

AC 2010-771: CONTROLLING A POWER SUPPLY VIA THE INTERNET PROVIDES A CAPSTONE DESIGN EXPERIENCE IN TOPICS OF APPLIED DESIGN

James Everly, University of Cincinnati

James O. Everly is an Associate Professor of Electrical and Computer Engineering Technology at the University of Cincinnati. He received a BSEE and MSEE from The Ohio State University in 1969 and 1970, respectively. He is a Senior Member of the Institute of Electrical and Electronics Engineers (IEEE) and a registered professional engineer in the state of Ohio. He is currently a member of the IEEE Cincinnati Section, and in 1997 he received the IEEE Professional Achievement Award. He has held several research and management positions in industry working for such companies as Battelle's Columbus Laboratories, Rockwell International, and Claspan Corporation. He joined the University of Cincinnati in 1985.

Delmer Nicholson, University of Cincinnati

Delmer W. Nicholson Jr is a Visiting Assistant Professor of Electrical and Computer Engineering Technology at the University of Cincinnati. He received a BSES and MSCS from The University of Cincinnati in 1982 and 2009, respectively. He has held several research and development positions in industry working for such companies as Cincinnati Milacron, Delco Electronics and Nicholson Computer Consulting. He has taught as an adjunct instructor in the Mathematics and also the Engineering Technology Departments at Cincinnati State Technical and Community College since 1989. He joined the University of Cincinnati in 2008.

Controlling a Power Supply via the Internet Provides a Capstone Design Experience In Topics of Applied Design

Abstract

This paper focuses on using an off-the shelf Programmable Power Supply Kit to illustrate the concepts required in successfully completing a junior level capstone project. The capstone Programmable Power Supply Project is structured to support course goals and objectives of "Topics of Applied Design" offered to junior students in Electrical and Computer Engineering Technology. The course is structured to meet the requirements of a Senior Design capstone experience with the exception that the project design is controlled by the faculty member. The paper presents the "project concept" together with the theory of operation, project schematic, and parts list. Each student is required to build a Programmable Power Supply (PPS) using a combination of through-hole and surface mount components. In addition, each student is required to implement a packaging concept, and write a complete project proposal. An additional goal of the course is to prepare students for the upcoming Senior Design Capstone Experience required for a baccalaureate degree in engineering technology. In support of this goal students are required to maintain a project journal to record ideas, data, test results, and experiences throughout the project. To date, student feedback on this project has been very positive and a representative selection of student comments is included in this paper.

Introduction

The course "Topics of Applied Design" is a three credit-hour lecture combined with a one credit-hour laboratory. The course is offered autumn quarter of the junior year to baccalaureate students in Electrical and Computer Engineering Technology. Minimum prerequisites for the course include three course sequences in electronics, electric circuits, and digital electronics. The primary focus at the beginning of the junior year is to introduce students to a course in applied design that utilizes a "capstone project" as a learning vehicle to bring together concepts previously introduced in both analog and digital electronics. In addition new technologies such as digital potentiometers, printed circuit board design, and surface mount parts are introduced. The objectives of the course include:

- Learning techniques that allow for an easy transition into the Senior Capstone experience by allowing students to experience the "Capstone Process" using a well-defined technical example.
- Reverse engineering the project and then developing a proposal to give a desired outcome. Learning to use a research journal to record ideas, data, test results, and experiences throughout the project.
- Prepare a poster board together with a ten minute technical presentation.

- Develop and implement one or more embellishments to the project. This is a natural part of a capstone project final report and allows students to offer possibilities for future development for others to consider.

The baseline project presented in this paper is the same as that presented by Lapsansky ^[1]. The paper describes a Programmable Power Supply (PPS) wherein the output voltage of the supply is controlled via a digital potentiometer and operational amplifier. A voltage divider samples the output of the operational amplifier which subsequently serves as an input to an analog-to-digital converter in the microprocessor. Students are not required to program microprocessors at this point but they are given a blank chip and asked to transfer the compiled program to the chip. The power supply firmware is written in about 100 lines of BASIC code and the microprocessor provides communication via an RS-232 serial link with a desktop computer running a Microsoft Visual Basic Application to select voltage values.

