University of Tehran
School of Electrical and Computer Engineering

Course:	**8101612 – Hardware/Software Codesign**
Course type:	EE*: Com ☐, E ☐, P ☐, B ☐, Con ☐, D ☐ — CE*: SW ☐, HW ☒, IT ☐ — Credit: 3 (Required / Elective)
Level:	Undergraduate ■ Graduate ☐
Co-requisite(s):	None.
Prerequisite(s):	Computer Architecture (8101423), Advanced Programming (8101119), Logic Circuits (8101367)
Prerequisite by topic:	Digital hardware design, architectures of computer systems, C programing language, hardware description language
Textbook(s):	[1] Patrick R. Schaumont, *A Practical Introduction to Hardware/Software Codesign*, Springer, 2010. [2] Daniel D. Gajski, S. Abdi, A. Gerstlauer, and G. Schirner, *Embedded System Design: Modeling, Synthesis and Verification*, Springer, 2010.
Coordinator:	Salehi Nasab , Professor, School of ECE
Goals:	This course provides an introduction to the design of electronic embedded systems that use hardware as well as software, with an emphasis on modern design methodologies and the transition area between hardware and software. This includes a study of modeling techniques of hardware and software components at different levels of abstraction and a study of interface techniques between hardware and software components. The course includes a practical component in the form of homework and practical computer assignments and projects.
Outcome:	Upon successful completion of the course, students will be able 1. Analyze and explain the control-flow and data-flow of a software program and a cycle-based hardware description; 2. Transform simple software programs into cycle-based hardware descriptions with equivalent behavior and vice versa; 3. Partition simple software programs into hardware and software components, and create appropriate hardware-software interfaces to reflect this partitioning; 4. Identify performance bottlenecks in a given hardware-software architecture and optimize them by transformations on hardware and software components; 5. Use simulation software to co-simulate software programs with cycle-based hardware descriptions.
Topics:	1) **Introduction and the nature of hardware and software**

	(fundamental properties of hardware and software and motivations for hardware/software codesign) 2) **System-level design challenges and methodologies (**surveys the basic concepts and principles of system design techniques and methodologies, including software and hardware which is particularly needed in today's embedded systems where software and hardware are interchangeable.) 3) **Data-flow and control-flow modeling** (introduces control-flow and data-flow as the common underlying properties of hardware and software) 4) **Microprogramming** (introduces micro-programmed architectures which are still very much like RTL machines, but have a flexible controller, which allows them to be reprogrammed with software.) 5) **General-purpose embedded cores** (reviews general-purpose embedded RISC cores that are the heart of typical contemporary hardware/software systems.) 6) **System synthesis** (deals with issues and possible solutions in synthesis and verification of software and hardware component needed in an embedded system platform.) 7) **System-on-chip** (ties the general-purpose embedded core back to the FSMD in the context of a System-on-Chip architecture.) 8) **On-chip busses** (discusses a typical on-chip bus structure and explains how it can efficiently move information between hardware and software) 9) **Hardware/software interfaces** (presents three different locations in the SoC architecture where a designer could attach custom hardware. This includes the memory-mapped interface, the coprocessor interface, and the custom processor datapath) 10) **Cosynthesis using reconfigurable hardware platform** (introduces a framework for high-level hardware-software cosynthesis and application development.) 11) **Application analysis tools for ASIP design** (presents application profiling and instruction-set customization tools and techniques.) 12) **Coprocessor control shell design** (shows how a designer can take an arbitrary hardware module and attach it to one of the three hardware/software interfaces.)
Computer usage:	C programing language and Hardware description language
Assignments:	10 homework assignments
Projects:	4 Computer Assignments
Grading:	Assignments: 10 % Computer Assignments & quiz: 20% Midterm exam: 30 % Final exam: 40 %
Further readings:	[1] Jingzhao ou and Viktor k. PraSanna, *Energy Efficient Hardware-Software Co-Synthesis Using Reconfigurable Hardware*, CRC Press, 2010. [2] K. Karuri and R. Leupers, *Application Analysis Tools for ASIP Design*, Springer 2011.

	[3] Ivan Radojevic and Z. Salcic, *Embedded Systems Design Based on Formal Models of Computation*, Springer 2011. [4] Giovanni De Micheli, R. Ernst, and W. H. Wolf, *Readings in hardware/software co-design*, Morgan Kaufmann, 2002. [5] Wayne Hendrix Wolf, *Computers as components: principles of embedded computing system design*, 2nd Edition, Morgan Kaufmann, 2008.
Prepared by:	Mostafa E. Salehi
Date:	Sept. 11, 2011

*EE: Electrical Engineering		CE: Computer Engineering	
Com	Communications	SW	Software
E	Electronics	HW	Hardware
P	Power	IT	Information Technology
B	Bioelectronics		
Con	Control		
D	Digital System		