Digital Communication Systems Education via Software-Defined Radio Experimentation

Alexander M. Wyglinski, Di Pu, Daniel J. Cullen
Wireless Innovation Laboratory
Department of Electrical and Computer Engineering
Worcester Polytechnic Institute
Worcester, MA 01609-2280, USA
alexw@ece.wpi.edu

Abstract

In this paper, we present an educational approach that employs "hands-on" software-defined radio experimentation in the instruction of digital communication systems theory to electrical and computer engineering students. By prototyping and evaluating actual digital communication systems capable of performing "over-the-air" wireless data transmission and reception, students enrolled in this course can attain a first-hand understanding of the design trade-offs and issues associated with these systems, as well as gain a sense of actual "real world" operational behavior. Given that the learning process associated with using an actual software-defined radio platform for prototyping digital communication systems is often lengthy, tedious, and complicated, it is difficult to employ this tool within the relatively short duration of a typical undergraduate course, e.g., 7-14 weeks. Consequently, we have devised a method where the laboratory experiments have been preassembled with examples provided in order to enable the student to successfully implement designs over a short period of time. Moreover, software-defined radio design software packages were selected based on their ease of use and their flexibility in implementing various digital communication systems. Initially, a combination of MATLAB, Simulink, GNU Radio, and GNU Radio Companion were employed in the laboratory experiments. However, by the second offering of this course and its associated laboratory experiments, all the laboratory material was migrated over to Simulink in order to provide a convenient, efficient, and easy-to-use graphical experimentation environment, including the use of a recently released collection of blocks capable of controlling software-defined radio platforms for "over-the-air" transmissions.

Introduction

Modern society is increasingly becoming dependent on digital communication systems in order to function properly, with a growing number of applications relying on these devices, e.g., personal health/body networks, defense/homeland security, navigation/localization, social networking, vehicular transportation. In order to provide electrical and computer engineering (ECE) undergraduate students with a solid foundation in digital communication theory while simultaneously enabling them to synthesize several fundamental concepts taught in class with the latest communication systems technology, an ECE undergraduate course was proposed, developed and instructed at Worcester Polytechnic Institute (WPI) that focused on a "hands-on" experimentation approach using both computer simulations and programmable radio prototyping hardware, i.e., software-defined radio (SDR). This course is entitled: *ECE4305 - Software-Defined Radio Systems and Analysis*.

The use of SDR technology can enable students to obtain valuable insights on the practical implementation issues involved in performing real-time "over-the-air" communication experiments while concurrently relating their design experiences to abstract theoretical concepts[7,8,9,10,11,12]. Ultimately, this course was devised to give students a systems-level understanding (i.e., breadth) of a modern digital communications device while focusing on several key aspects (i.e., depth) in the design and implementation of these systems. However, learning how to use an actual software-defined radio platform for prototyping digital communication systems is often a lengthy, tedious, and complicated process. As a result, this technology is usually not employed within the undergraduate curriculum due to the relatively short duration of a typical undergraduate course, e.g., 7-14 weeks. Conversely, if the laboratory experiments and SDR modules have been pre-packaged for the students enrolled in the course, this would allow for a controlled exposure to SDR experimentation while simultaneously enabling the hands-on learning process of several digital communication system concepts.

In this paper, we present the development, structure, and organization of an undergraduate-level digital communication systems course, including details on the "hands-on" laboratory component that employed both computer simulation and SDR experimentation. In particular, several software packages used in the design and development of SDR systems, as well as their application to the laboratory experiments via a collection of preassembled modules, will be discussed. Although the software packages initially used in this course included both commercially-available (MATLAB[2], Simulink[1,4]) and open-source (GNU Radio, GNU Radio Companion) programs, the laboratory experiments gradually evolved towards a computer-based graphical experimentation environment that is entirely implemented in Simulink. The reason behind this evolution is primarily the result of a recently released collection of Simulink blocks capable of seamlessly controlling software-defined radio platforms[5,6].

