

Project Description

CPRE 488 Embedded Systems Design
Spring 2009

Overview

Objectives. In this project you will build a prototype embedded system. You will exercise the skills and methods you have learned from the labs and lectures. You will use the design methodology in the textbook, which includes requirements, specification, architecture, component design, and system integration. You will also apply the knowledge and skills you learned on hardware design, software design and UML, component interfacing, performance profiling, performance analysis and optimizations, testing and validation, embedded OS, networked systems and other subjects.

Suggested Projects. You are given a list of suggested projects. You may also propose a project not listed here. Each project team should have no more than four members. Two teams may choose the same project and compete with each other. It is not encouraged that more than two teams work on the same project.

Schedule

Timetable. The following is a *suggested* working schedule. You are encouraged to move as fast as you can.

Week 11	Requirement analysis, specification and architecture
Week 12	Component design
Week 13	Component design/integration
Week 14	Integration and testing
Week 15	Integration, testing and demo

The demo is scheduled in Week 15 in your lab section.

Submissions. In the beginning of week 12, you should inform the professor and TAs your choice of project. In the beginning of week 13, you should submit a design document that includes the project requirement form, system architecture, hardware architecture and software architecture, and a draft system specification. You may revise those documents as the project progresses. All final documents are due with the demo, and only the final versions are graded. All documents should be submitted through WebCT.

Web Server. After you form a team with your project partners, the team should create a homepage for the project on the WebCT. The professor and TAs will help each team create a group account on WebCT.

Attendance Policy. Attendance is required through the project time. You will need to sign an attendance list in the beginning of each lab. Attendance will be part of the project grade.

Grading

The project weighs 30% in your lab grade or 12% in your overall grade. Your project grade is based on demo, documentations and source code. You should use UML to describe the system specification and component design. More details will be given in the progress of the project.

Suggested Projects

The following is a list of suggested projects. You may also propose your own project. To do so, write a one paragraph description of your proposed project and e-mail it to the professor and the TAs.

1. *Camcorder*: Write a VxWorks application to use the digital camera to record video (admittedly at a low frame rate). Have the camera periodically take pictures (at 5 frames per second or so) and encode the resulting pictures in MPEG format onto the compact flash card. The MPGE file should be playable on a Windows machine.

Risk level: Medium

Resources:

- a. [MPEG Home Page](#)
- b. [Xvid open source codec](#)
- c. [ffmpeg open source codec](#)
- d. [MPEG4IP open source MPEG4 streaming audio and video](#)

2. *Remotely controlled Webcam*: Use the board as a web server through VxWorks. Create an html through which a user may control the webcam. A user may connect to the board through a web browser, click a button to take a picture, and then view the picture in the web browser.

Risk level: Low

Resources:

- a. A simplest, sample Web Server program (will be put on web site).
- b. The JPEG encoder you used in the labs 4, 5 and 6.

3. *Mini Recording Studio*: Write a VxWorks application to record audio from the line in port on the audio codec, encode it as mp3, and save it to the compact flash card. Your application should also interface with your mp3 player so that you can play back the audio you recorded.

Risk level: Low

Resource: The mp3 player program in lab 8.

Resource: An external microphone with connection to the board will be provided to you.

4. *Internet Radio Player*: Write a VxWorks application to connect to a streaming MP3 source over the internet and play the stream using the board's audio codec and your MP3 player code.

To write such a program, you need to know the Internet Radio protocol. Shoutcast Protocol is a widely used one.

Risk level: Medium

Resources:

- a. [Internet Radio - Wikipedia](#)
- b. [Shoutcast Metadata Protocol](#)

5. *Internet Radio Station*: Use the board to provide Internet Radio service through Internet. Any Internet Radio program should be able to connect to the service. The music files are stored in MP3 format in the Compact Flash card.

Risk level: Medium.

Resources: see project 2.