Course Title:	Digital Systems Design				
Course Code:	ACOE361				
Course Type	Compulsory				
Level	Bachelor (1st Cycle)				
Year / Semester	3rd Year/ 5th Semester				
Teacher's Name	Dr. Konstantinos Tatas				
ECTS	6	Lectures/week	2	Laboratories/week	2
Course Purpose	The purpose of this course is to introduce students to practical aspects of the Digital System design flow using Hardware Description Languages as well as High Level Synthesis. The verification, synthesis and optimization flow are explained in detail. The course also introduces advanced topics in digital design such as verification methods and asynchronous input synchronization methods and design for reuse guidelines.				
Learning outcomes	By the end of this course, students should be able to: 1. Use EDA tools for ASIC/VLSI design. 2. Describe ASIC, PLD and FPGA technologies. 3. Design hazard free synchronous Digital Systems using ASMs and reliably synchronise asynchronous inputs. 4. Use Hardware Description Languages as well as High-Level Synthesis to design complex digital circuits 5. Implement, test and verify digital systems using FPGA prototyping boards.				
Prerequisites:	ACOE161, ACSC182		Co-requisites:	None	
Course contents:	• **Hardware Description Languages:** Top-Down Design. File organization. Entity and Architecture. Structural and Behavioural Description. Concurrent and sequential statements. Procedures and functions. Packages and design for reuse. • **Verification:** Basic verification methodology. Testbench design: directed and constrained-random testing, self-checking testbenches • **ASIC architectures and Implementation Options:** Synthesis and EDA tools for ASIC and FPGA implementation. Semi-custom / full custom ASICs. Gate Array, Standard Cell, Full Custom. CMOS/BI-CMOS technologies. PLDs and FPGAs.				

	• **Digital Systems Design - ASMs:** ASMs, Mealy and Moore machines. ASM charts. VEM minimization. IFL/OFL minimization and implementation. State machine implementation using PROMs and multiplexers. PLDs. Finite state machine implementation using FPLAs. Timing. Glitch minimization techniques. Asynchronous input systems. Implementation selection trade-offs: performance, cost, production size etc. • **Introduction to High-Level Synthesis**: FSM and datapath, DFGs and CDFGs. Scheduling and resource sharing. Design space exploration. High-level synthesis of arrays and loops. Loop unrolling and pipelining • **Laboratory work:** Individual or small group experiments using Xilinx tools, FPGA implementation and testing.
Teaching Methodology	The course is structured in lectures that are conducted with the help of both computer presentations and traditional means. Practical examples and exercises are included in the lectures to enhance the material learning process. Often short post-lecture quizzes are used to assess the level of student understanding and provide feedback. Student questions are addressed during the lecture, or privately after the lecture or during office hours. Lectures include tutorials of EDA tools and design case studies. Lecture notes are available through the web for students to use in combination with the textbooks. Students are assessed continuously and their knowledge is checked through tests with their assessment weight, date and time being set at the beginning of the semester via the course outline. Furthermore, guided individual and group design assignments are used to develop practical engineering skills while integrating the course theory. Laboratory experiments are carried out in small groups and lab reports are required two weeks after the laboratory class resulting in a cumulative mark. The first laboratory exercises are totally structured (cookbook) in order to familiarize the students with the equipment, while later exercises are less structured, allowing the student to create their own designs or programs for a given application.
Bibliography	**Textbooks:** • John F. Wakerly, "**Digital Design: Principles and Practices** ", 5th edition, Prentice Hall, 2018 • Michael D. Ciletti, **"Advanced Digital Design with the Verilog HDL"**, 2nd Edition, Pearson, 2011 **References:** • V. Pedroni, **The student's guide to VHDL**, Morgan Kaufmann, 1998.
Assessment methods and	The final assessment of the students is formative and summative and is assured to comply with the subject's expected learning outcomes and the

criteria:	quality of the course. In order to continuously assess students, and given the design-based nature of the course, coursework weight is set at 60%, which comprises assignments, an individual or group project, a mid-term exam and laboratory work assessment. Assignments range from simple problems to work out, to circuit design assignments that require demonstrate concept understanding as well as problem-solving skills. The assessment weight, date and time of each type of continuous assessment is being set at the beginning of the semester via the course outline. An indicative weighted continuous assessment of the course is shown below: • Assignments 10% • Group project 10% • Tests: 20% • Laboratory Work: 20% • Final Exam 40%
Language	English