The rest of this paper is organized as follows: A brief description of the undergraduate-level ECE course on digital communication systems employing the SDR-based laboratory experiments will be presented, followed by a detailed overview of the laboratory workflow applied to each of the experiments. A description of the software packages used in the laboratory experiments to design and control the SDR platforms will then be presented, followed by a discussion of lessons learned during the offering of this course as well as a description of future work activities to further refine and enhance the laboratory experience. Finally, several concluding remarks regarding this educational approach are presented.

An SDR-based Undergraduate Course in Digital Communication Systems

The inclusion of a digital communication systems course that emphasizes a "hands-on" laboratory component would significantly assist ECE undergraduate students pursuing a concentration in signals, systems, and information (SSI) to synthesize many of the concepts taught in class by realizing them in actual prototype systems and conduct real-time "over-the-air" digital communication system experiments. Given the complexity of these modern digital communication systems, especially those based on SDR technology, it was anticipated that the students will not only employ the fundamental concepts from courses within the SSI concentration but also integrate fundamental concepts from other concentrations, such as computer engineering, electromagnetics, control theory, and analog/RF circuit design. Consequently, this course would ultimately give students a systems-level understanding (breadth) of a modern digital communications device while focusing on several key aspects (depth) in the design and implementation of these systems.

To ensure that all the students possess the necessary background knowledge for taking a senior-level undergraduate course in digital communication systems, the following prerequisites were specified:

- A basic understanding of various fundamental communication systems engineering principles, such as the manipulation of amplitude, phase, and frequency for the representation of information (e.g., AM, PM, FM), synchronization and phase-lock loops (PLLs), and equalization.
- A basic understanding of probability theory is order to quantitatively assess the theoretical performance of communication systems operating in the presence of noise and other random sources of interference.
- Familiarity with Simulink or any general programming language, especially if it is graphical, is needed since the implementation of the digital communication system experiments will all be conducted in Simulink.

In addition to these prerequisites, it was also suggested to interested students that having some background knowledge in basic wireless communications and networking would be beneficial, especially since several of the laboratory experiments conclude with the application of the digital communication system to a wireless access scenario. Thus, understanding how digital communication systems can operate amongst each other within a wireless environment would provide the student with a clearer vision of how their experiments fit in the larger scheme of things.

The course consists of 26 lectures (4 per week) that are each 50 minutes in duration. The lecture materials are provided to the students prior to each class via the course website and cover the following topics: SDR technology, digital signaling and transmission, error performance in the presence of noise, receiver structures, multicarrier modulation, spectrum sensing and identification, and cognitive radio. The course material covered in class is closely synchronized with the collection of five SDR-based laboratory experiments, which are each scheduled once a week for a single 3-hour supervised laboratory session. Laboratory experiments are conducted via teams consisting of two students each. Finally, a course design project is conducted throughout the entire course by the same team of two students, where the teams are responsible for the design and implementation of a digital communication system solution to an open-ended communication problem.

Regarding the expected learning outcomes for this SDR-based digital communication systems course, students who successfully complete this course should be able to:

- Implement an end-to-end wireless data transceiver capable of performing “over-the-air” digital transmission.

- Understand the advantages and limitations of available digital technology when constructing SDR systems.
- Compare symbol and bit error probabilities of common digital communication systems in additive white Gaussian noise (AWGN) channels between theoretical, computer simulated, and empirical experimentation.
- Implement the optimum receiver structure for digital transmission through an AWGN channel.
- Devise high-speed wireless data transmission approaches based on the concept of multicarrier modulation.
- Understand how to characterize and identify various digital transmissions based on their frequency representation.

Evaluation of student understanding and mastery of the course material is handled via a combination of the laboratory experiments and the course design project, namely:

