LABORATORY MANUAL

EE 504. Software-Defined Radio.

California Polytechnic State University, San Luis Obispo

Latest Revision: April 5, 2022

Advisor: Steve Dunton

Author: RJ Macaranas

Contents

List of Figures

0 Course Information

0.1 Catalog Description

EE 504. Software-Defined Radio. **4 units**

Prerequisite: EE 314; and EE 328 or CPE 327; or graduate standing.

Introduction to software defined radios, including architectures of software defined radio receivers and transmitters, design principles and trade-offs, signal processing techniques, and applications of the technologies. 3 seminars, 1 laboratory.

0.2 Learning Objectives

- Become familiar with the configuration of software-defined radios.
- Develop an understanding of communication system components.
- Leverage understanding of communication systems to use MATLAB's Communication, LTE, and 5G toolboxes.
- Investigate a relevant topic of interest and present findings during seminar or lab session.

0.3 Laboratory Activities

Week	Lecture Topic	Lab Activity
1	Intro	Plutos and Links
2	Comm Systems, Sampling Theory	FM, Nyquist, and Antennas
3	Sampling Theory	TBD
4	Digital Filters	TBD
5	Timing/Contingency	TBD
6	Probability	TBD
7	Channel Encoding	project work
8	Channel Encoding	project work
9	Special Topics	project work
10	Special Topics	project presentations
F	final exam	submit report

1 Plutos and Links

"Pluto is not a planet"— M. Rodriguez.

1.0 Learning Objective

The purpose of this session is to introduce the software-defined radio (SDR) unit that will be used in this course, install the necessary software on your machine, and establish communication links.

1.1 The ADALM-PlutoSDR

For this course, we will be using the Analog Devices Active Learning Module (ADALM) PlutoSDR. This is a bit of a mouthful so we will just refer to them as **plutos**. The plutos are equipped with a transmit (Tx) port, a receive (Rx) port, a Micro-USB 2.0 data port, a power port (also Micro-USB 2.0), an indicator LED, and uncountably infinite configuration possibilities via software. The radio frequency (RF) ports, Tx and Rx, are SubMiniature version A (SMA) female connectors, so an attaching antenna will need to have an SMA male connector. The RF ports can be set to either half duplex, either talk or listen, or full duplex, talk and listen at the same time. I tend to operate better in half duplex so if you have a question try to send an interrupt first so my brain can make a context switch (bad software joke).

1.1.1 Specifications

The previous TA, Julio Tena, put together a wonderful presentation to outline the capabilities of the pluto. Some specifications of interest pulled from Analog Devices website are listed below:

- RF coverage from 325 MHz to 3.8 GHz
- Up to 20 MHz of instantaneous bandwidth
- 12-bit ADC and DAC
- One transmitter, one receiver, half or full duplex
- MATLAB and Simulink support

1.1.2 Kit Components

Your kit should come with the pluto shown in Figure 1, a USB data cable, an SMA male-male connector shown in Figure 2, and two antennas shown in Figure 3.

Figure 1: Pluto au naturel

Figure 2: Pluto kit SMA male-male connector

Figure 3: Pluto kit antennas

1.2 Lab Activity

We will install the necessary software components, run an example transmit-and-receive (tx and rx, respectively) script, and test a few communication links.

1.2.1 Equipment

- pluto kit
- laptop with MATLAB and Simulink

1.2.2 Procedure

Be sure to document your findings (pictures and screenshots) along the way for your lab notebook.

- Sign into MathWorks.
- Install MATLAB if you don't already have it on your machine.
- Install the ADALM-PLUTO Radio Support from Communications Toolbox.
- Follow the hardware installation and connect the pluto to your computer via USB.
- Open MATLAB and hack your pluto by running the following line in the command window:

  ```
  configurePlutoRadio('AD9364')
  ```

  This will expand the operating frequency range by configuring the pluto to the AD9364 RF transceiver architecture which supports frequency tuning from 70 MHz to 6 GHz. The pluto by default will follow the AD9363 architecture which has the advertised operating range of 325 MHz to 3.8 GHz.
- Verify that the receive-and-transmit MATLAB script works.

1.2.3 Deliverables

Document your set-up and results in your lab notebook and post next week before class.

1. (25 pts) Screenshot of transmitted signal from tx-and-rx example MATLAB script.
2. (25 pts) Screenshot of spectrum analyzer output from tx-and-rx with bare RF ports.
3. (25 pts) Screenshot of spectrum analyzer output from tx-and-rx with a hard-wire link.
4. (25 pts) Screenshot of spectrum analyzer output from tx-and-rx with an antenna air link.

2 FM, Nyquist, and Antennas

"Marconi plays the mamba, listen to the radio"— Starship.

2.0 Learning Objective

The purpose of this session is to introduce the software-defined radio (SDR) unit that will be used in this course, review concepts of frequency modulation (FM), implement an FM receiver in software, construct an antenna, compare antenna performance of audio fidelity, and observe the Nyquist-Shannon sampling theorem in the frequency spectrum.