In the sections that follow the authors describe revisions to the printed circuit board (PCB) for use in an academic environment. In particular the power supply PCB has been redesigned by the authors in conjunction with David Tashjian ^[2], the departmental Systems Engineer, to allow for additional voltage regulators, and the ability to use either a digital or analog potentiometer.

As part of the course students are required to extend the project by adding additional features. This requirement is normally taken lightly by most students and the resulting improvements generally take the form of adding a power on/off indicator lamp and for a few students, adding either an analog or digital capability to monitor the power supply voltage and current.

Over the last four offerings of the course the authors have become frustrated with how to best motivate student to think “outside of the box.” Hence, a large portion of the paper will deal with slightly more sophisticated, but still practical, embellishments to the existing project. For example, the paper shows:

- How the RS-232 serial cable can be replaced with a 433 MHz radio frequency link between the desktop client software and the power supply firmware. This will allow the power supply to be placed over 300 feet from the computer.
- Two techniques for controlling the power supply voltage via an Internet connection are presented. This allows for the possibility of changing the supply voltage from different parts of the world.

Subsequently, two basic technical discussions will be presented dealing with both the hardware and software used in the project. This will be followed by descriptions of how project embellishments are implemented. Student evaluation based on completion of specific tasks both in the lecture and the laboratory is given. The use of a project journal will be discussed along with the concern by the authors that current students do not easily accept the “paper notebook” concept. The authors have surveyed the Web and discovered that this problem exists at many

institutions and some suggestions on software replacement will be presented. Finally, student feedback on the project will be presented. The next two sections consist of technical discussions.

Hardware Technical Discussion

This section describes the modifications to the Programmable Power Supply schematic to include features that the authors considered desirable for class room use. In addition, the unique technical feature of the original design is highlighted. To date a simple power supply like this would have been constructed using an LM317 voltage regulator which the authors introduce in Linear Electronics offered to second year students.

Modified Printed Circuit Board

Figure 1 shows the modifications to the programmable supply that were added to the PCB for classroom use. A full-wave power supply is provided for the application. One improvement is the addition of a 16 volt regulator which stabilizes variations in the output. Also, to the right of the digital potentiometer is a PCB provision for replacing the digital potentiometer with a ten-turn analog device. The analog potentiometer can be switched into the circuit in order to reinforce concepts presented in previous courses. New devices are also introduced and considerable time is expended in teaching students how to extract information from device data sheets.

Student Exposure to New Devices

The Texas Instruments OPA547 high-voltage/high-current operational amplifier is an ideal component for this project. The device has a high output voltage swing together with a high current output. The output current has a maximum of 750 mA with a nominal value of 500 mA. Also, the device is internally protected against over-temperature and current overloads.

