Application Note

Low-Cost, Wide-Band RF Recording and Spectral Analysis with a USRP Device

Ettus Research

Introduction

The USRP™ (Universal Software Radio Peripheral) is a flexible and low-cost software defined radio used for spectral monitoring and RF recording. Many USRP users in these applications operate in a Linux environment, commonly within the GNU Radio software framework. However, the USRP device is supported by many pre-compiled, out-of-the-box software solutions not requiring additional development. These solutions are attractive for users that wish to access specific functionality without investing development time with new tools. The flexibility and widespread adoption of the USRP for applications is enabled by the USRP Hardware Driver (UHD).

HDSDR is one example of existing Windows application supporting the USRP product line. Developed by Alberto di Bene, HDSDR is a free downloadable program with an attractive fast-Fourier transform (FFT) and waterfall display. It is useful for observing signals of interest within the bandwidth capability of the USRP radio. HDSDR provides the following functionality:

1) Fast, configurable FFT and waterfall displays
2) Adjustable filtering, and selectable notch filtering for RFI mitigation
3) Demodulation of AM, ECSS, FM, LSB, USB, CW, and DRM
4) AGC and AFC
5) Wideband recording
6) Configuration and control of an RF frontend with intuitive user controls

This document explains the installation procedure for HDSDR and providse an introduction to some of the features available in the application.

UHD and the HDSDR Interface

The full functionality of this software package requires three components. The software component that interacts directly with the USRP device is UHD. UHD enables you to deploy software radios on Windows, Linux, or MacOS. Various software frameworks, such as GNU Radio, LabVIEW, MATLAB, and Simulink use the UHD driver to access to USRP device. These are ideal frameworks for if you want to rapidly develop custom applications. UHD supports all USRP devices without explicit abstraction to your application. Therefore, once you develop an application to work with UHD, all USRP models and daughterboards are supported.

The second component in the system is the ExtIO DLL feature provided with the HDSDR software. This is essentially an abstraction layer that maps the common functions of HDSDR to the API functions of UHD. This allows HDSDR to be used with a variety of frontends. Balint Seeber (spench.net) has developed the dynamic-linked library DLL to access UHD through this Ext IO DLL interface. This interface also provides the capability to pass along the same I/Q samples to an external program, such as a GNU Radio flowgraph.

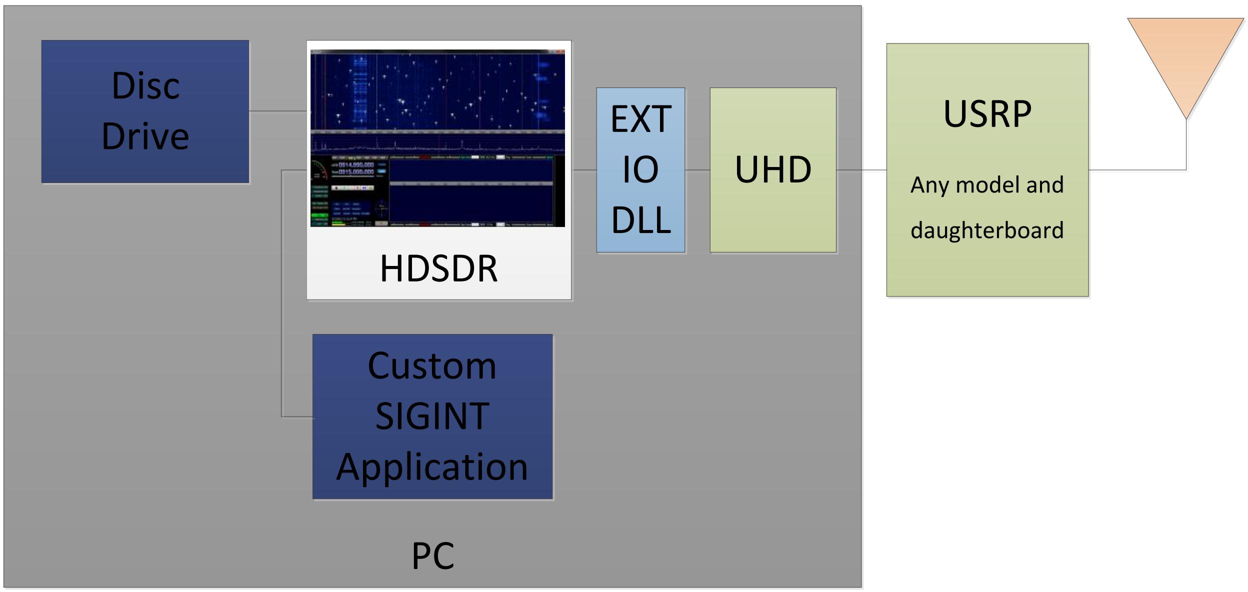

Figure 1 - Spectrum display and RF recording with HDSDR, UHD, and a USRP Device.

