Object detection using a software defined radio with FMCW RADAR technology

ANTON ESPEDALEN
SVANTE LYSMARK
JUREK PREISEGGER

DEPARTMENT OF ELECTRICAL ENGINEERING
Chalmers University of Technology
Gothenburg, Sweden 2024
www.chalmers.se

Bachelor's thesis 2024

Object detection using a software defined radio with FMCW RADAR technology

Anton Espedalen
Svante Lysmark
Jurek Preisegger

Department of Electrical Engineering
Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden 2024

Object detection using a software defined radio with FMCW RADAR technology

Supervisor: Jafar Banar, Department of Electrical Engineering
Examiner: Thomas Eriksson, Department of Electrical Engineering

Bachelor Thesis 2024
Department of Electrical Engineering
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Abstract

In today's society, the rate of development of sensory units in vehicles is increasing rapidly. This is a result of the autonomation of vehicles becoming increasingly more common. RADAR sensors are used in these autonomous vehicles, and due to safety, the RADAR sensor needs to be able to detect multiple different objects in front of the vehicle while driving.

This paper presents a method to detect and approximate the distance of objects in a controlled environment using FMCW RADAR technology. FMCW has multiple advantages to pulsed radar including being able to detect closer objects and not needing to be turned off during ranging. The ranging of targets was done with a frequency difference of the echoed signal and the direct signal between one transmitter and receiver module. It was found that multiple objects could be detected and the distance of them could be approximated in a single measurement. The accuracy of the approximation was 10.1 meters of the measured distance due to the limited sample rate, which impedes resolution.

The method presented in this paper works as a proof of concept that an FMCW RADAR system is possible by using the ADALM Pluto SDR module. However, for this system to be implemented in any real-life scenarios the sample rate needs to be increased which would lead to a better resolution and therefore a better approximation of the distance to a target.

Keywords

- Bandwidth - Difference between lowest and highest frequency of the signal
- Carrier Frequency - The frequency at which the electromagnetic field propagates
- FFT - Fast Fourier Transform
- FMCW - Frequency Modulated Continuous Wave
- Lobe - The radiation pattern and direction of the antenna
- Nyquist Theorem - $f_s \geq 2f$, where f_s is the sample rate and f is the wave's frequency
- RADAR - Radio Detection And Ranging
- RF - Radio Frequency
- Rx - Receiver
- Sampling Frequency - The rate at which a continuous signal is sampled
- SDR - Software Defeined Radio
- SWO - Sample Window Offset
- Tx - Transmitter

Contents

1 Introduction

In the modern transportation sector, sensory units in vehicles, especially cars, are becoming increasingly common. This is a product of many factors, the autonomation of vehicles and the price of components dropping being two of them [1]. The two main types of sensors used in automated vehicles are exteroceptive- and proprioceptive sensors, which are sensors for measuring internal and external properties, respectively. Looking at the exteroceptive sensors there are multiple ways of obtaining information where the most common ones are LiDAR, RADAR, cameras, and ultrasonic sensors. The RADAR systems are short range, physically robust, and mostly responsible for driver assistance systems such as cruise control and collision detection. In an attempt to mimic the RADAR system on a vehicle, this project will revolve around creating an affordable short range object detection using a RADAR system.

2 Technical Background

The goal of this project is to configure a radio module to be able to detect objects relative to the modules, essentially creating the software for a RADAR system. By processing the received signals through a software of choice, the distance of an object can be calculated. In general, a RADAR can calculate the range, velocity, and angle of a target relative to the RADAR station. The different components of a RADAR module are most often hardware, however, in this project, the base for the system will be an ADALM Pluto Software Defined Radio, SDR. The SDR module makes it possible to build radio equipment using software instead of hardware, in particular, to manage the functions of modulation and demodulation of the signals [2].

2.1 RADAR Technology

RADAR technology is a well-known concept in the civilian sector as well as the military sector and it is based on reflection, as depicted in figure 1. A transmitter emits a radio frequency (RF) electromagnetic (EM) wave to a region of interest and the receiver acquires the echo that appears after the wave is reflected from an object [3]. Since the speed of the wave is known, the distance to the target can be calculated by using the time delay of the echoed signal. The technology is evaluated from the electromagnetism theory first elucidated by James Clerk Maxwell, at Cambridge in 1855 [4]. Maxwell further developed the theory to predict that an electromagnetic field could be propagated across space in the form of a wave traveling with the speed of light. The first practical demonstration of this theory was not accomplished until 8 years after Maxwell's death, by the experiments of Heinrich Hertz in Germany in 1886. After Maxwell's innovation and Hertz's practical experiment, RADAR technology has split up into two main categories, pulsed and continuous wave.

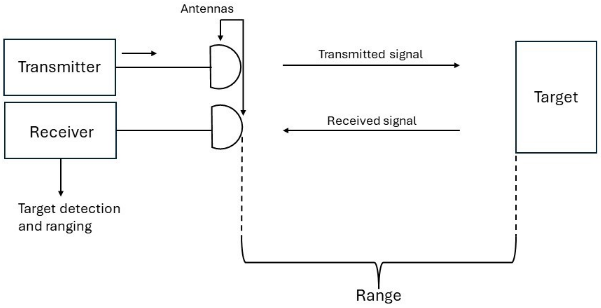


Figure 1: Basic radar principle

2.1.1 Pulsed RADAR

Pulsed radar technology emits a signal in repetitive cycles of short pulses which is radiated into space through either one or multiple antennas [5]. During the transmission of a pulse, the receiver antenna is turned off. When the transmission of the pulse is completed, the transmitting antenna is turned off and the receiver antenna is turned on. As a result of the receiver being turned off, there will be a short period where the RADAR is not getting any information, creating a blindspot. The theory of pulsed RADAR is depicted in figure 2. The received signal will have a time delay, denoted as τ. This time delay is used to obtain the range to the object by using the following equation $R = \frac{c \cdot \tau}{2}$, where c is the speed of light, R is the distance to the object and τ is the time delay of the received echo signal.

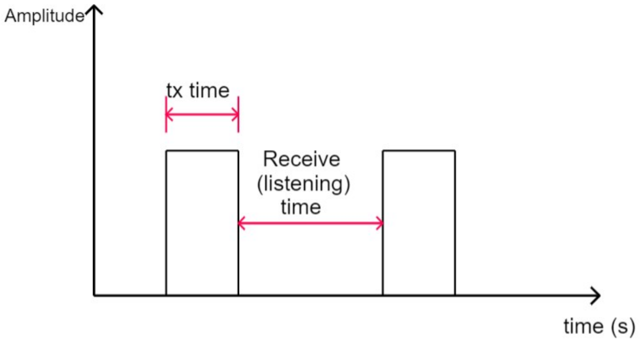


Figure 2: Pulsed radar principle

2.1.2 Continuous Wave RADAR

Continuous Wave (CW) RADAR transmits a high-frequency signal continuously while the echo signal is received and processed at the same time [6]. This is visualized in figure 3. CW RADAR technology works differently from pulsed RADAR technology, in particular, the receiver- and transmitter antenna are always turned on as opposed to the pulsed RADAR technology. The distance to a target is calculated in the same way as in pulsed RADAR, namely through the equation $R = \frac{c \cdot \tau}{2}$. The CW technology includes different types where the signal is modulated in different ways, one of them is Frequency Modulated Continuous Wave, FMCW.

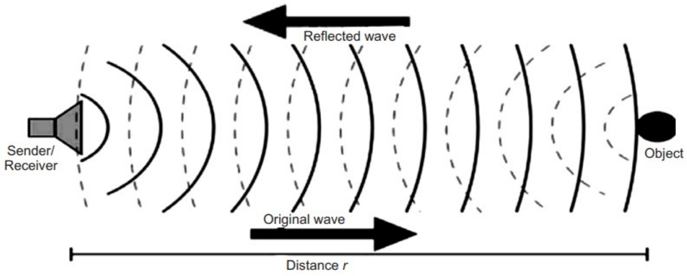


Figure 3: Continuous wave radar principle [7]

2.2 FMCW RADAR

A Frequency Modulated Continuous Wave (FMCW) RADAR is a technology for obtaining distance to an object by modulating the frequency of a continuous signal [8]. The frequency modulation of the continuous signal can be made in two different ways, linear and sinusoidal. Linear frequency modulation is the most versatile. For that reason, the FMCW radar is almost all concentrated on the linear type, and because of that this project will only include linear FMCW.

A FMCW RADAR works by emitting a continuous sinusoidal wave, where the frequency is modulated according to a sawtooth function. The frequency increases linearly from a base frequency and while reaching a defined maximum frequency it decreases very sharply back to the base frequency [9]. This procedure repeats multiple times creating frequency chirps, as depicted in figure 4. The transmitter and receiver are working simultaneously meaning there will always be two frequencies at any given time, the signal directly from the transmitter and the signal reflected from the target. As seen in figure 4, there is a frequency difference between the transmitted signal and the received signal which is used to calculate the distance to the target.