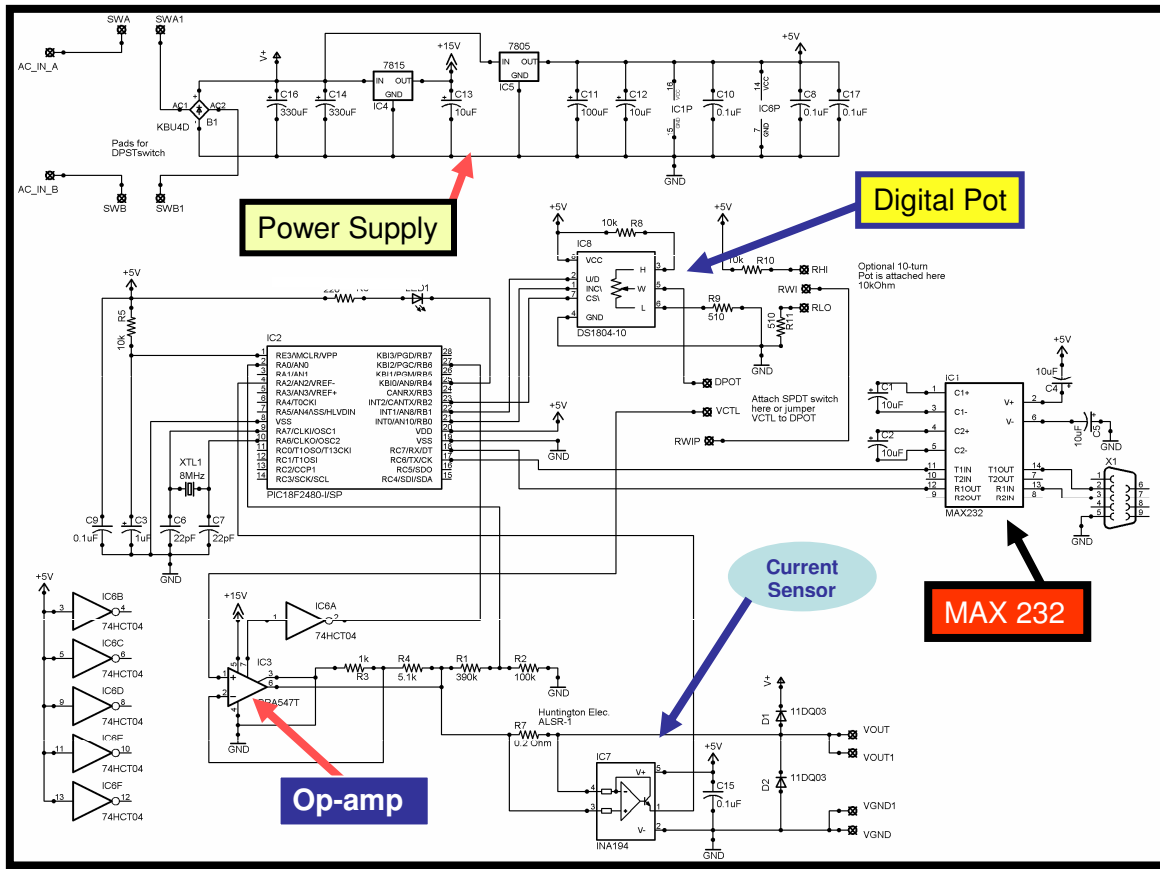


Figure 1– Revised Printed Circuit Board for the Programmable Power Supply

The use of a DS1804 digital potentiometer is a new device introduced in this course. The potentiometer has a value of 10K Ohms distributed over ten segments. This implies that the discrete potentiometer resistances produce a discrete output voltage for the supply output.

The last new device is an INA194 current shunt monitor. The device senses a voltage drop across a 0.2 Ohm shunt resistor and subsequently outputs the measured current as a voltage that is fed to one of ten A/D converters of a PIC18F252 microprocessor.

A Simple RF Hardware Embellishment

The industrial, scientific and medical (ISM) radio band is a small section of the RF spectrum set aside or reserved internationally for the use of RF electromagnetic fields for industrial, scientific and medical applications. Sensors operating in the ISM Band are used to collect data remotely for rebroadcast as an RF 433 MHz Data Acquisition System Radio. An RS232 signal serves as a common serial interface between these modules.

Data-Link Transmitter

The RF transmitter for this project is low power and occupies the frequency range from 433.05–434.79 MHz with a center frequency 433.92 MHz. The advantage of using the ISM Band is that no FCC license is required in order to operate the transmitter. This transmitter is available "off-the-shelf" ^[3] as a 433.92 MHz RF transmitter module and uses a modulation method known as "On-Off Keying (OOK)" which transmits data by simply switching the carrier signal on and off. The RF data-link serial receiver can be located up to 500 feet from the transmitter. A four pin RF module designated as TWS-BS3 is manufactured off-shore and is available at very low cost. The transmitter is operated from a 9-volt battery and is easily placed at a remote location for data collection. A photo of the data-link transmitter is presented in Figure 2.

