

EECS 123
" Introduction to Real-Time Distributed Programming "
Winter quarter, 2008

Class Hours: Tu/Th 8am – 9:20am

Classroom: ICS259

Lab Sessions: Students taking EECS123 must enroll in one of the two lab courses which will be held in ET 421. :

Instructors:

Kane Kim, Professor

Administrative Assistant: Jim Collins x7814

Office: ET 544

Office Hours: Tue 2:00-3:20pm

Other times by appointment

E-mail: khkim@uci.edu

TA:

Jing Qian

E-mail: JINGQ@UCLEDU

Will be available in the lab for consulting during lab hours and also available in ET444 during 9:30 – 10am. Tu & Th, if appointments are made in advance.

Catalog Data:	EECS123 Introduction to Real-Time Distributed Programming (Credit Units: 4) W , (Design units 2). Introduction to the techniques for programming applications involving timing-sensitive actions. Hands-on experiences with object-oriented programming styles. Timing requirements, timing specification, response times, deadlines, application programming interfaces to real-time operating systems and middleware, remote procedure call, and distributed objects.
Textbook:	Tutorial notes on High-Level Real-Time Distributed Programming prepared by Kane Kim A selection of research articles and book chapters
References:	The TMO Programming Toolkit including a user manual and other research articles available from the Web
Coordinator:	Kane Kim
Course Objectives:	(1) Introduce the techniques for programming applications involving timing-sensitive actions; (2) Introduce the timing requirements inherent in certain applications and the established approaches for timing

specification;

(3) Introduce the techniques for measuring and analyzing the timings of computing actions;

(4) Introduce the techniques for object-oriented real-time distributed programming; and

(5) Introduce application programming interfaces to real-time operating systems and middleware and give students hands-on experiences.

Course Outcomes and Relationships to Program Outcomes:

(1) Students will be able to measure the amount of time taken for execution of program-segments (Program Outcomes 1, 2, 11, 13).

(2) Students will be able to specify timing requirements to be imposed on various parts of real-time computing programs (Program Outcomes 1, 2, 5, 11, 13).

(3) Students will be familiar with application programming interfaces to real time related parts of operating systems (Program Outcomes 2, 11)

(4) Students will be able to design and program real-time distributed computing application programs using C++ (Program Outcomes 1, 2, 3, 4, 5, 11, 13).

(5) Students will be familiar with the principles of real-time distributed object based design and programming (Program Outcomes 1, 3, 5, 11, 13).

Prerequisites By Topic:

Understanding of

(1) Basic computer architecture including input and output subsystems and interrupts,

(2) Data structures and object-oriented programming techniques, and

(3) Introduction to operating systems.

Prerequisite courses: EECS 111

Class Schedule:

Each class meets 3 hours per week and students are assigned to a 2-hour laboratory session.

Computer Usage:

Exercises in real-time distributed programming are a part of this course. Personal computers or workstations running widely used operating systems such as Windows are appropriate for use in this course.

Laboratory Projects:

Projects are aimed for providing to the students experiences for hands-on real-time programming. Five or six project assignments are given in this course.

Professional Component:

Contributes towards the Computer Engineering major requirements for Engineering Topics courses.
This is a technical elective course for Computer Engineering majors.

Design Content Description: Real-time programming activities are essentially design activities.

Approach: Lectures emphasize relevant design principles and applicable measures and constraints. Laboratory projects provide hands-on design experiences to the students.

Lectures: **50%.**

Laboratory Portion: **50%.**

Grading Criteria: The grade in this course is based on home-works (10%), laboratory assignments (20%), a midterm examination (20%), and a final examination (50%).

Estimated ABET Category Content:

Engineering Science: 2 credit units or 50%

Engineering Design: 2 credit units or 50%

Prepared by: Kane Kim

Date: 2003.01.02

Last modified: November 2005

**EECS 123 - " Introduction to Real-Time Distributed Programming
Winter 2008**

Class Schedule

1st wk	Course overview Background Quiz Introduction including characteristics of RT applications
2nd wk	Resource scheduling & time measurement TCP, IP, UDP, RPC, and relevant APIs
3rd wk	Global Time and Clock Synchronization Time referencing and calculation facilities in the TMO Approach
4th wk	Time-triggered actions in single node systems - TMOs containing SpMs only Config.ini file and a GUI tool for creating Config.ini TMOs containing SpMs and ODSSs
5th wk	I/O Handling 1 Midterm Exam
6th wk	Real-time Multicast and Memory replication Channel (RMMC) - Part 1 TMOs containing SvMs only including Remote TMO calls
7th wk	Further on Config.ini file TMOs containing both SpMs and SvMs I/O Handling 2 - Use of callback functions issuing one-way service-requests for SvMs
8th wk	RMMC - Part 2 (including RMMC-to-SvM and emulation of ORT) Visual Studio for TMO
9th wk	Virtual Machine for Auxiliary Threads I/O Handling 3 - Use of slave service functions and slave time-triggered functions
10th wk	DTS (Distributed Time-triggered Simulation) - an introduction
11th wk	Final Exam