CG2271

Real-Time Operating Systems

Lecture 0

About This Course

colintan@nus.edu.sg

NUS
National University
of Singapore

School of Computing

Learning Objectives

- **Not a lot, just to give you an idea of who I am, what I do, and most of all what this course is about.**
- **Why?**
  - It's important for you to have a road-map that you can refer to when you can't connect the dots together. ☺

NUS National University of Singapore
School of Computing

About Me

- **The "garment" calls me: Tan Keng Yan Colin.**
- **I would prefer you to call me: Colin.**
- **Do not call me: Dr. Colin, Dr. Tan, Mr. Colin, Mr. Tan, Sir, Oi!, etc etc.**
- **I work at: COM2-02-08**
- **I can see you when: you email me to make an appointment AND you get a response from me.**
- **My email address is: colintan@nus.edu.sg**
- **You can call me at: 6516-7352.**
- **In my spare time I:**
  - Do research on artificial intelligence on embedded systems.
    - ✓**I have a really cool project on intelligent control of unmanned aerial vehicles. 100% Embedded Systems stuff!** ☺
  - Attempt to have some semblance of a life.

NUS National University of Singapore

School of Computing

About Me

- **Yes, I do (sort of) have a life outside of NUS. ☺**

NUS National University of Singapore
School of Computing

About Me

- **Sometimes I get a bit too adventurous:**

NUS National University of Singapore
School of Computing

What Is This Course About?

- **This course teaches you about Real-Time Operating Systems.**
  - Duh. We knew that from the course title!
  - But seriously; we're going to look at:
    - ✓**Hardware programming.**
    - ✓**Low-level microcontroller programming.**
    - ✓**Basic operating systems concepts like processes, scheduling, inter-process communication and coordination, etc.**
    - ✓**Construction and Customization of an RTOS.**

What Is This Course About?

- **This course is similar to CS2106 Operating Systems:**
  - We teach almost the same stuff in the second half.
- **But we're different!**
  - We're covering only the stuff that's relevant to embedded systems. So no virtual memory, etc, but we'll cover real-time scheduling algorithms like rate-monotonic scheduling.
  - We're looking at the topics from the point of view of microcontrollers.
  - We're covering topics like I/O programming and interrupt programming in MUCH greater detail.

What Is This Course About?

- **This is an introductory course, so it's more breadth than depth.**
- **However it prepares you for:**
  - CG3002 Embedded Systems Project (for CEG students)
  - CS3271 Software Engineering for Reactive Systems.
    - ✓**Fun course that teaches you how to engineer control software for aircraft, robots, cars, etc.**
  - CS4271 Verification of Embedded Systems
  - CS4272 Hardware/Software Co-Design
  - CS4273 Embedded Systems Design Project (compulsory for some CEC students. ☺)

Why Is This Course Important to You?

- **Embedded systems are the *most important* class of computers!**
  - Big Embedded Devices: Factory automation, oil refinery process management, aircraft autopilots…
  - Medium Embedded Devices: Smart phones, PDAs…
  - Tiny Embedded Devices: Your matric card, EZ-Link card, washing machine…
- **These systems often run sophisticated operating systems like MULTOS, MicroC/OS, LynxOS and RT-Linux!**
  - MicroC/OS is a Do178B compliant operating system: It can be used in critical applications like airplane autopilots!

NUS National University of Singapore
School of Computing

Course Map

- **This is a rough course map that is subject to change, covering 26 hours of lectures in 13 weeks.**
- **Lecture 1: Introduction (1 hour).**
  - What is a real-time system?
  - What is the Onion Model?
  - What is an Operating System?
  - How are RTOS different?
  - RTOS: An Overview.

NUS National University of Singapore
School of Computing

Course Map

- **Lecture 2: Input/Output Systems (2 hour).**
  - Basic hardware ideas:
    - ✓**Processor Design.**
    - ✓**Hardware input/output (I/O) design.**
    - ✓**Programmed I/O, Interrupt Driven I/O, Direct Memory Access.**

Course Map

- **Lecture 3: Microcontroller Programming (6 hours).**
  - Bit manipulation
  - Introduction to the Atmel Atmega328 Microcontroller.
  - General Purpose I/O Programming.
  - Analog to Digital Conversion.
  - Programming Timers.
  - Controlling analog actuators using PWM.

