AC 2011-2086: LABVIEW, THE USRP, AND THEIR IMPLICATIONS ON SOFTWARE DEFINED RADIO

Dr. Thad B. Welch, P.E., Boise State University

Thad B. Welch, Ph.D., P.E. received the B.E.E., M.S.E.E., E.E., and Ph.D. degrees from the Georgia Institute of Technology, Naval Postgraduate School, Naval Postgraduate School, and the University of Colorado in 1979, 1989, 1989, and 1997, respectively. He was commissioned in the U.S. Navy in 1979 and has been assigned to three submarines and a submarine repair tender. He has deployed in the Atlantic Ocean, Mediterranean Sea, and Arctic Ocean.

From 1994 1997 he was an Instructor and Assistant Professor teaching in the Electrical Engineering Department at the U.S. Air Force Academy, Colorado Springs, CO. During 1996 1997 he was recognized as the Outstanding Academy Educator for the Electrical Engineering Department.

From 1997 2007 he was an Assistant Professor, Associate Professor, and Permanent Military Professor teaching in the Electrical Engineering Department at the U.S. Naval Academy, Annapolis, MD. During 2000 2001 he was recognized as the Outstanding Academy Educator for the Electrical Engineering Department. During 2001 2002 he received the Raouf outstanding engineering educator award. During 2002 2003 he was recognized as the Outstanding Researcher for the Electrical Engineering Department. He was an invited scholar at the University of Wyoming, fall 2004, where he was recognized as an eminent engineer and inducted into tau beta pi. In 2006 he co-authored "Real-time Digital Signal Processing, from MATLAB to C with the TMS320C6x DSK.”

From 2007 2010 he was a Professor and the Chair of the Electrical and Computer Engineering Department at Boise State University, Boise, ID. His research interests include real-time digital signal processing (DSP), the implementation of DSP-based systems, communication systems analysis, IED defeat, efficient simulation of communication systems, spread-spectrum techniques, and ultra-wideband systems.

Sam Shearman, National Instruments

Sam Shearman is a Senior Product Manager for Signal Processing and Communications at National Instruments (Austin, TX). Shearman started with NI in 2000 and has worked in product management and R&D roles related to signal processing, RF/communications, and measurement. Prior to working with NI, Shearman worked as a technical trade press editor and as a research engineer. As a trade press editor for Personal Engineering & Instrumentation News, he covered PC-based test and analysis markets. His research engineering work involved embedding microstructures in high-volume plastic coatings for non-imaging optics applications. Shearman holds a bachelor of science with honors (1993) in electrical engineering from the Georgia Institute of Technology (Atlanta, GA).
The universal software radio peripheral (USRP) family of products has become a popular platform for hardware-based research and test bed validations conducted by universities in the software defined radio (SDR) and cognitive radio (CR) fields. With the recently released universal hardware driver (UHD) and related support for Microsoft Windows and National Instruments LabVIEW, the USRP now offers a scalable, simpler, and easier to use combined platform. The new software support broadens the accessibility of the USRP platform for teaching applications and will spur further adoption within university communication systems classrooms, teaching laboratories, and their natural follow-on coursework (e.g., SDR, CR, Digital Communications, Wireless Communications, and Satellite Communications).

This paper will discuss the utilization of National Instruments (NI) LabVIEW-based virtual instrumentation with the USRP and a UHD-based software driver to rapidly create real-time communication systems demonstrations for the classroom and/or laboratory settings. The combination of the USRP, UHD, LabVIEW, and Windows support enables implementation and exploration of both foundational and more advanced concepts related to signal processing and communications.

1 Introduction

Universities have been teaching software defined radio (SDR) courses and related communication system/signal processing topics for a number of years. Unfortunately, these courses typically experience one or more of the following problems:

a. The required equipment is expensive.
b. The signal is not actually processed in real-time, but rather recorded and processed at a later time.
c. The required equipment or its software, have a steep learning curve and the application of these tools may not be possible for a typical undergraduate student.
d. System development takes a significant amount of time, requiring several hours, if not weeks, to achieve a functioning prototype.

The universal software radio peripheral (USRP) based system we are describing addresses all of the previously mentioned problems. Specifically,

a. The complete USRP2 system that we used cost less than $2000 – an order of magnitude less expensive than a number of the systems discussed in the literature.
b. Real-time processing is possible when data is streamed to or from the USRP2 via a gigabit per second Ethernet connection, to a reasonably powerful computer running LabVIEW software.
c. The required equipment and software is easy to install, connect, and learn.
d. System development, even for the poorly prepared student, proceeds quickly.