RF Data-Link Receiver

At the heart of the data-link receiver is the RWS 374-6 receiver module operating at 433.92 MHz. The modulation mode is amplitude shift keying and the unit supports data rates up to 4800 bps. The receiver selectivity is -108 dB and the channel spacing is 500 kHz. At the heart of the data-link receiver is the RWS 374-6 receiver module operating at 433.92 MHz. The modulation mode is amplitude shift keying and the unit supports data rates up to 4800 bps. The receiver selectivity is -108 dB and the channel spacing is 500 kHz. This module can be used separately or integrated into a package that that supports RS232 serial interface. The module selected for prototyping is shown in Figure 3 and is available in kit form (K174) and retails for \$44.95. This unit supports receiver addressing and two selectable baud rates, namely: bps or 4800 bps. Communication synchronization is controlled by an AT89C2052 microcontroller. The kit contains a MAX232 interface chip and a 78L05 voltage-regulator. The module selected for prototyping is shown in Figure 2 and is available in kit form (K174)^[3] and retails for \$44.95. This unit supports receiver addressing and two selectable baud rates, namely: bps or 4800 bps. Communication synchronization is controlled by an AT89C2052 microcontroller. The kit contains a MAX232 interface chip and a 78L05 voltage-regulator.

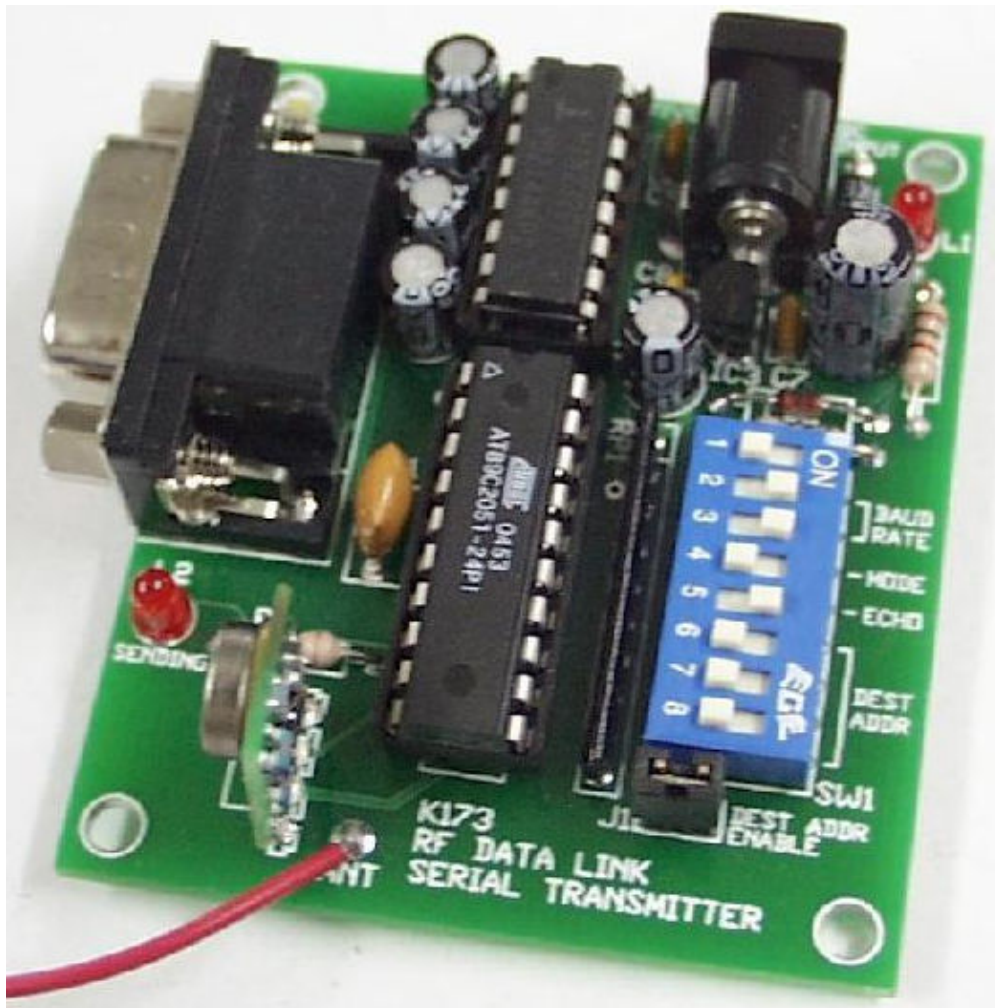


Figure 2– RF Data Link Serial Transmitter Supports the RS232 serial interface.