- Five pre-laboratory assignments that should be completed prior to the start of the laboratory experiment in order to help the students prepare for the material to be covered. These assignments consist of both an analytical and a Simulink component, and each submission is expected to be entirely the work of each laboratory team.
- Each laboratory team is expected to present a 15-minute group demonstration to the course instructor prior to the start of the next laboratory experiment session. Each team member is expected to equally participate in the demonstration by thoroughly explaining the team's experimental implementation, demonstrate its operation and functionality, and provide clear, concise answers to questions posed by the course instructor.
- A comprehensive laboratory report is expected for each of the five laboratory experiments. Reports should include a description of all steps taken throughout the experiment, observations of the performance and behavior of the communication system, answers to the questions located within the laboratory handout, and experimental results demonstrating the operation of the system.
- An open-ended course design project that involves the design and implementation of a SDR system incorporating and synthesizing many of the topics covered in class will be conducted by each student team,. The final design and implementation are to be documented in a final report that is due on the last day of class, and 15-minute laboratory presentation/demonstration to be given during the last two days of lecture to the rest of the class and the course instructor. The problem to be solved via the course project will be announced at the beginning of the course, with 2-page proposals due within a couple of weeks from the time of announcement. A midterm review with the course instructor is also conducted to ensure that steady progress is being made towards the completion of the project.

Note that to ensure the students are attending each of the lectures, as well as keeping up with the material, a short 5-minute quiz is administered to the entire class at the beginning of each lecture. Although the overall percentage value of these quizzes is nominal, they are intended to keep the class engaged with the course material covered during each lecture.

Regarding the topics for each of the five laboratory experiments conducted throughout the duration of the course, they include the following:

- *Digital Signaling and Data Transmission*: Inphase/quadrature and magnitude/phase representations of digital transmissions, waveform/vector representations, deterministic and probabilistic signals, noise.
- *Error Performance of Digital Communications in Noise*: Modulation/demodulation, bit error rate, error bounds, forward error correction coding, performance comparison between various modulation schemes.
- *Receiver Structures*: Correlator realization, matched filtering realization, orthonormal basis functions, Gram-Schmidt Orthogonalization, coherent/non-coherent receivers.
- *Multicarrier Data Transmission Techniques*: Orthogonal frequency division multiplexing, cyclic extension, peak-to-average power ratio, "waterfilling" concept and power allocation, adaptive modulation.
- *Spectrum Sensing and Identification*: Energy detection, matched filtering implementation, cyclostationary implementation, radio frequency characterization, opportunistic wireless access.

Proposed Work Flow of Laboratory Experiments

SDR is versatile tool for both education and development of digital communication systems. However, many curricula in communication systems engineering still lack the "hands-on" experience needed to highlight some of the numerous practical considerations that occur during the design and implementation process. Moreover, the complexity associated with many SDR platforms combined with intricate user interfaces makes this tool difficult to use by beginner communication system engineers.

Consequently, the SDR-based digital communication systems course presented in this paper consists of a conventional digital communication system theory component synchronized very closely with five "hands-on" experiments and a course design project. As a result of this close coupling between theory and practice, the students will be provided with a solid theoretical background in digital communication systems, which can then be readily used in the implementation of actual transmission systems based on SDR technology. In turn, these implementations will reinforce many of the concepts taught in class, as well as illustrate many of the non-ideal, realistic phenomena that occur during the transmission and reception of wireless data.

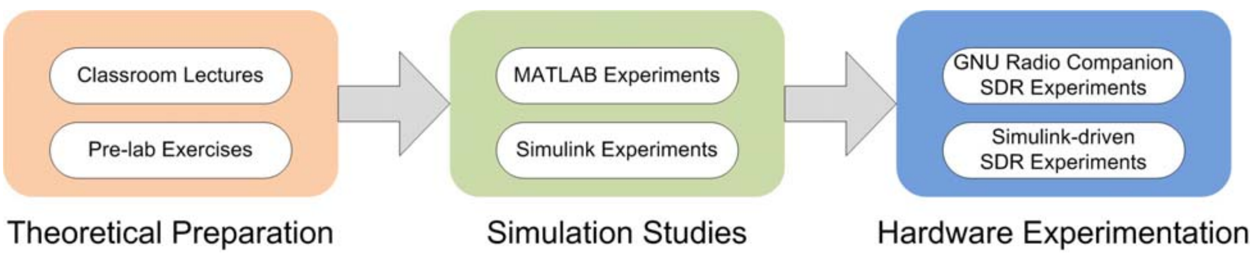

Figure 1: Work flow of the senior undergraduate level digital communication systems course employing SDR experimentation.

