COURSE OUTLINE

(1) GENERAL

SCHOOL	ENGINEERING		
ACADEMIC UNIT	ELECTRICAL AND COMPUTER ENGINEERING DEPT.		
LEVEL OF STUDIES	Undergraduate		
COURSE CODE	**ECE_ELE810**	**SEMESTER**	**8**
COURSE TITLE	EMBEDDED SYSTEMS I		

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		
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*	Specialised
PREREQUISITE COURSES:	No. Students are advised to have already attended the course: COMPUTER ARCHITECTURE
LANGUAGE OF INSTRUCTION and EXAMINATIONS:	Greek
IS THE COURSE OFFERED TO ERASMUS STUDENTS	Yes
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*

The main goal of the course is to introduce the students to the basic concepts of embedded systems and the practical issues of their design. In this context, the main structural components of modern embedded systems and their corresponding architectures will be presented and analyzed thoroughly. In parallel, the students will be trained in using and programming development boards through which they will be able to focus on practical aspects of embedded system design.

Keywords: Embedded systems, embedded system design, embedded system architecture

The knowledge and the skills acquired will form the basis for the courses of next semesters like the course EMBEDDED SYSTEMS II, for which the current course is prerequisite.

Learning Outcomes

After the successful completion of the course, the students will be able to:

At the knowledge level:

1. To describe the basic structural components of an embedded system
2. To understand the architecture and the way an embedded system works
3. To know the interfacing capabilities between the constituent components of an embedded system
4. To be able to analyze the performance of an embedded system

At the skills level:

1. Familiarize with the development boards used for embedded system design
2. Be trained in programming modern embedded systems using state-of-the art development boards
3. Be trained in using specialized software for designing embedded systems

At the level of abilities:

1. Select the proper components for designing an embedded system considering the functional and non-functional requirements
2. To solve practical problems that appear during embedded system design
3. To evaluate the properties, the capabilities and the features of an embedded system
4. To program embedded systems with specific requirements and design goals

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 for, analysis and synthesis of data and information, with the use of the necessary technology
- Working independently
- Team Work
- Working in an international environment
- Production of free, creative and inductive thinking
- Production of new research ideas

(3) SYLLABUS

Why we embedded microprocessors in systems? Which are the difficulties in embedding microprocessors? Description of an embedded system using UML. Review of architecture classification and assembly language. Input/output mechanisms. Supervisor mode, exceptions and traps. Memory management and address translation. Cache memories. How the system architecture influences the performance of a program? How the system architecture influences the power consumption of a program? CPU buses. Interfacing I/O devices. The system of CPU as a framework for understanding design methodologies. Development environments and debugging.

The course lectures cover the following thematic areas:

1. **Complex systems and microprocessors** Embedding computers. Features of embedded computing. Why do we use microprocessors? Challenges embedded system design.
2. **The process of designing embedded systems** Requirements. Specification. Architecture design. Design of hardware and software components. System integration. Formalisms for embedded system design. Design example.
3. **Processors for embedded systems** Taxonomy of architectures. ARM processor: Memory organization, data functions, flow control.
4. **Programming input/output Input/output devices** Basic elements of input/output. Busy wait input/output.
5. **Interrupt handling Interrupts** Priorities and vectors. Interrupts in ARM. Supervisor mode, exceptions and traps.
6. **Cache memory mechanisms** Cache memories. Memory management units and address translation.
7. **CPU performance Pipeline** Superscalar execution. Using cache memories. CPU power consumption. Examples of embedded system performance.
8. **CPU buses** Bus protocols. Direct memory access. Bus configuration. ARM buses.
9. **Memory devices** Organization of memory devices. Random Access Memories (RAM). Read Only Memories (ROM).
10. **Input/output devices – Device interconnection** Timers and counters. A/D and D/A converters. Keyboards. LEDs. Displays. Touch screens. Device interconnection: Memory interconnections, device interconnection.
11. **Designing with microprocessors** System architecture. Hardware design. PC as and embedded platform. Development and debugging. Development environments. Debugging techniques. Challenges in debugging embedded systems. Design testing. Examples of embedded system design using microprocessors.
12. **Methodologies for embedded system design** Why do we need design methodologies? Design flows. Specifications: Specification languages, advanced specifications. System analysis and architecture design.
13. **Systems on Chip Systems on chip/circuit** Design challenges. Analysis tools. Synthesis tools. Co-simulation tools.

(4) TEACHING and LEARNING METHODS - EVALUATION

DELIVERY *Face-to-face, Distance learning, etc.*	Face-to-face in-class lecturing
USE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY *Use of ICT in teaching, laboratory education, communication with students*	• Slides (ppt) of the presentation of the theoretical part of the course, which will be available from the beginning of semester through e-Class. • Guidelines for the exercises (one per exercise), which will be available from the beginning of the semester through e-Class. • Suggested solutions for each exercise will be provided following the completion of each exercise. • Support of teaching procedure through the e-Class platform (notification of the teaching procedure, distribution of slides, supplementary material, announcements, relative links and literature, provision of test and the final examination)

<table>
<tr><td></td><td>

- Specialized software relevant to the course.

</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>

Activity	***Semester workload***
Lectures	39
Practical examples and exercises – focusing on problem solving	13
Study of lectures and bibliography	53
Project implementation	20
Course Total	***125 hours (5 ECTS)***

</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>

A. Evaluation of theoretical part:

Final written exam that includes:

1. Solving exercises
2. Multiple choice questions
3. Comparative evaluation of theory elements

B. Evaluation of exercises/projects

Written exams take place throughout the semester and include:

1. Solving exercises
2. Multiple choice questions

Comments:

- The final grade is the weighted result of the grades of theory and assignments. The weights will be defined and the beginning of each semester and they will be announced via e-Class.
- The final exams are in Greek language
- The examination process and the evaluation criteria are publicly available to the students through e-Class.

</td></tr>
</table>

(5) ATTACHED BIBLIOGRAPHY

- *Suggested bibliography:*

1. Wayne Wolf, «High Performance Embedded Computing", 2nd edition, Morgan Kaufman, 2014
2. Κωνσταντίνος Καλοβρέκτης, «Βασικές Δομές Ενσωματωμένων Συστημάτων», Εκδόσεις Βαρβαρήγου, 2012
3. Peter Marwedel, «Embedded System Design», Springer, 2011
4. Wayne Wolf, «Οι Υπολογιστές ως Συστατικά Στοιχεία», Εκδόσεις Ελληνικών Τεχνολογιών, 2008

- *Related academic journals:*

1. ACM Transactions on Embedded Computing Systems
2. ACM Transactions on Design Automation of Electronic Systems