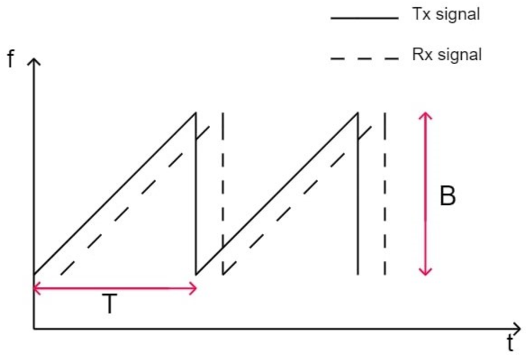


Figure 4: Chirps of linear FMCW-radar

2.3 ADALM Pluto SDR

The SDR module, seen in figure 5, is an educational tool available from Analog Devices Inc that can be used to introduce fundamentals of Software Defined Radio, Radio Frequency, or Communications as topics in electrical engineering in a self-lead setting [10]. It offers one receive channel and one transmit channel that can operate in full or half duplex, which means data can go in both directions simultaneously or just in one direction at a time.

The SDR module can be configured through software in multiple ways to act independently or as a part of a system. The SDR can be used to experiment with different wireless communication protocols like Wi-Fi or Bluetooth because it can measure different RF signals' amplitude, frequency, and phase.

Figure 5: ADALM Pluto Software Defined Radio [10].

2.3.1 ADALM Pluto SDR Specifications

From factory the SDR module can generate and measure frequencies between 325 MHz to 3.8 GHz at up to 61.44 Mega Samples per second and with an instantaneous bandwidth of up to 20 MHz, which makes it a good tool for all kinds of topics the user wants to use it for [10]. The module is compatible with many different software programs like MATLAB, Python, C#, and C++ over a USB connection, making it suitable for multiple target groups.

2.3.2 Configuration Options of ADALM Pluto

The described factory default with its properties are shown in table 1 where LO means Local Oscillator. Beneath, another configuration is shown with expanded range for both center frequency and bandwidth.

Table 1: Available tuning options for ADALM Pluto [11]

RF Transceiver	**LO Tuning Range**	**Bandwidth**
AD9363 (factory default)	325 – 3800 MHz	0 - 20 MHz
AD9364	70 – 6000 MHz	0 - 56 MHz

2.3.3 Local Oscillators

Inside each SDR module there is a local oscillator that generates the RF signal, the oscillators will due to manufacturing have minor frequency differences. "Because of manufacturing and environmental effects, such as aging, temperature, and power supply fluctuations, platform acceleration, radiation effects, etc., an oscillator's true output frequency often deviates from its nominal value specified by the manufacturer" [12]. When using two different modules, both are configured to the same center frequency but the two oscillators could have small differences as stated earlier. This leads to when the SDR modulates the center frequency to the bandwidth, the sent frequency and the received could be different.

2.4 Analythics of an FMCW RADAR System

As described in section 2.2 the transmitted frequency increases over time. The steepness of the linear increase is a sweep over bandwidth B in a time T, where T is the chirp time. The linear increase of the frequency can be written as equation 1 where f_c is the carrier frequency [9].

$$f(t) = f_c + \mu t \tag{1}$$

As described above the steepness of the linear increase depends on bandwidth and chirp time, therefore the chirp rate is calculated as equation 2.

$$\mu = \frac{B}{T} \tag{2}$$

To determine the appropriate angle $\Phi(t)$ for achieving the desired frequency sweep, equation 1 must be integrated with regard to time as equation 3 describes. In the equation below, Φ_0 is the phase at t=0.

$$\Phi(t) = 2\pi \int_0^t f(t)\,dt = 2\pi\left(f_c t + \frac{\mu t^2}{2}\right) + \Phi_0 \tag{3}$$

As described in section 2.2, the transmitted waveform is a sinusoidal function of the form described in equation 4.

$$s_{tx}(t) = A_{tx}\text{exp}(j\Phi(t)) = A_{tx}\text{exp}\left(j(2\pi(f_c t + \frac{\mu t^2}{2}) + \Phi_0)\right) \tag{4}$$

In equation 4, A_{tx} is the transmitted wave amplitude.

The received signal will have a time delay, τ, which is the time it takes for the wave to propagate to a target and bounce back to the rx-antenna. The received function will therefore follow equation 5.

$$s_{rx}(t) = A_{rx}\text{exp}(j\Phi(t-\tau)) + w(t) \Rightarrow$$

$$\Rightarrow A_{rx}\text{exp}(j(2\pi(f_c(t-\tau) + \frac{\mu(t-\tau)^2}{2}) + \Phi_0)) + w(t) \tag{5}$$

In equation 5, $w(t)$ is the Additive White Gaussian Noise, AWGN, which is a noise that is located in all transmitters and receivers. The transmitted signal and the received signal are mixed to obtain $z(t)$, the mix is made in such way that the transmitted wave is multiplied by the conjugate of the received wave.

$$z(t) = s_{tx}(t) \cdot s_{rx}(t)^* = A_{tx}A_{rx}\text{exp}\left(j2\pi(\mu t\tau + f_c\tau - \frac{\mu\tau^2}{2})\right) + w_z(t) \tag{6}$$

Time delay, τ, can be calculated for 2 cases, if the target is stationary or if it is moving. The two cases is calculated according to equation 7.

$$\begin{cases} \tau_{stationary} = \frac{2R}{c} \\ \tau_{moving} = \frac{2vt}{c} \end{cases} \tag{7}$$

R is the distance from the antenna to the target, v is the velocity of the target and c is the speed of light. To obtain a total time delay, both cases are added together. This results in $\tau_{total}=\frac{2(R+vt)}{c}$. By substituting τ in equation 6 with τ_{total} the following expression of z(t) is obtained.

$$z(t) = A_{tx}A_{rx}\text{exp}\left(j2\pi\left(\mu t\frac{2(R+vt)}{c} + f_c\frac{2(R+vt)}{c} - \mu\frac{(2(R+vt))^2}{2c^2}\right)\right) + w_z(t) =$$

$$A_{tx}A_{rx}\text{exp}\left(j4\pi\left((\frac{\mu R}{c} + \frac{f_c v}{c} - \frac{2\mu Rv}{c^2})t + (\frac{\mu v}{c} - \frac{\mu v^2}{c^2})t^2 + \frac{f_c R}{c} - \frac{\mu R^2}{c^2}\right)\right) + w_z(t)$$

Since it is known that c≫v, c^2 ≫c, c≫R and because of the very short chirp time $vt \ll R$, some simplifications of the expression above can be made and the final expression of the mixed signal $z(t)$ is described in equation 8.

$$z(t) = A_{tx}A_{rx}\text{exp}\left(j2\pi\left(\frac{2f_c v + 2\mu R}{c}\cdot t + \frac{2f_c R}{c}\right)\right) + w_z(t) \tag{8}$$

The frequency of the mixed signal, $z(t)$, is a sum of the Doppler frequency caused by a moving target and the beat frequency. The Doppler frequency of a FMCW radar system is $f_D = \frac{2f_c v}{c}$ and the beat frequency is $f_{beat} = \frac{2\mu R}{c}$.

For this project, a moving target will not be included. This summarizes that the doppler frequency will be equal to zero and the beat frequency is the only frequency that will be analyzed to obtain the distance to the target. Beat frequency is the frequency difference between the sent and the received signal at any given time.

$$f_{stationary} = \frac{2\mu R}{c} \tag{9}$$

2.5 FMCW Ranging

In theory, the beat frequency, f_{beat}, calculated in section 2.4 is the difference between the echo signal's frequency and the direct signal's frequency [9]. In figure 6, the red arrow is the direct signal and the black arrow is the echo signal. To obtain the appropriate f_{beat} in order to calculate the correct range to the target, a thorough analysis of the received and transmitted signal has to be made.

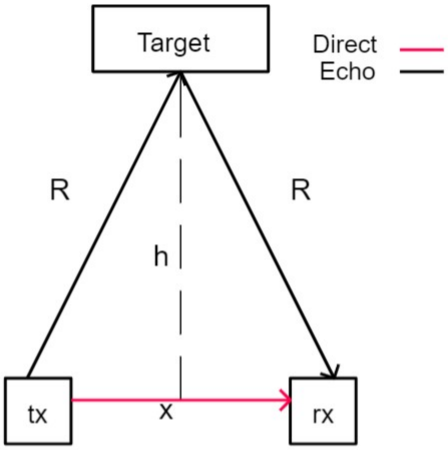


Figure 6: Illustration of direct signal and echo signal of the RADAR system

The SDR module works by recording a set number of samples. In theory, the recording of samples at the receiving end of the system starts at the same time as the start of the transmission. However, if there is a time delay between the recording of samples and the start of transmission an offset in the sample window will appear. Throughout this project it will be referred to as Sample Window Offset, SWO.

If the computational performance tends to infinity and delays of information are equal to zero, the relation between the transmitted signal and received signal can be as visualized in figure 7a. Where the SWO is equal to zero, and therefore the sample window starts at the same time for both the transmitted signal and the received signal [13]. In practice, there are some delays in the code and the hardware used for the testing which leads to a sample window starting slightly after the transmission has started, as visualized in figure 7b. There are also two other kinds of delays in practice, delay of the echo signal and delay of the direct signal which need to be taken into account while calculating the range.

(a) Tx/Rx matched chirps

(b) Tx/Rx unmatched chirps

Figure 7: Tx/Rx matched/unmatched chirps.

In equation 6 in section 2.4, the transmitted signal is multiplied with the conjugate of the received signal even if there is a SWO. This will cause a frequency shift in the recorded signals, moving both the echo and direct signal on the bandwidth spectrum an amount related to the size of the SWO. However, since they are both shifted by the same frequency and only the difference between them is of interest the SWO does not impact the range calculation.

