Final Project Overview

In this lab assignment, you will do the following:

- Develop a significant final project individually or as part of a team.
- Demonstrate your project to the class and document your work in a professional final report.

The lab portion of this assignment is due by **April 28th**.
The documentation for this project is due by 5:00 p.m. on **TBD (tentatively 5/01)**.
Students should have their preliminary final project ideas formed by mid February.

This assignment is weighted as 25% of your course grade. This assignment is meant to give students the opportunity to explore an area which is of interest to them. For those students who would prefer final project suggestions, several ideas will be given.

Lab Details

Hardware, Firmware, and Software

The **Due Date** for all final projects is **April 28th**. By that day, each person or team must be prepared to demonstrate their final project hardware, firmware, and software, if applicable. In addition, schematics for the final project must be complete, and a copy of these schematics must be available during the final project demonstration. Project demonstrations will be limited to approximately 7-10 minutes per project, so the team members should carefully plan how they will demonstrate that they met the project goals presented to the class earlier in the semester. Each team member should be prepared to answer questions related to his/her contribution to the project. Demonstrations will take place inside the lab.

Basic Final Project Guidelines:

- Students may work individually or in groups of up to three. Students may want to work with other students who have complementary skills; however, since the goal of the class is to learn as much as possible, students should consider accepting responsibility for a component of the project which forces them to learn new skills.
- Final projects must be completed before the due date at the end of the semester. Choose a project which has several milestones. **Do not use an 'all or nothing' approach.** Make sure you have something working to demonstrate on the due date. Strongly consider showing incremental progress to the TA or instructor before the due date to guard against project failure on demo day.
- Projects must have sufficient complexity. Projects which are too easy will not be eligible for 'A' grades. There will be a difficulty factor and a quality factor associated with the grade for each project. Focus should be on electronics, firmware, and software; not on mechanical elements or packaging.
- Projects must include both a **new** hardware and a **new** firmware component; however, students may choose to focus more heavily on either the hardware or firmware aspects of the project. Remember that the 8051 has limited bandwidth and projects should be defined with this in mind. Projects may also include a software component which allows communication between a host computer application and the project hardware and firmware, but the project focus should be on the embedded HW/FW.
- Students/teams will need to make a brief presentation (~5 minutes) at the preliminary design review (PDR) around week 10 of the semester. This presentation will give a brief project overview, define project milestones, and describe project deliverables. Project teams will need to generate a final report and give a project demonstration at the end of the semester. Each student on a team will be expected to participate in the presentation and documentation activities. Final projects will be presented during the last class periods. Scheduling of demos will depend on the number of projects to be shown.
- Projects must not pose a safety risk to students. In addition, projects must not make significant noise or otherwise disturb other students during development and during the presentation.

As examples of past 8051 projects, student teams have developed a SCSI interface to a storage device, an MP3 player (difficult) with an interface to a compact flash memory card, an EPROM programmer, a graphic calculator, a tank battle game, an embedded multitasking operating system, a robotic checkers game, a mobile robot, a home security system, a universal billboard programmer, a feedback control system, a web-based remote temperature sensor, an electronic battleship game, an X-10 home control system, a bar code scanner, a MIDI file player, and an infrared remote control sound effect generator.

Suggestions for final projects include:

- Device Programmer: Design a programmer which can program a limited number of devices, such as a PLD, an EPROM or a programmable microcontroller, such as an 8751 or a Microchip PIC microcontroller. A beginning goal could be to program 1KB of an EPROM and then run an 8051 program from that programmed EPROM. More advanced goals could be to program multiple types of EPROMs, parallel EEPROMs, or programmable microcontrollers.
- Alternate Microcontroller: Design a project using an additional microcontroller, such as a Microchip PIC. This alternate microcontroller could communicate with the 8051 board built during the semester.
- Home Network: Design a system which will allow multiple devices to communicate using a protocol such as X-10 or RS-485. Each device on the system could be responsible for storing some particular pieces of data which could be accessed across the network. Alternately, each device could be responsible for controlling some element like a motor, and commands could be sent across the network to sequence the operation of multiple devices.
- Storage Device Controller: Design an interface which can communicate with an external device, such as a floppy drive or optical storage device. Bandwidth limitations of the 8051 should be kept in mind if a project like this is attempted. **Projects should focus on functionality rather than performance.**
- Web Page Server: Create a web page server that responds to TCP/IP requests coming through a serial port attached to a modem which is in turn connected to a network. Alternately, design a system which monitors some device like a thermometer, and sends an e-mail message across the network if temperature limits have been exceeded.
- Operating System: Design a cooperative multi-tasking operating system which allows multiple tasks to share processor bandwidth. For students with a background in operating systems, this system could be extended to support real-time functionality, or to support forms of interprocess communication.
- MIDI Controller: Design a Musical Instrument Digital Interface which communicates with MIDI-enabled music equipment. This device could sequence different electronic instruments, merge MIDI channel data and filter unwanted MIDI events, etc.
- Digital Control System: Design a control system using DACs and amplifiers to drive motors, and a feedback mechanism such as an optical encoder or position sensor with an ADC. Balance a bar on its end. Maintain the position of a object on which a varying load is placed.
- Plotter: Use stepper motors to control the movement of pens and paper and write a device driver to allow simple pictures to be drawn.
- USB Device: Design a low speed HID-class Universal Serial Bus peripheral. Perhaps use a USB interface chip or use an 8051 derivative which has an integrated USB interface. If this project is pursued, definitely consider a chip which integrates and automates as much USB functionality as possible. One suggested option is the Cypress EZ-USB 8051 derivative. (Note: Difficult Project)