Installation Procedure

At this time, this software can only be installed in a Windows environment. This was tested on a PC running Windows 7. It may be possible to achieve the same functionality with virtual machines in a Linux environment, but this has not been tested.

To install this software, there are three major steps:

1. Install UHD
2. Install HDSDR
3. Install packaging that includes BorIP/ExtIO functionality for USRP device

Installing UHD

Presently the software has been tested with UHD versions 3.3.0 and 3.4.0. Ideally, HDSDR will support all future versions of UHD. However, if any issues arise, you can revert to these two versions. Binary installers for windows can be found under the "Binary Downloads" section on this page:

http://code.ettus.com/redmine/ettus/projects/uhd/wiki#Binary-downloads

Each UHD release includes links to binary installers for several operating systems. Select the binary installer labeled:

UHD-[uhd_version]-win32.exe

After completing the installation, connect a USRP device to the host PC and test the functionality. This can be accomplished with a few easy steps. Open a command prompt and type the following command:

uhd_find_devices

This should locate the USRP devices, and output various parameters for the device such as name (blank by default) and IP address. If this is a new installation and any problems are encountered, see some of the troubleshooting hints in the networking and/or appropriate USRP sections:

http://files.ettus.com/uhd_docs/manual/html/

Once you have completed this step, a more in-depth test is:

uhd_usrp_probe

If you have multiple USRP devices connected to the host, you may need to specify an address. Use the "—help" argument in the uhd_usrp_probe command for more information.

This step may reveal the FPGA image of the USRP needs to be changed to match that of the driver. Once again, refer to the uhd_docs manual for more information on how to complete this. If you have executed uhd_usrp_probe and received a full report on the device and any integrated daughterboards, UHD and the USRP interface are functioning properly.

Installing HDSDR

Installing HDSDR is a fairly straight forward process. Testing for this application note was completed with Version 2.11. The latest version of this software can be accessed on the HDSDR website:

http://www.hdsdr.de/download/HDSDR_install.exe

Ettus Research encourages you to use the latest version of the software because it will likely provide new features and an improved experience. If the software is updated and any issues arise, you can revert to Version 2.11 by downloading the legacy HDSDR installer from:

http://files.ettus.com/app_notes/hdsdr/HDSDR_install.exe

Executing this Windows installer will start an installation wizard. This wizard will navigate through several steps typical for a windows installer: install directory selection, shortcut selection, etc.

Installing ExtIO Interface Developed by Balint Seebger

A Windows installer is also available for the ExtIO DLL to enable USRP integration. The file can be downloaded from:

http://spench.net/drupal/files/ExtIO_USRP+FCD+BorIP_Setup.zip

Once again, if the link is broken or there are problems with a newer version, you can revert back to Version 1.1, which was used in this application note. This back-up file is available at:

http://files.ettus.com/app_notes/hdsdr/ExtIO_USRP+FCD+BorIP_Setup.zip

Once you arrive at the "Choose Components" screen, you must enable the LibUSB, VC++ Runtime, and ExtIO_USRP+FCD + BorIP selections. BorIP is not required for this software use-case, so you do not need to select "Autorun BorIP." HDSDR is already installed.

Other steps in the install process are fairly straight forward. If issues are encountered in this step, refer to this site:

http://wiki.spench.net/wiki/USRP_Interfaces

Running the Application

After the installation is complete, reboot the PC. Ensure the USRP device is plugged into the computer and power on. Start the HDSDR application. At start-up, HDSDR should recognize the presence of the USRP ExtIO DLL and open this window:

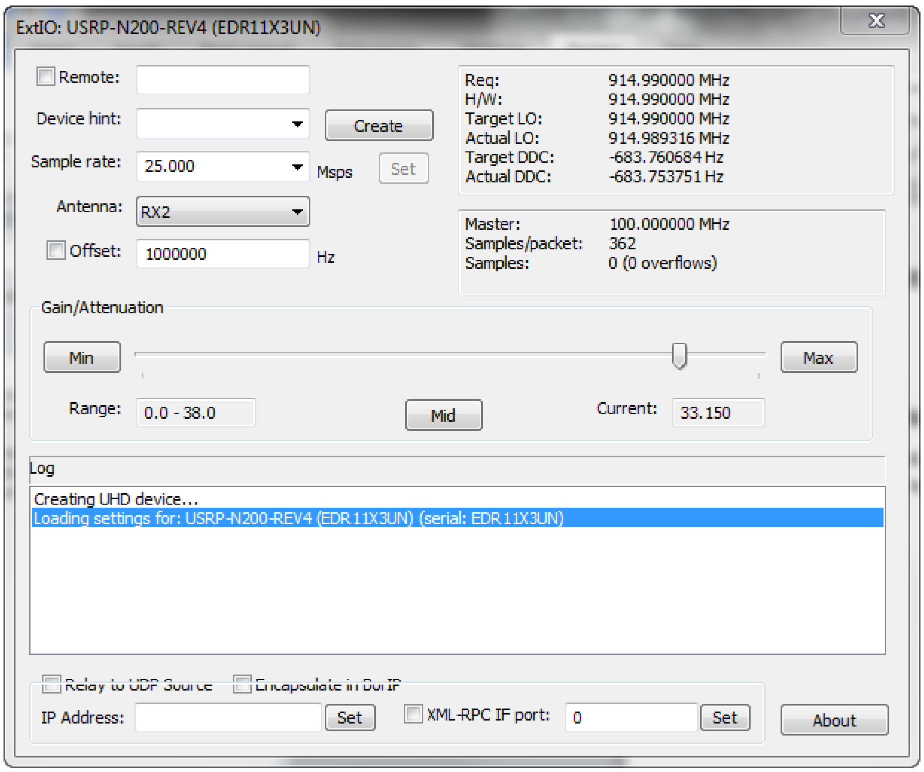

Figure 2 - ExtIO Configuration Screen

This window is used to configure the USRP device for operation. The remote function allows HDSDR to access a USRP device on a remote machine through the BorIP server. The device hint includes various parameters such as address, frame sample sizes, etc. In most cases, you can leave this area blank if you only have a single USRP device connected to the host machine. Refer to this site for additional details on device hints:

http://wiki.spench.net/wiki/ExtIO_USRP

The sample rate sets the rate of the complex baseband samples transferred to the host from the USRP device. This sample rate must be an integer multiple of the ADC sample rate of the USRP device. The ADC sample rate is 64 MS/s for the USRP1, and 100 MS/s for the USRP N200/N210. The default ADC sample rate for the USRP E100/110 and USRP B100 is 64 MS/s, but this is configurable. The maximum sample rate is limited by the interface bandwidth between the USRP device and the host machine. It may also be limited by the processing power of the host machine. In other words, even though the Gigabit Ethernet interface of the USRP N200/N210 can support 25 MS/s, the host machine many not be able to perform the DSP operations quickly enough.

If the offset checkbox is selected, UHD will offset the LO setting accordingly and compensate for the offset within the DDC in the FPGA. This makes it possible to move the LO leakage spike away from signals of interest. The gain setting controls the analog gain of the daughterboard mated to the USRP device. All remaining settings are not critical for this application example.

After setting the ExtIO component correctly, the USRP device should be ready for operation. Clicking the “Start” button will initiate receive streaming. The waterfall and FFT displays should begin. If the sample rate is below 10 MS/s, the secondary waterfall, FFT and demodulator will be enabled.