2.1 Before the Lab Session

1. Read all sections up to the lab activity.

2.2 FM Theory

Imagine you are at a party. Music is playing, everyone is either laughing, talking, or screaming across the room, and it seems impossible to have an intelligible conversation. You want to talk to your friend so you walk over to another room with them where it is quieter. Well this is the same idea behind signal modulation. We just walk our **message** to a different space in the frequency spectrum. The **carrier frequency** of our modulated signal is where we want to stand. A **channel** is the room that we are in which is inside some building, or **frequency band**. The **bandwidth** of our message is the range (varying timbre) of our voices. You and your friend can hear all the people in the party room still so their bandwidth extends the **guardband**, walls, and prohibits complete **channel separation**. FM is one technique we can use to relocate our message.

2.2.1 Definition

We define an FM signal as Eq. 1. Modulated signals are typically represented as $\phi(t)$ with some subscript describing the nature of modulation. An FM signal can be described as a sinusoid with constant carrier amplitude and time-varying frequency that is proportional to some integral gain of the message signal.

$$\phi_{FM}(t) = A_c \cos\left[\omega_c t + k_f \int_{-\infty}^{t} m(\lambda) d\lambda\right] \tag{1}$$

2.2.2 Modulation Index

The FM modulation index β, also known as the frequency deviation ratio, is defined as the ratio of the frequency deviation, of the modulated signal, to the message signal bandwidth as given by Eq. 2. The frequency deviation describes how far our modulated signal can be displaced from the carrier which effectively constrains the spectrum.

$$\beta = \frac{\Delta f}{B} = \frac{\Delta\omega}{2\pi B} \tag{2}$$

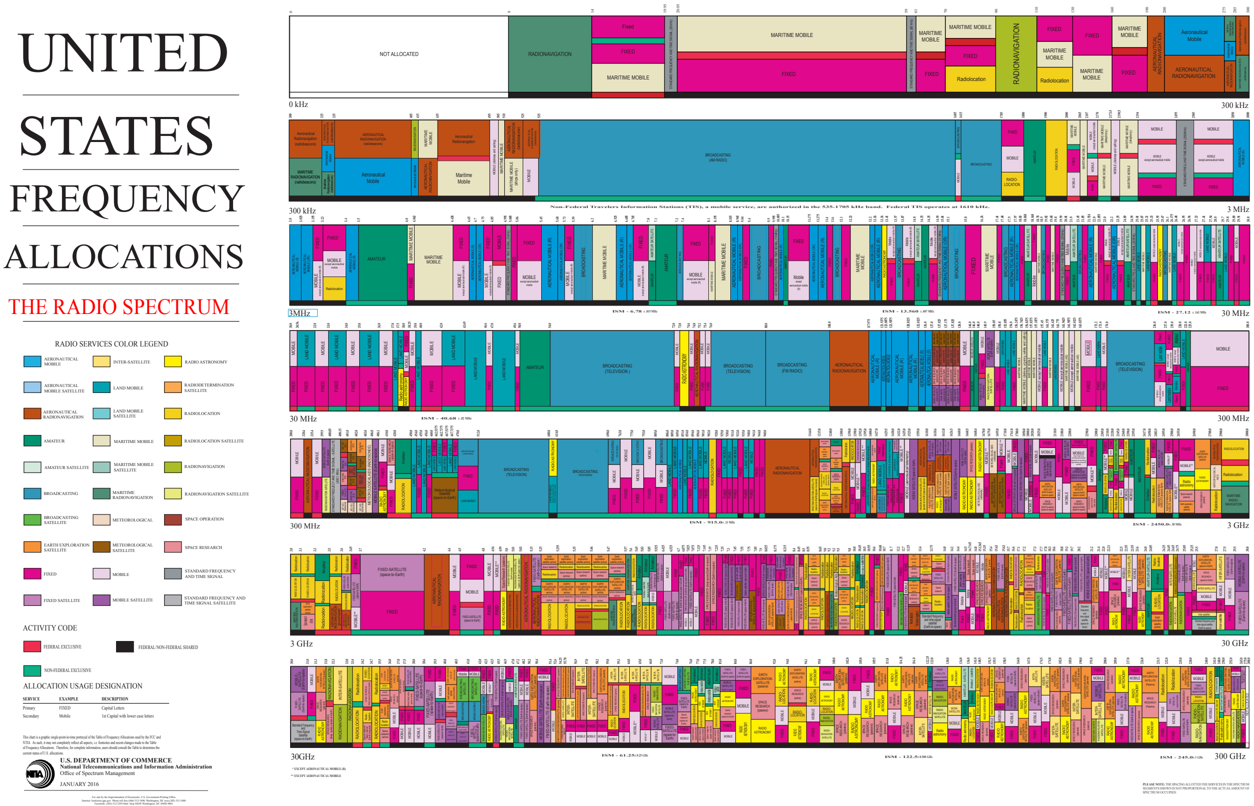

Figure 4: United States Frequency Allocations, Federal Communications Commission (FCC)