To overcome the significant learning curve that is often associated with the initial usage of SDR technology, each laboratory was devised using the work flow shown in Figure 1, where the necessary skills required to successfully complete a laboratory experiment is gradually built up. Specifically, each laboratory experiment starts off with the theoretical preparation and the pre-laboratory exercises in order to ensure that the students are familiar with the key concepts before they commence with the actual implementation and experimentation process. Once the theoretical foundations have been established, the students are then responsible for designing digital communication systems and evaluating their implementations within a computer simulation environment such as Simulink. The rationale for the computer simulations is to ensure the students observe the operational behavior of a digital communication system within a controlled environment before conducting real world "over-the-air" experiments, which possess numerous non-ideal characteristics.

Finally, once the skill set of the students has been reinforced by the combination of theoretical preparations and simulation studies, they are ready to commence experimentation with digital communication systems operating within actual wireless environments. At this stage in the laboratory experiment, the students should be familiar with the behavior of the digital communication systems that will be implemented, which will greatly assist them when debugging their designs. Moreover, at this stage in the laboratory experiment, the students would be exposed to software tools that would enable them to utilize the SDR platforms. Given the recent migration to an "all-Simulink" laboratory environment, the ease of use associated with this software will help all the students quickly surmount the learning curve of this technology.

Hands-On SDR Experimentation

SDR platforms are increasingly being employed in educational, research, and development activities that focus on the design and evaluation of digital communication systems due to their ability to facilitate rapid prototyping and evaluation of new transceiver designs and architectures. Nevertheless, not all SDR platforms possess the same capabilities and characteristics, and factors such as design flexibility, user accessibility, platform cost, and radio frequency transmission capabilities usually define which platforms are suitable for a specific application or activity. In the case of an undergraduate course with an SDR-based laboratory component, some of the key factors that will influence the selection of a particular SDR platform are cost and user accessibility. The former factor is important since universities usually possess limited budgets and a significant number of SDR platforms are required in order to fully equip an undergraduate laboratory, e.g., 14 SDR platforms were acquired for the course described in this paper. The latter factor is equally important since it is essential to keep the learning curve for all the students enrolled in the course to a minimum due to limited time frame in which the laboratory experiments can be conducted. Thus, when selecting a suitable SDR platform, both the hardware and software should be able to facilitate a high level of user accessibility.

In terms of SDR hardware, one candidate platform that ended up being selected due to its reasonable cost and relatively decent level of user accessibility is the *Universal Software Radio Peripheral – Version 2* (USRP2) by Ettus Research LLC. Figure 2 shows the front panel of a USRP2 platform[3]. This SDR platform depends on the computational horsepower of a host computer workstation for executing software implementations of the baseband radio functions, while the USRP2 itself acts as an actual peripheral that converts the baseband digital signal into an analog waveform that is then upconverted and modulated at radio frequencies. Note that a field programmable gate array (FPGA) is also available onboard each USRP2 in order to perform filtering and sampling functions on the incoming and outgoing data streams.

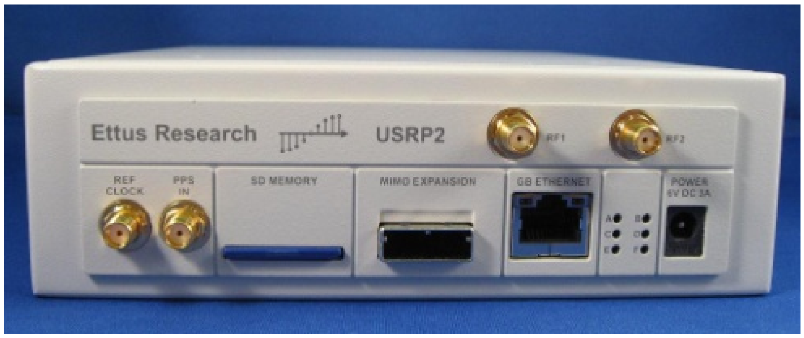

Figure 2: Front panel of a Universal Software Radio Peripheral, Version 2 (USRP2) SDR platform.

