CSCE 313: Embedded Systems

1. Course number and name: CSCE 313: Embedded Systems

2. Credit: 3-hrs; Contact: 3 lectures of 50 minutes each or 2 lectures of 75 minutes each per week

3. Instructor: Spring 2011: Jason Bakos

4. Text book: Ronald Sass and Andrew G. Schmidt, *Embedded Systems Design with Platform FPGAs: Principles and Practices*, Elsevier, 2010.

1. Specific course information
   a. Catalog description: Fundamentals of embedded systems: hardware components, software components, hardware/software interface design, and hardware/software co-design.
   b. Prerequisites: CSCE 211 and 212
   c. Required in CE curriculum

2. Specific goals for the course
   a. Specific outcomes of instruction:
      1. Perform hardware/software co-design for a programmable embedded system;
      2. Write software that directly interfaces with I/O peripherals such as LEDs, LCD panels, buttons, monitors, and remote consoles;
      3. Write software that performs real-time processing of audio and video data;
      4. Use high-level synthesis tools to develop coprocessor architectures in an embedded environment.
   b. Relation of course outcomes to Student Outcomes: CE: see page 2; CS & CIS: see page 3

3. Topics covered and approximate weight (14 weeks, 3 hours/week, 42 hours total)
   1. Design constraints for embedded systems (1 week);
   2. Platform FPGA design methodology for programmable system-on-a-chip (2 weeks);
   3. Image processing (3 weeks);
   4. Audio processing (3 weeks);
   5. Video processing (3 weeks);
   6. Embedded application acceleration using special-purpose logic (2 weeks).

c.

Computer Engineering

Relation of Course Outcomes to EAC Student Outcomes*

Course Outcomes (CE)	Student Outcomes											
	(a) apply knowledge of mathematics, science, and engineering	(b) design and conduct experiments, … interpret data	(c) design a system, component, or process to meet desired needs …	(d) function on multidisciplinary teams	(e) identify, formulate, and solve engineering problems	(f) an understanding of professional and ethical responsibility	(g) communicate effectively	(h) the broad education to understand the impact of engineering solutions …	(i) a recognition of the need for, and an ability to engage in life-long learning	(j) a knowledge of contemporary issues	(k) use the techniques, skills, and modern engineering tools ….	(CE) demonstrate knowledge of discrete mathematics [CE]
Criteria	a	b	c	d	e	f	g	h	i	j	k	CE
1. Perform hardware/ software co-design for a programmable embedded system.	1		3		3			3			3	
2. Write software that directly interfaces with I/O peripherals such as LEDs, LCD panels, buttons, monitors, and remote consoles.	1		3		3			3			3	
3. Write software that performs real-time processing of audio and video data.	1		3		3			3			3	
4. Use high-level synthesis tools to develop coprocessor architectures in an embedded environment.	1	2	3		3			3			3	

* 3 = major contributor, 2 = moderate contributor, 1 = minor contributor; blank if not related

d.

Computer Science & Computer Information Systems

Relation of Course Outcomes to CAC Student Outcomes*

Course Outcomes (CS & CIS)	Student Outcomes											
	All									CS		CIS
	(a) apply knowledge of computing and mathematics appropriate to the discipline	(b) analyze a problem, and identify and define the computing requirements …	c) design, implement, and evaluate a computer-based system, …	(d) function effectively on teams to accomplish a common goal	(e) An understanding of professional, ethical, legal, … responsibilities	(f) communicate effectively with a range of audiences	(g) analyze the local and global impact of computing on … society	(h) Recognition of the need for … continuing professional development	(i) current techniques, skills, and tools necessary for computing practice	(j) apply mathematical foundations, algorithmic principles, and CS theory …	(k) apply design and development principles	(j) An understand processes that support the information systems environment.
Criteria	a	b	c	d	e	f	g	h	i	j	k	j
1. Perform hardware/ software co-design for a programmable embedded system.	3	3	3	3					3	1	3	3
2. Write software that directly interfaces with I/ O peripherals such as LEDs, LCD panels, buttons, monitors, and remote consoles.	3	3	3	3					3	2	3	3
3. Write software that performs real-time processing of audio and video data.	3	3	3	3					3	2	3	3
4. Use high-level synthesis tools to develop coprocessor architectures in an embedded environment.	3	3	3	3					3	2	3	3

* 3 = major contributor, 2 = moderate contributor, 1 = minor contributor; blank if not related