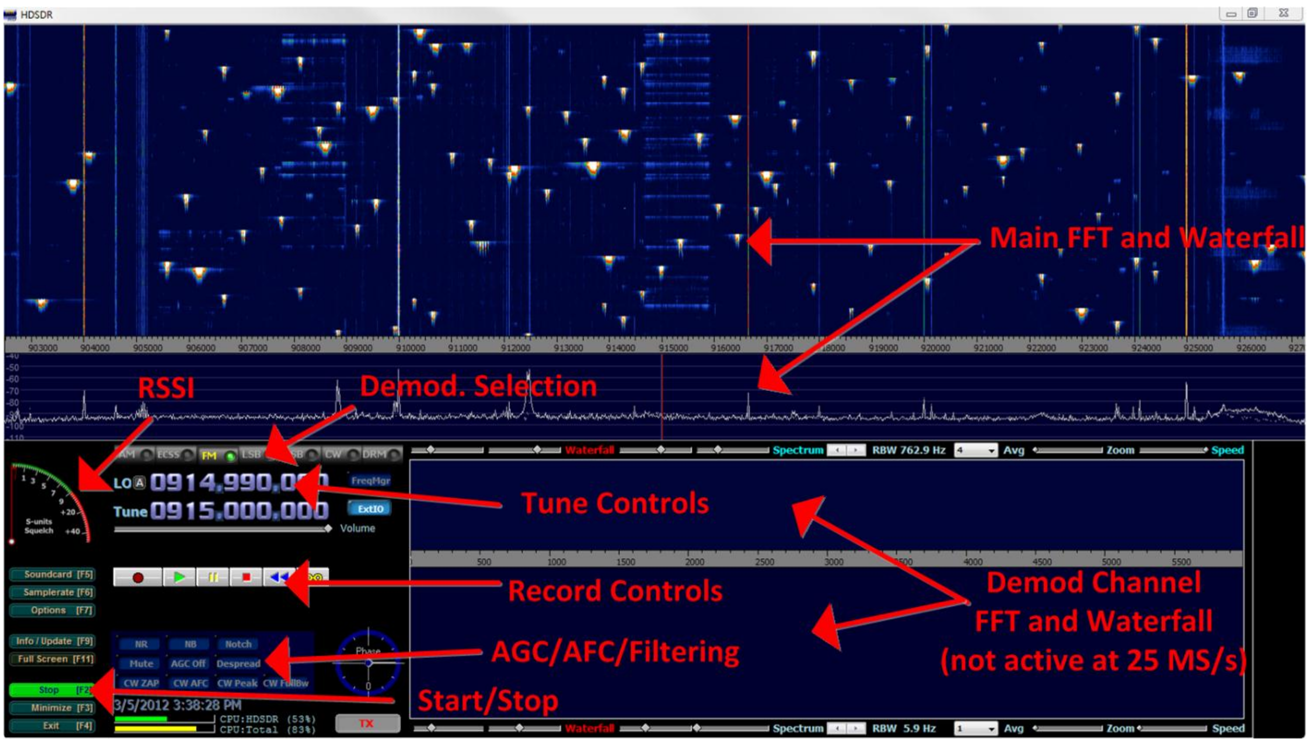

Figure 3 - Some Items to Navigate on the HDSDR Display

Important Points About Navigation

A full set of instructions for this software is outside the scope of this document. Ettus Research encourages you to refer to any instructions provided by HDSDR developers. However, it is worth providing a quick review of some helpful features:

1) Point-and-Click Tuning – The HDSDR developers have made frequency navigation very intuitive. You can click on any point along the frequency axis of the main waterfall or FFT plots to tune the receiver. A Left-Click will tune HDSDR's demodulator to a specific frequency. A Right-Click will automatically tune the USRP receive chain, including the LO and FPGA DDC. This enables you to rapidly tune to a signal of interest after observing it on the waterfall.

2) Options Button – Provides access to several configuration parameters, ranging from the appearance of the waterfall, to configuration for automatic tuning capability through DDE, to I/Q manipulations.

3) Tuning Controls – The controls provide tuning functionality via mouse or keyboard. To enter a new frequency, click on the controls and begin typing the desired frequency.

4) Recording Controls – Located just below the tuning controls, the recording controls enable you to quickly and easily start recording. File names are automatically generated. Playback to the DSP chain is also enabled with these controls.

5) Waterfall and FFT Scaling – The controls just below the Waterfall and FFT allow you to adjust units per division, reference level, resolution bandwidth, speed, and averaging.

About Recording

The recording functionality provided with this package is one of the most attractive features. RF recording is an important capability for users in a number of application areas such as GPS research, signals intelligence, interference mitigation, etc. This software package shows the USRP device can be easily employed for these applications. HDSDR allows you to identify potential signals of interest in real time, record them in real-time, and then post process the data in other, flexible environments such as GNU Radio.

The software also supports RF playback. Currently, this playback is not routed back to the USRP device for transmission. However, the ability to display recorded signals on the waterfall and FFT display allows you to examine various signals for telltale signatures and patterns.

The reference system was able to record 20 MS/s throughput while running the Waterfall and FFT at full speed. This compares favorably to performance of other software or development environments. There are several components in systems that may limit the performance of the recording feature. In many cases, the hard drive will constrain the maximum sample rate for recording. Remember every complex sample is a set of two sixteen bit number. Each complex sample includes two, 16-bit integers. Recording at 20 MS/s translates to 80 MB/s to the disc.

Some Interesting Screen Shots

In order to show the potential for this system, screenshots of several signal types and bands are shown below. The software is able to display the entire FM broadcast band and a portion of the HF band. There are also screenshots of a 802.11n signal and the 900 MHz ISM band, which shows several frequency hopping spread spectrum (FHSS) and direct spread spectrum (DSS) transceivers.

