COURSE OUTLINE

(1) GENERAL

SCHOOL	ENGINEERING		
ACADEMIC UNIT	ELECTRICAL AND COMPUTER ENGINEERING DEPT.		
LEVEL OF STUDIES	Undergraduate		
COURSE CODE	**ECE_ELE920**	**SEMESTER**	**9**
COURSE TITLE	FPGA-based Systems Design		

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	2	
Seminars / Practice exercises	1	
Laboratory	1	
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: • Hardware Description Languages (ECE_ELE830)
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 aim of the course is to introduce students to the Basic Concepts of FPGAs. FPGA design and design methodologies are analyzed. The basics of VLSI technologies are briefly presented. It also analyzes FPGA fabrics, mainly FPGAs Architecture such as SRAM based FPGAs and anti-fuses based FPGAs. FPGA inputs and outputs, placing and routing, embedded circuits in FPGAs. The physical design of FPGAs is analyzed. Describes how to design combinatorial circuits as well as finite state machines. Analyzing FPGA programming with structural logic and behavioral description. It also described the Tools and the design flow for FPGAs while analyzing power consumption issues and clock distribution rules-networks. In addition, ways for simulation, test, and verification of the designs in FPGAs are described.

Learning outcomes

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

- Knows the basic FPGA technologies
- Designs a digital system for implementation in FPGA
- Implements a digital system in FPGA
- Programming in an FPGA
- Describes the internal structure of FPGAs
- Measure the performance of an FPGA
- Optimizes a design and an implementation in an FPGA in terms of power consumption and performance
- Designs with various types of FPGAs
- To design circuits based on system-on-chip

Keywords: FPGA Integrated systems, Basics on VLSI, SRAM -based FPGAs and Anti-fuse based FPGAs, Workflow for FPGAs Programming

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
- Adapting to new situations
- Working independently
- Team Work
- Working in an international environment
- Production of new research ideas.
- Project planning and management
- Criticism and self-criticism
- Production of free, creative and inductive thinking

(3) SYLLABUS

Lectures:

1st: Introduction: Basic Concepts of FPGAs
• Design issues
• FPGA and ASIC differences

2nd: VLSI technology revisited
• Basics of VLSI
• CMOS logic gates
• Quality metrics
• Semiconductor Element
• The CMOS inverter
• Static complementary CMOS logic
• Power consumption of CMOS circuits

3rd: FPGA Design and Fabrics Methodologies: FPGAs Architecture
• FPGA Architecture
• Inputs / Outputs (I/Os)
• Structural circuits of FPGAs
• FPGA design parameters

4th: SRAM based FPGAs and anti-fuse based FPGAs
• FPGA based on SRAM memories
• Permanently scheduled FPGAs

5th: Placing – routing and embedded processing units
• Types of routing
• Logic elements
• Placing logical elements
• Routing logical elements
• Basic processing units

6th: Combinatorial logic for FPGAs
• Delay of gates and cables
• Arithmetic logic
• Logic synthesis for FPGAs

7th: Sequential Machines
• Sequential machine design process
• Types of sequential machines
• Timing rules
• Performance analysis

8th: Tools and design flow for FPGAs
• Tools for various FPGAs
• Design workflow for FPGAs

9th: Power consumption issues. Clock distribution rules
• Power consumption management in FPGAs
• Methods and techniques to reduce power consumption
• Cables for clock distribution

10th: Simulation testing and verification of FPGAs
• Simulation of HDL code
• Simulation during FPGA synthesis
• Simulation during FPGA implementation
• FPGA operation verification

11th: Introduction to e-FPGAs
• Embedded FPGAs
• Uses
• Applications
• Implementations
12th: Multi-processors in FPGAs: design and implementation
• Types of multi FPGA systems
• Multi FPGA interface networks
• Multi-FPGA segmentation

13th: Embedded microprocessor architecture
• Examples of embedded processors (Picoblaze, Microblaze, NIOS)
• Implementations with embedded processors
• Integrated processor programming

Laboratories The laboratory part of the course includes practical exercises that aim at the application and consolidation of the knowledge of the theory and concern: 1st: Introduction and Familiarity with the FPGA BASYS3 board and Xilinx software tools. Implementation of a FPGA System by determining inputs and timing 2nd: Validation of Operation of a Circuit Using Test Benches and Do Files. Power Consumption Assessment in FPGA 3rd: Programming a Counter Circuit in FPGA and post-Map, post Place and Route simulation 4th: Complex Time and Space Limitations and how to program circuits in FPGA 5th: Embedded processors (Softcore) for programming in FPGA 6th: Introduction to the Xilinx EDK tool environment - Design and configuration of the Microblaze processor softcore - Microblaze programming. Communication and data exchange program with PC Attendance of laboratory exercises is mandatory.

(4) TEACHING and LEARNING METHODS - EVALUATION

DELIVERY *Face-to-face, Distance learning, etc.*	Face to face in class and in the laboratory. Distance learning support via e-Class system
USE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY *Use of ICT in teaching, laboratory education, communication with students*	• Slides (ppt) for teaching the theoretical part, which have been posted since the beginning of the semester in the e-Class. • Laboratory guides for the laboratory part (one for each laboratory exercise), which have been posted on the e-Class since the beginning of the semester. • Support of learning process through the e-Class platform (for notification of the course operating regulations, for distribution of slides, supplementary material, announcements, links and bibliography, for the conduct of the intermediate and final examination of the laboratory part, etc.). • Specialized software e.g VIVADO for the laboratory part freely available to each student.
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*	(see table below)

Activity	*Semester workload*
Lectures	26 (=13×2)
Seminars	13 (=13×1)
Laboratory Exercises (in Lab)	12 (=6x2)
Preparing Laboratory Exercises	26hours
Lecture & bibliography study (at home)	48 hours
Course Total	***125 hours (5 ECTS)***

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.*	A. Assessment of Theoretical Part: • Intermediate exam (30%), which includes solving exercises and multiple-choice questions with graded difficulty. • Written final exam, that includes solving exercises, multiple choice questions and comparative evaluation of theory elements, graded difficulty. B. Evaluation of Laboratory Part: • Oral examination during laboratory exercises (40%) • Written final exam (60%) which includes solving exercises. Remarks: • The final grade results from the weighting of the theory and laboratory grades with weights of 60% and 40%, respectively. • The evaluation is done in the Greek language • The evaluation process and evaluation criteria are published on the course's website in the e-Class.

(5) ATTACHED BIBLIOGRAPHY

- Suggested bibliography:

In Greek

1.Wayne Wolf, "FPGA-based Systems Design", New Technologies Publishing, 2014.

2. Stavros Souravlas, Ioannis Roumeliotis, “Digital Systems”, Version 5th, Tziolas, 2008.

3. Steve Kilts “Advanced FPGA Design: Architecture, Implementation, and Optimization”, Wiley-IEEE Press; 1 edition (June 29, 2007).

4. Peter Wilson, “Design Recipes for FPGAs: Using Verilog and VHDL”, Newnes; 2 edition (October 9, 2015)