| | | | | | |
|---|---|---|---|---|---|
| Course Title | Computer Architecture & Hardware | | | | |
| Course Code | CSC620 | | | | |
| Course Type | Compulsory | | | | |
| Level | Master (2nd Cycle) | | | | |
| Year / Semester | 1st year/1st semester | | | | |
| Teacher’s Name | TBA | | | | |
| ECTS | 10 | Lectures / week | 3 hours / 14 weeks | Laboratories / week | N/A |
| Course Purpose and Objectives | The objective of this course is to provide a comprehensive introduction to the fundamental architectural and organizational concepts of computer systems, and to equip students with the necessary skills which will allow them to understand the major trade-offs during the design of a modern computer system. | | | | |
| Learning Outcomes | Upon successful completion of this course, students should be able to:<br>• Identify a digital system and its main characteristics, and differentiate between digital and analogue systems<br>• Describe the concepts of binary numbers and binary encoding and perform basic mathematical operations using binary numbers<br>• Describe and utilize theorems and axioms of Boolean Algebra for the analysis, design and simulation of combinational and sequential logic circuits<br>• Explain the concept of memory in digital systems, and explain how computer memory is organized hierarchically in a modern computer system<br>• Explain and dicuss how hardware parts interact in a typical CPU in order to execute a computer instruction<br>• Describe the Von Neumann architecture and explain the operation of the fetch-decode-execute cycle<br>• Describe and compare the major characteristics of Instruction Set Architectures (ISA)<br>• Describe and explain the operation of the basic types of I/O architectures used in modern computer systems<br>• Classify instruction set architectures and describe their main characteristics.<br>• Describe the different CPU performance metrics<br>• Classify the different types of pipelining.<br>• Analyze the effect of different design methodologies on CPU performance | | | | |
| Prerequisites | None | Co-requisites | None | | |

| | |
|---|---|
| Course Content | Overview and History: History, overview and applications of digital systems, digital signals and analogue signals, basic concept of a digital system and its main characteristics, definition of a computer system.<br>Number Systems & Digital Arithmetic: The binary number system, binary counting, the hexadecimal number system, integer and fixed-point conversions between binary, hexadecimal and decimal numbers, utilization of binary quantities, binary encoding schemes, encoding and decoding using BCD, Grey, ASCII and Unicode binary codes. Signed and unsigned binary numbers, representing signed binary numbers using the 2's complement method, performing basic mathematical operations using binary numbers (addition, subtraction, multiplication, division)<br>Boolean Algebra: Axiomatic definition of Boolean algebra, the AND, OR, NOT logic operations, representation of Boolean operations using logic gates. Derived Boolean algebra operations (NAND, NOR, XOR, XNOR), description and effective utilization of Boolean algebra theorems, DeMorgan's Theorem.<br>Combinational Logic Circuits: Mathematical modelling of logic circuits using Boolean functions and truth tables, implementation of logic circuit from given Boolean function expression (logic circuit design), derivation of Boolean function from given logic circuit (logic circuit analysis), derivation of truth table from given logic circuit (simulation). Description of standardised combinational logic circuits such as decoders / encoders, multiplexers / demultiplexers, adders, and digital displays. Use of standardised logic circuits as building blocks for modular design purposes.<br>Sequential Logic Circuits: Description and operation of basic flip-flops (D, JK, T). Serial registers, parallel registers, shift registers, synchronous and asynchronous counters.<br>Memory Circuits: Introduction to digital memory terminology, basic operation of a memory module, RAM memory and types (SRAM and DRAM), ROM memory and types.<br>The Central Processing Unit (CPU): main parts of a CPU (Arithmetic Logic Unit, Control Unit, Registers), System Bus, Clock, Main Memory, description and explanation of the Von Neumann architecture as the basis for the implementation of computer systems, limitations of the Von Neumann architecture, the fetch-decode-execute cycle, the Von Neumann bottleneck, representation of a computer as a hierarchical set of computing levels, recent developments in the implementation of computer systems (including Harvard architectures, parallel computers, multicore architectures, embedded systems).<br>Computer Organization: Arithmetic Logic Unit implementation, Control Unit implementation, hardware implementation of the Control Unit's decoding circuit, software implementation of the Control Unit's decoding circuit, trade-offs between hardware and microprogrammed implementation of the Control Unit's decoding circuit.<br>Memory Organization: Hierarchical organisation of computer memory, access time, cost, and capacity trade-offs, cache memory, virtual memory, secondary storage systems (including magnetic, optical, and solid-state systems).<br>Input / Output (I/O): Definition and design considerations of the I/O subsystem in a modern computer system, I/O control methods (including |

| | |
|---|---|
| | programmed I/O, interrupt-driven I/O, Direct Memory Access (DMA) and channel-attached I/O), basic interrupt service routines.<br>Instruction Set Architectures (ISAs): Basic characteristics and functions of ISAs, design parameters of instructions (including instruction length, variable vs fixed-length instructions, addressing modes, number of operands, and endianness), internal storage (including accumulator, stack, and register-based architectures), types of instructions, Complex Instruction Set Architectures (CISC), Reduced Instruction Set Architectures (RISC), modern implementations of CISC and RISC architectures.<br>Pipelining: Parallel processing, pipelining, arithmetic pipeline, instruction pipeline, RISC pipeline, pipeline hazards, pipeline implementation.<br>Instruction Level Parallelism (ILP): introduction to instruction-level parallelism, dynamic scheduling, dynamic hardware prediction, high performance instruction delivery, ILP limitations, ILP with software approaches, compiler techniques for exposing ILP, Hardware VS software speculation mechanisms.<br>Multiprocessors: Symmetric shared-memory architectures, distributed shared-memory architectures, performance issues, synchronization, multithreading |
| Teaching Methodology | Face-to-Face |
| Bibliography | “The Essentials of Computer Organization and Architecture”, by Linda Null and Julia Lobur<br>“Computer Organization and Architecture” by William Stallings<br>“Structured Computer Organization” by Andrew Tanenbaum<br>“Computer Architecture: A Quantitative Approach” by John L. Hennessy<br>“Digital Design and Computer Architecture” by David Harris and Sarah Harris |
| Assessment | Mid – Term Examination 30%<br>Final Examination 40%<br>Assignments/Lab 20%<br>Class Participation and Attendance 10%<br>100% |
| Language | English |