After the conjugate multiplication has been carried out, the frequency shift between the echoed signal and the direct signal is used to calculate f_{beat}. However, when the direct signal is approaching the receiving antenna, its frequency will increase due to the nature of FMCW technology. This frequency increase needs to be calculated because the beat frequency should be equal to the frequency difference between the echoed signal and the frequency that is currently being sent. Therefore the frequency increase needs to be subtracted from the frequency of the direct signal. The frequency increase will depend on μ, f_s, and the number of samples the delay is modeled after, N_x, according to equation 10.

$$f_{increase} = \frac{N_x \cdot \mu}{f_s} \tag{10}$$

During the ranging, the distance x as seen in figure 6 is known, as it is equal to the distance between the Tx- and Rx-antenna. However, N_x will only be known during simulation meaning another equation for $f_{increase}$ will be needed for tests in practice. Equation 9 for f_{beat} in section 2.4 can be used for this by substituting R for x and because $f_{increase}$ is not a round trip as f_{beat} is, the integer 2 in equation 9 is removed which leads to an expression for $f_{increase}$ as equation 11.

$$f_{increase} = \frac{x\mu}{c} \tag{11}$$

To obtain an appropriate expression of f_{beat} the direct signal has to be modulated by subtracting the direct signal with the calculated $f_{increase}$. Since f_{beat} is the difference between the echo signal's frequency and the direct signal's frequency the expression of the beat frequency is as equation 12.

$$f_{beat} = f_{echo} - (f_{direct} - f_{increase}) \tag{12}$$

By substituting f_{beat} in equation 9 with f_{beat} as expressed in equation 12, equation 13 is obtained.

$$\frac{2\mu R}{c} = f_{echo} - (f_{direct} - f_{increase}) \tag{13}$$

Deriving R from equation 13 the distance to the target can be calculated according to equation 14.

$$R = \frac{c}{2\mu} \cdot (f_{echo} - f_{direct}) + \frac{x}{2} \tag{14}$$

2.6 Range and Resolution

The range and resolution of the RADAR is dependent on the bandwidth, chirp time, and the sampling frequency. However, due to the nature of RADAR, a greater bandwidth yields a greater resolution but a worse range. In this project the resolution is the more important factor, therefore the bandwidth will be as large as possible. According to the Nyquist Theorem, the sample frequency needs to be at least double that of the bandwidth to avoid aliasing.

The sampling frequency is limited to 30 MHz by hardware which yields a maximum bandwidth of 15 MHz. To guarantee no aliasing, a bandwidth slightly smaller than the maximum available was chosen. By using a bandwidth of 14.8 MHz, a chirp time of 8 μs, a sampling frequency of 30 MHz, and the speed of light, c, as $3 \cdot 10^8$ m/s the maximum range and resolution could be calculated using equation 15 and 16.

$$R_{max} = \frac{cT_{acq}}{4B_{eff}} \cdot f_s \qquad [9] \tag{15}$$

$$\Delta R = \frac{c}{2B_{eff}} \tag{16}$$

With the usage of the values above the resulting range and resolution is seen below.

$$\begin{cases} \Delta R = 10.1 \text{ m} \\ R_{max} = 1216 \text{ m} \end{cases}$$

2.7 MATLAB Toolboxes

To use the ADALM-Pluto SDR with MATLAB multiple toolboxes had to be installed, note that all versions of the toolboxes were at their most recent version during this project. The following four toolboxes are the ones used for this project.

1. Communications Toolbox
2. Communications Toolbox Support Package for Analog Devices ADALM-Pluto Radio
3. Signal Processing Toolbox
4. Simulink

The first toolbox is needed for SDR and all antenna communications, it also makes it possible to create multiple different waveforms. The second toolbox is an extension of the communications toolbox tailored for the Analog Devices ADALM-Pluto Radio which is required to get MATLAB to be able to communicate with and control the SDR that is used for this project. The Signal Processing Toolbox is being used to process the received signals. The most important algorithm enabled by this toolbox is the Fast Fourier Transform (FFT). This algorithm computes the Discrete Fourier Transform (DFT) of a sequence which essentially takes it from the time domain to the frequency domain making it easier to distinguish frequency differences. Finally, the Simulink toolbox was used to access the block-based system of Simulink.

2.8 Internal Leakage

One prominent issue with the ADALM Pluto SDR is its leakage inside the module. Since the frequencies used in this project are high, namely 70 MHz to 6 GHz, there will be a significant leakage inside the module. The cause of this is that all conductors can act as antennas at higher frequencies meaning they can radiate electromagnetic waves [14]. This means practically that the conductor associated with the Tx part inside the module will transmit its signal within the module and the conductors associated with the Rx part will receive these signals within the module. The effect of this is that the received signal will consist of what the Rx antenna has picked up as well as what the Rx conductor inside the module has picked up. If the power of the reflected signal is low relative to the leakage it will be hard to analyze because it is impossible to differentiate the leakage from the received signal [15].

3 Aim and Scope

This project requires a linear development of different technologies and theoretical knowledge to be completed. For faster progress, a clear structure will be needed. This will be achieved through adding boundaries and subgoals which will be presented in the following section.

3.1 Aim

The purpose of this report is to explore and analyze the usage of an SDR module along with FMCW radar technology for object detection. By integrating these techniques, we aim to develop an effective method for detecting objects in a given environment.

A significant aspect of the work will involve developing and implementing software to control and process signals from the SDR module. This includes designing and optimizing signal processing algorithms to extract relevant information from the received signals.

The subgoals for the project are presented, with numbers in chronological order.

1 Establish transmission and reception on the SDR through Simulink

2 Transmit a basic vector and receive it

3 Transmit a sine wave and receive its corresponding FFT

4 Transmit a sine wave with varying frequency (FMCW) and receive its corresponding FFT

5 Identify echo signal of a transmission

6 Target ranging and detection

7 Target triangulation

8 Create a User Interface, UI

3.2 Scope

As described in section 2.1.1 pulsed RADAR technology could create a blind spot while detecting objects at shorter distances. Because of this, it is decided that FMCW RADAR technology will be used for this project.

For this project to be feasible, certain limitations needs to be set when choosing targets. The targets will have a distance between each other larger than the resolution and the distances will not exceed the maximum range calculated in section 2.6. The targets will also be placed directly in front of the center of the RADAR system.

The SDR module needs software to be able to transmit and receive different types of data. The used software for this project will be Simulink and MATLAB. Simulink will be used for the first subgoal in section 3.1, and the rest of the subgoals will be accomplished using MATLAB.

4 Method

This project requires a well-structured method to be able to complete each subgoal presented in section 3.1. The subgoals for the project were solved in numerical order and the way the subgoals were achieved will be presented in this section.

4.1 Simulink

Simulink was chosen as a starting ground for the project because of the lack of necessary coding. It instead works by building block diagrams where each block has predetermined inputs. The first step was to set up a spectral analysis of the recorded noise from the receiver with the block diagram depicted in figure 8.

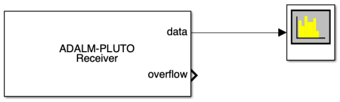
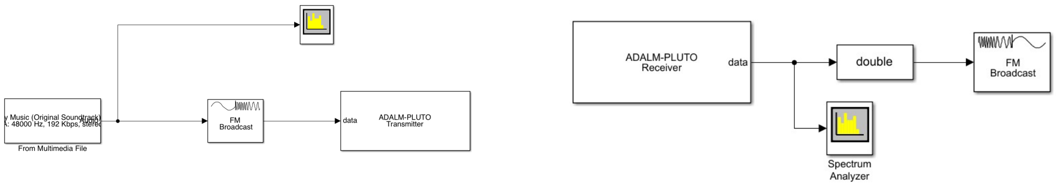


Figure 8: Spectral analysis of received noise

Afterward, the block diagrams in figure 9 were used to send data from one SDR module to another. In the transmitter, a audio file was translated with the FM broadcast block and thereafter broadcasted with the ADALM-PLUTO transmitter block. The signal was transmitted with a center frequency of 2 GHz and a sample rate of 240 KHz. In the receiver, the ADALM-PLUTO receiver block was set up with the same center frequency and sample rate as the transmitter. The received data was transformed into the data type double, afterwards, the FM-Broadcast block could decode the data to a sound file.



(a) Transmitter

(b) Receiver

Figure 9: Simulink block diagram of transmitter and receiver

4.2 Square Wave Transmission

To set up a transmitter and receiver in MATLAB the first step was to create a radio object tied to the physical SDR. Secondly, the radio attributes were created, such as center frequency and sample rate. An important requirement is the Nyquist Theorem, this needs to be fulfilled to not get aliasing. The previous steps were carried out with the help of the MATLAB documentation for various toolboxes stated in section 2.7. The next step was to create a signal to transmit from one device and receive on another. A square wave as seen in figure 10 was chosen, essentially a vector where half of the elements are negative ones and the other half are positive ones.

Since the input for the SDR needs to be complex, a small imaginary number was added to each element in the vector thus allowing it to be transmitted. The vector is just a single step but since the "transmitRepeat" function is used it becomes a continuous square wave. The transmission of a square wave was done with the MATLAB code shown in listing 3, Appendix A.

