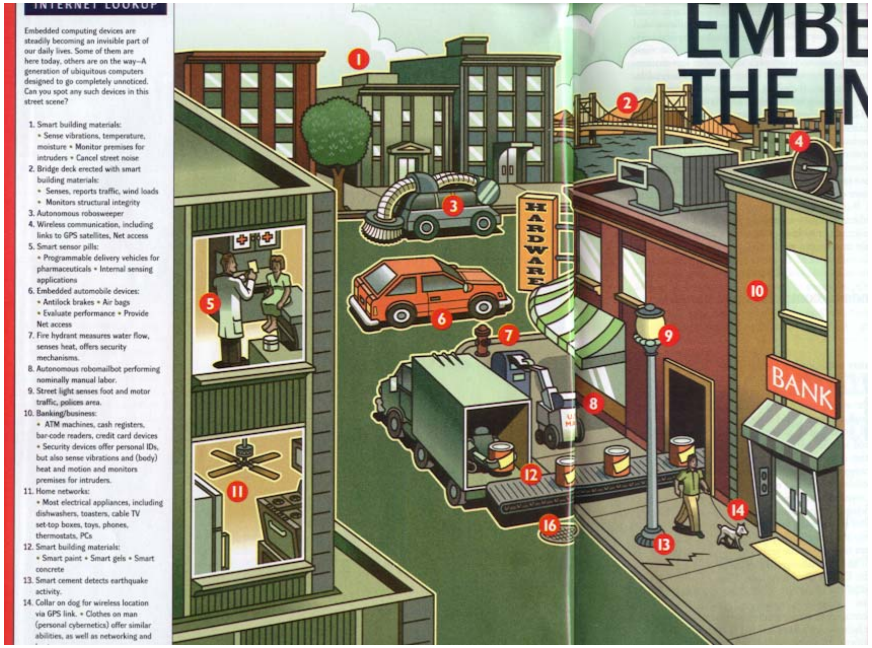
Embedded Systems: EmNets

April 15, 2003

Class Meeting 25

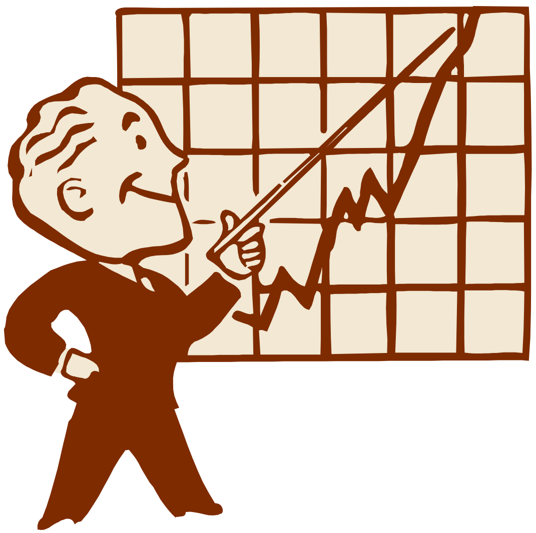


Announcement

- CORRECTION:
  - Reading for today should have been Chapters 1 and 2 of Embedded Everywhere!!
  - Reading for Thursday should have been Chapter 3 of Embedded Everywhere!!

Student Paper Presentation

- *Embedded, Everywhere*, Chapter 1, National Academy Press, 2001.
- Presented by David Resseguie



Another Motivating Example

- Partially collapsed building in an earthquake

- Multiple robots, people trying to locate places where people might be trapped

- Queries
  - Where are people trapped ?
  - How can one get there ?
  - What does the space look like ?

