A Design Project Approach to Microcontrollers

Paul D. Johnson
Padnos School of Engineering
Grand Valley State University

Introductory microprocessor and microcontroller classes often seem to be taught with a minimum of realistic applications and experience. Either the courses have no laboratory experience at all, or the laboratory exercises are far removed from practical consumer or industrial applications. A two course sequence has been developed in the Padnos School of Engineering at Grand Valley State University which is designed to ensure that the students have extensive hands-on experience in practical microprocessor and microcontroller applications. This experience will prepare them for the demands of industry, both in their co-op employment and in their post graduation employment. An important part of this experience is the use of extensive design projects emphasizing embedded control applications.

Background

It is possible to discuss the topic of microprocessors without the use of any actual hardware. Projects can be executed as software simulations only with no application to hardware or actual systems. Alternatively, hardware laboratory exercises can be introduced as simple procedures where individual and isolated microprocessor features are demonstrated. The Padnos School of Engineering at Grand Valley, however, is very closely tied to local industry and strives to provide an education which will prepare the student to meet the engineering design needs of the extensive manufacturing industry of west Michigan. The school requires one year of co-op experience of every student. Students work in a local industry heavily involved in automotive applications, office furniture and aerospace. As a result there is a great demand from both students and employers for an ability to design products with embedded control systems and to design manufacturing processes which use embedded control.

In order to meet these demands, a design oriented microprocessor sequence has been developed to ensure that students will have the skills to carry out the design of consumer and industrial control projects. Prior to taking the two courses in microprocessor applications, the students have had one semester of introductory digital systems and several analog electronics classes. This background allows the junior/senior level microprocessor classes to integrate both analog and digital design. Initial lab exercises in the first microprocessor course are traditional laboratory exercises designed to familiarize the students with the basics of the microprocessor, the software development tools and basic interfacing. After familiarity with these basic tools has been developed, however, the students are able to begin working on more complex projects which integrate and expand the skills which they have learned. The more advanced features of the microcontroller and peripheral devices which can be interfaced to the microcontroller are mastered through the medium of these projects.

Project Approach

Design projects are employed to introduce the student to a wide variety of skills involved in hardware and software design. The following design requirements are integrated into the various projects:

- Design specifications and requirements analysis
- Hardware and software tradeoff analysis
- Algorithms for software solutions
- Structured analysis of problem and structured flowchart development
- User interface development
- Sensor and actuator interfacing

Design projects are generally presented to the student as a textual description of the problem to be solved. Technical specifications are presented only to the extent necessary to ensure that the final solution will meet the design requirements. The students are responsible for developing the detailed specifications by talking with the user, i.e., the instructor, and by analyzing the system which is to be controlled. The class includes extensive discussions of software versus hardware solutions for various problems and the advantages and disadvantages of each solution. Software solutions are emphasized when there are alternative methods of carrying out a design.

Students are required to produce a structured flowchart describing the problem solution prior to actually beginning their coding. Top down structured analysis, as described by flowcharts, can greatly improve the student's ability to see flaws in their analysis of the problem and can make the actual coding of a program quite easy. The biggest difficulty is ensuring that students really do complete the flowchart prior to coding and that the flowchart is properly structured. There seems to be an almost irrepressible urge to begin writing program code before the problem is thoroughly understood. A freely shareable version of SnapGraphics™ is used for creating the flowcharts so that students can carry out the work on their own without purchasing an additional software package. (The software can be downloaded from Micrografx at http://www.micrografx.com.)

Since many of the projects require input from a user, the student must design a user interface. This generally consists of switches, keypads and encoders as inputs and LED or LCD displays for feedback to the user. The student is responsible for the interface design along with the software to control the external devices. Students are encouraged to make the interface intuitive and easy to use. Additional interfacing is often required to provide data from an external system and to provide control signals to that system. The devices used vary from year to year as there is not enough time for the student to get experience with every type of sensor. Typical input data comes from keypads, switches, optical couplers, encoders and temperature sensors. Output data is typically sent to LED and LCD displays, RS232 drivers, heaters and motor drivers.

Each of the projects requires the students to use manuals and data sheets from the manufacturers to determine software and hardware requirements. The students are expected to be able to use the data sheets from both analog and digital devices to determine interfacing requirements. At the start of the semester, each student is provided with a set of manuals for the microcontroller being used in the class. Other data sheets and manuals are made available as needed throughout the semester.

