Lab 8 - SDR: Software Defined Radar

ECE531 – Software Defined Radio

Spring 2025

Daniel Gallagher, danielgallagher@arizona.edu
Department of Electrical & Computer Engineering
University of Arizona, Tucson AZ 85721

1 Overview and Objectives

This laboratory constructs and demonstrates a simple continuous wave (CW) Doppler radar system using the PlutoSDR. The continuous unmodulated transmit signal will provide no direct range measurements, but the Doppler beat frequency will be easily measured as a proportional shift to the transmit frequency. The MIT coffee can radar system designed by Gregory L. Charvat is a popular example of a CW radar implemented with discrete Mini-Circuits components in the RF front-end [1]. In this lab, we replace most of these components with a single software-defined radio.

2 Theoretical Preparation

The continuous wave radar technique is used with the least expensive kinds of radar, such as those used for traffic monitoring and sports [2]. CW radar is simple to understand and implement without the need for the coherent timing required for pulsed radar architectures. A block diagram of a simple CW radar system is shown in Figure 1. The radar receive antenna is located nearby the radar transmit antenna and is therefore a monostatic radar configuration.

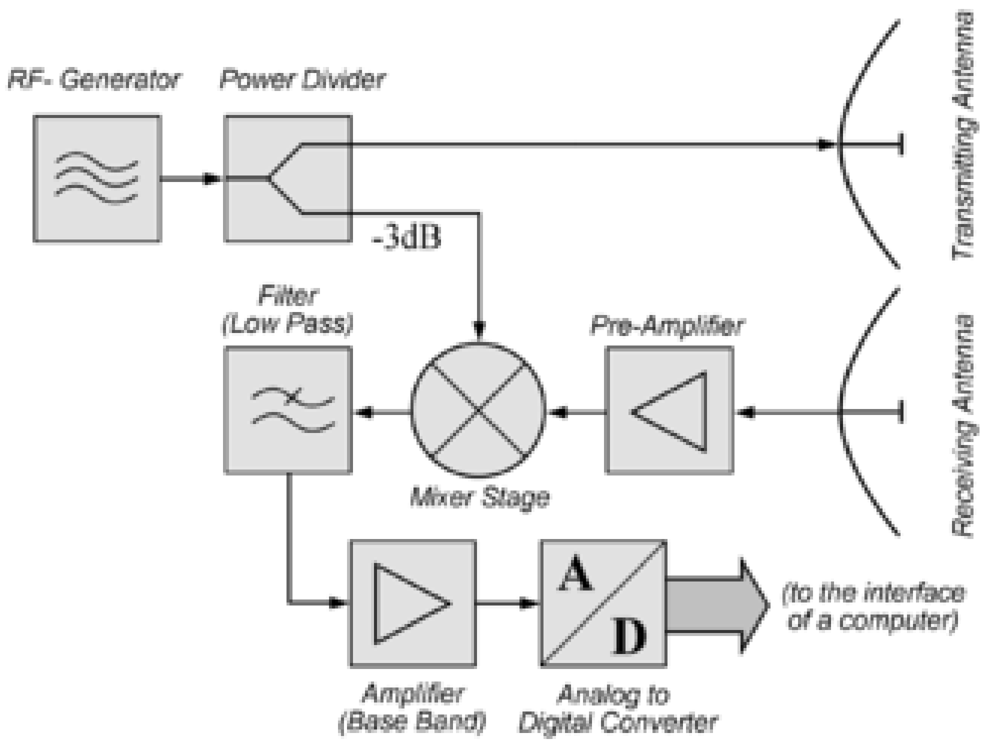

Figure 1: RF block diagram for a simple continuous-wave radar [2].