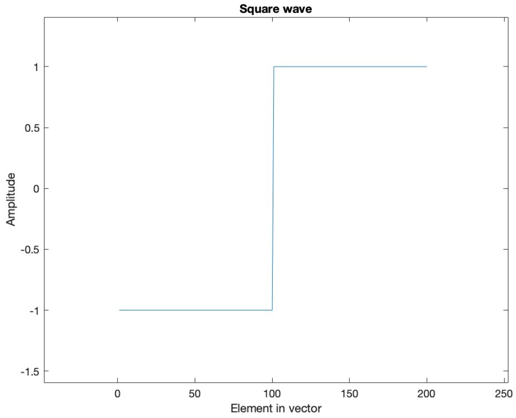


Figure 10: Transmitted square wave

4.3 Frequency Correction of Oscillators

To correct the possible frequency offset between two SDR modules a square wave was sent from one SDR to another using the same center frequency. An underlying sine wave was detected in the received signal and the frequency of it was calculated by using the number of samples of one period, N_f, and the sampling frequency. This was done through equation 17 where the underlying sine wave frequency, f, was derived.

$$f = \frac{f_s}{N_f} \tag{17}$$

Where N_f is the number of samples for one period of the underlying sine wave and f_s is the sample rate. To obtain the desired frequency offset in ppm, equation 18 was used.

$$\text{Offset} = \frac{f}{f_c} \cdot 10^6 \quad \text{[ppm]} \tag{18}$$

Where f_c is the carrier frequency. The offset can thereafter be corrected by using the code shown in listing 1, as a value between -200 and 200 in ppm.

```
'FrequencyCorrection','offset'
```

Listing 1: MATLAB command for synchronizing oscillators

4.4 Configuring ADALM Pluto in MATLAB

To increase the range for center frequency and bandwidth according to section 2.3.2, the MATLAB command in listing 2 was used.

```
configurePlutoRadio('configurationname','usbport')
```

Listing 2: MATLAB command for SDR configuration

Where "configurationname" is wich configuration the SDR should switch to and "usbport" is wich USB port its plugged into the computer.

4.5 Sine Wave Transmission

To transmit a sine wave, the only major changes in the code from section 4.2 was the transmitted vector and the display of the received signal. Instead of transmitting a square wave, a sine wave was transmitted and the received signal was displayed in the frequency domain instead of the time domain. It was created using the built-in sine wave function in MATLAB and transmitted both within the same module and between two different modules by using the “transmitRepeat” command in MATLAB. A sine wave with a frequency of 900 kHz was transmitted within a single module, and a sine wave with a frequency of 400 kHz was transmitted between two different modules. The received signal of both tests was displayed in the frequency domain using an FFT as seen in the MATLAB script in listing 4, Appendix A.

4.5.1 Linearly Increasing Frequency

To modulate the frequency for the sine wave linearly, to obtain FMCW technology, the built-in sine wave function in MATLAB was changed to a complex baseband equation which is illustrated in equation 19.

$$T_{x,wave} = Ae^{i2\pi ft} \tag{19}$$

Where A is the amplitude and the frequency, f, is modulated as described in section 2.4. The equation of the frequency sweep is seen in equation 20. The chirp rate, μ, is calculated by dividing the bandwidth, B, through the chirp time, T.

$$f = f_c + \mu t \tag{20}$$

The transmitted wave was created as equation 4 in section 2.4. Since this only represents one chirp the wave vector was repeated to become as long as the received vector. This was done since there could be possible transmission delays when using “transmitRepeat” and for MATLAB to be able to do the conjugate multiplication as described in section 2.4. The transmission of an FMCW signal was made by using the MATLAB code in listing 5, Appendix A.

4.6 Internal Leakage Measuring

As described in section 2.8, the SDR modules could have internal leakage that could impact other signals while testing. To measure this leakage a test with both a single module and two modules was made.

To ensure that the only received signal was from the possible leakage, the transmitting antenna was enclosed. This was done by encapsulating it in two tin cans. In the first tin can a small hole was made to fit the antenna, the small gap was then sealed using tin foil. The second can was then placed over the first one, essentially enclosing the antenna in metal. Subsequently, two types of experiments were made. In the first one, only the encapsulated SDR module was used as seen in figure 11a. In the second test, one module transmitted a signal to the encapsulated SDR module as visualized in figure 11b. Both of the tests were carried out using the MATLAB script in listing 4, Appendix A.

(a) Setup of leakage test with one module

(b) Setup of leakage test with two modules

Figure 11: Setups of SDR modules during leakage tests

In the test with the setup as figure 11a, the SDR was transmitting a sine wave of 300 kHz, and the resulting FFT was analyzed to check if there was an internal leakage from the module. In the test with the usage of the setup in figure 11b the encapsulated module transmitted a signal of 300 kHz, and the other module transmitted a signal of 400 kHz. The resulting FFT of the encapsulated SDR was analyzed to see how strong the leakage was relative to a direct signal from another module.

4.7 Simulated Ranging

To determine the distance to an object, the mathematics described in section 2.4 and 2.5 was first tested in a simulated environment. This was done to eliminate unwanted and uncontrollable variables by simulating the received signal. This will allow for quick modulation of the distance to the target in relation to the RADAR system as well as the distance between the Rx and Tx antenna.

The simulation was carried out with the MATLAB code in listing 6, Appendix A. Lines 29 through 44 are where the distances alongside the SWO are decided. Simulating varying distances and SWO is done by delaying the signals of a set amount of samples, resulting in a time and frequency difference. The distance the electromagnetic wave travels in N_{ew} number of samples can be calculated according to equation 21, where f_s is the sampling frequency, and c is the speed of light, here approximated to $3 \cdot 10^8$ m/s.

$$R = \frac{N_{ew}}{f_s} \cdot c \tag{21}$$

The received signal was simulated by adding a delayed echo signal with a delayed direct signal. By varying the delay of the echo the distance of the target to the radar system could be simulated, while the delay of the direct signal represents how far apart the Tx and Rx antennas were. During the simulations, the parameters seen in table 2 were used.

Table 2: Parameters of the simulated FMCW transmission

Parameter	Value
f_c	4.2 [GHz]
f_s	30 [MHz]
B	14.8 [MHz]
T	8 [μs]
μ	1.85 [THz/s]

In the first test, ranging of a target with a single module was simulated. The delay in the SWO signal was set to zero meaning instantaneous computational power. The direct signal strength was set to zero since the distance between the Tx and Rx antenna was smaller than the resolution computed in section 2.6. A delay of ten samples was added to the echo signal by adding a vector of ten zeroes in front of the echo signal vector. This means the simulated electromagnetic wave has traveled 100m according to equation 21. Since this signal represents the echo, the distance is a round trip resulting in the target being stationed 50m away.

In the second test, the delay of the echoed signal was kept at ten samples. The simulation was still made with the theory of using one SDR module but not instantaneous computational power. This means the strength of the direct signal was kept at zero while the SWO was set to 300 samples. This represents a time delay of 10 μs between the start of transmission and recording of the received signal according to equation 22.

$$t_{delay} = \frac{N_{SWO}}{f_s} \tag{22}$$

In the third test, two SDR modules were simulated, one acting as a transmitter and the other as a receiver. The direct signal was therefore accounted for since the distance between the two SDR modules was larger than the resolution. With the current settings, the resolution of the RADAR system was determined in section 2.6 to 10.1 meters, meaning the shortest delay possible that relates to a distance greater than the resolution is a delay of two samples. This is because the simulated sample delay is in integers. Therefore the delay of the direct signal was set to two samples. The distance between the two SDR modules was set to 20 meters by adding a delay of two samples to the direct signal. The delay of the echoed signal was kept at ten samples while the SWO was set to 200 samples. The last test was carried out with the same parameters as the third test but the strength of the direct signal was scaled down to one percent of the echoed signal. This was done to observe how different strengths of signals impact the resulting FFT.

In all tests, the conjugate multiplication of the received and transmitted signals was done. The resulting vector was then processed with an FFT and displayed to identify the direct and echoed signals. The frequency peaks of the direct and echo signal in the FFT were then used with equation 14 to calculate the value of the Range to the target.

4.8 Multi-Module Reflection Measuring

The decision to measure outside was made and the location chosen was a football field since the lack of objects close by will lower reflections and clutter. Because of the lack of accurate measuring tools, such as a measuring wheel, the distance to the targets was made using an approximated 1 meter steps. To avoid the reflected signal being overpowered by the leakage inside the module, the following tests will be conducted by using two separate SDR modules. This means two separate modules will act as the Rx- and Tx-antenna respectively for the RADAR system. The target will be in front of the two SDR modules as visualized in figure 12. For both of the following tests, the x-value was kept at 22 meters to have a margin to the resolution as described in section 4.7. The distance to the targets will be the h-value and the measured distance will be the R-value.

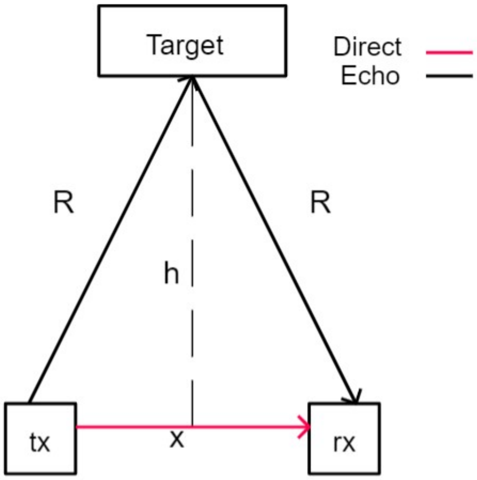


Figure 12: Illustration of RADAR system setup

Both tests were made with a RADAR system based on FMCW using the MATLAB code in listing 7, Appendix A. The transmitted signal will be an FMCW with the parameters according to table 2. To calculate the R value the same calculations and signal processing as in section 4.7 were used. To acquire the measured h-value Pythagoras theorem was thereafter used.