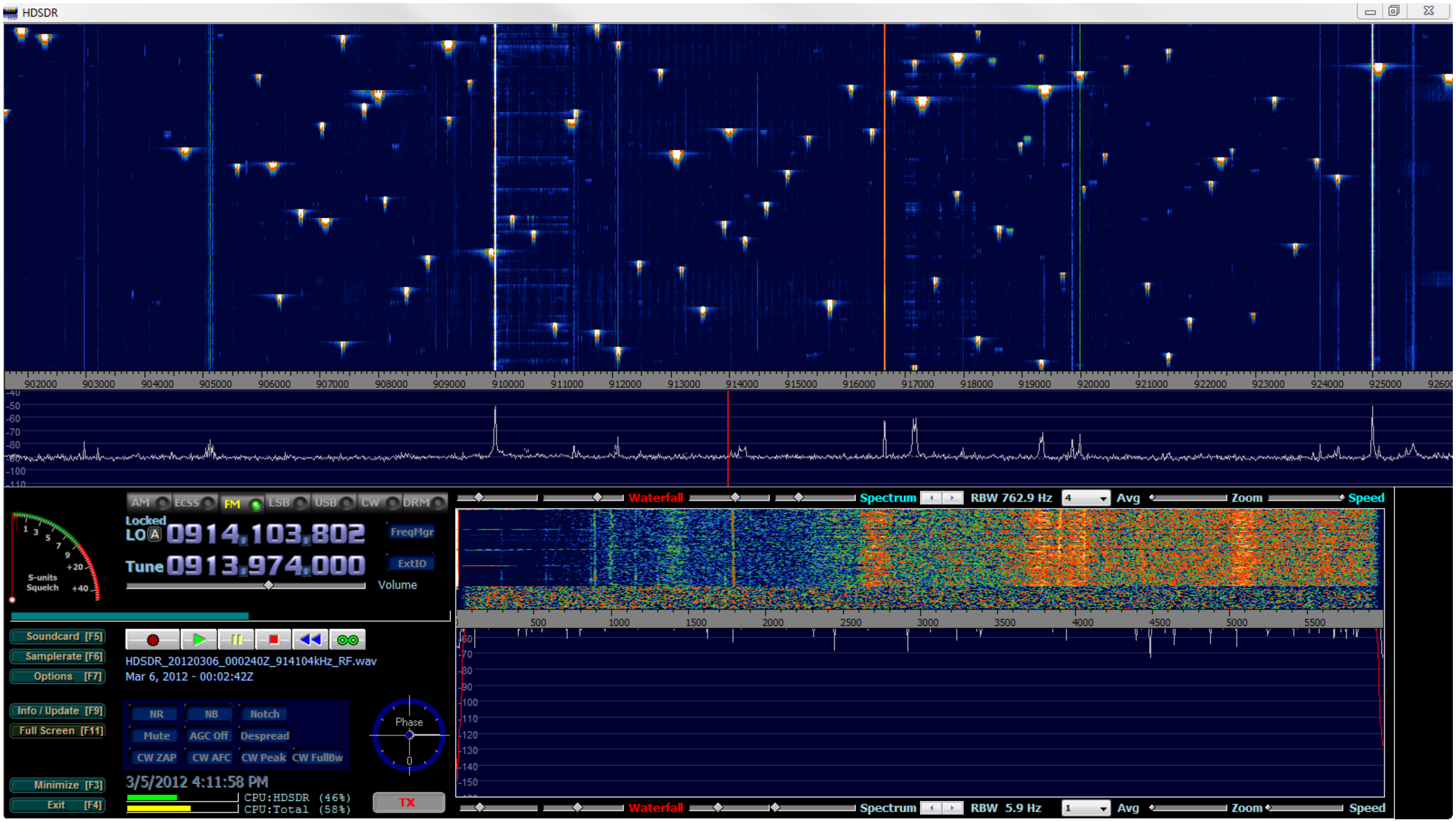

Figure 4 - Screenshot of FHSS and DSSS Signals in the 915 MHz ISM Band(N200, SBX)