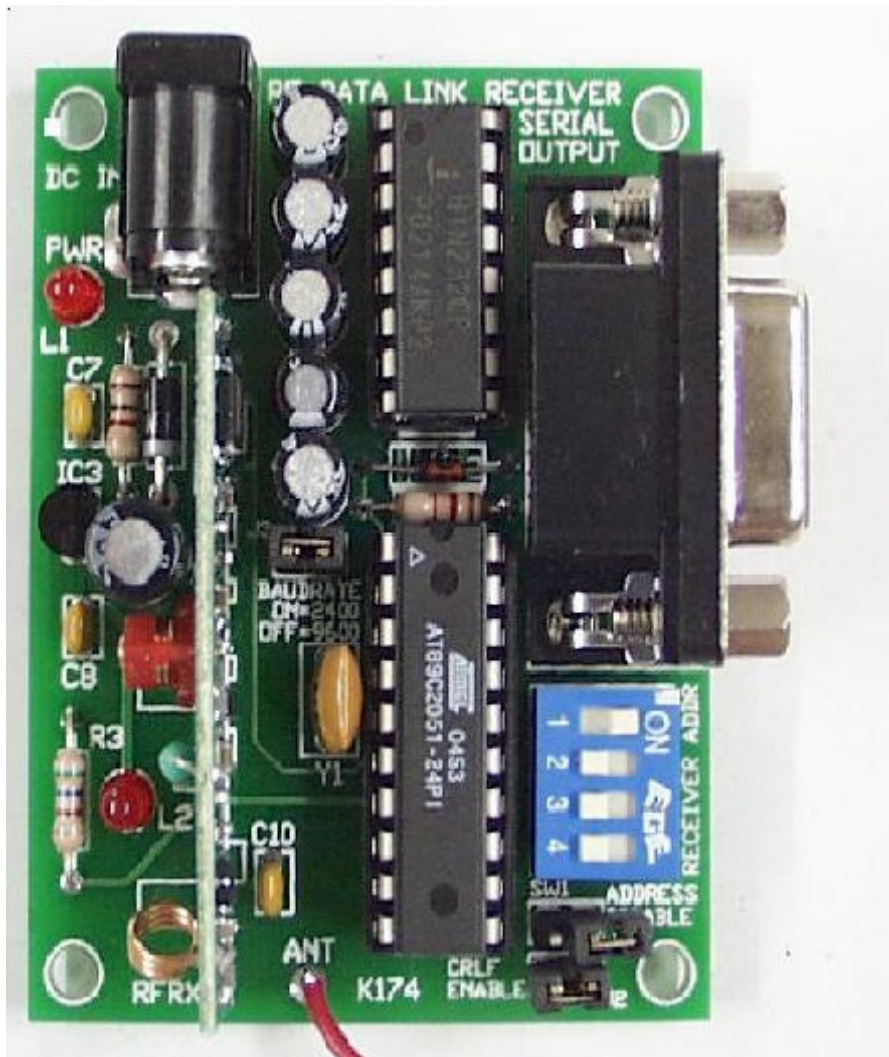


Figure 3– RF Data Link Serial Receiver

The software components associated with the Programmable Power Supply are discussed in the next section.

Software Technical Discussion

In this section the authors introduce the software components of the Programmable Power Supply project. They also discuss the technology employed to control the PPS via TCP/IP and across the Internet. They introduce the WIZ110SR Serial to Ethernet adapter and the LogMeIn Free software product. The Authors discuss the benefits to the students in Applied Design of studying the software components of the project.

Academic Goals and Details for the Software component of the Project

Several lectures were presented focusing on the software aspects of the project. As instructors, our student expectations were as follows:

- Recognize the necessity of developing software along with hardware.
- Understand the basic structure of a simple embedded software program.
- Acquire some experience in reading and reverse engineering a functional embedded software program.