The USRP family of SDR platforms was developed in conjunction with an open source software package called GNU Radio since 1998. The GNU Radio software combines both C++ and Python in order to connect signal processing blocks via a flow graph to create baseband digital communication system designs. The signal processing and digital communication blocks are implemented in C++ in order to achieve the best possible performance. Moreover, these functions are wrapped together using a SWIG interface that enables them to be tied together using Python. In Python, a flow graph is constructed by connecting several of these blocks together. Hierarchies of blocks can also be created for more complex systems by creating flow graphs with inputs and outputs, as well as using it as a single block higher up in the hierarchy.

Although this software package possesses a highly degree of flexibility, and its abilities are closely coupled with the USRP2 hardware platform, there are several significant disadvantages. One of the primary disadvantages of GNU Radio is the amount and quality of the documentation. Although a graphical front-end for GNU Radio is available, which is referred to as GNU Radio Companion (see Figure 3), the quickly changing codebase due to an active open source contributor community has ultimately resulted in the software documentation being either unclear or even non-existent. In fact, the code is largely “self” documented by sparsely commented code. More documentation would make the use of this software easier for users new to the platform, especially for students who are trying to develop digital communication systems using GNU Radio. However, the steep learning curve associated with GNU Radio is not only difficult for people familiar with digital communication systems, but also students who are relative novices in the area of SDR experimentation. Thus, there is a need for a commonly used, commercially documented, SDR platform that would serve as an ideal tool for both development and learning.

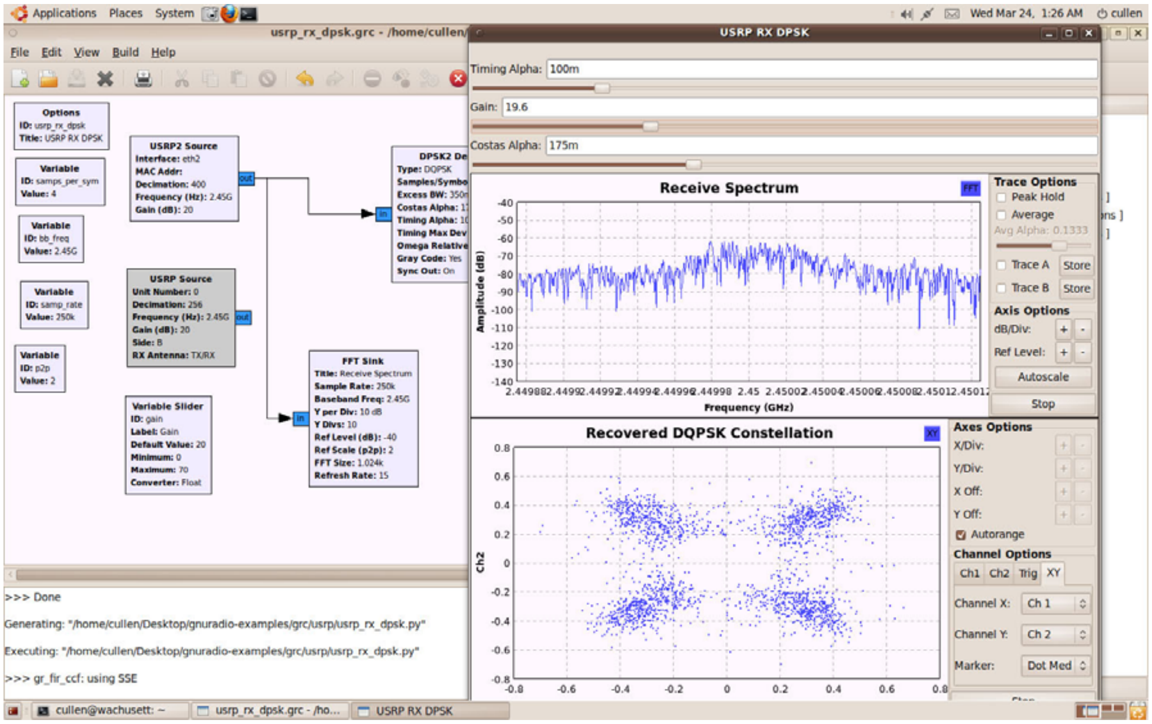

Figure 3: Screen capture of the GNU Radio Companion software (i.e., a graphical front-end to GNU Radio) employed in a digital communications experiment using the USRP2 as an actual wireless receiver.