As a suggestion, students should determine an area which they find interesting and then try to structure a project around that area. The goal of the final project is more focused on learning something new and building something significant rather than designing and building something completely practical. One can get ideas for a project by looking through electronic parts catalogs, embedded systems magazines, application notes/data sheets, and walking through electronics surplus stores.

Projects could include the use of graphic LCD displays, input devices such as keypads or computer keyboards, ADCs, DACs, Flash memory, FPGAs, PLDs/PALs, and/or motors. One option for projects requiring higher bandwidth is to use a drop-in replacement processor such as the Dallas DS80C320.

Stationary projects are preferable to mobile projects (robots, vehicles).

Final Report and Presentation

The **Due Date** for all final project reports is **TBD (tentatively 5/01**). Before 5:00 p.m. on that day, each person or team must submit a project report to the instructor. Guidelines for the report are given below.

1. The final report must represent a professional effort on your part. **Plagiarism will not be tolerated.** The report must be well organized, easy to read, free from spelling and grammatical errors, and must include page numbers. The report must be bound using a wire/plastic spiral or comb binding method which enables the report to remain open (flat) on a table without reader intervention.
2. The report must be typed on a word processor and must include a cover page. The cover page must include a descriptive title of the project, the team members' names, the course number and title, and the due date. The report must describe the project, and detail its design and implementation. Assume that the reader is familiar with the 8051 architecture and instruction set.
3. One possible way to organize the report follows: cover page; table of contents; introduction and overview; technical description/details, results/error analysis; conclusions; future development ideas for the project; appendices. Hardware, firmware, and software design (if applicable) must be detailed.
4. The details section should include small figures, diagrams, or code fragments if needed; however, large schematics and code listings should be placed in an appendix. All schematics and appendices must be titled and numbered, and referred to by number in the report. All schematics and graphics must be large enough and clear enough to be read easily.
5. Appendices should include copies of any special data sheets or references. URLs for web sites containing data sheets, application notes, and/or other specifications must be listed in the report.
6. A full circuit schematic must be included in an appendix. The schematic should be done with a schematic capture program such as Orcad, which is available in the lab. Don't put too much information on any one sheet. Label all signal names and include pin numbers.
7. Source code listings for project firmware and software must be included in an appendix. Code must follow general good coding standards. Useful, descriptive comments must be included in the code. Use blank lines and white space wisely in your code to improve readability. It is suggested that code listings are printed "2-up", with two pages of code per side of paper. Double sided printing is encouraged, as long as the resulting appendix is easy to read without requiring that the report be twisted 180 degrees to read successive pages of the code. Duplex and 2-up printing will save paper and result in a lighter and more compact document.
8. A project parts/cost list (bill of materials or BOM) must be included. This BOM should indicate the source, part number, and cost of items specific to the final project, not parts used for Labs 1-4.
9. Firmware, software, or hardware which is highly leveraged or copied from another source may be used if legal; however, credit **must** clearly be given to the original author/designer in code comments and project documentation. Student-created code and hardware will be counted more heavily in the project grade and should be clearly identified in the report. Company confidential information must not be included in the report, since these reports may be viewed by a wide population.
10. Each project team must provide the instructor an electronic copy of all project source code (both firmware and software), include files, makefiles, the final report and any other documents, such as unique data sheets and app notes. Teams may provide the files on CD (preferable), floppy, or by e-mail. If providing by e-mail, use WinZip to combine all your files before e-mailing the zip file to the instructor.
11. All reports submitted will become the property of the instructor, so if you would like a copy of your report materials, make a copy before submitting your report.
12. Each team member must participate in the final project presentation. At least one transparency showing a detailed block diagram of the project must be presented.
13. During the project demonstration, the project team should talk about design and implementation details. The team should discuss any major engineering challenges encountered during the implementation, and discuss what the team learned during the implementation.