COURSE OUTLINE

(1) GENERAL

SCHOOL	ENGINEERING		
ACADEMIC UNIT	ELECTRICAL AND COMPUTER ENGINEERING DEPT.		
LEVEL OF STUDIES	Undergraduate		
COURSE CODE	**ECE_K410**	**SEMESTER**	**4**
COURSE TITLE	COMPUTER ARCHITECTURE		

INDEPENDENT TEACHING ACTIVITIES *if credits are awarded for separate components of the course, e.g. lectures, laboratory exercises, etc. If the credits are awarded for the whole of the course, give the weekly teaching hours and the total credits*	**WEEKLY TEACHING HOURS**	**CREDITS**
Lectures	3	
Seminars / Practice exercises	1	
Laboratory	0	
Add rows if necessary. The organisation of teaching and the teaching methods used are described in detail at (4).	4	5

COURSE TYPE *general background, special background, specialised, general knowledge, skills development*	Special background
PREREQUISITE COURSES:	No. It is suggested that students have already attended the course: ECE_K140 Digital Logic Design
LANGUAGE OF INSTRUCTION and EXAMINATIONS:	Greek
IS THE COURSE OFFERED TO ERASMUS STUDENTS	No
COURSE WEBSITE (URL)	https://www.ece.uop.gr/

(2) LEARNING OUTCOMES

Learning outcomes

The course learning outcomes, specific knowledge, skills and competences of an appropriate level, which the students will acquire with the successful completion of the course are described.

Consult Appendix A

- *Description of the level of learning outcomes for each qualifications cycle, according to the Qualifications Framework of the European Higher Education Area*
- *Descriptors for Levels 6, 7 & 8 of the European Qualifications Framework for Lifelong Learning*

and Appendix B

- *Guidelines for writing Learning Outcomes*

This course aims to the in-depth understanding of computer architecture and organization of RISC processors as well as their operation and the interconnection/communication with the memory and external Input/Output units. Emphasis is given to the interaction between software and hardware, the instruction set architecture, the pipelined design and the programming in assembly language.

At the end of this course the students should:

<u>At the knowledge level:</u>

1. Have understood the interaction between software and hardware and the control of hardware functional units through software.
2. Have understood the procedure of compilation of high-level programs to machine language.
3. Know the hardware and software parameters that affect the performance.

4. Have understood the techniques applied by the hardware and software designers for the improvement of the performance.
5. Have understood the architecture of a MIPS central processing unit, the data path, the pipeline and the hazards (structural, data and control).
6. Know the addressing modes, the memory hierarchy, the operation of cache memory (mapping functions) and the operation of virtual memory.
7. Know the instruction set of MIPS processors.
8. Have understood the integer and float-point arithmetic.
9. Know the algorithms for performing operations in fixed- and floating-point arithmetic (addition, subtraction, multiplication, division) and their hardware implementations.

At the skill level - abilities:

1. Calculate the performance of a processor based on both hardware and software parameters.
2. Develop programs in MIPS assembly language.
3. Detect hazards (structural, data, control) in a program executing in a pipelined processor and handle them both in hardware and software.
4. Evaluate and improve the performance of cache memories.

General Competences

Taking into consideration the general competences that the degree-holder must acquire (as these appear in the Diploma Supplement and appear below), at which of the following does the course aim?

Search for, analysis and synthesis of data and information, with the use of the necessary technology	*Project planning and management*
Adapting to new situations	*Respect for difference and multiculturalism*
Decision-making	*Respect for the natural environment*
Working independently	*Showing social, professional and ethical responsibility and sensitivity to gender issues*
Team work	*Criticism and self-criticism*
Working in an international environment	*Production of free, creative and inductive thinking*
Working in an interdisciplinary environment	*……*
Production of new research ideas	*Others…*
	…….

- Search, analysis and synthesis of data and information, with the use of the necessary technology.
- Working independently.
- Teamwork (at the laboratory).
- Working in an international environment.
- Production of free, creative and inductive thinking.