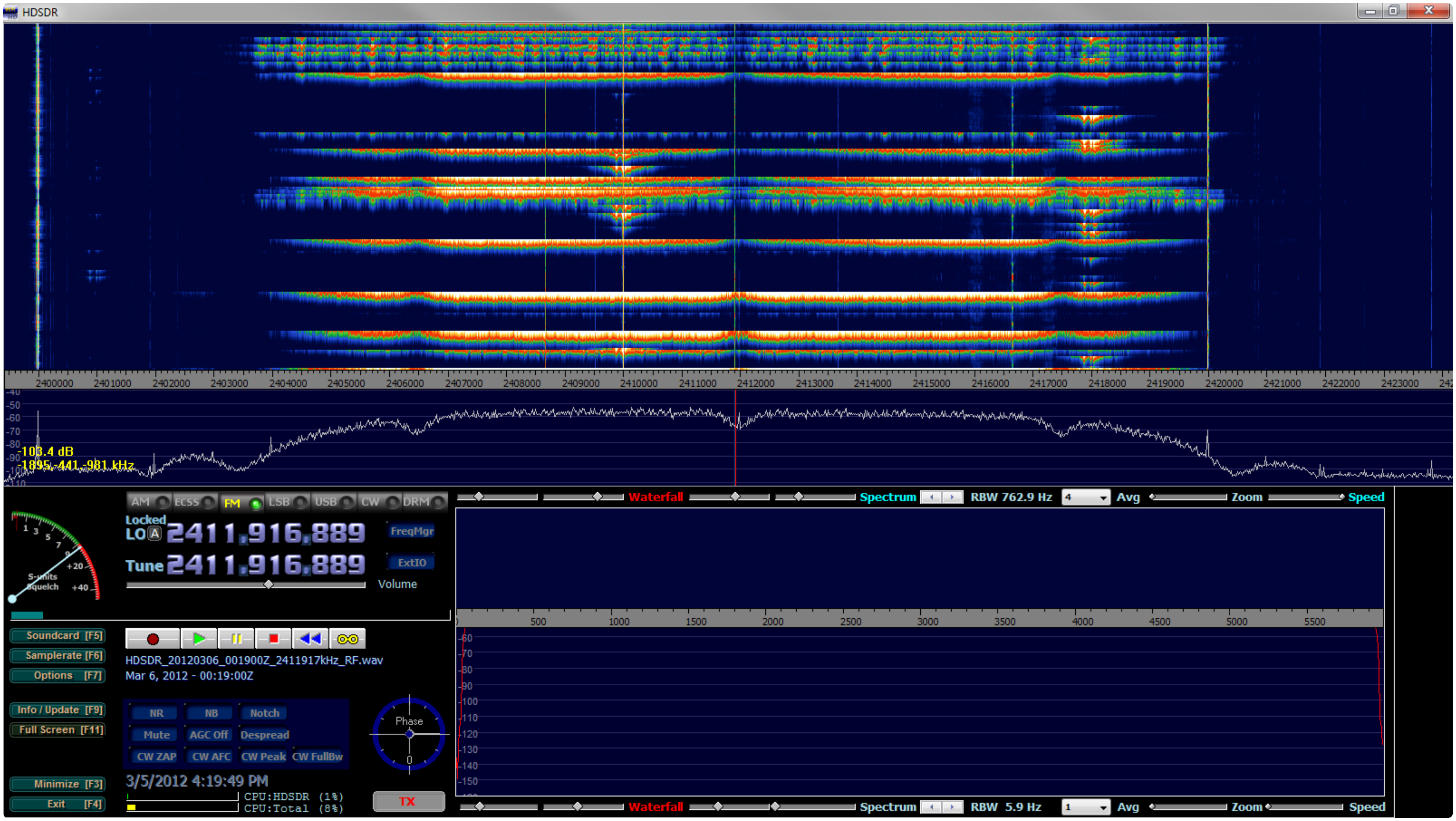

Figure 5 - 802.11n Playback(N200, SBX)

