in signal processing.
- GPU and Heterogeneous Architectures: Massive core parallelism (thousands of ALUs), SIMD/MIMD hybrid models, unified memory for graphics/compute workloads; integration with RISC-V for hybrid CPU-GPU systems
- Multi-processor FPGAs and Tensor Processing: System on Chip (SoC), PSoC FPGAs, Versal FPGAs, and TPUs

Memory Technology in Various Architectures

- Cache Engineering: On-chip SRAM-based caches for latency reduction, multi-level hierarchy (L1/L2/L3), associativity impacts on hit rates; cache coherence protocols (MESI/MOESI) in multiprocessing.
- Cache Mapping and Policies: Direct-mapped, fully associative, set-associative organizations; write-through vs. write-back strategies, victim cache for pollution reduction, performance/latency profiling.
- Virtual Memory Systems: Hardware address translation (TLB - Translation Lookaside Buffer), paging mechanisms (fixed/variable page sizes), segmentation for protection domains; demand paging, page faults, thrashing mitigation.
- Page Replacement Algorithms: Least Recently Used (LRU), First-In-First-Out (FIFO), Optimal (Belady's); working set model, reference bit tracking for efficient replacement in engineering workloads.

I/O Subsystems and Interfacing

- I/O Engineering Fundamentals: Device controllers, bus standards (PCIe, AXI), polling vs. interrupt-driven I/O, programmed I/O for simple peripherals.
- Direct Memory Access (DMA): Hardware DMA engines for high-bandwidth transfers, scatter-gather operations, integration with RISC-V I/O extensions; latency optimization in multiprocessing environments.
- Interrupt Handling: Vectored interrupts, priority levels, nested interrupt support; performance implications on system responsiveness and real-time constraints.

Reading List

Compulsory Readings

	Title
1	John L. Hennessy and David A. Patterson, Computer Architecture: A Quantitative Approach, Morgan Kaufmann, 6th edition, 2017.
2	Beuchat, René, et al. Fundamentals of system-on-chip design on arm cortex-M microcontrollers. Arm Education Media, 2021.
3	Sterling, Thomas, Maciej Brodowicz, and Matthew Anderson. High performance computing: modern systems and practices. Morgan Kaufmann, 2017.

Additional Readings

	Title
1	Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer Organization, McGraw Hill, 5th Ed, 2014.
2	Kai Hwang, Advanced Computer Architecture: Parallelism, Scalability, Programmability, McGraw-Hill, 3rd Ed, 2015
3	Vanderbauwhede, Wim, and Khaled Benkrid, eds. High-performance computing using FPGAs. Vol. 3. New York: Springer, 2013.