ICS 212: Introduction to Embedded Computing Systems

Rajesh K. Gupta
Information and Computer Science
rgupta@ics.uci.edu, *824-8052*

Course Overview and Goals:

Increasing integration of communications, multimedia and processing and relentless digitization of data (including even RF data) continues to expand the scope and complexity of embedded systems. To appreciate these advances, and to productively contribute to future advances of these systems, a *critical* appreciation of the underlying technology underpinning is a must. The goal of this course is to develop a comprehensive understanding of the technologies behind the embedded systems, particularly, those using computing elements (processor, DSP, or ASSPs). The students develop an appreciation of the technology capabilities and limitations of the hardware, software components for building embedded systems, and methods to evaluate design tradeoffs between different technology choices.

Course Rationale and Relationship to ICS Curriculum:

Continuing advances in system software and hardware components now present exciting opportunities in building embedded systems for applications ranging from embedded control, multimedia, networking and information and biomedical appliances. Building these systems, particularly for highly integrated micro-electronic technologies and mobile applications, presents a challenge at every of level abstraction from gate-level designs to complex runtime systems. Even with a detailed technical knowledge in a specific technology area that make up an embedded system, a good system design would require understanding of the design tradeoffs across choice in technologies that make up the system. For instance, is it better to a particular interface as a gate-level logic or build the functionality into device driver software. This course fills this gap by presenting basic characteristics and *usage model* of the technologies that make up an embedded system and describing their relationship to system analysis tools and methods. Continuing digitization offers a unique advantage for the embedded system designer in using software to develop both systems and the applications that must run on them. This course provides the training necessary for ICS graduates to be effective contributors to embedded systems and be on the leading edge of the system integration by using digitization and software in innovative ways.

This course is recommended for all entering students who are affiliated with the Center for Embedded Computing Systems.

Topical Outline:

1. Introduction to embedded systems: classification, characteristics and requirements.
2. Modeling and characterization of embedded computing systems.
3. Interfacing basics: signal frequency spectrum, and sampling, digitization (ADC, DAC), signal conditioning and processing (modulation, filtering, transformation), sensors and actuators.
4. Communication strategies for embedded systems: encoding, buses, protocols and flow control.
5. Mobility in embedded systems: system and medium characteristics
6. Computing technologies for embedded systems: microprocessors, microcontrollers, DSP, application-specific signal processors. Their implementation as re-usable core blocks.
7. Memory technology for embedded systems: RAM and ROM components, intelligent memories. Performance characteristics and design trade-offs.
8. Component interfacing: interface timing, synchronous versus asynchronous interfacing, polling, interrupt and DMA, design methodologies for interface circuits particularly using re-programmable devices (FPGAs).
9. Pre-designed and pre-verified components for embedded systems: processors, network interfaces, wireless (RF/analog), DSP, encryption, compression components.
10. Using core components in system design. Case studies from: building a spread-spectrum wireless terminal, multimedia engine, vehicular LAN and controller. Role of standards in component socketing.

Course Text Book:

- H. Kopetz, “Real-time Systems,” Kluwer, 1997
- Class lecture notes.

Recommended Reading:

- Additional reading from selected journal papers.

Prerequisites:

An undergraduate degree in ICS is required for this graduate course. Basic courses in digital hardware, algorithms and data structures, elementary calculus, and probability. ICS 51, ICS 152, Math 3A or 6C, ICS 161.