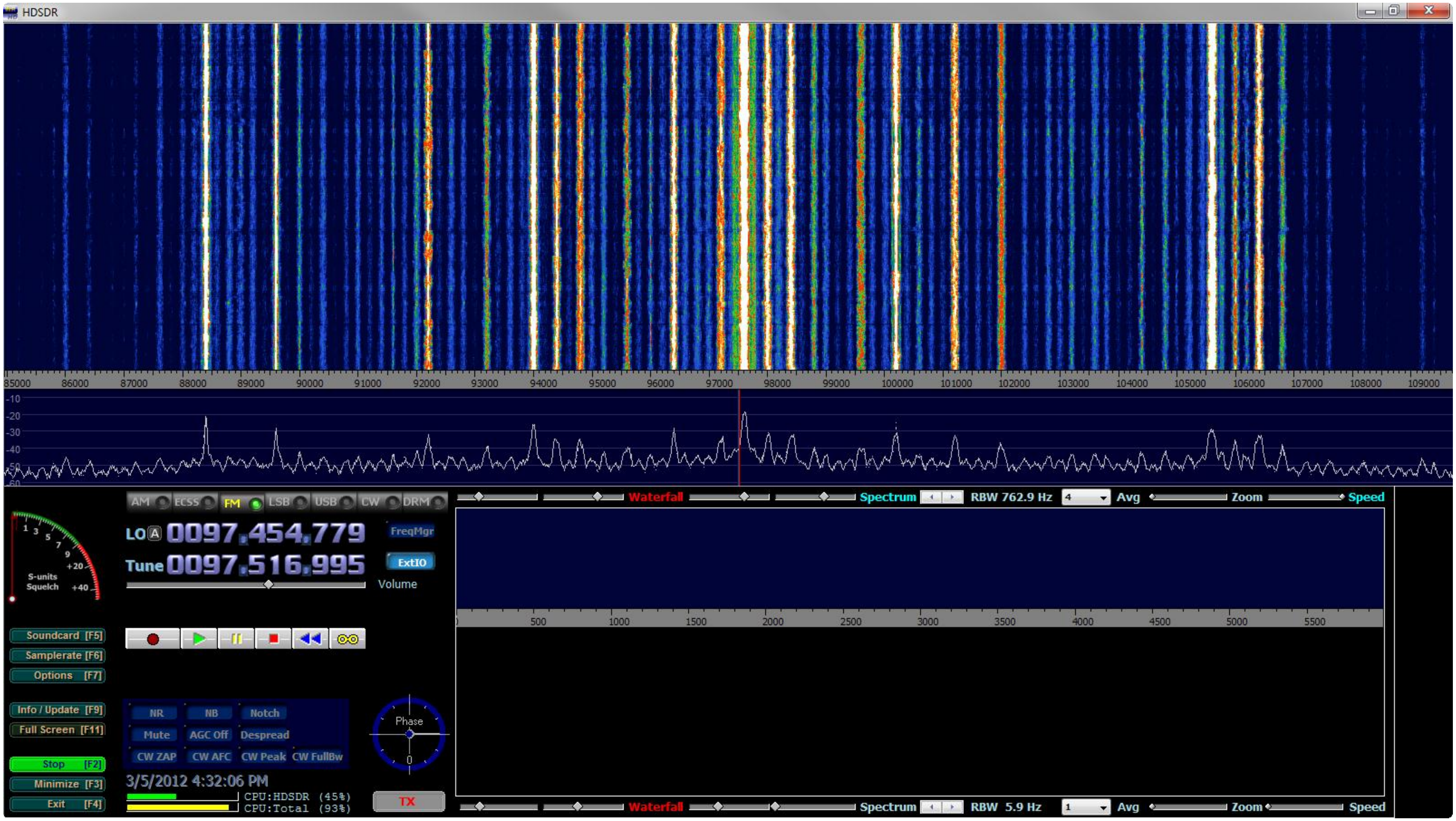

Figure 6 - Capture of Entire FM Broadcast Band - 25 MS/s(N200, WBX)