The first test was conducted at the short end of the field with only one target approximately 82 meters away. The target in question was a large building, however, there were some smaller objects in front of it causing cluttering. Since the antennas used on the SDR module are isotropic, the waves radiate in a sphere meaning there will be reflections from all directions. Therefore a tin-can was placed over the Rx-antenna to limit received reflections from the desired direction. Since the power of the received signal is relatively low, the gain settings were set to "AGC Fast Attack" meaning the module will amplify low-powered signals that quickly change frequency. For the direct signal, the leakage from the SDR modules will be utilized. The direct signal from the Tx-module will penetrate the casing of the Rx-module and radiate into the circuit. With this setup, a total of 5 measurements were made.

The second test was a replication of the third simulation test but with two targets as seen in figure 40 in Appendix B. The setup was similar to the aforementioned test with tin-cans over the Rx-antenna, however, for the first measurement the gain setting was set to "Manual" and the gain was thereafter set to zero. This was done for testing purposes of the different gain settings. For the remaining five measurements the gain setting was set to "AGC Fast Attack". This test was performed with two targets, one car approximately 50 meters away and another large building approximately 84 meters away from the SDR module. This test was designed for ranging with multiple targets instead of just one. With this setup, a total of 6 measurements were made.

5 Result

The following section will include the deliverables of the project according to section 4. This will include, but not be limited to MATLAB and Simulink plots as well as approximated distances of targets in relation to the RADAR system.

5.1 Simulink

The setup described in section 4.1 was used to analyze the spectrum of the received signal. The frequency spectrum of the noise is visualized in figure 13, it is a result of the setup as the block diagram in figure 8.

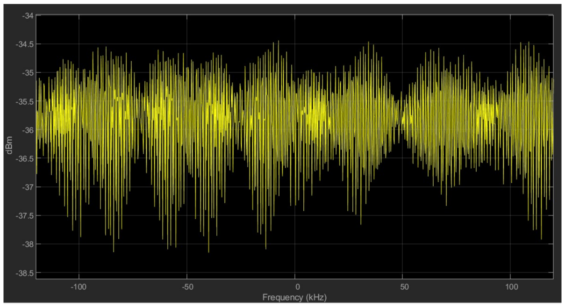
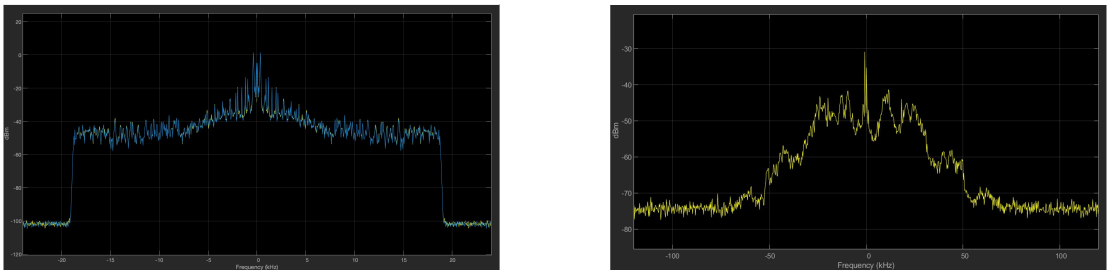


Figure 13: Spectrum of received noise

Transmitting an audio file between two SDR modules was done with the setup visualized in figure 9. When the audio file was played in the receiver it was possible to distinguish the transmitted audio, although it was noisy. The transmitted and received signals' spectrum is depicted in figure 14.



(a) Spectrum of the transmitted signal

(b) Spectrum of the received signal

Figure 14: Spectrum of received and transmitted signals in simulink

5.2 Square Wave Transmission

The transmission of a square wave within the same SDR module according to section 4.2 resulted in a received signal depicted in figure 15.

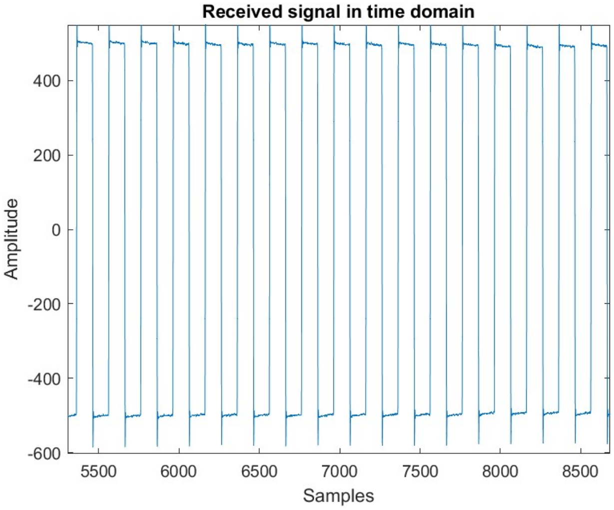
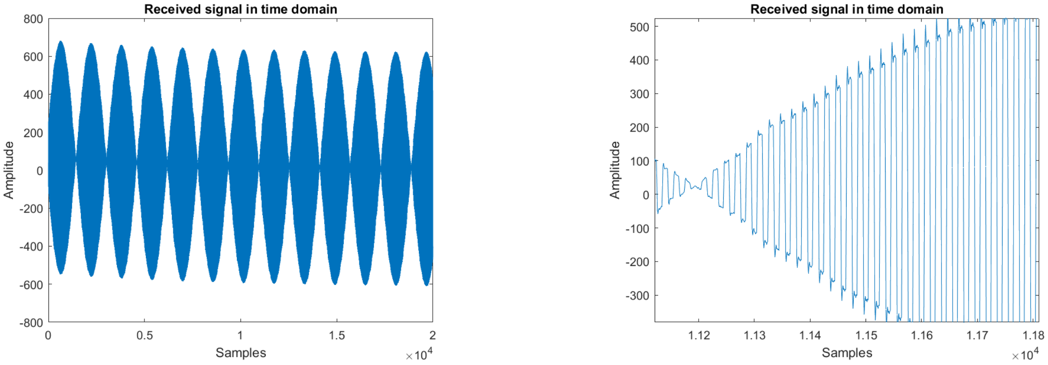


Figure 15: Square wave transmission within the same module

The square wave being transmitted and received between two different SDR modules is displayed in figure 16. Figure 16b is a close up of figure 16a to visualize that it is still a square wave being received.



(a) The received square wave

(b) The received square wave zoomed in

Figure 16: Square wave transmission between two SDR modules

5.3 Frequency Correction of Oscillators

Figure 17 is the received square wave between two SDR modules without regard for frequency correction. The marked points in the figure were used to find the wavelength of the underlying sine wave.

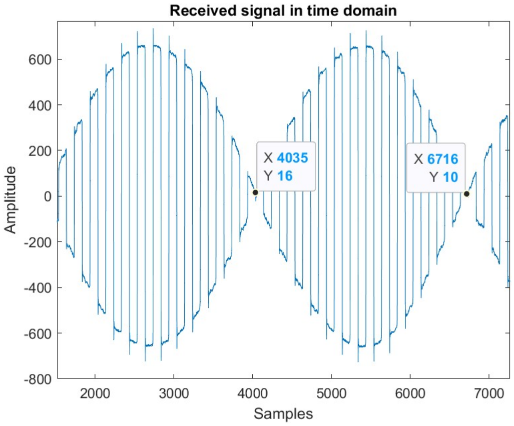
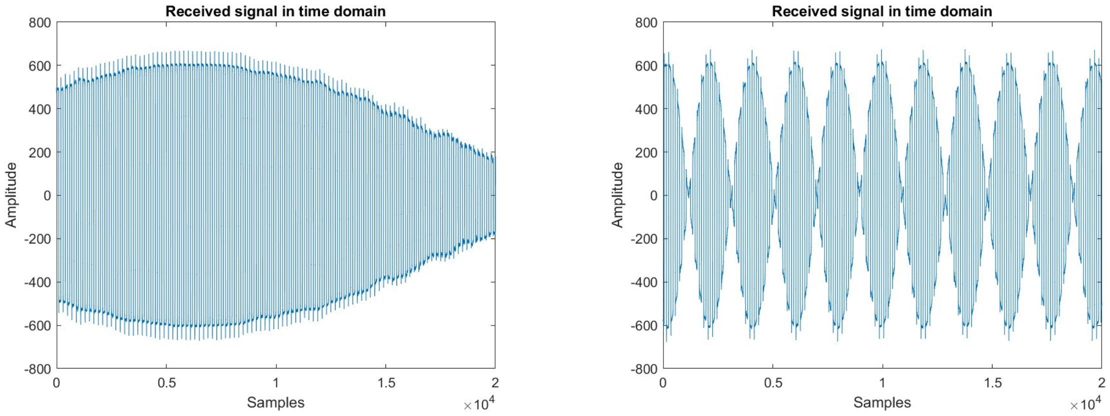


Figure 17: Received square wave

With equations 17 and 18 the frequency offset in parts per million, ppm, was calculated to 1.33. This value is used to synchronize the oscillators of the two SDR modules and the resulting received square wave is seen in figure 18. Figure 18a and 18b are two different recordings using the same frequency correction.