Leveraging the existing, user friendly, graphical programming environment of the Simulink software package, a collection of Simulink blocks were recently developed that can be used to control and access any USRP2 hardware platform attached to the host computer workstation. These Simulink blocks, which consist of a transmitter block (see Figure 4) and a receiver block (see Figure 5), have been part of the Communications Blockset since the release of MATLAB 2010b. Providing full control over the USRP2 and fully using the capabilities of the hardware is the objective of these interface blocks. The basic controls need to include center frequency, signal power, and gain and interpolation and decimation rates. Providing these controls in Simulink requires tunable parameters or additional input ports for each of these properties.

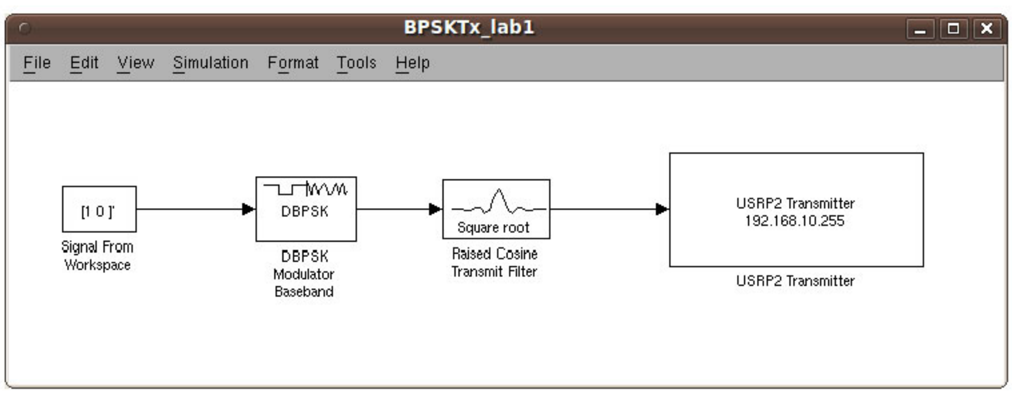

Figure 4: Sample Simulink implementation of a DBPSK transmitter employed in Laboratory 1 of the digital communications course. Notice the use of the Simulink USRP2 transmitter interface block.

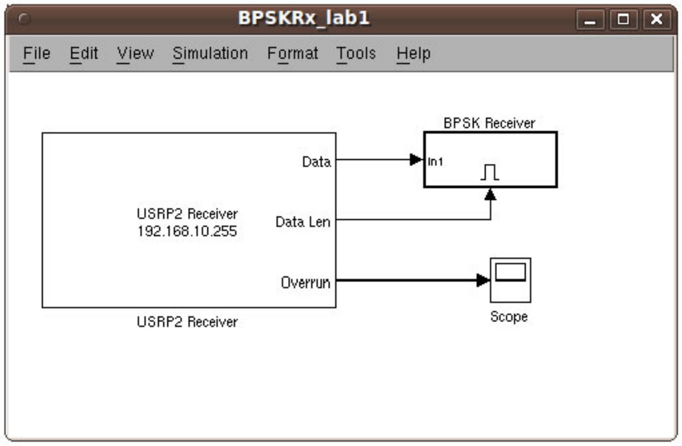

Figure 5: Sample Simulink implementation of a DBPSK receiver employed in Laboratory 1 of the digital communications course. Notice the use of the Simulink USRP2 receiver interface block.