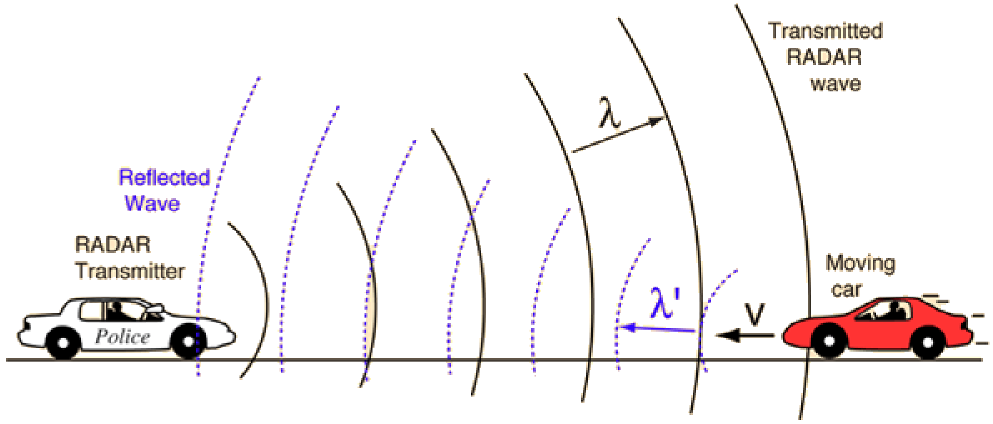

Figure 2: Illustration of Doppler shift in radar.

If the object is moving either toward or away from the transmitter, there is a slight equivalent change in the frequency of the radio waves, caused by the Doppler effect. The sign of the shift depends on the direction of the movement of the target. The sign is positive if the target is approaching the radar.

The Doppler shift can be defined as the difference between the transmit frequency and the frequency reflected off the moving object.

$$f_d = f_r - f_t \tag{1}$$

The Doppler frequency can be approximated for electromagnetic waves from the following equation,

$$f_d \approx \pm \frac{2v}{\lambda} \tag{2}$$

where the positive sign corresponds to an approaching target and the negative sign to a receding target.

Rearranging, the velocity of the moving object can be calculated from the measured frequency shift and the transmitted wavelength, λ, using Equation 3.

$$v \approx \pm \frac{f_d \cdot \lambda}{2} \tag{3}$$

CW radar without frequency modulation (FM) only detects the Doppler of singular moving targets. Stationary targets and slow moving clutter will not cause a doppler shift and will not be detected. Frequency-modulated continuous-wave radar (FM-CW) is capable of determining the distance, or range, to multiple targets.

3 Radar Implementation

Our goal is to implement this monostatic CW radar with the PlutoSDR. Students may use MATLAB, Simulink, or GNU Radio to implement the software-defined doppler radar. The instructor has chosen to approach the experiment using GNU Radio. A more complex radar system could be implemented using the GNU Radio Out-of-Tree (OOT) module *gr-radar* [3], however for this simple radar we can use primitive GNU Radio blocks.

3.1 CW Radar Transmitter

The transmitter section can be implemented with as little as two blocks; the signal source and PlutoSDR sink. The signal source provides complex I/Q samples to the hardware sink. The signal source is a sinusoidal waveform with baseband frequency of 10kHz, and is deliberately chosen to be slightly offset from DC or the Local Oscillator (LO). This offset helps distinguish the target return from static reflections. The software-defined radio should operate in the international 2.4 GHz ISM Band for regulations. This just happens to be S-band. A GNU Radio flowgraph for the transmitter is shown in Figure 3.

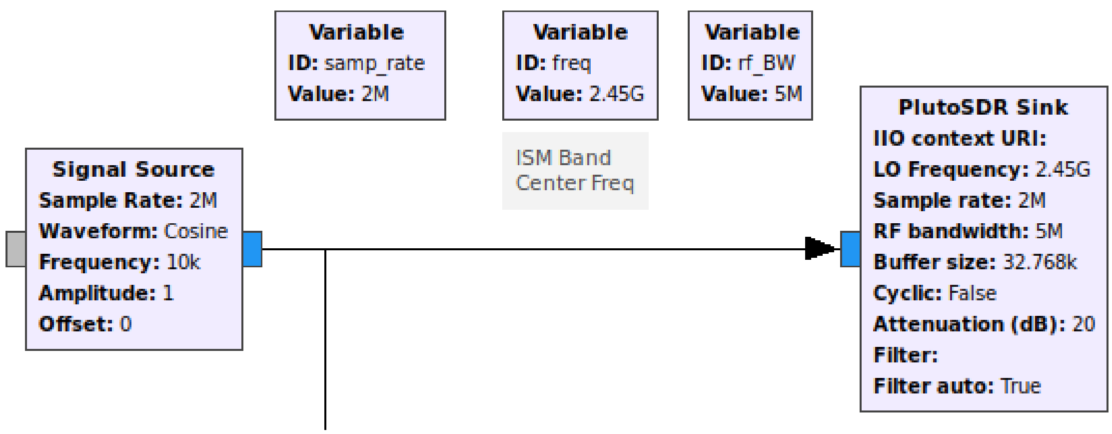

Figure 3: GNU Radio flowgraph for the CW radar transmitter.