Design Project Examples

A number of projects have been used to ensure the students' mastery of the microcontroller applications. In recent years, all of the designs have been carried out using the Motorola 68HC11 microcontroller, but the examples can be easily adapted to other microprocessors and microcontrollers. The 68HC11 EVBU evaluation board from Motorola has been used to allow the students to develop their external circuitry either on the attached PC board development area or on a separate breadboard. Although the hardware is provided for student use, many of the students choose to purchase their own EVBU so that they can continue to experiment after the class is complete. The following are a few of the projects which have been used successfully to stimulate students' interest and improve their skills in developing embedded computer applications.

Traffic Control Systems

One of the oldest and simplest type of control problems involves developing a standard traffic light system. Students are assigned to develop a control system for a given intersection near campus and are required to develop their project based on an analysis of the traffic at the intersection. The project can be given varying degrees of complexity. At its simplest, the project may involve using delay loops to set the times and light the lights in the correct sequence. If the exercise is given later in the course, a programmable timer may be used to establish more accurate timing cycles. The traffic control system can also be demand driven with traffic sensors (generally interruptible optocouplers) used to control the traffic signal based upon the traffic volume. Demand driven turn lights may also be added. Control of railroad crossing signals has also been used as an alternate problem, with or without an adjacent automotive traffic light control system. This project is a useful one for stimulating discussions about interrupt driven versus delay loop timing of events.

Clock Design

Various 12 and 24 hour clocks have been used as design projects. The students are required to use the programmable timers of the 68HC11 to produce accurate timing of the clock. A portion of the grade is based upon clock accuracy, although it is essential to verify the accuracy of the crystal oscillator as well. The clocks must have a user interface that allows the clock to be easily set (fast and slow set features). An alarm clock feature has also been added in some implementations. This project has also been used to get students familiar with the requirements of producing a true embedded application that can initialize itself rather than relying on the operation of an underlying monitor program. The clock is required to operate independently of the EVBU monitor software and to carry out all required initialization of stack, registers, interrupts and memory on its own.

Motor Control

The control of DC motors is useful for discussion of high power devices from a microprocessor and for discussion of isolation techniques. Students are required to isolate the high power portion of their circuit from the processor by the use of optocouplers. A pulse width modulated output from the processor is integrated and amplified to provide smooth control of a motor. The motor speed may be set by external switches or by either an analog or digital potentiometer. In most cases the students have also been required to provide active feedback to the controller so that motor speed can be maintained steady under varying loads. Speed displays may be output from the processor and verified by an external speed sensor.

Oven Controller / Heating Cooling Thermostat

Several variations of thermal control systems have been used. A temperature sensor is interfaced to the A/D converter of the microcontroller and used to provide output signals to control an external heating and/or cooling circuit. A simple oven controller may be implemented by using the processor output to vary the intensity of a light bulb in an insulated container. The student must write a program to heat the chamber to the set temperature using appropriate proportional control to minimize overshoot while providing rapid heating. A less complex variation is a heating and cooling thermostat design where a simple on-off output signal is provided to the climate control system based on the thermal sensor input. Both designs require a user interface to allow the desired temperature to be set. This can be an analog or digital input and requires that the input data be displayed so that an accurate set point may be entered.

DC Voltmeter

A/D inputs can also be used to create a simple voltmeter from the microcontroller. The output can be sent to a set of 7 segment displays to provide a digital output or to a bargraph level indicator display using individual LED's.

Keypad interfacing

Keypad interfacing using software decoding and debounce algorithms can be very useful. Students are required to look at the switch bounce characteristics of the keypad and to consider possible changes in those characteristics during the lifetime of a product. The keypad may then be used to provide input data to the controller for other projects. The keypad interfacing provides an excellent opportunity to discuss alternative hardware and software solutions by comparing the lengthy software code required to a relatively easy to implement, but more expensive keypad decode integrated circuit.

Summary

Only a selection of the above projects can be given to a group of students each semester as the projects may require from 2 to 6 lab periods to complete. Extensive outside analysis and design are also expected from the students. Although the project approach limits the number of individual exercises that the students can complete, it provides an opportunity to integrate material from numerous courses from each of the engineering disciplines as well as to become comfortable with the use of data sheets and data books from various manufacturers.

PAUL D. JOHNSON
Paul Johnson received his BA from Houghton College (NY) and his MS and PhD from the University of Notre Dame. He has worked as an engineer for Allied Signal Corporation and is currently an Associate Professor in the Seymour and Esther Padnos School of Engineering at Grand Valley State University. Interests include environmental issues in engineering design, embedded controller applications and web site design.