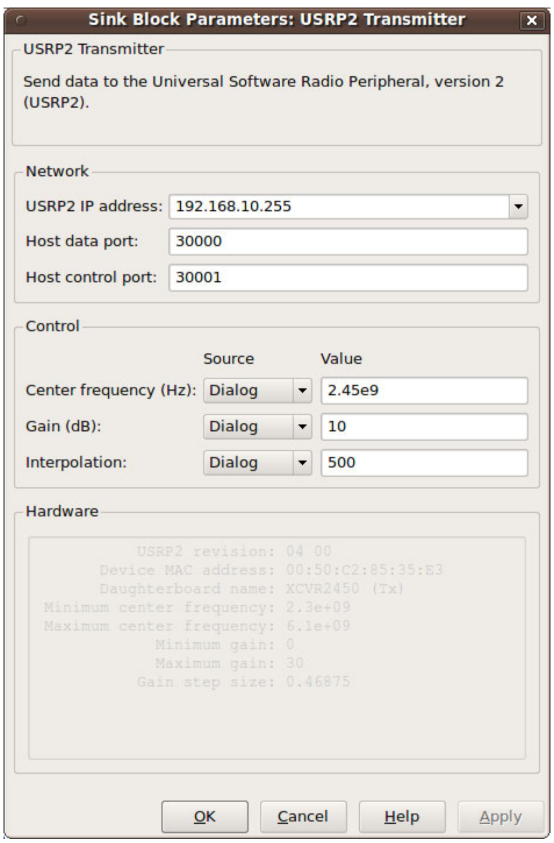

Figure 6: Properties window for the Simulink USRP2 transmitter interface block.

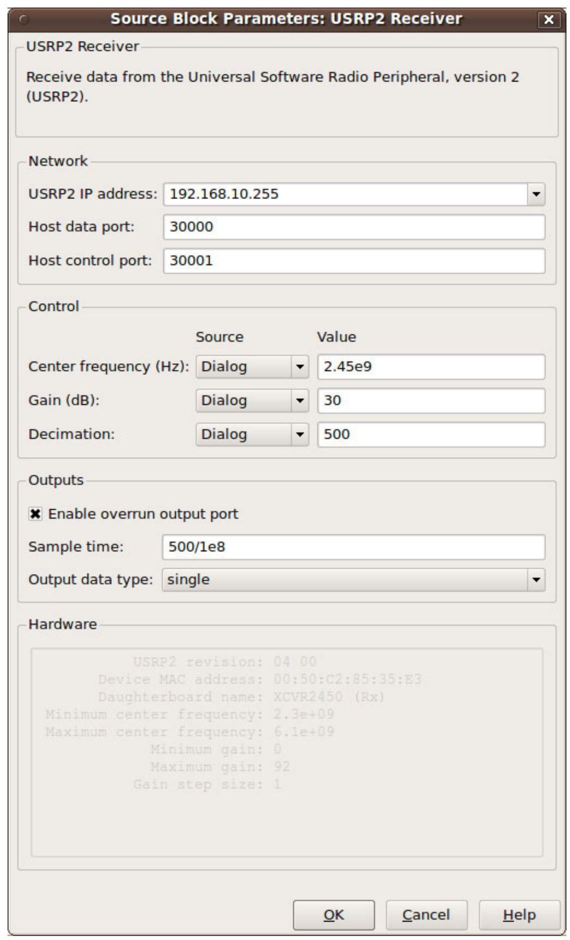

Figure 7: Properties window for the Simulink USRP2 receiver interface block.

Simulink blocks are published as sets in libraries. Additional input ports are made available through the mask, enabling the model to calculate and adjust the parameters of the radio. Each

USRP2 hardware platform attached to the host computer workstation is identified by an IP address, along with the data and control ports, using the mask. The mask has some help text and all of the parameters needed to configure the USRP2. The USRP2 has three parameters for the user to control: center frequency, gain, and decimation/interpolation factor. These are shown in Figure 6 for the transmitter block and Figure 7 for the receiver block. The range for the center frequency and the power are dependent on the attached daughterboard. The interpolation and decimation factors configure filters on the USRP2. By default, the parameters of the USRP2 will be statically set in the mask of the block. The blocks will be able to input these parameters from a port, so the parameter can be calculated in the simulation and changed on the device.

Lessons Learned and Future Work

At the time this paper is written, the second offering of this SDR-based digital communication systems course was already underway (see Figure 8). Based on student feedback from the first offering of this course, it was readily observed that the limitations and steep learning curve associated with GNU Radio and GNU Radio Companion made the SDR experimentation component of this course less accessible to senior undergraduate students that possess no prior experience with SDR. Consequently, all the laboratory experiments have been updated for the second offering of this course to be entirely based in Simulink (both simulation and SDR experimentation).