2 The Universal Software Radio Peripheral (USRP)

The USRP is a family of computer-hosted radios developed by Ettus Research\(^\text{14}\) that allows users to create a software defined radio using any computer with a USB 2.0 or Gigabit Ethernet port. A wide variety of plug-on daughterboards allow the USRPs to be used at different radio frequency bands. Daughterboards are currently available from DC to 5.9 GHz.

Each USRP device consists of a motherboard housed within a rugged, ventilated case. The case is ventilated by several openings and a small fan to promote component cooling. Once assembled, access to the installed daughterboards is provided via front panel mounted, SMA connectors. As shown in Figure 1, the USRP 2 also has front panel connectors for

a. Low voltage direct current (DC) power.
b. Gigabit Ethernet, firmware programming (2 GB secure digital, SD card) of the installed field programmable gate array (FPGA).
c. Multiple-input, multiple output (MIMO) cable.

For convenient use of the host computer system, a second network interface card (NIC) should be installed and used for a dedicated USRP2 connection. This second NIC should be configured to the same subnet of the static internet protocol (static IP) of the attached USRP device. This will allow the host computer to perform other Internet and network functions without the necessity to repeatedly reconfigure the system’s NIC. Multiple USRP devices can be controlled from a single host computer if a network switch is used. This will allow, for example, a single host computer to control both a transmitter and receiver USRP device. All of the network devices and the associated interconnection components must support the Gigabit Ethernet standard.

\(\text{Figure 1, Photograph of an Ettus Research USRP2.}\)
Since the FPGA’s primary purpose on the mother board is to provide signal filtering and rate conversion from the system’s analog-to-digital converter (ADC) and to the system’s digital-to-analog converter (DAC), its firmware can be modified to meet your particular needs or just downloaded without modification, as a preconfigured file. The decision to be personally involved in, or knowledgeable of, the algorithm development and/or programming of the FPGA depends on the skill set that you are trying to impart to your students. Depending upon your academic program, this may or may not be an appropriate topic for undergraduate students.

Some of the USRP family support multi-radio cooperation using multiple-input, multiple output (MIMO) techniques. This is enabled by installing a MIMO interconnecting cable between two USRP devices. For example, four USRP devices with two MIMO interconnecting cables would allow for two transmit and two receive radios. When properly configured and programmed, this could form a “2 x 2” MIMO link.

In this development effort, we exclusively utilized several USRP2 systems configured with a WBX daughterboards. The WBX daughterboard is capable of

- Full-duplex transceiver
- 50 MHz to 2.2 GHz coverage
- Interchangeable front end boards to allow for higher power amplifiers, custom filters, antenna switches, etc …
- Applications include:
  - White spaces
  - Mobile phones
  - Public safety radio
  - Land mobile
  - Broadcast TV and FM radio
  - Satellite navigation
  - Covers six amateur radio bands
- Transmit specifications:
  - 50-100 mW (17-20 dBm) from 50 MHz to 1.2 GHz
  - 30-70 mW (15-18 dBm) from 1.2 GHz to 2.2 GHz
  - 25+ dB Output power control range under software control
- Receive specifications:
  - Noise figure of 5-7 dB
  - IIP3 of 5-10 dBm
  - IIP2 of 40-55 dBm
  - At every frequency there is a gain setting which gives a noise figure of less than 8 dB while simultaneously giving an IIP3 of better than 0 dBm and an IIP2 of better than 40 dBm

We selected the WBX daughterboard to provide a wide range of both transmitter and receiver frequencies, as well as offering access to a large number of real-world radio signals for students to explore. For most students, the utilization of real-world signals during projects, homework assignments, and/or laboratory exercises, has proven very motivational.
Since no single daughterboard will cover all of the possible usages and frequency ranges, a wide variety of standard and custom configuration boards are available from Ettus Research.16

Selection of a specific daughterboard can involve some compromises. In our case, the WBX is a compromise in that,

a. The board’s lower frequency limit is 50 MHz and a number of signals and their respective bands are therefore excluded. This includes, for example, the U.S. commercial AM radio band (540 – 1700 kHz). This is unfortunate in that commercial AM is one of the simplest widely available communication signals to understand.

b. Additionally, the board’s upper frequency limit is 2.2 GHz and a number of signals and their respective bands are therefore excluded. This includes, for example, the 2.4 GHz industrial, scientific, and medical (ISM) band, which is one of the most commonly used unlicensed bands in the U.S. (2.4 – 2.5 GHz). It is important to remember that in the U.S. there are twelve ISM bands that extend from a low frequency of 6.765 MHz to a high frequency of 246 GHz. This wide variety of bands and bandwidths allows “something for everyone.” Eight of these twelve ISM bands can currently be covered using available daughterboards. The WBX board covers two of these eight bands.