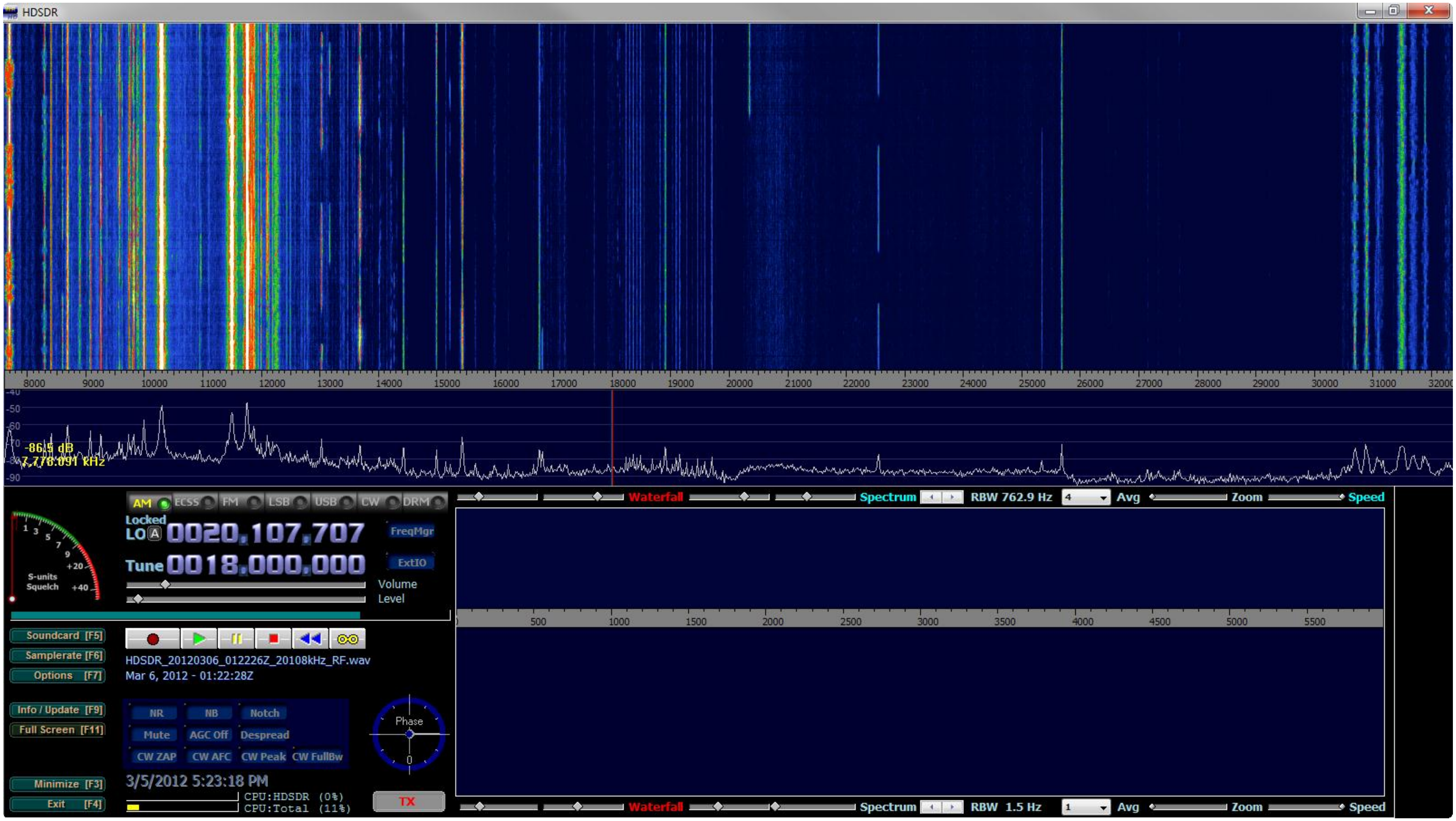

Figure 7 - Capture 25 MHz of HF Band showing strong Short Wave Broadcasts, CW, USB and other Signals(N200, LFRX, External LNA)

About the Reference System

Note performance varies across various computers. Processing performance impacts the maximum speed and throughput of the DSP operations. The speed of the hard drive will limit record and playback speed. In limited cases, the USB controllers or Gigabit Ethernet adaptors have limited throughput as well. Table 1 shows key parameters of the reference system used for this application note.

USRP Device	USRP N200, Rev 4
Daughterboard(s)	WBX, SBX, LFRX
	Computer
Make/Model	Lenovo Thinkpad
Computer Model Number	W520 4270-CTO
Processor	Intel® Core™i7-2620 CPU @ 2.7 GHz
Memory	4.00 GB
Gigabit Ethernet Chipset	Intel®82579LM
OS	Windows 7 Home Premium, 64-bit, SP1
Hard Drive	WDC WD3200BEKT-08PVMT1
Video	NVIDIA Quadro 1000M
	Software
UHD	3.4.0
HDSDR	2.1
ExtIO DLL	1.1.0

Table 1 - Reference System Characteristics

Spurs

Many of the spurs present in the spectral displays are man-made sources external to the USRP radio. There are also spurs generated by the USRP itself. Dealing with these spurs is one of the challenges presented by a low-cost, wideband software radio. It is possible to relocate this spurs in the baseband signal by using the LO offset function provided in UHD the ExtIO control. Filtering of the RF signal will reduce this spurs.

USRP Hardware Portability

The USRP N200 was used in this reference design. However, it's worth mentioning again the lower-cost radios, including the USRP B100, USRP1, or N210, can easily replace the N200. Of course, selecting a USRP device with a USB 2.0 host interface will reduce the bandwidth capability.

Since this software package only supports Windows, the USRP E100 and E110 are not compatible. However, Ettus Research is investigating the potential of using the BorIP remote USRP server to easily share the embedded UHD functionality of the USRP E100/E110 across a network connection.

Linux Functionality

While this software package won't operate in a Linux environment, there are several other paths allowing for similar functionality. You could implement a similar application using a GNU Radio flowgraph in Linux. As mentioned, it may also be possible to run this application on a virtual machine, although performance generally suffers in such implementations. Ettus Research has not tested this on a virtual machine.

Warranty

The intent of this application note is to show you the flexibility offered by the UHD. HDSDR is a very helpful example of what can be accomplished with the USRP software defined radios. However, Ettus Research does not claim any affiliation with the HDSDR or, USRP ExtIO developers. Ettus Research does not warrant these products. Ettus Research is not liable for any damages these programs may cause. USRP hardware warranty information for Ettus Research products can be found on the website at www.ettus.com

Conclusion

This application note provides a brief overview of how to install and operate an application called HDSDR. This application provides wide-band spectral analysis, demodulation, and record/playback capability. Through the UHD driver, this software is able to access and configure the USRP for wideband signal collection. If you have any questions on USRP-specific items, please send them to support@ettus.com.