4 Completing the SDR Radar

Complete the CW Doppler radar with your PlutoSDR and software of your choice. Refer back to the CW radar block diagram in Figure 1; the final implementation should function similarly. Below is some implementation guidance.

- In Figure 1, the transmit reference and receive signals are then mixed together by an Analog Mixer. The digital counterpart of an analog mixer is the Multiply Conjugate Block, which produces the beat frequency proportional to the Doppler shift.
- It may prove helpful to normalize the reference and receive signal. In GNU Radio this can be achieved with an AGC block.
- The pre-amplifier would ideally be a low noise amplifier (LNA), but the PlutoSDR hardware AGC will help achieve similar functionality for the purposes of this lab.
- The observable Doppler frequency will be much lower than the sampling rate. Consider using a decimating low pass filter to focus on the frequency range of interest.

4.1 Example Radar Test Results

Below are example results for the CW Doppler radar implemented using a PlutoSDR, GNU Radio, and a small fan as a target, as seen in Figure 4. The fan offers a good repeatable Doppler target, but the SDR radar can also detect waving hands or aluminum foil. The example results shown in Figure 5 clearly show the three fan speeds. The slowing fan blades are also apparent when turning off the fan. In this plot, the DC component was removed digitally using a DC-block and the waterfall display thresholds adjusted.

Figure 4: Doppler Radar Test Setup using PlutoSDR and fan.

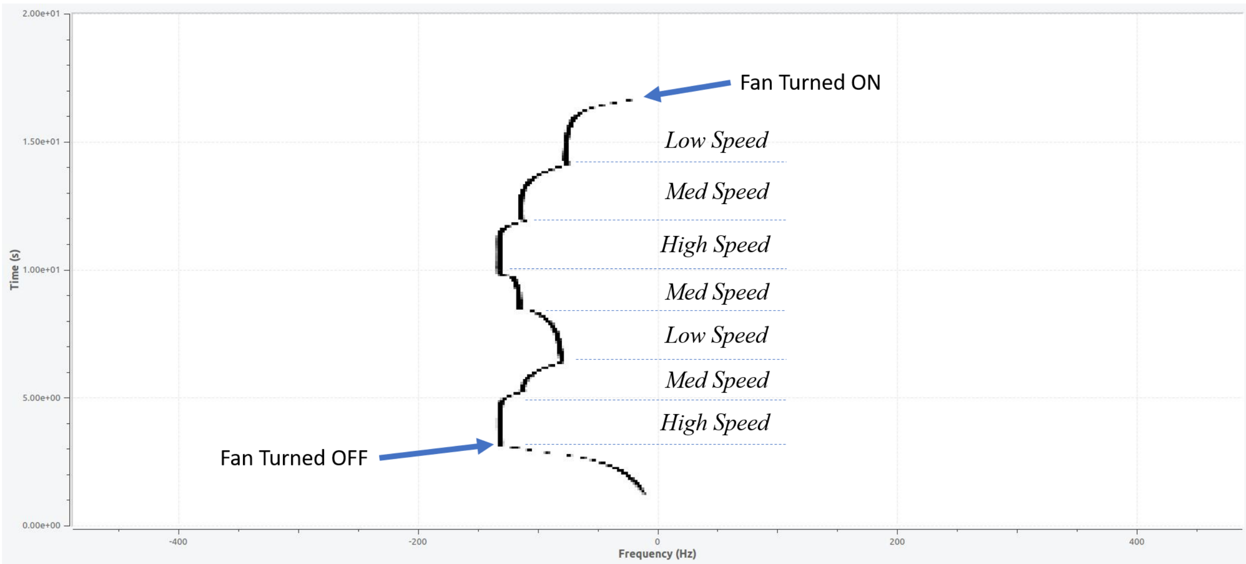

Figure 5: GNU Radio waterfall plot with fan turned on at low, medium, and high speed, with time on y-axis and frequency on the x-axis. The frequency shift is the result of the fan blade velocity.

