AME 3623: Embedded Real-Time Systems

Andrew H. Fagg
Symbiotic Computing Laboratory
School of Computer Science
University of Oklahoma

Teaching Assistant: Di Wang
What is an Embedded System?
What is an Embedded System?

• Computing system with a non-standard interface (often no keyboard or screen)
• Often involved in sensing and control (and may not even talk to a human)
• Typically a custom system for a very specific application
What is an Embedded System? (cont)

• Limited processing capabilities:
  – Can be extremely small
  – Can require a small amount of power
• Can have significant real-time constraints
  – Act on inputs very quickly
  – Generate high-frequency outputs
• Often a higher expectation of reliability
Examples of Embedded Systems
Robotics

Mark Tilden
Los Alamos National Labs
and Wowwee

picture from
Robosapiens
Humanoid Robotics

NASA/JSC Robonaut

UMass Torso
Real-Time Robotic Control
Dual-Limb Coordination
Personal Satellite Assistants

NASA Ames Research Center

picture from *Robosapiens*
Intelligent Prosthetics

Hugh Herr
MIT Leg Lab

picture from *Robosapiens*
Sensor Networks

1000 sensor nodes
Embedded Systems Challenges
Embedded Systems Challenges

• Sensing the environment:
  – Sensors are typically far from ideal (noise, nonlinearities, etc.)
  – Sensors/subsystems can fail
  – Hard to get a ‘complete’ view of the environment

• Affecting the environment through “actuators”
  – Application can require fast, precise responses
Embedded Systems Challenges (cont)

• Testing/debugging can be very difficult:
  – Hard to identify and replicate all possible situations
  – Often involves the interaction of many different components
  – Often no standard user interface
  – Limited on-board resources with which to record system state

• Competing requirements of cost, complexity, design time, size, power…
Embedded Systems Challenges (cont)

• Lack of reliability can be a killer ..... literally
My Assumptions About You

• Circuits and sensors class (or equivalent):
  – Boolean logic and circuits (AND/OR/NOT gates)
  – Analog circuits (in particular, resistive-capacitive circuits)

• Some background in programming
  – We will be using C for all projects

• Everyone has a laptop that can be used for the projects
Course Goals

By the end of this course, you will be able to:

• design and implement embedded circuits involving microcontrollers, sensors and actuators,

• design, program and debug embedded sensing and control software,

• work in collaborative teams to solve system design and implementation problems, and

• communicate in both oral and written forms with team members.
Sources of Information

• Primary readings: web pages and book sections (posted on D2L)
• Other textbooks:
  – Optional references
  – Mostly available for download (legally!)
• Class web page: www.cs.ou.edu/~fagg/classes/ame3623
• Desire2Learn: learn.ou.edu

You are responsible for making sure that you have access to all of these resources
Class Schedule

www.cs.ou.edu/~fagg/classes/ame3623/schedule.html

- Lecture plans
- Required reading
Channels of Communication

• Lecture
• Class email list: time-critical messages to the class
• Desire2Learn announcements
• Desire2Learn discussion group: you may post questions (and answers)
• Private email or office hours for non-public questions/discussions
Grading

• Components of your grade:
  – Midterm exam: 10%
  – Final exam: 20%
  – Four homework assignments and several pop quizzes: 25%
  – Five projects: 40%
  – In-class participation: 5%

• Grades will be posted on the Desire2Learn

• Final grades boundaries will be selected based on the overall class distribution
Exams

• Closed book/closed notes
  – Exception: you are allowed 1 page of your own notes

• Assigned seating

• No electronic devices

• Grading questions must be addressed before the returned exams leave the classroom
Homework Assignments

• Individual work

• Hand-in:
  – Through the digital dropbox of Desire2Learn or hardcopy
  – By 5:00 on the due date (no exceptions)

• Grading questions must be addressed within one week of being returned
Group Projects

• Five group projects will focus on sensor processing and design of robot control circuits
  – Control of a hovercraft

• Project Topics:
  – Sensor interface and processing
  – Intra-processor communication
  – Finite-state machines and microcontrollers
Project Grading

Group grades are a function of:
- Code correctness and readability
- Documentation
- Demonstration and presentation

Individual grades:
- Group grade scaled by your personal contribution, plus
- Personal contribution (must have 2 significant contributions over the course of the semester)
Group Projects (cont)

• Groups will be of size 3-4 and will be assigned
• Be ready to demonstrate project by the due date
• Projects require more than a day to complete
• Project reports in pdf or postscript format
• Projects may be late (but I do not recommend this):
  – 0-24 hrs: 10% penalty
  – 24-48 hrs: 20% penalty
  – 48+ hrs: 100% penalty
Classroom Conduct

• Ask plenty of questions
• Contribute to the discussions

• No: cell phone use (including texting)
• No: laptop use (except for classroom exercises)

• More details in the syllabus
Academic Conduct/Misconduct

Homework assignments:

- All work must be your own: no looking at or copying solutions from other students or from the net

- General discussion is OK (e.g., the fundamental skills that we are learning)

- When in doubt: ask
Academic Conduct/Misconduct

Projects:
• All work must be that of your group: no looking at, discussing or copying solutions from other groups or from the net
• General discussion is (again) OK

Secure your data
Next Time

• Analog circuits review
• Review readings: see the schedule page