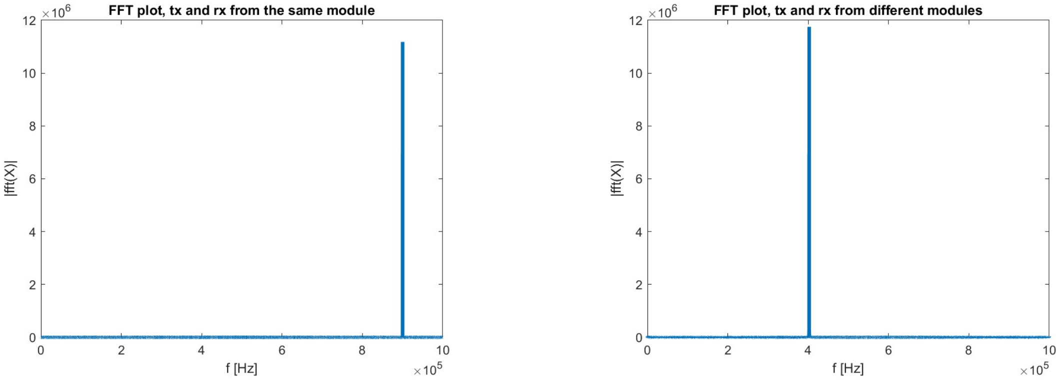
(a) The received square wave

(b) The received square wave zoomed in

Figure 18: Square wave transmission between two modules

5.4 Sine Wave Transmission

The transmission of a sine wave with a constant frequency was made according to section 4.5. Both tests FFT of the received signal are depicted in figure 19. Figure 19a is the received signal in the frequency domain of a single module while figure 19b is between two SDR modules.



(a) FFT of a sine wave within the same SDR module

(b) FFT of a sine wave between two SDR modules

Figure 19: Sine wave transmission with one and two SDR modules

5.4.1 Linearly Increasing Frequency

As described in section 4.5.1, the frequency had to be modulated to obtain FMCW RADAR technology. It was modulated to increase linearly and the result of the sine wave with a modulated frequency is visualized in figure 20. It is important to note that the chirp rate, μ, has been scaled down to be able to visualize the frequency increase. In practice, the rate of the frequency increase is magnitudes faster.

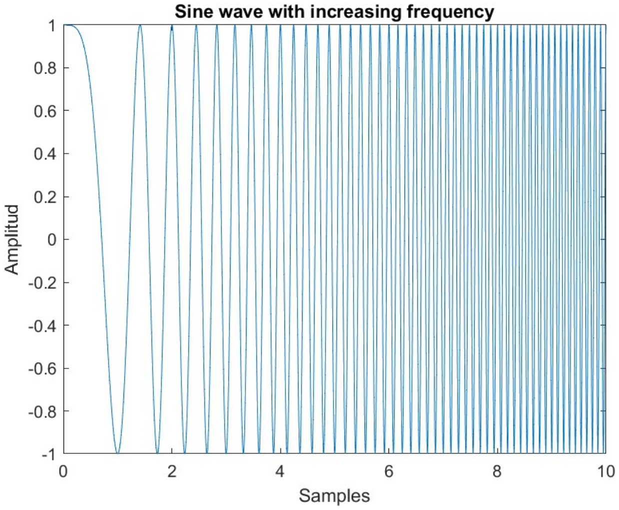


Figure 20: Sine wave with increasing frequency

5.5 Internal Leakage

Using a single SDR module with the Tx antenna encapsulated as described in section 4.6, the SDR module transmitted a sine wave with a constant frequency of 300 kHz. In figure 21 the received signals FFT is visualized. In theory, the received signal should be zero since the transmitting antenna is covered, however, a clear signal at 300 kHz is visible.

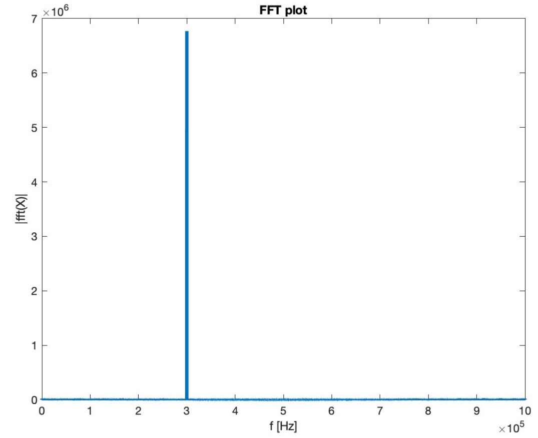


Figure 21: Internal leakage measuring with one SDR module

Using the same receiver setup as aforementioned, but adding another SDR module with its antennas fitted normally, the received signal is depicted in figure 22. The SDR with its antennas fitted normally is transmitting a sine wave with a frequency of 400 kHz. The encapsulated SDR module is still transmitting a sine wave of 300 kHz. It is apparent that the 300 kHz signal from the internal leakage is stronger than the 400 kHz signal from the second SDR module. It is worth noting that the 400 kHz signal is a direct signal from a SDR module in line of sight from the receiving SDR module.

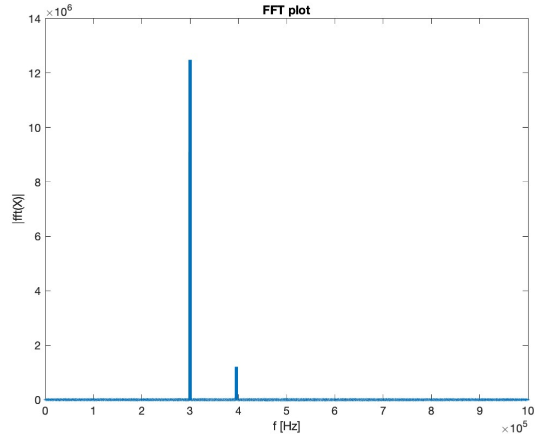


Figure 22: Internal leakage measuring with two SDR modules

5.6 Simulated Ranging

Throughout all the tests in the following section, parameters for calculation according to table 3 will be used.

Table 3: Parameters of the simulated FMCW transmission

Parameter	Value
f_c	4.2 [GHz]
f_s	30 [MHz]
B	14.8 [MHz]
T	8 [μs]
μ	1.85 [THz/s]

The FFT plot of the first test mentioned in section 4.7 is depicted in figure 23.

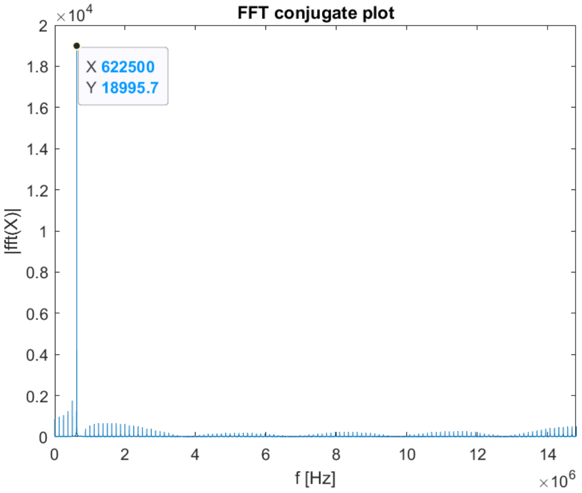


Figure 23: FFT for the first test

Using the frequency at the peak and equations 10 through 14 in section 2.5, the distance is calculated to 50.47 meters. In theory, the delay of 10 samples would be the equivalent of a distance of 50 meters to the target.

In the second test, a SWO of 300 samples was added and a frequency shift appeared as seen in figure 24. This shift causes a distortion when calculating the distance to the target. With the frequency from the peak, the range could be calculated to 959 meters while it should still be 50 meters.

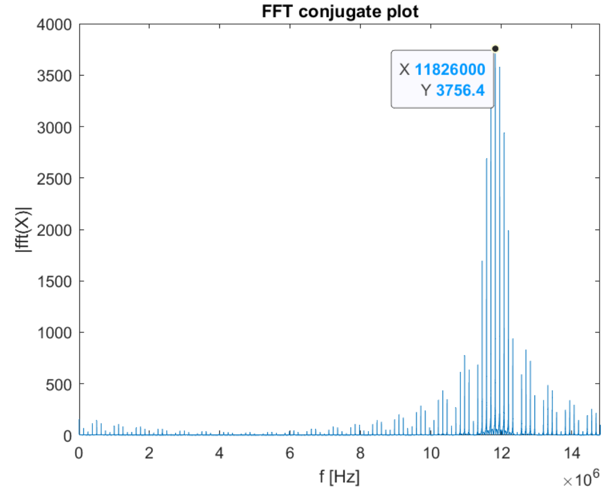


Figure 24: FFT for the second test

The FFT for the third test in section 2.5 resulted in figure 25. By using equation 14 with the value of x derived from the two sample delay, meaning 20 meters, the distance is 50.37 meters.

