

CS4220 Embedded Systems CS6235 Real-Time Systems

1A: Introduction

Instructor: Calton Pu
calton.pu@cc

(tentative) TA: Younggyun Koh (young@cc)

Intro Class: 1/10/2005

- Class format and content
- Projects
- Embedded and Real-Time Systems
 - Centralized and distributed
 - Why embedded and RT are “the same”
- Related courses
 - Advanced OS (6210)
 - Distributed Computing (7210)

Class Format

- Format of Class
 - Read chapters/papers before class
 - Lecture on the topic, perhaps beyond the paper
 - Discussions and interaction in the class
- Your Part
 - Reading papers and writing comments
 - Project: proposal, execution, demo
 - Participation in the class
 - Active thinking is encouraged

Class Goals

- Traditional learning
 - Concrete RTE concepts and techniques
- Problem solving in RTE
 - Case studies: how these concepts and techniques were used in problem solving (Pathfinder Mars Rover example)
 - Practice in your project
 - Project presentations: share your experience
- Learning to learn

Pathfinder Mars Rover

- Landing: July 4, 1997; initial successes
- Intermittent software system resets
 - Delay of mission, serious loss of data
 - Happens when “too much” data are sent over a shared information bus
 - Low priority data collection task locks the bus, gets interrupted by medium priority tasks
 - High priority data distribution task fails to complete: cannot get shared bus
 - Scheduler detects pending high priority task and resets all the hardware and software

RTE Techniques

- Problem modeling
 - Priority inversion: high priority task delayed in a critical section by low priority tasks
- Solutions proposed
 - Priority inheritance: low priority tasks entering critical section will inherit the highest priority of waiting tasks
- Solved the Pathfinder reset problem

Learning to Learn

- Acquiring information
 - Important goal: information from diverse sources
- Acquiring meta-information
 - General methods that apply to new situations
- Methods to acquire (meta-)information
 - Important goal in our fast-changing world
 - Very important for the Internet environment

Lecture Content

- Traditional real-time systems topics
 - Time constraints and guaranteed properties (e.g., schedulability)
- Traditional embedded systems topics
 - Resource constraints (e.g., optimized code)
- “Newer” topics
 - Quality of service, dynamic optimization
 - Mobile computing environments

Grades

- Comments on papers (20%)
 - Try to relate to it. Summary/abstract is OK.
- Project (60%)
 - Be interesting: something you are good at
- Student presentations (20%)
 - Leading class discussions
 - Project presentation
- A is about 70%

Projects

- Proposals (due early Feb)
 - Your own project (can be group)
 - Standard projects
- Examples
 - Write some system software components and/or applications
 - Standard project: evaluate hardware/software
 - Theoretical studies: wait-free synchronization