If it is your intention to actually transmit and/or receive an RF signal you must use an appropriate antenna. Since antennae are inherently bandpass structures, this will often require different antennae for different radio bands. At this point it should be clear that having a single daughterboard to cover the entire radio frequency (RF) spectrum is no more achievable than having a single antenna to couple these signals to that single daughterboard.

3 Host-based Processing with the Universal Hardware Driver (UHD)

The USRP2 relies on host-based software for processing and synthesis of communications signals for transmit or receive. Programmable control of the USRP hardware and software transfer of the data for transmit or receive is provided through the Universal Hardware Driver (UHD), newly developed by Ettus Research. The UHD is the “enabling function” or “glue” that allows for this seamless, bi-directional communication between the USRP and the host computer. It offers an object-oriented application programming interface (API) that can be called directly or indirectly from a variety of programming languages, operating systems, and development environments. For example, the UHD can be used with GNURadio a widely used open-source library and programming framework that is popular for Linux-based development of host-based SDR /USRP applications for the USRP with C++ and Python-based programming.

The UHD is the basis for a newly released USRP driver for LabVIEW developed by National Instruments. With this new driver, users can program the USRP using LabVIEW graphical programming on Windows. The LabVIEW development environment allows for rapid and flexible system design. While rapid and flexible system design is desired it is routinely not available when developing traditional communication systems. In a traditional communication system which was designed to meet a set of specifications, a number of engineering design
decisions are made that optimize the system’s performance. These specifications must be balanced with such other concerns as cost, time-to-market, scalability, weight, form factor, upgradability, reliability, maintainability, and manufacturability. If the system is fairly modern, it most likely has at least one processor in it that was selected to be able to quickly perform the required transceiver and user interface operations (e.g., modulation/demodulation, coding/decoding, interleaving/deinterleaving, etc ...) without exceeding the product’s bill of materials (BOM) target. This processor must therefore, be fast and powerful enough, but not too capable, since this will eventually drive up the component’s cost.

A straightforward comparison of traditional communication system design and a USRP/LabVIEW-based design follows. The USRP performs the traditional analog signal conditioning functions required of almost any radio, the required signal conversion (ADC and DAC), and then performs the rate conversion necessary to provide in-phase and quadrature (I & Q) signal samples, both to and from, the processor. In a USRP/LabVIEW-based system the powerful, but cost effective processor is replaced by, for example, a multi-core processor, with a phenomenal amount of processing resources, memory, and flexibility. In some ways, this approach is wasteful, but from an educator’s perspective, it adds flexibility to the variety of systems that can be quickly designed merely by reconfiguring the existing system.

Any host computer capable of running the latest version of LabVIEW, that also supports Gigabit Ethernet, should be able to support a USRP-based development effort. High speed, multi-core processors, with extra system memory, will permit higher performance projects to be implemented.

4 Possible Uses for this System

In an academic setting, USRP’s can be used in courses, laboratories, and research involving

a. Software defined radio (SDR).
c. Digital signal processing (DSP).
d. Multi-rate DSP.
e. Communication systems.
f. FPGA system design and implementation.
g. A wide variety of other areas (e.g., homeland security applications, satellite systems, radio astronomy, wildlife tracking, radio frequency identification (RFID), medical imaging, sonar systems, and customizable test equipment development).

5 Example of a Student Project Progression

5.1 Week One – a basic transmitter

In the first week, an independent study undergraduate student with no prior experience with LabVIEW was introduced to the USRP2/LabVIEW setup and tasked with creating a binary phase shift keying (BPSK) transmitter. Within days the student was comfortable enough with LabVIEW and the provided examples to create a working system.
5.1. Possible uses for such a system include

a. With rectangular data pulses (bits), the Sinc squared spectrum of a random bit stream becomes apparent.
b. The relationship between bit rate and first null bandwidth becomes apparent.
c. The fact that nulls occur at the spacing of the bit rate becomes apparent.
d. Discussions of the numerous definitions of bandwidth becomes possible.
e. Effect of varying signal and carrier parameters (e.g., amplitude, data rate, and carrier frequency can be explored).

5.2 Week Two – a more sophisticated transmitter

By the end of the second week, this student had implemented

a. Pulse shaping with variable roll-off factor (raised cosine).
b. Pulse shaping with variable roll-off factor (root raised cosine).
c. Pulse shaping with variable roll-off factor (Gaussian).
d. Random data sources from a number of sequence generators.
e. Quaternary phase shift keying (QPSK).
f. 8-ary PSK (8-PSK).
g. 16-ary PSK (16-PSK).
h. Explored the relationship between data rate, symbol rate, and transmitted bandwidth.

