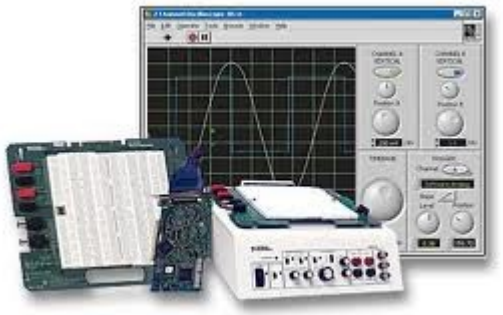
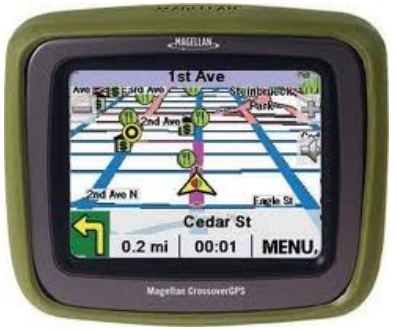
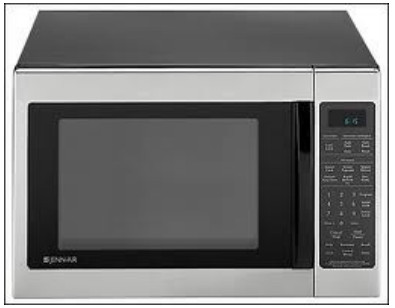
Introduction to Embedded Computer System

Chapter 1

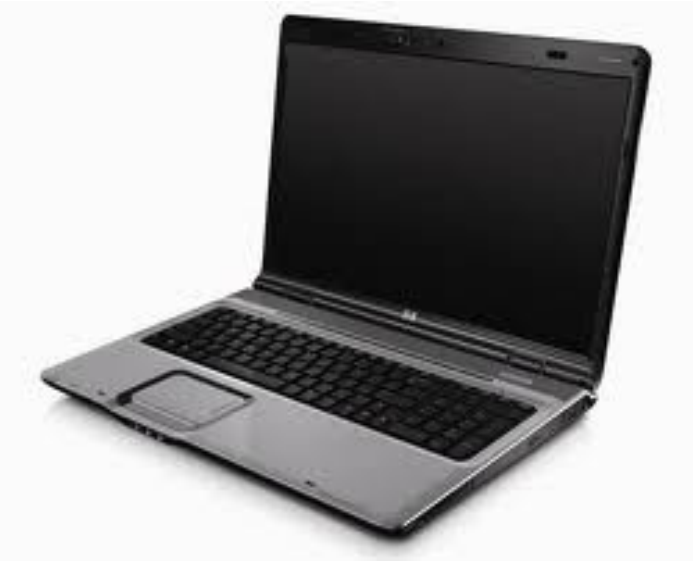
They are Embedded Systems

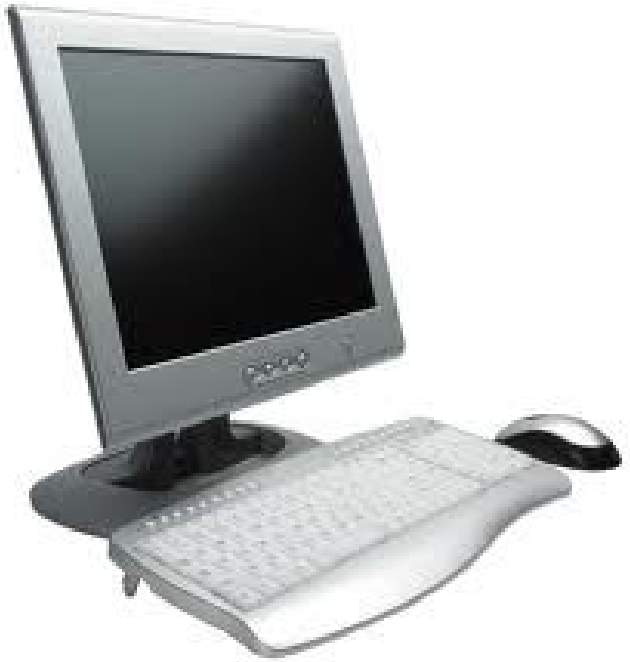




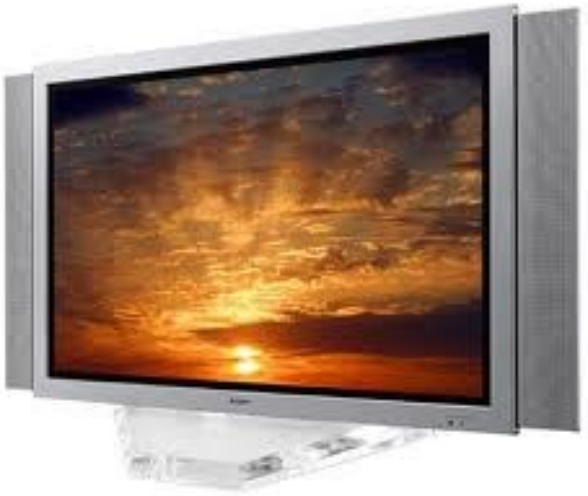


Is it an embedded system?





