CS [45]163: Embedded 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)

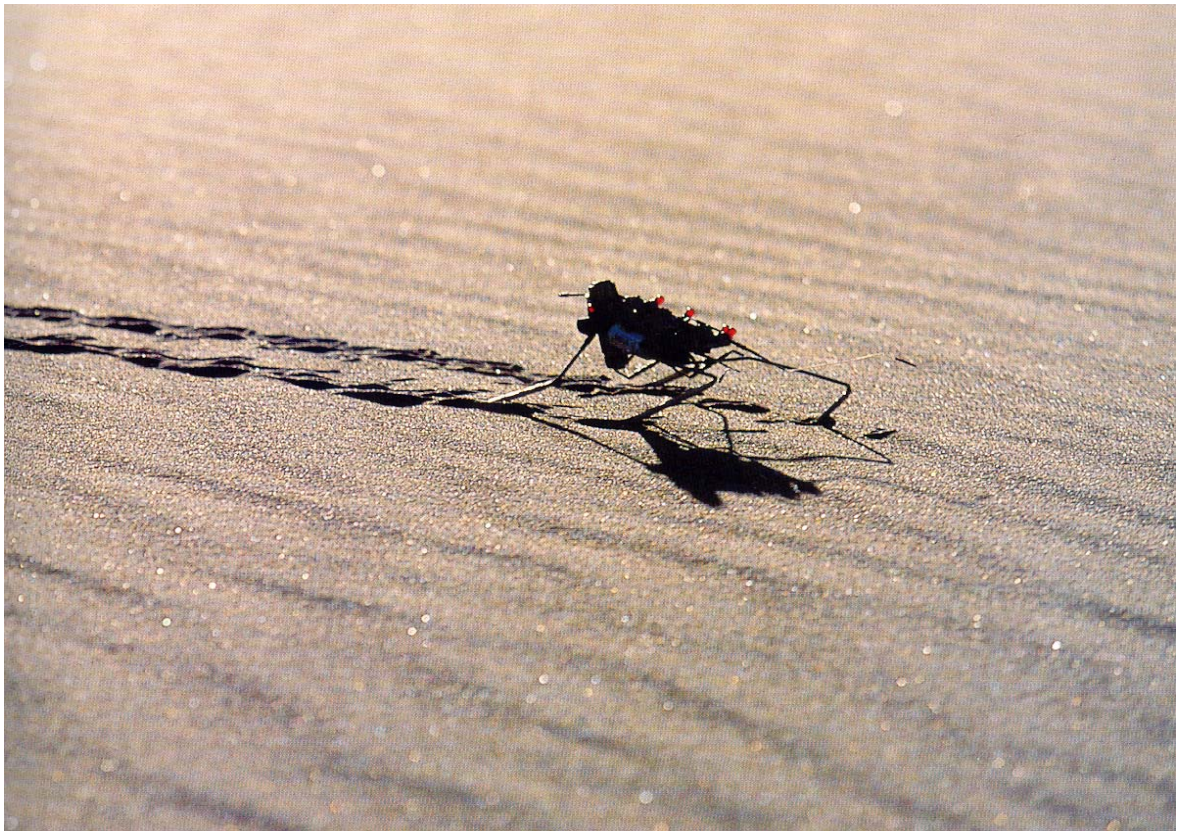
- Limited processing capabilities:
  - Can be extremely small
  - Can require a small amount of power
- Can have significant real-time constraints
  - Act on inputs very quickly
  - Generate high-frequency outputs
- Often a higher expectation of reliability