Course Map

- **Lecture 4 Real-time Software Architectures. (2 hours)**
  - Round Robin Architecture.
  - Round Robin with Interrupts.
  - Function Queue Scheduling.
  - Real-Time Operating Systems.

Course Map

- **Lecture 5: Task Management (4 hours).**
  - Task Models.
  - Task creation and destruction.
  - Task States.
  - Relationship between Task states and hardware.
  - Task Implementation.

Course Map

- **Lecture 6: Scheduling (2 hours)**
  - Fixed Scheduling.
  - Round Robin Scheduling.
  - Rate Monotonic Scheduling.
  - Earliest Deadline First Scheduling.

Course Map

- **Lecture 7: Inter-process Communications (6 hours).**
  - Race conditions.
  - Critical Regions.
  - Mutual Exclusion.
  - Producer-Consumer Problem.
  - Semaphores.
  - Monitors.
  - Queues.
  - Mailboxes.
  - Pipes.

NUS National University of Singapore
School of Computing

Course Map

- **Lecture 8: Non-RTOS Issues (2 hours)**
  - Threads.
  - Virtual Memory and Memory Management.
  - File Systems.

Course Structure

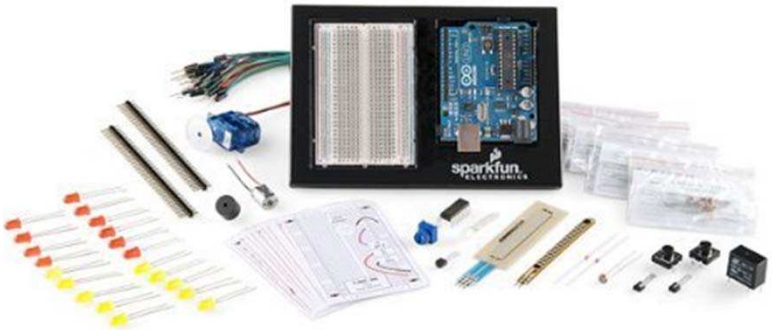
- **Lectures: 2 hours a week, total of 26 hours.**
- **Tutorials: 1 hour a week, 5% of course grade for participation, NOT ATTENDANCE!**
  - You will get a 0 if you are not able to answer a question.
  - This will affect your average grade for this component.
- **Labs: 2 hours a week, 15% of course grade.**
- **Mid-term test: 15% of course grade.**
- **Term Assignment: 15% of course grade.**
- **Final Exam: 50% of course grade.**
  - OPEN BOOK!

Other Course Information

- **Textbooks:**
  - Modern Operating Systems, Andrew S. Tanenbaum, Prentice Hall, ISBN-013-031-358-0.
  - Atmel Atmega328 Data Sheet from Workbin.
  - Embedded Real-Time Systems: Introductory Concepts and Tools. Graham Leedham, Kiam-Tian Seow, Pearson, ISBN 981-244-737-7.
  - An Embedded Software Primer, David E. Simon, Addison-Wesley, ISBN 020-161-569-X.
- **IVLE:**
  - As a matter of policy, teaching staff will not reply to queries on IVLE until 2-3 other people have at least attempted to reply.
    - ✓**This is to encourage discussion and interaction amongst you.**

Other Course Information

- **What you need for this course:**
  - Good knowledge of data structures.
  - Programming methodology.
  - The C Programming Language.
  - Ability to learn assembly language.
  - Basic mathematical skills.



- **Labs/Assignments**
  - Organize yourselves into teams of 2.
  - First lab begins in Week 3.
    - ✓**Each team will draw out a Sparkfun Inventor's Kit from Mr. Chan at the Help Desk. Check IVLE for details.**