2.3 Antenna Theory

We are not going to worry too much about antenna theory here since this is a study of software-defined radio. You might notice that the antennas provided in your kit are fairly small. There are many types of antenna and in general the length of an antenna is inversely proportional to its intended operation frequency. Recall that the frequency operating range of plutos goes from 324 MHz to 3.8 GHz. The antennas that are provided in our kit were designed for the range of operation specified, but we want to *hack* the pluto to go to the lower frequency FM broadcast band. We might achieve better performance by creating an antenna that is designed for the lower frequency. There are a number of different antenna parameters that we can measure to characterize performance such as: gain, bandwidth, radiation pattern, polarization, and impedance. You should have some familiarity with gain and impedance concepts from prerequisite coursework, so we will stick to observing gain in the frequency spectrum and impedance in the antenna construction.

2.3.1 Whip Antenna

We will create a whip, or monopole, antenna. Recall from circuit theory that maximum power transfer occurs when the impedance of the source and load are matched. For a monopole topology, we can approximate an impedance inverter by creating an antenna that is of quarter-wave length, $\lambda/4$, for reasons that are beyond the scope of this class (electromagnetism and our good friend Maxwell). The quarter-wave length of our monopole antenna is given in meters by Eq. (3) where f is our target frequency in Hertz and c is the speed of light, approximately $3 \times 10^8 m/s$.

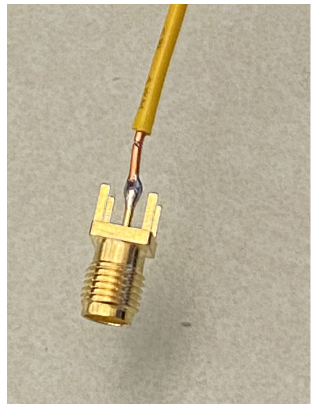

Figure 5: Whip antenna construction

$$l = \frac{c}{4 \times f}(meters) \tag{3}$$

There is a ten-week treatment of antenna theory offered at Cal Poly, EE 533 usually taught by Professor Arakaki in the spring, if you are interested in expanding your mind.

2.3.2 Antenna Construction

The antenna that I intend for us to design is composed of an insulated copper wire soldered to the center conductor of an SMA female connector. The copper wire is cut to a length such that the total length of hard-wire link and copper wire matches the monopole quarter-wave antenna length. Again this is probably not the greatest antenna ever made, but it does a fair job of showing us the importance of the length of an antenna. My whip antenna is shown in Figure 5.

2.4 Nyquist-Shannon Sampling Theorem

You should know by now that if we do not sample a signal at least twice the highest frequency present then we introduce the possibility for aliasing. In the frequency spectrum, this manifests itself as a remapping of content. To observe this effect, we will downsample the received signal until we notice changes in the frequency spectrum. Due to signal processing constraints, I have provided a Simulink model to capture the signal and a separate model for the spectrum analysis.

2.5 Lab Activity

We will create an FM broadcast receiver using a pluto, MATLAB, Simulink, one of the provided antennas, and a whip antenna that you create. Our target frequency will be Cal Poly's radio station KCPR, which is 91.3 FM in the San Luis Obispo area and has a carrier of 91.3 MHz.

2.5.1 Equipment

- pluto kit
- laptop with MATLAB and Simulink
- SMA connector and insulated copper wire
- solder and soldering station

2.5.2 Procedure

Be sure to document your findings (pictures and screenshots) along the way for your lab notebook.

- Connect the antenna from your kit to the Rx port of your pluto.
- Download the lab 2 source files and open the example FM capture Simulink model.
- Double-click the ADALM-PLUTO Receiver and tune the center frequency to KCPR.
- Lower the volume on your device so as to not damage your speakers and ears.
- Run the program and observe audio fidelity of the pluto kit antenna for FM band receiving.
- Determine the antenna length for a whip antenna optimized for 91.3 MHz, cut a piece of wire to approximately that length, and solder it onto the center conductor of an SMA.
- Connect your whip antenna to the Rx port and try running the Simulink model again.
- Try to get a good audio sample to use for spectrum analysis.
- Using the FM baseband plotter model, adjust the downsample factor until aliasing occurs.
- Discuss your findings in your lab notebook (set-up, results, conclusion).

2.5.3 Deliverables

1. (25 pts) Whip antenna length calculation and picture of hardware set-up with whip antenna.
2. (25 pts) Comparison of audio fidelity between antennas. You can tune in to KCPR here.
3. (25 pts) Spectrum analyzer screenshot of original signal and of aliased signal.
4. (25 pts) Feedback on the activity. Maybe there is something that you want changed or you came up with a simple hack that improved antenna performance. Or maybe you found the assignment fun, or too short/long.