Examples of Embedded Systems

Robotics

Mark Tilden

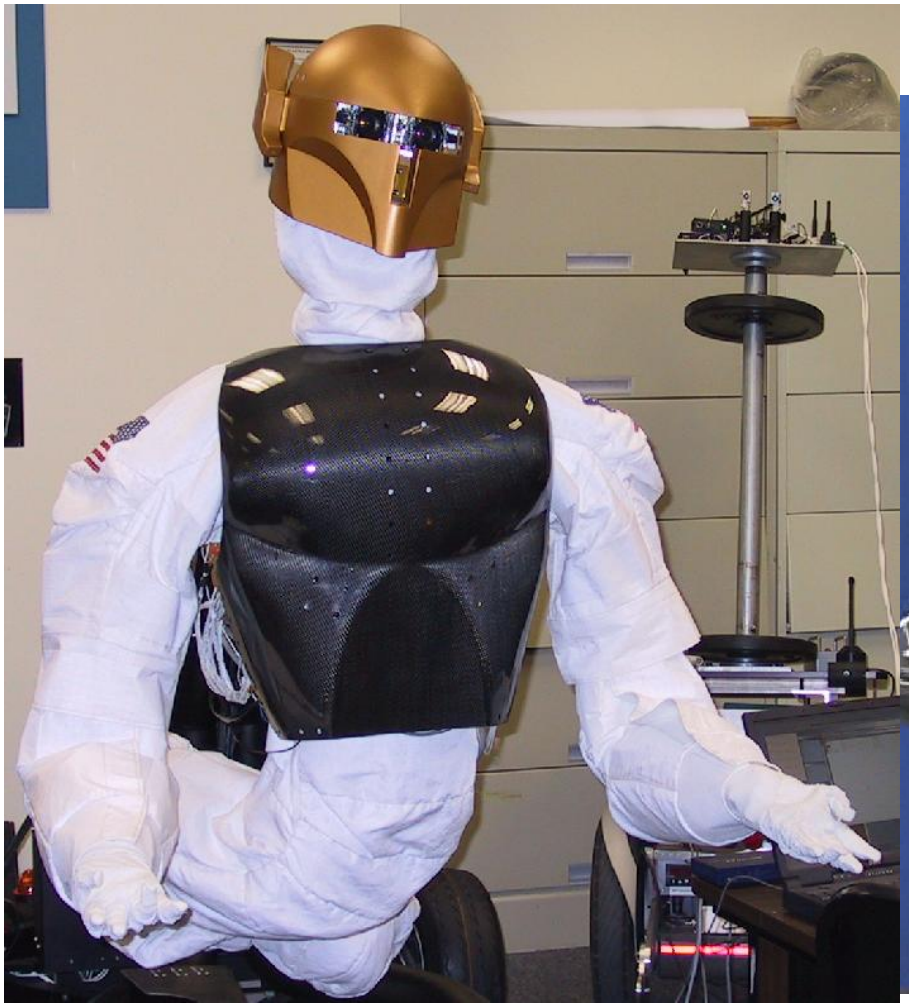
Los Alamos National Labs and Wowwee



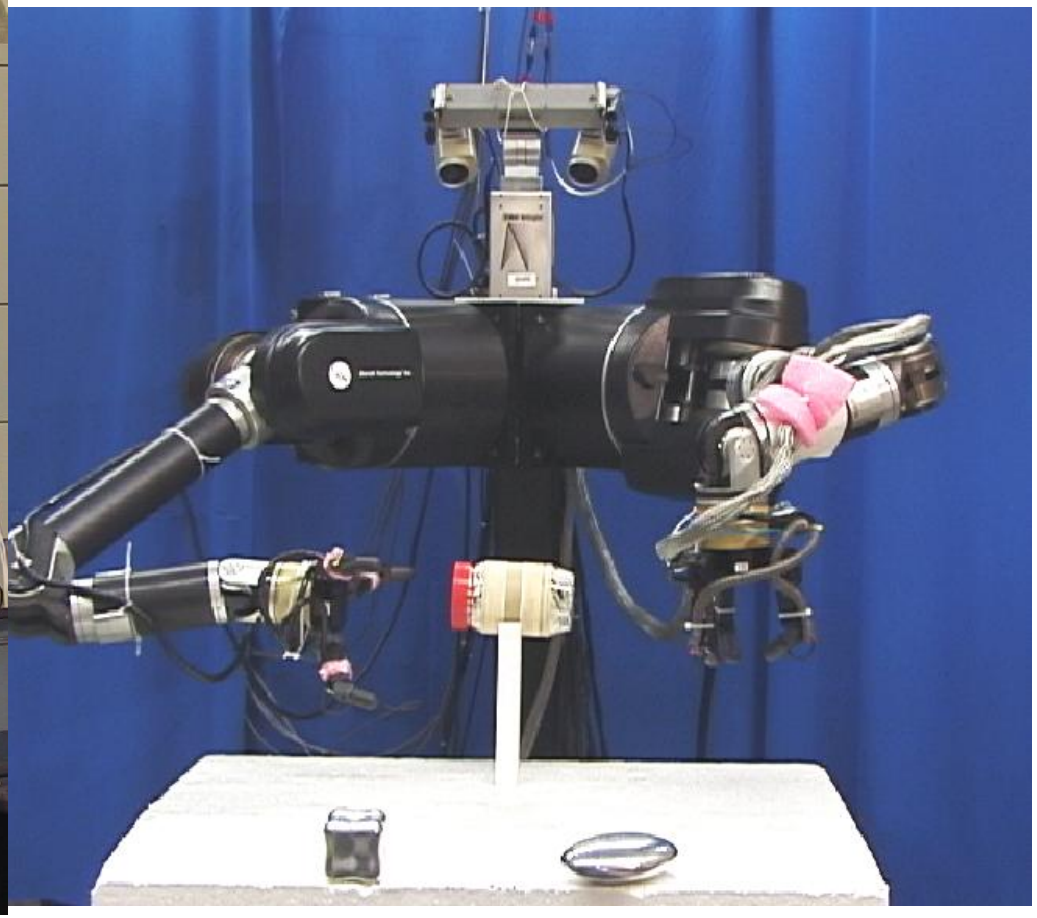
picture from *Robosapiens*

Humanoid Robotics

NASA/JSC Robonaut



UMass Torso



Dual-Limb Coordination

Personal Satellite Assistants

NASA Ames Research Center

picture from *Robosapiens*

Wearable Computing

Intelligent Prosthetics

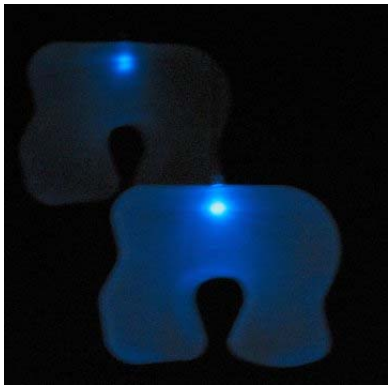
Hugh Herr

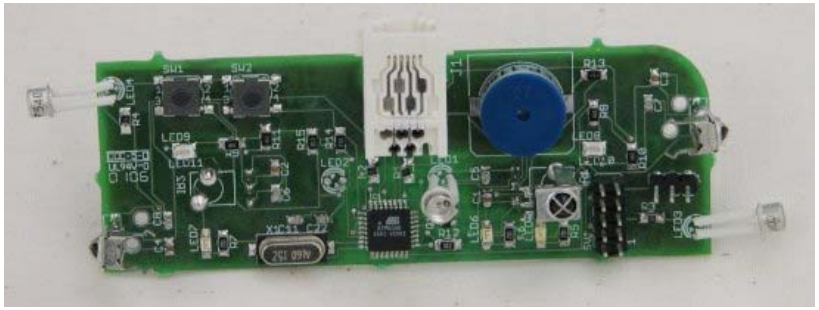
MIT Leg Lab

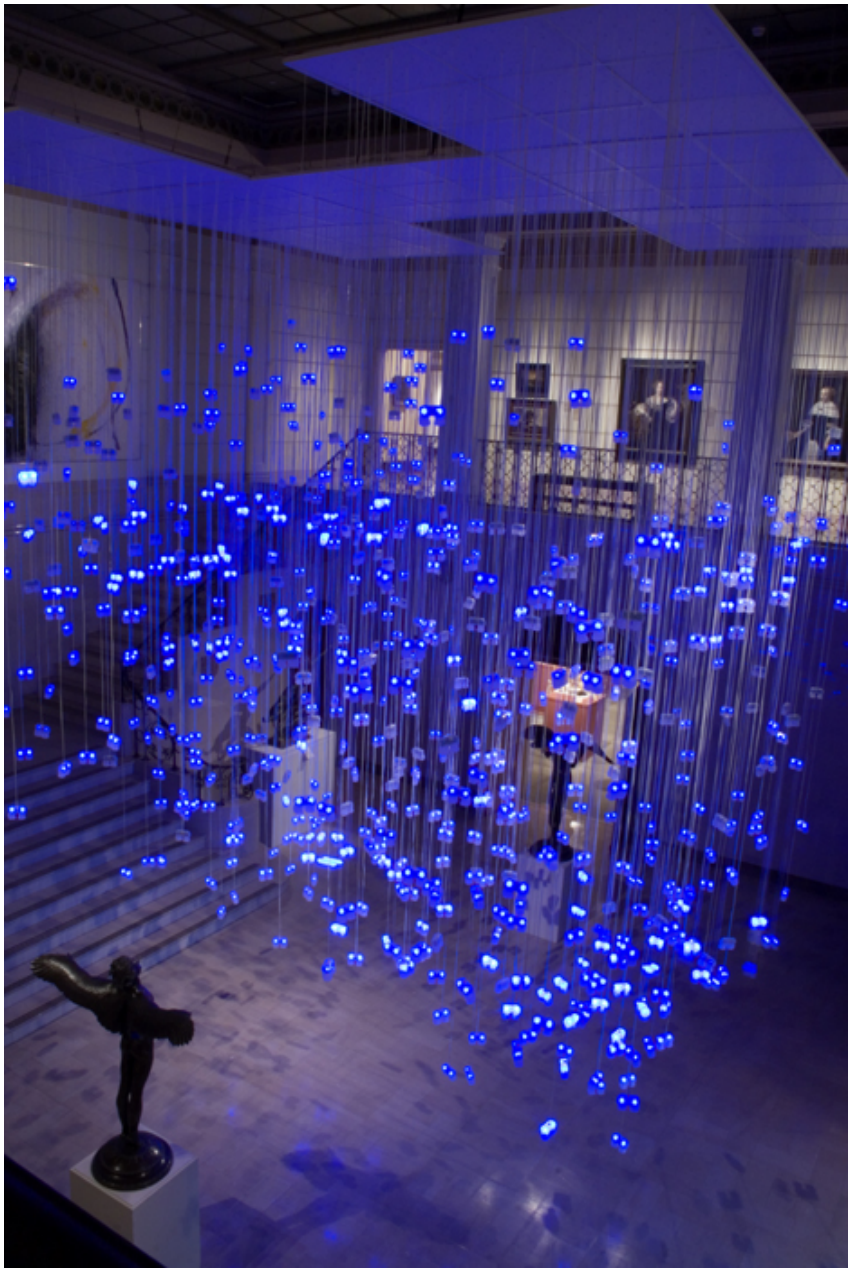
picture from *Robosapiens*

Sensor Networks

1000 sensor nodes







Embedded Systems Challenges

Embedded Systems Challenges

- Sensing the environment:
  - Sensors are typically far from ideal (noise, nonlinearities, etc.)
  - Sensors 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

- Background in Computer Organization and Operating Systems
- Programming in C
- Everyone has a laptop that can be used for the projects

Skills You Should Learn Here

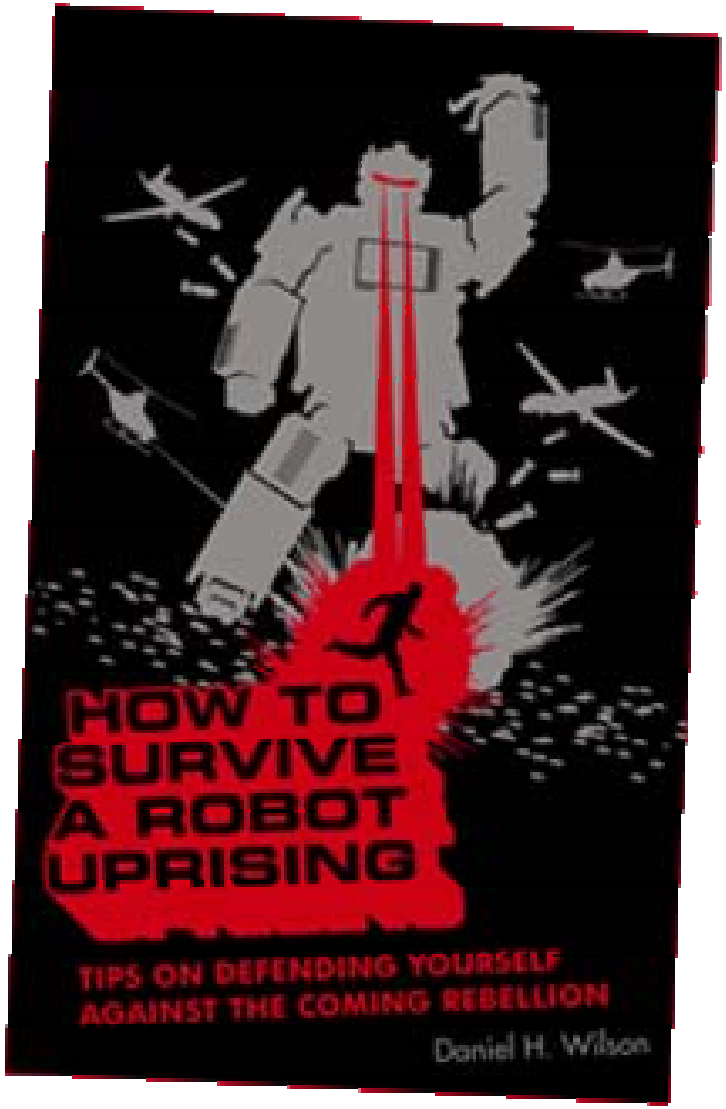
- Read (and understand!) technical documentation
- Design and implement embedded systems involving a microcontroller, sensors, actuators, and the necessary “glue”
- Design, program, and debug embedded software for sensing and control
- Work in collaborative teams

Sources of Information

- Required textbooks:
  - Designing Embedded Hardware, John Catsoulis, O'Reilly, 2005, **2nd Edition**, ISBN: 0-596-00755-8
  - **(optional)** Embedded C Programming and the Atmel AVR, Richard H. Barnett, Sarah Cox, Larry O'Cull (2006), **2nd Edition**, Thomson/Delmar Learning, ISBN: 1418039594
  - Also reading the Atmel Mega 8 specification (downloadable)
- Class web page: www.cs.ou.edu/~fagg/classes/ame3623_s07/
- Desire2Learn: learn.ou.edu

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

(available at the Engineering Library)



Class Schedule

www.cs.ou.edu/~fagg/classes/embedded_systems_2008/schedule.html

- Lecture plans
- Required reading

As changes are made, they will be posted here

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