The student workload for the software aspects of the project included a homework assignment to reverse engineer the firmware source code. During a preceding lecture, we went over the source code in great detail. We discussed macro functionality and we also presented line-by-line functionality for much of the program. Students took notes and asked questions. After the lecture they were given the assignment of writing a flow chart as a reverse engineering exercise. Instructor preparation time was significant, given that the original source code was thoroughly annotated and a Power Point presentation was prepared in advance of the lecture. The lecture was prepared with the assumption that students had little or no experience in embedded software. This preparation time can be amortized only if the same project is used in subsequent courses. If a project other than the PPS is used as the focus of the class, the presentation would need to be extensively revised.

Students were also assigned homework in which they proposed enhancements and modifications to the PPS as a whole. To this end we discussed in class the impact that specific changes might have on the embedded software. This discussion helped the students to focus on the impact that even what might appear conceptually as a simple functionality change could have on the software programming.

Student workload was reasonable for the software component of the course, given that they were asked to write a flow chart from an existing program that was only a page or two in length. The students were not required to have embedded systems software courses as a prerequisite for the Applied Design course. Therefore, the authors intentionally kept the workload light.

Software Components

The software components include a client program and a controller program. The two programs are written in different versions of 'BASIC' (Beginners All-purpose Symbolic Instruction Code). Both programs were provided by the author of the original programmable power supply project, Paul Lapsansky. The authors designed and implemented enhancements to the programs as an exercise and as demonstration projects for the students in Applied Design. The authors added enhancements to the firmware program and we rewrote the client-side program in Microsoft Visual Basic.

The authors' rewritten client-side program is written in Microsoft Visual Basic. We chose Microsoft Visual Basic because our programming courses in the College of Applied Science Electrical and Computer Engineering Technology department are standardized on Microsoft

products on the commercial side. We also emphasize Java proficiency on the open-source side. Visual Basic was chosen over C# because the code is easier to read and write by novice programming students. The tools for working with serial communications and TCP/IP communications are portable across several languages in the Microsoft.Net/ Visual Studio development environment. The client-side code is not complicated and could be translated to another high level language such as Java or C# by a student competent in that target language.

The original client-side program provided by the Lapsansky included functionality for adjusting voltage and current along with querying voltage. The authors duplicated that functionality in our version of the program. The authors also added some features such as TCP/IP communication capability, which is discussed below.

The authors used the PPS controller code provided by Lapsansky as an object lesson in their Applied Design course. The code was very effective as reverse-engineering pseudo code exercise, for illustrating embedded system programming concepts, and for discussing possible enhancements. Most of the students have had little or no exposure to embedded systems. The authors took advantage of our overhead projector in the computer lab to display the annotated source code. They presented the code from two different perspectives. First, they discussed the overall architecture of a general control program in an embedded system: the initialization, the infinite loop, and the clean-up. Finally, they went through the code line-by-line and discussed each line in detail.

Architecture of the PPS Control Software

The initialization component of the architecture includes configuring I/O ports, establishing values for variables, configuring the serial port, and writing a start-up message to the serial port for consumption by the client-side software. This code 'block' executes one time as a predecessor to the infinite loop. Many assumptions are made in this code regarding the mapping of input and output ports and the communication parameters of the serial port. The assumptions are common to a lightweight embedded system; we emphasized the sparse nature of the code as compared to a desktop application that might offer more flexibility due to the much larger pool of programming resources available in that environment. The student's typical programming experience consists of small programs with Graphical User Interfaces that they completed as assignments in programming courses. The authors felt that emphasizing the architecture of a program such as this would provide a good foundation for the Embedded Systems course that all students would be taking in an upcoming quarter. The authors also know from experience that most students select a Senior Design project that requires them to develop embedded system-type software "from scratch."

