Course: (Spring 1999)
Room 2161 CEBA
Monday Wednesday Friday 8:40–9:30
Call Number 1321
Web Page: http://www.ee.lsu.edu/ee4770
Prerequisite: EE 3750, Microprocessor Systems (or equivalent.).

Offered By:
David M. Koppelman
349 EE Building
(225) 388-5482
koppel@ee.lsu.edu
http://www.ee.lsu.edu/koppel
Monday and Thursday 13:30–16:00 (tentative office hours).

Teaching Assistant:
Jian Zhang
zjian@ee.lsu.edu
Room 150 EE Building, Desk G56.
Office Hours: Mon, Wed 10:30-12:00, Fr 10:30-12:30
Phone: 388-4835.

Graded Material:
40% Midterm Examination.
40% Final Examination. (Cumulative.)
20% Homework.
   About one assignment every two weeks.
   Lowest homework grade will be dropped.
Real Time System (RTS)
A computer-controlled mechanism in which there are strict timing constraints on the computer’s actions.

Examples:

- Automobile.
- Chemical reactor.
- Home bread maker.

Material to be Covered in The Course

Hardware:

- Sensors.
  For detecting light, temperature, etc.

- Conditioning circuits.
  For converting sensor output to a useful form.

- Computer-interrupt hardware.
  For getting the computer’s attention.

Software:

- Real-time software organization and features.

- Estimating timing of RT programs.

- Scheduling RT programs to meet deadlines.
Background and Prerequisites

Background Needed for Course

Prerequisite: 3750, Microprocessor Systems

Digital logic and computer organization.

Computer programming (no particular language).

Design and analysis of electronic circuits.

Types of Problems to be Assigned

Circuit design. (Design a circuit to meet some specification.)

Explain how a certain part works.

Write pseudocode to perform a certain function.

Types of Problems not to be Assigned

Laboratory projects.

Semester-length project.
Parts of a Real Time System

A RTS consists of four parts:

- *Physical process.*
  That which is controlled.

- *Sensors.*
  Observe.

- *Computer.*
  That which perceives and plans.

- *Actuators.*
  Act.

... for example, consider an anti-lock braking system...
Anti-Lock Braking System (ABS)

System that controls braking in a car, preventing wheel lock.

- Normally, surface of wheels move at same speed as road.
- Braking force can cause one or more wheels to slip or lock.
- Usually, one wheel will lock before the others.
- If ABS detects locking at a wheel it will reduce braking pressure to stop locking.

ABS as RTS

Physical process.

- Tire/wheel,
- brakes and brake hydraulic system,
- car and road, and
- driver.
- and perhaps the wind.

Sensors.

- Detect speed that wheels are spinning,
- force driver exerts on brake pedal,
- pressure of brake fluid, etc.
Computer

Hardware: Special *embedded* microprocessor:

Fewer components needed than general-purpose microprocessor and made to withstand vibration and temperature extremes.

System Software:

System runs without (computer) operator.
No computer terminal needed.
Easier (less hard) to predict timing of software.

Process-Control Software:

Reads wheel speed (and perhaps other data) at regular intervals.
Based on speed of wheels, detects if a wheel is locking.
If so, adjusts pressure of brake fluid.

Actuators

Brake-pressure valve.
Dashboard light.
Role of Parts of a Real Time System

A RTS consists of four parts:

- **Physical process.**
  That which is controlled by the computer for some productive end.
  *The thing the computer is controlling.*

- **Sensors.**
  Converts state of physical process into information (analog or digital).
  *Sensors see what’s going on.*

- **Computer.**
  Based on information from sensors, deduces state of physical process and issues commands to control the process.
  *The computer figures out what’s going on and issues commands to keep things running properly.*

- **Actuators.**
  In response to commands issued computer, modifies the physical process.
  *Carries out the commands issued by the computer.*
Other Example Real Time Systems

Washing Machine:

**Physical process:** (Presumably) dirty clothes, water, detergent, tub, agitator, etc.

**Sensors:** Water level, water temperature, control panel.

**Computer:** Embedded microprocessor.

Computer runs through pre-programmed cycles.

Might modify actions based on water temperature.

**Actuators:** Water valves, tub-rotation motor, and control-panel lights.

Aircraft Autopilot

Performs many functions, for example, maintaining level flight.

**Physical process:** Airplane, surrounding air, navigation radio sources.

**Sensors:** Airspeed, attitude, control-surface positions, control panel, etc.

**Computer:** Embedded microprocessor or general-purpose computer.

**Great care taken in writing software.**

**Actuators:** Hydraulics and servos for positioning control surfaces (ruder, flaps, etc.).
Complexity and Reliability

Complexity Range

Very simple: kitchen appliances.
Moderately complex: automobile engine control.
Most complex: aircraft control system, factory assembly line control system.

Managing the complexity of these systems is a major aspect of RTS design.

Safety Concerns

People’s safety depends on correct functioning of many RTS.

For example, aircraft control systems, automobile control systems, pharmaceutical-production machinery.
Reliability-Assurance Problem

Acceptable error rate must be very low.

Example: if an avionics system causes a plane to crash one out of a million landings then how many would die per year?

Testing cannot assure a sufficiently reliable system.

Example: How much would it cost to land an airliner one million times (to test a device)?

Solutions:

Use proven design methodologies.

Introduce new techniques slowly.

Design systems to be fault tolerant.

A fault-tolerant system can continue to operate properly despite faults.

Design systems to fail safe.

Failure will result in minimal damage.
Challenging (Hard) Part of Real Time Systems

- Writing Specifications for RTS
  For large systems this is harder to do than it sounds.

- Writing Software
  If can be difficult to ensure that timing deadlines are met under all circumstances.

- Testing for Bugs in Software
  Bugs could result in injury so cannot depend on customers to test product.

- Evaluating Reliability
  This includes software bugs, hardware failure, and specification errors.