

ECE4180 Course Syllabus

ECE4180

Embedded Systems Design (3-0-3-4)

CMPE Degree

This course is Elective for the CMPE degree.

EE Degree

This course is Selected Elective for the EE degree. * (Selected Elective means this course is one of a few choices that are required for the degree.)

Lab Hours

3 supervised lab hours and 0 unsupervised lab hours

Course Coordinator

Hamblen, James O

Prerequisites

CS 1372 and (ECE 3035 or ECE 3055/3056) or ECE 2031/20X2 and ECE 2035/3035 and (ECE 3040 or ECE 3055)

Corequisites

None

Catalog Description

Processors, chipsets, busses, and I/O devices for high-end embedded systems. Embedded operating systems; device drivers and applications for embedded systems.

Textbook(s)

Mbed Handbook and Cookbook, ARM Corporation, 2012. (required) (comment: Available free online at <http://mbed.org/cookbook>)

Eckel, *Thinking in C++, Introduction to Standard C++, Volume 2* (2nd edition), Prentice Hall, 2000. ISBN 0139798099, ISBN 9780139798092 (optional) (comment: text is available free on line at <http://original.jamesthornton.com/eckel/>)

Deitel and Deitel, *C++, How to Program* (9th edition), Prentice Hall, 2011. ISBN 9780133378719 (optional)

Hamblen, *Introduction to Embedded Systems Using Windows Embedded CE* (3rd edition), Microsoft, 2009. (required) (comment: This text is available free on-line at http://users.ece.gatech.edu/~hamblen/wemap3/CDROM_IMAGE/IntroEmbSysIII.pdf)

Course Outcomes

Upon successful completion of this course, students should be able to:

1. Design and implement embedded systems using the techniques and tools developed in this course.
2. Develop evaluation methods for embedded software, I/O devices, and embedded systems and analyze and interpret the resulting data.

3. Prepare reports, documentation, presentations, and web-based materials conforming to appropriate technical standards.
4. Design and implement hardware and software interfaces to connect standard I/O devices to a computer.
5. Develop C/C++ firmware and software applications for embedded devices utilizing common I/O devices and OS APIs.
6. Develop a basic stream interface OS I/O device driver.

Student Outcomes

In the parentheses for each Student Outcome:

"P" for primary indicates the outcome is a major focus of the entire course.

"M" for moderate indicates the outcome is the focus of at least one component of the course, but not majority of course material.

"LN" for "little to none" indicates that the course does not contribute significantly to this outcome.

1. (M) An ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics
2. (P) An ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors
3. (LN) An ability to communicate effectively with a range of audiences
4. (LN) An ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts
5. (M) An ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives
6. (LN) An ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions
7. (M) An ability to acquire and apply new knowledge as needed, using appropriate learning strategies.

Topical Outline

1. Introduction to Embedded Systems (1 week)
 - a. Embedded Products (i.e., Cell Phones, Robots, GPS, Cameras,
 - b. The Design and Development Process for a new embedded product
 - c. Software Development and Debug Tool Flows
2. Hardware for Embedded Systems Design (5 Weeks)
 - a. Processors, Chipsets, and Memory
 - b. ARM and X86 ISA
 - c. I/O devices and bus interfaces
 - d. Example Design (i.e. basic parallel I/O port)
 - e. Common Bus Standards (i.e., ISA, PCI, AMBA, PCI Express)
 - f. Common I/O interface Standards (i.e., Parallel, RS-232, SPI)
 - g. Analog I/O using A/D and D/A convertors
 - h. Driving high current and high voltage I/O devices (i.e., hi
 - i. Using PWM to efficiently control external I/O devices (i.e.
 - j. Programmed I/O
 - k. Interrupt driven I/O
 - l. Using DMA for I/O transfers
 - m. Example System Designs (i.e., small 32-bit ARM and X86-base

3. Software for Embedded Systems Design (4 Weeks)
 - a. Role of an Embedded Operating System
 - b. Hard and Soft Real-time systems
 - c. Multitasking, Threads, and Synchronization
 - d. Operating Systems used in Embedded Devices (Windows Embedded)
 - e. Overview of an example RTOS
 - f. Building an OS for a new device
 - g. Application Development using OS APIs for I/O devices and G
 - h. I/O device examples (i.e., A/D, RS-232, cameras, GPS, displ
 - i. BSPs and developing OS Device Drivers for new I/O devices
 - j. Developing Software for Safety Critical Systems
4. Proposed Project Design Review (1 week)
5. Design Project Implementation (3 weeks)
6. Project Demo and Presentation (1week)