5.2.a Possible uses for such a system include

a. Exploring the relationship between roll-off factor and bandwidth.
b. Exploring the roll-off factor’s effect on the transmitter’s constellation diagram and trajectory diagram.
c. Exploring the differences between raised cosine, root raised cosine, and Gaussian pulses have on the transmitter’s constellation diagram and trajectory diagram.
d. Exploring the relationships between data rate, symbol rate, and bandwidth.
e. Calculation of spectral efficiency.

The results of these efforts are shown in Figures 2 and 3. Notice that a significant amount of control over the transmitter’s parameters is available. This student made use of the modulation toolkit.
Figure 2, VI from a student generated, 915 MHz, root-raised cosine, 8-PSK transmitter.

Figure 3, A partial block diagram of the system behind the VI in Figure 2.
5.3 Week Three – a basic receiver

By the end of the third week, this student had learned about matched filters, phase-locked loops, eye-patterns, symbol timing recovery, and had implemented a BPSK and QPSK receiver. The transmitter and receiver were both controlled from the same host computer after the student added a Gigabit Ethernet switch to the system.

5.3.a Possible uses for such a system include

a. Importance of a properly matched filter.
b. Effect of transmitter or channel impairments on the receiver’s performance. This could include I&Q gain imbalance, quad skew, error vector magnitude, SNR effects, inband interference, weak signals, very strong signals, or the effects of a multipath environment.
c. Bit error rate (BER plots) or bit error probability plots (BEP).

5.4 Week Four – a more complicated receiver and deeper development

By the end of the fourth week, this student had extended his work to 8-PSK and 16-PSK receivers and was exploring 16-QAM and 32-QAM. The results of his 8-PSK receiver are shown in Figures 4 and 5. All of the signals were received over a wireless indoor channel. A channel equalizer was not used.

5.4.a Possible uses for such a system include

a. Discussion of the relationship between the need for an increased signal SNR while increasing spectral efficiency.
b. The need for a channel equalizer to overcome channel impairments.
c. Effect of phase noise on system performance.
d. The possible introduction of error correcting codes and interleavers to help mitigate the effects of fading.

At this point in the course we decided to remove the reliance on the modulation toolkit and asked the student to create his own data source using a state machine (feedback shift register, maximal length sequence, or m-sequence generator). This student had significant course work in the digital systems/embedded systems area. It was believed that this previous course work would make the design and implementation of a finite state machine a straight forward effort. This task was accomplished in very little time. The student without prompting created an arbitrary order (up to order 16) linear feedback shift register system with reprogrammable feedback connections.

We believe that given the proper introduction this, and another motivated, student would be able to create the functionality of almost any modulation toolkit block. This powerful process of allowing very rapid student progress is not hindered by a lack of ability to peel back the programmatic layering to add significant depth to the laboratory exercise or the course in general.
5.5 Week Five – the continuing effort

With the vast majority of the topics associated with a first course in “Digital Communications” now completed, the student began to explore additional areas of personal interest. He is currently working on a system that will display a histogram of the received signals power levels.

5.5.a Possible uses for such a system include

a. An opportunity to explore receiver architecture.
b. An opportunity to discuss and explore actual fading channel parameters compared to existing channel models.
c. Effect of the laboratory environment on the channel’s fading parameters (e.g., movement, reflectors, directional antennae, etc …)
6  A Future Full of Possibilities!

The LabVIEW / USRP combination presents an opportunity to enhance communications education by enabling a low cost, hands-on approach with live signals for realistic, real-world demonstrations, laboratory exercises, capstone design projects, and cutting-edge research. You are invited to participate in what we hope will become an ever increasing community of LabVIEW/USRP users.

7  Conclusions

The universal software radio peripheral (USRP) family of products has become a popular platform for hardware-based research and test bed validations conducted by universities in the software defined radio (SDR) and cognitive radio (CR) fields. The recently released universal hardware driver (UHD) as part of the LabVIEW, along with the USRP now offers a simpler, scalable and easier to use combined platform that will both broaden the accessibility of the technology and platform for hands-on applications and spur further adoption and use within university communication systems classrooms, teaching laboratories, and their natural follow-on coursework (e.g., SDR, CR, Digital Communications, Wireless Communications, and Satellite Communications).

Given the rapid systems implementation that can be achieved using this new LabVIEW(with the new UHD) and USRP2, coupled with the dramatic cost reductions that are possible when compared to more traditional SDR platforms, this is a timely and appropriate solution to a wide variety of education and research challenges.
References


