Application of Signal Processing in GPS Signal Detection Using USRP

Ehsan Sheybani, University of South Florida, College of Business, Sarasota, FL, USA
Giti Javidi, University of South Florida, College of Business, Sarasota, FL, USA

ABSTRACT

The Universal Software Radio Peripheral (USRP) is slowly becoming a very popular piece of hardware used in different universities and research labs across the world. It is inexpensive, which attracts a lot of attention, along with its variety of applications and capabilities. The USRP connects to a host-computer through a high-speed USB or Gigabit Ethernet interface. Another reason for the increasing popularity of the USRP is its ability to respond to multiple programming software such as GNU Radio, Matlab/Simulink, and LabView. There is a broad range of capabilities of the USRP one of which includes receiving GPS signals. Each GPS satellite transmits data on two frequencies, L1 (1575.42 MHz) and L2 (1227.60 MHz). We focused on the L1 band that transmits at 1575.42 MHz. The main objective was to engage in signal analysis with the carrier signal of the L1 band. This task proved to be challenging but not impossible. The carrier signal is essentially to carry information modulated on to it; in this case the navigation message from the GPS satellite along with other codes that are irrelevant to use at this point. After receiving the signal via USRP and with a simple extraction of the carrier signal, we were able to record the signal and reconstruct it using its In-phase and Quadrature phase (IQ) data. With just the carrier signal, one is able to do multiple things. One can modulate his/her own information onto the signal and transmit it through the USRP. Further analysis on the characteristics of the signal can be done. For example, one can compare the strength of a direct to the indirect signal. In theory, one can determine the characteristics of the surrounding area when comparing direct and indirect carrier signal. The possibilities are endless.

KEYWORDS

GNU, GPS Signal, Signal Processing, USRP

INTRODUCTION

What follows is a research report detailing the steps taken to apply signal-processing algorithms to USRP. One should be able to follow these steps with the specified set of hardware and software to get the same configuration on the proper USRP device. These projects were implemented under the supervision of NASA scientists and faculty advisors in dedicated NASA labs. Following lessons learned here and authors’ previous experiences in data visualization and signal processing research and training, (Javidi & Sheybani, 2008; Sheybani & Javidi, 2006; Sheybani, Ehsan; Garcia-Otero, 26)
Adnani, Javidi & Deshpande, 2012; Ouyang et al., 2010; Sheybani, Javidi & Garcia-Otero, 2008; Javidi & Sheybani, 2010; Varde et al., 2007; Sheybani, Javidi, Hardy, Denton & Campbell, 2007; Sheybani & Javidi, 2007; Badombena-Wanta & Sheybani, 2010; Sheybani & Sankar, 2002; Sheybani & Arora, 1992, Sheybani, 2011; Garcia-Otero & Sheybani, 2011) relevant labs were designed to enhance the Computer Engineering program at the Virginia State University (VSU).

To accomplish our overall goal of retrieving the carrier signal of the L1 band from a GPS satellite using USRP we had to be familiarized with the hardware. The USRP needs a daughterboard, which is used to hold the radio frequency receiver interface or the transmitter interface. Daughter boards are selected based on the project that one is set on doing. In many cases, the selection of an RF daughterboard is made solely on the application requirements for frequency coverage. Figure 1 shows the ranges of different daughter boards (Anon, 2014).

The next step was to determine what software can be used to program the USRP. GNU Radio is easy to use software that consists of mostly GUI blocks programmed by Python. The user has to fill in the parameters for each block depending on the application parameters. To become familiar with GNU Radio we constructed a simple dial tone. A dial tone is generally just two Sine waves, one with a frequency of 350 Hz and another with a frequency of 440 HZ. Combining the two waves into an audio sink will produce a dial tone. Figure 2 shows the simple flow graph used to make the dial tone.

**METHODOLOGY**

Constructing a working dial tone was only the beginning. The dial tone worked with just GNU Radio. Now it was time to incorporate the USRP. When using USRP2 and its Gigabit Ethernet connection a static IP has to be set. GNU Radio is used on a Linux operating system called Ubuntu. Instructions on how to set a static IP address can be found on the Ettus Research site. Programming the USRP to become a FM Radio receiver is considered to be a good way to become acquainted with using the USRP and its programming software. If we were able to construct a working flow graph and record the station we chose to listen to then we knew we were familiar enough with the software and hardware to proceed with our project. Figure 3 shows the GNU implementation of the FM radio.

After implementing the FM Radio, two things were proved. The first was the fact that we knew how to receive a signal with a known frequency and the other thing accomplished was the fact that we knew how to record what was received to a file for post-test analysis. A Stanford Research Systems Synthesized Function Generator was connected to the USRP to record a sine wave. Several recordings...
Related Content

Advance and Innovation in Wireless Power Transmission Technology for Autonomous Systems
www.igi-global.com/chapter/advance-and-innovation-in-wireless-power-transmission-technology-for-autonomous-systems/164168?camid=4v1a