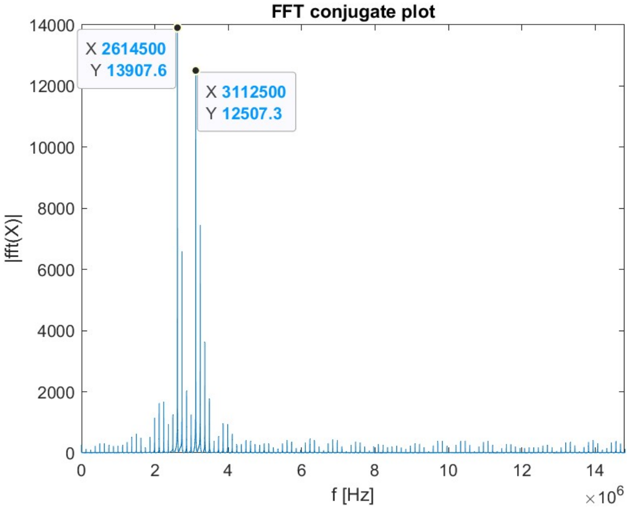


Figure 25: FFT for the third test

For the final test, the power of the echoed signal was scaled down to 1% of the direct signal which resulted in the FFT depicted in figure 26.

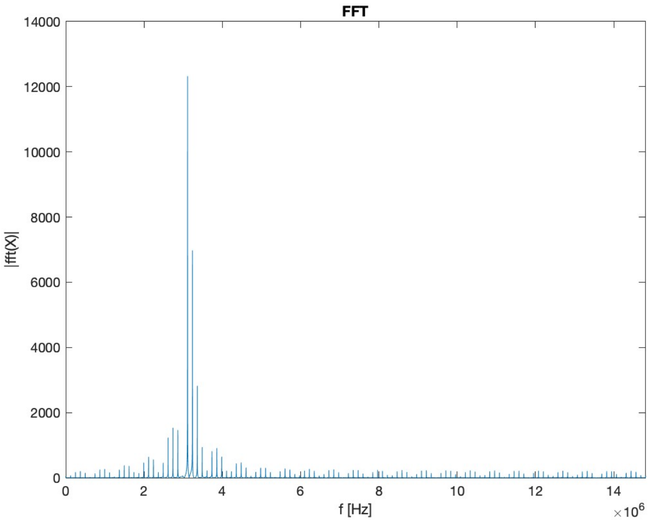
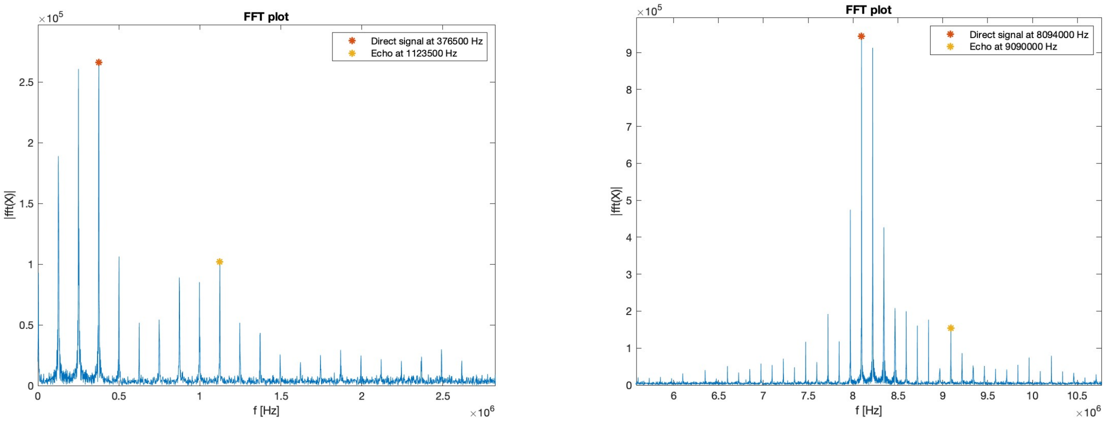


Figure 26: FFT for the final test

5.7 Multi-Module Reflection Measuring

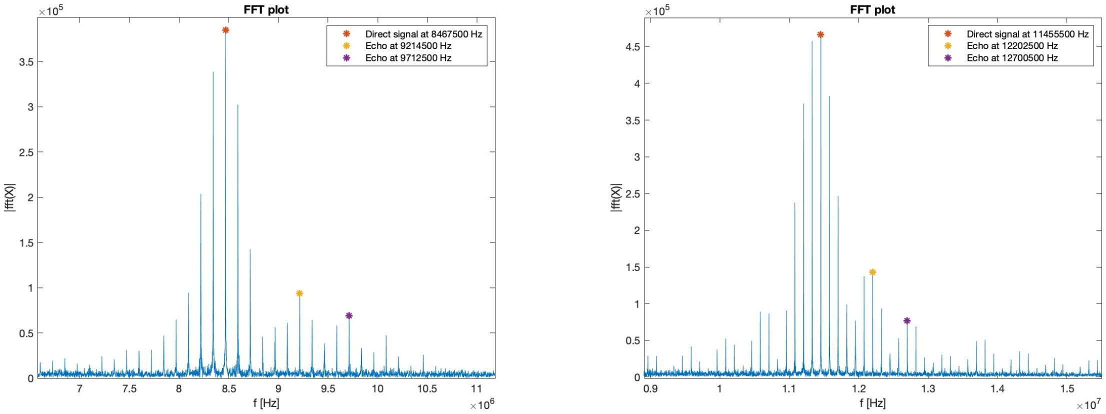
All measurements from tests one and two as described in section 4.8 resulted in the FFT plots as depicted in figures 29 to 39, in Appendix B. The figures displayed in this section are the ones that were deemed to have the most distinct peaks. The subsequent figures have enlarged areas of interest and marked peaks.

For all frequency peaks in the figures below equation 14 in section 2.5 was used to calculate the range R from figure 12. The range h was then calculated as described in section 4.8 and all values are compiled in table 5 and 4.



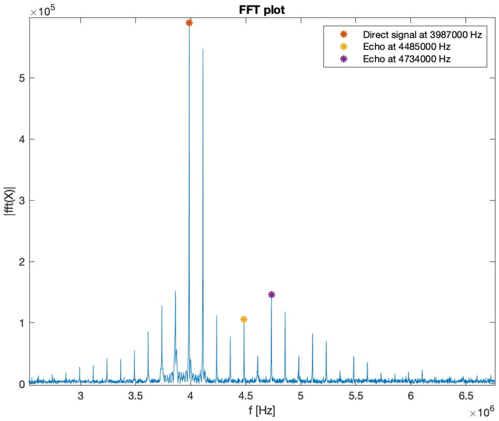
(a) Test 1, measurement 1

(b) Test 1, measurement 2



(c) Test 1, measurement 3

(d) Test 1, measurement 4

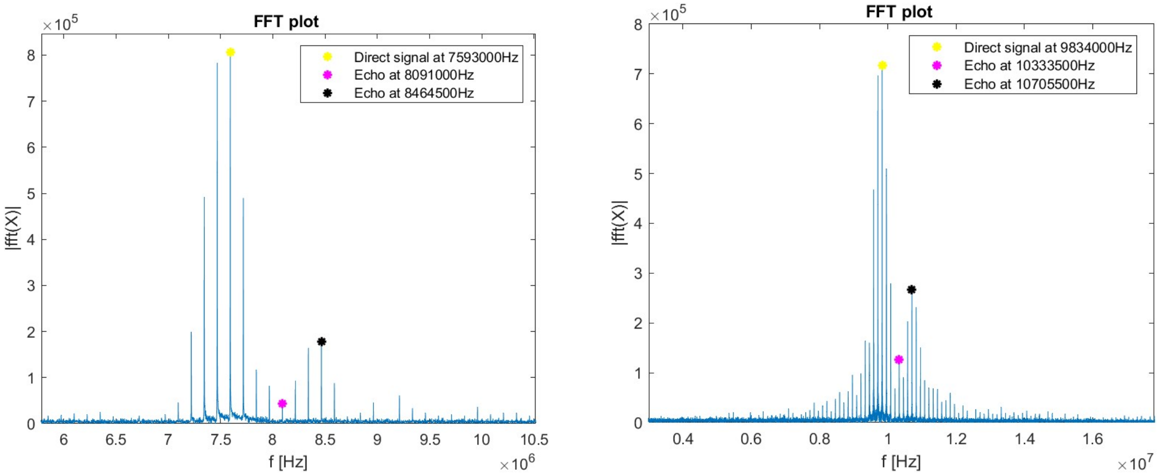


(e) Test 1, measurement 5

Figure 27: FFT plots from test 1 with marked frequency peaks

Table 4: Calculated R- and h-value of test 1

Measurement	R (First peak) [m]	h (First peak) [m]	R (Second peak) [m]	h (Second peak) [m]
1	71.57	70.72	-	-
2	91.76	91.1	-	-
3	71.57	70.72	111.95	111.4
4	71.57	70.72	111.95	111.4
5	51.38	50.2	71.57	70.72



(a) Test 2, measurement 2

(b) Test 2, measurement 6

Figure 28: FFT plots from test 2 with marked frequency peaks

Table 5: Calculated R- and h-value of test 2

Measurement	R (First peak) [m]	h (First peak) [m]	R (Second peak) [m]	h (Second peak) [m]
2	51.4	50.2	81.66	80.9
6	51.5	50.3	81.66	80.9

6 Discussion

A multitude of matters regarding the project will be discussed, concerning both decisions made and possible explanations for issues discovered during the project. Potential solutions for the problems will be presented and further development of the radar system will be discussed.

