

Embedded Topics

Getachew Teshome
(120-B)
AAIT, Department of ECE

March 9, 2011

Topics

- Introduction to Embedded Systems
- Real-Time Systems
- Systems-on-Chip (SoC, NoC)
- Embedded Systems Design and Tools
- Design and Verification

What is an embedded system?

3

What does an Embedded System mean?

A computer System
Embedded within another system
And defining/ controlling its functionality.

Examples

- Phone, Camera, TV, Game Console
- Printer, Copier, Fax, ...
- Microwave, Refrigerator, AC
- Car Engine, navigator, ...
- Fly-by-wire, Missile guidance, ...
- Robots, Automation Units in Factories
- Toy cars, Game Consoles,


Application areas

1. Automotive electronics
2. Aircraft electronics
3. Trains
4. Telecommunication


6

Application areas

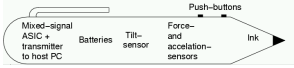
5. Medical systems
e.g. "artificial eye"



6. Military applications



7. Authentication



7

Application areas

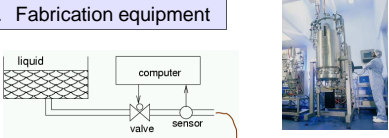
8. Consumer electronics




8

Application areas

9. Fabrication equipment



10. Smart buildings



9

- μ -processor
- μ -controller
- Embedded system

Characteristics of an ES









- Tight interaction with the embedding system.
- Real-time response to events.
- Dependable
 - Should handle exceptions properly.
 - Should be able to cope with failures.
- Concurrent processing.
- High performance.
- Power efficient.

Characteristics of Embedded Systems (1)

- Must be **dependable**,
 - **Reliability $R(t)$** = probability of system working correctly provided that it was working at $t=0$
 - **Maintainability $M(d)$** = probability of system working correctly d time units after error occurred.
 - **Availability $A(t)$** : probability of system working at time t
 - **Safety**: no harm to be caused
 - **Security**: confidential and authentic communication

Even perfectly designed systems can fail if the assumptions about the workload and possible errors turn out to be wrong. Making the system dependable must not be an after-thought, it must be considered from the very beginning

Characteristics of Embedded Systems (2)

- Must be **efficient**
 - Energy efficient 
 - Code-size efficient (especially for systems on a chip) 
 - Run-time efficient 
 - Weight efficient 
 - Cost efficient 
- **Dedicated** towards a certain **application**
 Knowledge about behavior at design time can be used to minimize resources and to maximize robustness
- **Dedicated user interface** (no mouse, keyboard and screen) 
- **Hybrid systems** (analog + digital parts).  

13

Trends

- Appear in every walk of life.
- 100 times PC market size.
- 25% annual growth.
- Accounts for 25-40% cost of a modern automotive.
- Must be dependable and at the same time affordable.

Application Types

	embedded	real time	safety critical	cost sensitive
▶ general purpose – PC, server, etc.	-	-	-	-
▶ embedded – consumer electronics	x	x	-	x
– automotive	x	x	x	x
– avionics	x	x	x	-
– military	x	x	x	-

Embedded Software

- TV, mobile phone, car: > 10 MLOC
- Code complexity is growing exponentially
- Number of bugs is growing exponentially
- Despite good SW eng'g ~10 bugs / KLOC
- 100 G\$ / yr on bug repair
- Embedded SW is difficult!

Embedded System Design

- Knowledge of the Embedding System.
- Hardware design
 - High performance (speed)
 - Less hardware cost (chip area, IO pins,)
 - Power consumption (may have to run for years on a single battery)
- Software design
 - Interactive
 - Real time
 - Efficient
 - Bug-free (testable with reasonable coverage)