Based on the end-of-term evaluations from the first offering of this digital communication system course, all the student respondents (n=5) agreed that the intellectual challenge presented by the lab assignments was “much more” than what was expected relative to undergraduate laboratory experiments from other courses. In fact, an average of 4.8 out of 5.0 was achieved (values can range from 1.0 to 5.0), where a score of 5.0 indicates “much more”.

Figure 8: Photograph of a supervised laboratory session for the proposed digital communication systems course. Note the availability of a USRP2 platform at each lab workstation.

There was also consensus for all the student respondents who enrolled in the first offering of this course with respect to the intensity and appeal of the laboratory experiments. Specifically, all the student respondents (n=5 students) indicated that they occasionally found the labs excessive in terms of work load. However, all the student respondents also indicated that they found the laboratory material very interesting. Although there is only a very small number of student respondents from whom to obtain feedback from about the first offering of this course, it is hoped that the second offering of this course will provide a greater level of statistically reliable feedback based on the 22 students currently enrolled in the course. Upon the completion of the second offering of this digital communication systems course, there should be a substantial amount of feedback information available in order to make more reliable statements about the education approach employed in this course.

Conclusion

In this paper, we present the development, structure, and organization of an undergraduate-level digital communication systems course at WPI, including details on the “hands-on” laboratory component that employed both computer simulation and SDR experimentation. Based on the feedback from the first offering of this course, it is observed that a combination of a specific work flow for each laboratory experiment that builds up the SDR experimentation and design skill set of each student, coupled with a software package that is relatively straightforward to use, can result in a meaningful and positive learning experience for the students enrolled in this course.

Ultimately, it is hoped that the lessons learned from the creation and offering of this course can provide the necessary insights to educators from other academic institutions that are considering to include similar courses in their ECE undergraduate curriculum.

Acknowledgements

The authors would like to thank The MathWorks, Natick, MA, USA for their generous technical and financial support of these educational activities and innovations.

References

1. The MathWorks Simulink Website. URL: http://www.mathworks.com/products/simulink/
2. The MathWorks MATLAB Website. URL: http://www.mathworks.com/products/matlab/
3. Ettus Research LLC Wesbite. URL: http://www.ettus.com/
4. D. Silage. Digital Communication Systems using MATLAB® and Simulink®, Bookstand Publishing, 2009
5. M. J. Leferman. Rapid Prototyping Interface for Software Defined Radio Experimentation. M.S. Thesis, Worcester Polytechnic Institute (Worcester, MA, USA), October 2009.
6. M. J. Leferman, A. M. Wyglinski. “Taming Software-Defined Radio: A Graphical User Interface for Digital Communication System Prototyping.” DSP-FPGA.com – The Journal of Embedded Signal Processing, January 2010.

7. H. Arslan. “A Wireless Communication Systems Laboratory Course.” Proceedings of the 2nd International Conference on Engineering Education & Training, April 9-11, 2007, Kuwait.
8. C. B. Dietrich, F. E. Kragh, S. M. Hasan, C. Aguayo Gonzalez, A. A. Adenariwo, H. I. Volos, C. C. Dietrich, D. L. Miller, J. Snyder, S. H. Edwards, J. H. Reed. “Implementation and Evaluation of Laboratory/Tutorial Exercises for Software Defined Radio Education.” Proceedings of the 2010 ASEE Southeast Section Conference, 2010.
9. H. Arslan. “Teaching SDR through a laboratory based course with modern measurement and test instruments.” Proceedings of the SDR Forum Technical Conference, November 2007.
10. S. Bilen. “Implementing a Hands-on Course in Software-defined Radio,” Proceedings of the 2006 ASEE Annual Conference, June 2006.
11. J. Kang, B. Olson, A. Felzer, R. Chandra, S. Oldak. "Simulink Based Real-Time Laboratory Course Development." Proceedings of the IEEE International Conference on Microelectronic Systems Education, pp.15-16, 3-4 June 2007.
12. S. Katz. “Using Software Defined Radio (SDR) to Demonstrate Concepts in Communications and Signal Processing Courses”, Proceedings of the 2009 ASEE Annual Conference, June 2009.