(3) SYLLABUS

The course consists of the following sections:

1. **Basic principles:** History of computer systems, hardware and software, arithmetic systems, structure, organization and operation of computers, central processing unit, memory system, random access memory, memory address decoding, types of machine-language instructions, addressing modes of main memory, instruction set architecture, complex- and reduced instruction set computers, control unit, arithmetic/logic unit, timing.
2. **Performance:** Performance metrics, performance evaluation, uniprocessors and multiprocessors.
3. **The language of the computer**: Operations of the hardware, instruction set of MIPS (arithmetic, logical, control, flow control), procedures, addressing modes, compiling and program execution.
4. **Arithmetic for computers**: Algorithms for addition, subtraction, multiplication and division for integers and floating-point numbers.
5. **Central Processing Unit (CPU):** Design of central processing unit, datapath, control, memory units and their organization, pipeline, hazards (structural, data and control) and their handling, design of a pipelined CPU.
6. **Memory:** Memory types, memory hierarchy, cache memory (organization, operation and implementation), virtual memory.

(4) TEACHING and LEARNING METHODS - EVALUATION

<table>
<tr><td>DELIVERY
Face-to-face, Distance learning, etc.</td><td>• Face-to-face in the class.
• Distance learning through the e-Class platform.</td></tr>
<tr><td>USE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY
Use of ICT in teaching, laboratory education, communication with students</td><td>• PowerPoint presentations of the course lectures, which have been uploaded to the e-Class page of the course.
• Support of educational procedure through the e-Class platform for discussion and solving of relevant problems.
• QtSpim simulator: Simulator for editing and running programs in MIPS assembly.</td></tr>
<tr><td>TEACHING METHODS

The manner and methods of teaching are described in detail.

Lectures, seminars, laboratory practice, fieldwork, study and analysis of bibliography, tutorials, placements, clinical practice, art workshop, interactive teaching, educational visits, project, essay writing, artistic creativity, etc.

The student's study hours for each learning activity are given as well as the hours of non-directed study according to the principles of the ECTS</td><td><table><tr><th>Activity</th><th>Semester workload</th></tr><tr><td>Lectures</td><td>39</td></tr><tr><td>Exercises</td><td>13</td></tr><tr><td>Personal study</td><td>73</td></tr><tr><td>Course Total</td><td>125 hours (5 ECTS)</td></tr></table></td></tr>
<tr><td>STUDENT PERFORMANCE EVALUATION

Description of the evaluation procedure

Language of evaluation, methods of evaluation, summative or conclusive, multiple choice questionnaires, short-answer questions, open-ended questions, problem solving, written work, essay/report, oral examination, public presentation, laboratory work, clinical examination of patient, art interpretation, other

Specifically-defined evaluation criteria are given, and if and where they are accessible to students.</td><td>The student is evaluated by a final written examination at the end of the semester, in which the student must answer questions and solve exercises based on the course’s content. The grade ranges from 0 to 10.

The evaluation is performed in Greek language.</td></tr>
</table>

(5) ATTACHED BIBLIOGRAPHY

- *Suggested bibliography:*

1. D. A. Patterson, J. L. Hennessy, Οργάνωση και σχεδίαση υπολογιστών: Η διασύνδεση υλικού και λογισμικού, 4η έκδοση, εκδόσεις Κλειδάριθμος, 2010.
2. C. Hammacher, Z. Vranesic, S. Zaky, Οργάνωση και αρχιτεκτονική ηλεκτρονικών υπολογιστών, εκδόσεις Επίκεντρο, 2007.
3. W. Stallings, Computer Organization and Architecture, 9th edition, 2012.
4. A. S. Tanenbaum, Η αρχιτεκτονική των υπολογιστών: Μια δομημένη προσέγγιση, 4η έκδοση, εκδόσεις Κλειδάριθμος, 2009.
5. Δ. Νικολού, Αρχιτεκτονική υπολογιστών, εκδόσεις Γκιούρδα, 2009.
6. Κ. Πεκμεστζή, Συστήματα μικροϋπολογιστών Ι: Μικροεπεξεργαστές 80x86, Pentium και ARM, Εκδόσεις Συμμετρία, 2009.