The control loop component of the firmware program executes repeatedly until power is removed from the PPS. The authors discussed the functionality of the control loop; poll a device for external commands and execute those commands. They also discussed the general functionality of closed-loop system and pointed out that our example code lacked such functionality. Since the hardware controlling the power supply output voltage is the digital

potentiometer, the firmware writes to the device, but does not make use of a feedback loop to adjust the output voltage.

The authors' example firmware does not include any 'power-down' code. In fact, there is no logic at all beyond the bottom of the control loop. The system is designed to continue executing the control loop until power is removed from the PPS. During the discussion of the firmware the authors brought out this point and we discussed the possibility of adding 'power down' functionality to the software. We solicited suggestions from students as to what functionality might be appropriate. Students suggested that data might be saved to EEPROM, the LED might be flashed in some manner, or the voltage might be forced to its' minimum value. The authors also noted that the control loop would have to be modified to include a power down command that would be received from the client software.

The firmware code is programmed in mikroBasic from the mikroElektronika Corporation^[3]. MikroElektronika provides the development environment as a retail product for 199.00 USD^[4]. A free trial version is also available. The free trial version is limited to building executable programs that are less than 2K of program words. The authors' modified version of Lapsansky's firmware builds to a footprint that is under that limit. They did not need to purchase the retail product and were able to provide students with a professional-level development tool at no cost to them.

The mikroBasic development environment includes features that are specific to the PIC that the authors used in our PPS. Instead of hand-codeing assembly language routines to perform port I/O, we were able to use the built-in features of the mikroBasic language to make development much more straightforward. The authors saw no useful purpose in reprogramming the firmware in another language or another development environment. Adapting another development environment would have been problematic in a course taught to students at this level of experience.

As developed by Lapsansky, the original client-side software was programmed in a dialect of BASIC. Since the authors rewrote the client-side application, they spent no time discussing the original code in class. The authors assigned as an exercise to the students the task of reverse-engineering the firmware source code. Each student was required to develop their own pseudo-code based on the annotated source code listing we discussed in class. This exercise obligated them to focus on a software program from a third party and decipher it. The authors' expectation is that the students will understand the value of a high-level design and apply their experience to their Senior Design project.

Client -Side Software Enhancements

The authors' enhancement to the client-side software was the capability of TCP/IP communication. To this end they added an Ethernet to Serial controller manufactured by WizNet. The WizNet WIZ110SR plugged directly into the serial port of the PPS. It also connected via Ethernet to a standard desktop PC. Their modified version of the PPS client

software opened a TCP/IP socket to communicate with the PPS. No modifications to the PPS hardware or software were necessary. This exercise exposed the students to another communication technology beyond the serial port originally used by Lapsansky.

Implementing the serial to Ethernet enhancement required modification to the authors' client-side program. They used TCP/IP programming classes provided in the Microsoft .Net development environment. The WIZ110SR includes a configuration program. For their development system they set up a very small network consisting of two devices; our laptop computer and the WIZ110SR, connected by a standard Ethernet cable. They worked to keep the system as simple as possible. They 'hard-coded' the IP address and subnet of each device, thereby avoiding the need for a DHCP server or a network router. Note: by default, the WIZ110SR is configured to read and write port 5000. Be sure your firewall(s) does not block that port.

Engineering Project Notebook

Each student is required to maintain an Engineering Project Notebook ^[11] in the laboratory to record the student's journey through the course with emphasis being placed on recording test data, transceiver construction notes, and final calibration and testing of the transceiver. Many students also use the notebook as a journal and often record their frustrations as well as their successes. Laboratory time is also used for informal discussion groups which deal with problems encountered during the building and testing phase of the project. The concept of "test as you build" is emphasized in the discussions and students share ideas and problems during group discussions. Also, each student is required to develop a simple "Gantt Chart" to show project progress during the building and testing phases of the project. A handout is given in lecture that describes the "Gantt Chart" technique for implementing the project schedule, but the actual development and updating of the chart occurs during the laboratory period. The chart is used as a project management tool and must be included in the student notebook.

Student Assessment

To date, qualitative and quantitative student feedback on the course structure and project has been very positive and representative selections of student comments are presented in the following student assessments.