5 Questions

1. Build your own CW radar with the PlutoSDR. What target(s) did you test against? Discuss results and observations in your lab report.
2. Explain more than one reason for the strong return at zero Hertz (DC)?
3. What physical changes can be made to reduce the DC return?
4. For CW radar, as implemented in this lab, how would an interfering signal affect its operation? What design changes can be made to improve its resilience to interference?
5. How can the system be changed to achieve range measurements?
6. What is the approximate angular velocity of the objects measured?

Possible Implementation Solution

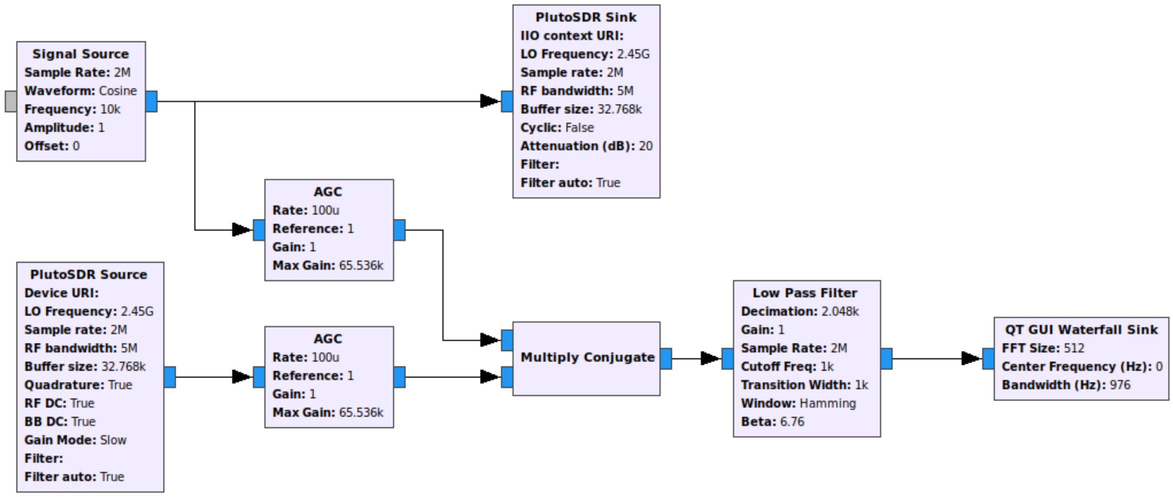

Figure 6: GNU Radio flowgraph for the CW radar.

6 Lab Report Preparation & Submission Instructions

Laboratory reports should be uploaded as PDF or Word documents outside of a zip archive and formatted as follows:

- Cover page: includes course number, laboratory title, name, submission date.
- Table of Contents, List of Tables, List of Figures. (Optional)
- Lab Results:
  - Introduction to the laboratory experiment, including a brief description of the objectives and goals.
  - Detailed explanation of the laboratory experiment, including the design, implementation, and testing of the system.
  - Results and discussion of the laboratory experiment, including captured outputs, observations, and responses to laboratory questions.
  - Conclusions to the overall lab that discuss meaningful lessons learned and other takeaways from the assignment. (Important)
- Upload source files with report submission. You may additionally include select code source listings in your report to highlight techniques used.
  - Note: Python files autogenerated from GNURadio do not need to be uploaded if the .grc files are included.

Remember to write your laboratory report in a detailed and descriptive manner, explaining your experience and observations in such a way that it provides the reader with some insight as to what you have accomplished. Furthermore, please include images and outputs wherever possible in your laboratory report document.

References

[1] G. L. Charvat, J. H. Williams, A. J. Fenn, S. Kogon, and J. S. Herd. (2011) Build a small radar system capable of sensing range, doppler, and synthetic aperture radar imaging. Massachusetts Institute of Technology: MIT OpenCourseWare. License: Creative Commons BY-NC-SA. [Online]. Available: https://ocw.mit.edu

[2] Wikipedia, "Continuous-wave radar — Wikipedia, the free encyclopedia." [Online]. Available: https://en.wikipedia.org/wiki/Continuous-wave_radar

[3] S. Wunsch, "GitHub: GNU Radio Radar Toolbox," September 2014. [Online]. Available: https://github.com/kit-cel/gr-radar