Characteristics of EmNets

- Multiple interacting nodes (> 2, preferrably thousands)
- Embedded in control systems operating w/o human intervention
- Tightly coupled to physical world
- Purpose other than general computing and communications
- Natural or engineered contexts

Distinctions within EmNets

- Energy-constrained vs. non-energy constrained
- Fixed topology vs. flexible topology
- Safety-critical applications vs. non-safety critical applications
- Highly engineered vs. unconstrained, ad hoc systems

Differences with Traditional Systems

- Combination of constraints on solutions, including:
  - Tight coupling to the physical world and each other
  - Resource-constrained environment
  - Persists for long periods of time
  - Many interacting components
  - Use by nonexpert users

Key Trends and Developments Enabling Embedded Computers (EmNets)

- Silicon Scaling
- Computing:
  - Growing complexity
  - Simpler processors
  - Power dissipation
- Communication
  - Wireline
  - Wireless -- Consumer devices with 802.11b wireless; small radios
- Geolocation
- Small/real time OS
- MEMS sensing and actuation
- Better battery technology

- A one inch form factor mote combines sensing communication and even actuation

Silicon Scaling

- Reduces size, cost, power
- Improves performance, reliability
- Enables cheap computing, so that it is economical to embed computing inside devices not traditionally thought of as computers
- Rapidly decreasing cost curve creates and expands a huge market for embedded computing
- As communication becomes cheaper, allows embedded computeres to talk with each other and outside world

Computing

- Increasingly complex chips → increased design cost
  - More engineers needed to design state-of-the-art chip
  - Leads to very expensive initial tooling

- Therefore, chips are inexpensive only if they can be produced in large volumes to amortize design costs

Computing Issues

- Growing Complexity
  - Logical complexity → possibility for design errors → reduced reliability
  - Cost: Intel's Pentium 4 required design team of 100s, several years, and up-front investment of $100s M
  - Rate of improvement slowing, due to difficulty of extracting more parallelism at instruction level
- Simpler Processors
  - Alternative to increased performance:
    - Use added transistors for other functions (e.g., communications)
    - Or, don't use them at all → cheaper and smaller and less power consuming
- Power Dissipation
  - Difficult to add expensive power supplies and cooling systems in embedded processors
  - Techniques for reducing power: turn off processor when machine is inactive, use more time to complete a task (➔ less energy)
  - Use custom chips to solve specific application with minimal overhead (but raises cost issues)

Communication

- Cost driven down
- Improved technology → more sophisticated coding and detection algorithms → either decreased power or increased bandwidth
- Constraint:
  - Bandwidth increasing
  - Cost decreasing
  - Power demands about same
- Wireless communications:
  - While flourishing, involves overcoming many problems inherent in over-the-air communication:
    - Radio-frequency spectrum is a scarce resource, must be shared:
      - In time, space, and/or frequency
      - Lowering range → reduced space overlap and reduced power, but requires multiple hops
      - Multiple hops → opportunity for data aggregation and collaborative processing

Software – Operating Systems and Applications

- In EmNets, line between OS and application is blurry
- EmNets: very complex, heterogeneous distributed system
- Requirements: security, safety, reliability, usability, privacy
- Cost of failure corrections – often much higher than in traditional desktop and server environments
- Real-time performance requirements: difficult to add new components if they weren’t designed in from the beginning

MEMS

- MicroElectroMechanical systems
- Use mechanical properties of device, in conjunction with electronic sensing, processing, and control, to achieve real-world physical sensing and actuation
- Difficulties:
  - Devices must be self-monitoring and adaptive
  - Have to integrate electronics (for control and communication) and MEMS sensors in same silicon
  - Below a certain size, MEMS devices don't work well

Summary of Core Technologies

- Large systems → thousands or millions of sensing, computing, and actuating nodes
- Feasible because of cumulative effects of silicon scaling, advances in computing hardware/software, and wireless communications, plus MEMS
- Constraints:
  - Communication is costly on-chip and between chips
  - Power dissipation is a problem
  - Battery life is limited
  - Design complexity is high

Next Time

- Chapter 3 of Embedded Anywhere
- Self-configuration and Adaptive Coordination