<table>
<tr><td>Course Title</td><td colspan="5">Embedded Systems</td></tr>
<tr><td>Course Code</td><td colspan="5">WSS531</td></tr>
<tr><td>Course Type</td><td colspan="5">Specialization (Elective)</td></tr>
<tr><td>Level</td><td colspan="5">Master (2nd Cycle)</td></tr>
<tr><td>Semester</td><td colspan="5">2 or 3</td></tr>
<tr><td>Teacher’s Name</td><td colspan="5">Konstantinos Tatas, PhD</td></tr>
<tr><td>ECTS</td><td>10</td><td>Lectures/week</td><td>3</td><td>Laboratories/week</td><td>0</td></tr>
<tr><td>Course Purpose</td><td colspan="5">Embedded computing systems are becoming more and more prevalent as the number computing devices that are not desktop computers or servers are increasing exponentially with now users and households possessing a large number of them. The ability of many of these devices to connect to the internet is bringing about the Internet-of-Things revolution, with incredible potential benefits for healthcare, entertainment, social interaction and more. This requires an increasing number of capable application developers, familiar with the unique requirements and characteristics of embedded computing.

This course aims to provide you with the knowledge of the essential tools and techniques to:
• Analyze embedded system requirements and develop realistic yet innovative embedded system requirements and specifications
• Be well acquainted with all aspects of the multidisciplinary process of embedded systems design
• Recognize the importance of embedded systems design in the smart system ecosystem
• Identify important future trends and strategies along with areas of research</td></tr>
<tr><td>Learning Outcomes</td><td colspan="5">• By the end of the course the students are expected to:
• Assess the differences between computers and embedded systems in terms of implementation and constraints
• Identify the unique challenges, opportunities and trends in embedded system design
• Combine and Synthesize aspects of key technologies involved in embedded systems and Internet-of-Things design
• Use appropriate mathematical tools to model sensors and the embedded system physical environment
• Model embedded applications using appropriate models of computation
• Develop requirements and specifications for innovative, yet realistic, embedded systems</td></tr>
</table>

	• Select appropriate implementation platforms based on the strengths and limitations of microcontrollers, RISC processors, DSPs and FPGAs • Analyze how implementation platforms affect performance, cost and power consumption • Optimize code for efficient embedded system implementations • Evaluate potential candidate scheduling algorithms for a given embedded system • Evaluate and validate partial and full designs with respect to design objectives • Analyze potential security threats in a given embedded system • Devise and employ countermeasures for security threats		
Prerequisites	None	Required	None
Course Content	1. **Introduction to embedded systems:** Differences between embedded and computer systems, constraints present in typical embedded systems, typical components of an embedded system, embedded system classification 2. **Embedded System Requirements and Specifications:** Requirements documents, functional and non-functional requirements, specification documents and verification plans, executable specifications 3. **Embedded System Modeling:** Modelling based on first principles, state machines, control and dataflow diagrams, Fourier, Laplace and z transforms. 4. **Models of Computation used in Embedded Systems:** Data Flow Graphs, Finite State Machines, Petri nets. 5. **Embedded System Implementation Platforms:** Architecture and ISA of the microcontroller, RISC processor and DSP. FPGAs as embedded system implementation platforms 6. **Programming for embedded systems:** Analog and Digital Input and Output, programming in high and low-level languages, program optimization 7. **Real-time operating systems:** task soft and hard deadlines, scheduling aof periodic and non-periodic tasks, static and dynamic memory management 8. **Testing, validation and evaluation:** Program traces. Instruments used in verification, waveforms generators, oscilloscopes, multimeters, logic analyzers Debugging using breakpoints, LEDs and logic analyzers. 9. **Embedded system dependability:** Reliability metrics and analysis. Mean Time Between Failures 10. **Embedded system security:** Embedded systems limitations regarding security. Physical and other attacks unique to embedded systems and IoT		
Teaching Methodology	The course is structured in three-hour lectures that are conducted with the help of material available online. The primary resources are presentations that introduce the course material together with practical examples and exercises to enhance the material learning process based on the textbook(s). Other resources include research papers and online tutorials in presentation or video format. Online short post-lecture quizzes are used to assess the level of student		

	understanding and provide feedback. Student questions are addressed through online interaction both synchronous and asynchronous (chat sessions and forum discussions). The online forums are also used for further student participation activities such as short group exercises. Examples are developing preliminary requirements and specification documents. Two assignments are part of the requirements of the course. The first one concerns writing a survey paper on some area of embedded system design while the other one is typically a programming/design assignment. Other assessment methods include tests with their assessment weight, date and time being set at the beginning of the semester via the course outline.
Bibliography	**Textbook:** • Peter Marwedel, "**Embedded System Design: Embedded Systems Foundations of Cyber-Physical Systems, and the Internet of Things**", Fourth Edition, Springer, 2021 **References:** • Edward Ashford Lee and Sanjit Arunkumar Seshia, "Introduction to Embedded Systems, a Cyber-Physical Systems Approach", Second Edition, MIT Press, 2017 • Peter Hintenaus, "**Engineering Embedded Systems: Physics, Programs, Circuits**", Springer, 2015 • Hermann Kopetz, (2011), "**Real-Time Systems: Design Principles for Distributed Embedded Applications**", Springer • M. Margolis, **Arduino Cookbook**, O'Reilly, 2011**.**
Assessment	• Assignments 40% • Tests/quizzes: 20% • Final exam: 40%
Language	English