The following comments ^[7] are extracted from course assessment forms developed by the authors. Four questions were presented to the students during the last class of Topics of Applied Design during summer/autumn quarter each academic year. Overall, the comments are very encouraging:

1.0 Did the class project illustrate the concepts presented in the course?

”The project followed the concepts that were introduced at the beginning of the class. We were expecting to build a power supply during the term and we

understood that the parts would be available so we could actually put it together. The instructor covered a lot of details about how the hardware components were selected and sourced and how packaging could be done. We also learned concepts regarding the software through reverse-engineering exercises and lectures in class. We did not get to write any software, probably because the parts were not available until the last few weeks of the class and we had to spend our time on assembly and testing. The software concepts that were presented in the class were interesting but we did not get to work on them as part of the project.

2.0 Was the class project effective in enhancing your technical skills?

"The project introduced the soldering of many different components onto a printed circuit board, which I had no prior experience with in the past. I was able to work on packaging and also on obtaining components for a project. The presentation at the end of the term was something I was nervous about but I am glad that I was made to do it. I had not done a poster before and it was helpful to see the posters of the other students in the class. The feedback from the instructor was good as we looked at each poster. I was exposed to the software components of an embedded system but I didn't get to spend much time on that because we were waiting for parts to come in and we didn't have power supplies to work with. When I did get my power supply built, my technical skills in circuit board troubleshooting were enhanced, with the help of the instructor and the lab assistants and my fellow students."

3.0 Was the class project rewarding?

"We could have accomplished a lot more if the parts had come in sooner. The instructor covered a lot of useful material about component sourcing, analog circuit components, and packaging, but we needed more hands-on activities. It was very interesting to see the power supply come together once the parts came in. Creating a poster and giving a presentation were things that I didn't want to do but I know they are important for engineers to be able to do."

4.0 Rate this course overall, based on its effectiveness and helpfulness in utilizing your past coursework experiences and preparing you for senior design.

"This course was interesting to attend and I appreciate all the information that the instructor provided during class. It did prepare me for Senior Design, I think, because we will be doing circuit design and probably some software. Senior Design has poster presentations and the Tech Expo also, so we will have to know how to do that. I didn't have much experience in software in

embedded systems so that part of the course did not build on anything I have seen before. We will be writing software at some point in our careers, I think.”

Conclusion

This junior level course relies heavily on knowledge and skills previously learned in both the analog and digital electronic sequences at the University of Cincinnati. Only a modest amount of mechanical construction is required for the project. On the other hand a reasonable amount of soldering thru-hole components, electronic testing, and troubleshooting are required by each student. Also, calibration and testing skills are required for successful completion of the project. Over the past four years the authors have found that student attitude towards the course and subsequent feedback to be most rewarding, and I was very encouraged by the level of student involvement in the project. In addition, the effort expended in hardware/software development and testing offered students an opportunity to gain a “broader technical view” of project development.

Bibliography

- [1] Lapsansky, Paul, “Remotely Programmable Power Supply,” *Nuts and Volts Magazine* , pp. 39, July 2008. retrieved from the World Wide Web March 5, 2010, http://www.nutsvolts.com/index.php/?magazine/article/remotely_programmable_power.
- [2] Tashjian, David (private communication), September 30, 2009.
- [3] “Serial Data link Transmitters, and Receiver”, retrieved from the World Wide Web March 5, 2010, http://www.electronickits.com/kit/complete/RF_Transmitters/k173.htm.
- [4] “Sparkfun Electronics,”retrieved from the World Wide Web March 5, 2010, http://www.sparkfun.com/commerce/product_info.php?products_id=9476 <http://www.mikroe.com/>
- [5] “Log me in free software” retrieved from the World Wide Web December 10, 2009, <https://secure.logmein.com/products/free/>
- [6] Portland State University, *Engineering Notebooks*, Retrieved February 1, 2009, from the World Wide Web: http://web.cecs.pdx.edu/~warren/Capstone/index.cgi?PAGE=engineering_notebook
- [7] “Topics of Applied Design Student Survey” University of Cincinnati, data collected September 4, 2009