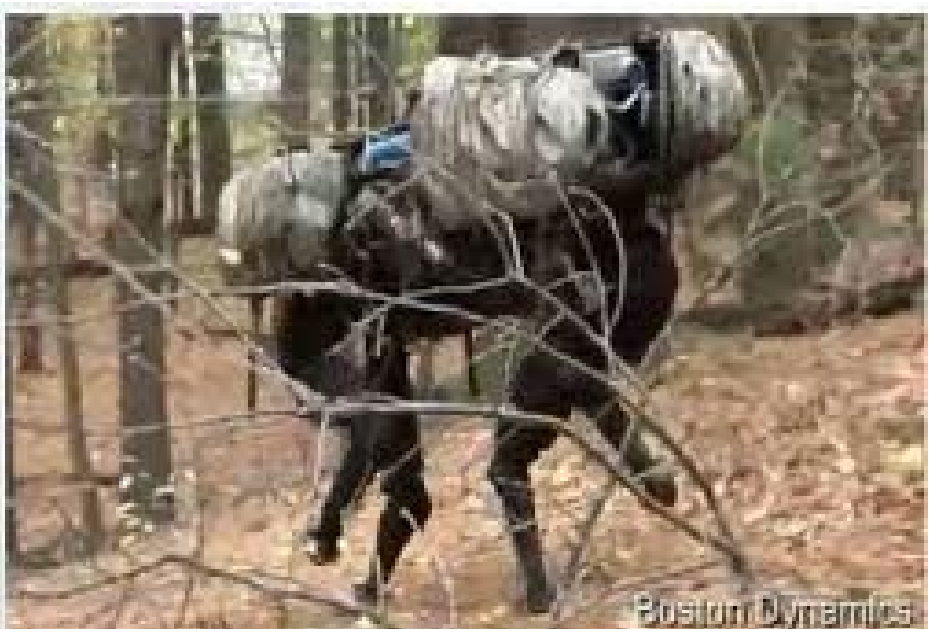
Is it an embedded system?



Is it an embedded system?

Smartbird

Big dog



Definition

- No clear definition !!!
  - Non-computer, **small**, regular, big, super
- Non-computer : no computation
  - The transmitter of car remote controller
- Small computer : embedded system
  - The receiver of car remote controller
- Regular computer : 1 CPU (a few cores)
- Big computer : a few CPUs
  - Servers
- Super computer : thousands of CPUs
  - IBM Roadrunner, 1.105 PFLOPS

Characteristics

- More limited in hardware and software functionality.
  - Sensor
  - Hardware : 4KByte ram, 1MIPS CPU
  - Voice recording
    - 8 bit per sample
    - 9.6K samples per second (only fair quality)
    - Without compression : 0.41 seconds of recording
    - With compression, each sample should be compressed within 104 instructions.
  - But, not always so resource constrained

Characteristics

- Designed to perform dedicated functions
  - Digital LCD
  - A remote controller, a signal receiver, a panel
  - Although the controller has many buttons, the TV only has a fixed menu with a fixed set of functions.
    - Change channel
    - Recording
    - Video on demand
    - Change color
  - So, it is feasible to have a manual with a TV, which describes all functions provided by the TV.

Characteristics

- Designed with specific quality requirements
  - Car's brake controller
    - ABS, ESC, ...
    - 60 mph = 87.5 feet per second
    - A response time of 0.1 second in the brake system means a distance of 8.75 feet.
    - High real-time requirement
  - Car's engine controller
    - Gas and ignition control in response to current speed
    - High reliability requirement : cannot fail over years
  - Airplane : higher
  - TV : lower

Characteristics

- Modern general purpose embedded systems
  - PDAs, cell phones, ...
  - They were NOT general purpose embedded systems, but are now.
  - Limited resources
    - ?
  - Dedicated functionality
    - ?
  - Specific quality requirements
    - ?

Market

- Automotive
  - Engine, ignition, brake, stability
- Consumer electronics
  - TV, DVD, PDA, refrigerator, oven, camera
- Industrial control
  - Robot
- Medical
  - Monitor, analyzer, pump
- Networking
  - Router, access point
- Office automation
  - Fax, printer, copier, scanner

Market

- Defined in the following order
- Dedicated functionality
  - Each device is designed for a particular market.
- Specific quality requirement
  - Each device satisfies a particular needs.
- Limited resources
  - Resources are trimmed to save cost.

System Design

- Same as software engineering principles
- Big-bang
  - Beginners
- Code-and-fix
  - Small applications
- Waterfall
  - Large applications
- Spiral
  - For a continuous business
- Examples in the book

System Design

- Marketability
  - Who need it?
- Functionality
  - Does it provide what people need?
- Cost limitation
  - Can people afford it?
  - Does it only provide what people need?
- System's quality
  - How does it provide what people need?
- Performance limitation: computation power, memory size, battery life
  - Finally, we, computer professionals, start to design a system to answer the four questions.

System Architecture

- A system is composed of interacting elements.
  - Elements/functionality
  - Relationship/interaction
- Every embedded system has an architecture.
  - Why?
  - Except all good reasons, it is hard to apply the design of one system on another.
- Defining and understanding the architecture is the essential component of good system design.

System Architecture

- System model
  - Figure 1-2
  - Application software layer (required)
    - Application programs, common libraries
  - System software layer (optional)
    - OS, system libraries, system calls
  - Hardware layer (required)
    - CPU, memory, peripherals,