6.1 Simulink

As aforementioned, the project started with Simulink. At this phase, it was uncertain which software would be used to control the SDR module, meaning Python, MATLAB, and Simulink were tried. Simulink was first chosen as the block-based system is simple to begin with and there are plenty of documentation and example systems. After the mp3 file was successfully transmitted and received on two different SDR modules it was established that communication between SDR modules was possible and an attempt to send a sine wave was made. However, the block system also became a liability as it restricted functions to existing blocks. This in conjunction with the fact that Simulink requires vast amounts of computing power led to the decision to use MATLAB instead.

6.2 Square Wave Transmission

As seen in section 5.2 the square waves were similar to the transmitted vector when transmitting and receiving within the same SDR module. At the extreme points, where the transmission changes from negative to positive ones there is an overshoot. This overshoot comes from MATLAB trying to create a square wave from a sine wave. When transmitting and receiving between two SDR modules the shape of the signal is still a square, however, an underlying sine wave is visible. The reason for the sinusoidal shape of the received signal is due to frequency offsets between the respective oscillators in each SDR module, as described in section 2.3.3. The offset was then corrected as section 4.3 describes to get both SDR modules to operate at the same center frequency. However, as seen in figure 18 this did not work as intended since both figure 18a and 18b show varying degrees of frequency offset even when the same frequency correction is applied. The reason for this could be that the offset does not just depend on set value due to production differences but there is also a degree of time- and space-dependent offset. This could be caused by varying temperatures of the oscillator or it could simply be random. However, as the sampling frequency in the latter parts of this project is high, the offset was never recorded at a value high enough to impact the results of the FMCW RADAR system.

6.3 Sine Wave Transmission

As seen in section 5.4 the transmission of a sine wave with a frequency of 900 kHz by using one SDR module as transmitter and receiver worked well as the FFT in the receiving end visualized a peak at the transmitted frequency. The test where one SDR module acted as a transmitter and another as a receiver was also successful since the received FFT had a peak at 400 kHz which was the frequency that had been transmitted. For both cases, with the usage of one SDR module and two SDR modules, the frequency peak is the result of a sum of the echo, direct, and leakage signals.

6.4 Internal Leakage

As seen in figure 21 in section 5.5, the SDR module receives signals even when its transmitting antenna is encapsulated. This indicates that the SDR module has internal leakage, meaning the electromagnetic waves are propagating out of the Tx circuit and being received without the use of a Tx antenna. Since the Tx antenna was covered, this could have happened in several ways. Firstly the waves could be propagating into the Rx circuit directly from the Tx circuit or the waves could be penetrating the hull of the SDR module and travelling from the Tx circuit into the Rx antenna. A third option could be that the setup with the cans did not completely block the waves.

The test depicted in figure 22, section 5.5, shows how the measured leakage is greater in amplitude than a direct signal from another module. This could cause problems since the power of the echo signal will be much lower than the direct signal and therefore even lower than the internal leakage, thus making it hard or impossible to detect. The power of the leakage could drown out the reflected signal which is a problem since the reflected signal is the signal of interest in this project. Therefore a decision was made to continue the project using multiple SDR modules instead of a single one. Using two separate SDR modules to act as Rx- and Tx antenna respectively with an adequate distance between them. The problem with the internal leakage is then diminished since the power of the leakage signal is lower as it has to propagate through free space. Although the leakage problem is solved by adding another module, new problems arise such as syncing them which will be further discussed in section 6.5.

6.5 Object Detection Using FMCW RADAR

Due to the leakage discussed in section 6.4 it was concluded that using a single module as Tx and Rx would not work. Instead, a setup of multi-module reflection measuring was used. This setup requires two SDR modules, one transmitting in an unrestricted lobe and a receiver having an antenna with an aimed lobe in the direction of the target of interest. This makes use of the leakage from the transmitting SDR module to the receiving SDR module as the direct signal penetrates the hull of the SDR module while the echo is received with the aimed antenna. These two frequencies can then be used to obtain the beat frequency as equation 12 in section 2.5.

This concept was proven in section 5.6 where the distance to the target could be calculated with the help of the frequency difference between the echo and the direct signal. However, only in a controlled environment, meaning a large open area with minimal reflective objects to cause noise and clutter. It was also discovered that as long as the amplitude of either signal was no less in amplitude than 1% of the other it was detectable as depicted in figure 26. However, in practice, the signal probably needs to be closer in amplitude than 1% because of noise which was not simulated.

Observing the figures in section 5.7, it is apparent that there are periodic peaks through the sample window. The frequency difference between each peak is equal to 124500 Hz. This frequency difference, calculated in terms of how far the wave has traveled, according to equation 11 equals 10.1 meters which is equal to how far the wave travels during one sample. This confirms our calculation for the resolution due to the inability to get a more accurate approximation of distance to a target than 10.1 meter intervals. It can also be calculated that our maximum range in practice is in fact equal to our theoretically calculated maximal range if the maximal frequency difference is used with equation 14.

In test one, the distance to the building that was attempted to measure was approximately 82 meters away. In four of the five measurements, a distance of 70.72 meters was calculated which could represent the building. Due to the resolution of 10.1 meters calculated in section 2.6 the deviation from the calculated distance to the measured distance could be explained. The measured distance of exactly 70.72 meters was obtained in four of five measurements which strengthens speculation that these results strongly depend on the resolution. Because the RADAR was not fully stationary but held by hand during the tests it should deviate the results by a couple of centimeters from measurement to measurement if the results were accurate. There were also multiple other buildings farther away that explain the distances of 91.1 and 111.95 meters that were calculated. In measurement five there was a tractor that drove in front of the building that the radar was aimed at. This explains the distance of 50.2 meters that was calculated. It is also noteworthy that three of the five measurements from test one managed to pick up two large echo peaks in the same measurements which means multiple targets are detectable at the same time. Multiple smaller peaks were also visible in many of the measurements but these were deemed as not noteworthy enough due to their low amplitude.

In test two, the most significant frequency peak came from the building located 84 meters from the modules as seen in figure 27. That frequency peak gave a distance of 80.9 meters according to table 5 in section 5.7. This value is an accurate result since it deviates just 3.1 meters from the measured distance. However, the deviation is lower than the previously mentioned resolution which means that the system should not be able to measure this accurately. Meaning the low deviation is probably due to pure randomness and luck. In this test, there were multiple targets, for example, the previously mentioned building and a car. The car was located approximately 50 meters from the modules and the frequency peak that came from the car was low in amplitude but still distinguishable. The peak is marked in figure 28b and is clearly visible, however, in figure 28a the amplitude is lower but can be found if searched for nonetheless. From these peaks, a distance of 50.2 and 50.3 meters was calculated. This result has a deviation of just 0.2 and 0.3 meters respectively. The reason that the frequency peak that came from the reflection on the car was hard to distinguish is because of the resolution. Around the car, there were multiple different smaller targets within the resolution that the signal could also reflect at which made these reflections hard to distinguish in the resulting FFT. The amplitude of the peaks from the car was lower than the amplitude of the peaks from the building. This is due to that the surface of the building had a larger area than the car. It must also be noted that all the previously mentioned deviations from the calculated distances to the measured also depend on how accurate the physical measured distances were.

In section 5.7 it can be observed how the SWO works in practice even though the R-value in both tables were relatively close to the correct range for each measurement the peaks ranged the whole bandwidth over the different measurements.

6.6 Simulated Minimum Distance

While simulating different distances to the target it was discovered that if the target was closer than 50 meters to the RADAR system, the frequency peaks from the direct signal and the reflected signal merged into one peak. This makes it impossible to calculate the distance to the target by using equation 14 since it requires two frequencies. This phenomenon happens because each signal consists of multiple frequency peaks, and when the signal gets stronger it also gets wider. The resolution only describes how far apart each peak can be, but if each signal consists of multiple peaks, the real minimum distance to objects will be decided by how close the two signals can be to each other without any overlap in the frequency domain.

Therefore, the system has a resolution and a minimum range which are different. The resolution is the distance at which two targets can be apart without being detected as one. The minimum distance is usually the same as the resolution, however in this project due to the width of the frequency peaks, it is different. During testing, it appears that 50 meters is the minimum distance to a target where it is still possible to distinguish two peaks. If the distance was shorter than 50 meters the two signals start to overlap resulting in one larger and broader peak. This means the 50 meter minimum range only applies to the first target, due to the strength of the direct signal, thereafter the resolution decides how far apart objects can be distinguished.

This problem could most likely be solved by using additional signal processing to further differentiate the signal peaks. This would make them narrower, resulting in the peaks being able to be closer to each other without any overlap. This could shorten or all together remove the difference between the resolution and minimum range.

6.7 Hardware Limitations

As aforementioned in section 2.6 the Pluto SDR module has hardware limitations of certain parameters, the parameter that impacts this project the most is the sampling frequency. The maximum sampling frequency should, on paper, be 61.44 MHz but despite that when tested, the system was affected by aliasing. The highest sampling frequency that could be used without being affected by aliasing was 30 MHz and to fulfill the Nyquist theorem the bandwidth was set to 14.8 MHz. Why the sampling frequency specified by Analog Devices causes aliasing could be because of documentation differences between their MATLAB toolbox and their physical products.

Another limitation caused by the sampling frequency is the chirp rate, μ, which depends on the bandwidth and the chirp time. As aforementioned the bandwidth was set to 14.8 MHz during the project. Typically the chirp time, T, is set to a short period in FMCW RADAR situations to make the chirp rate as large as possible. This will increase the time difference between the direct and echoed signals. However, the wave vector in the transmitted signal depends on the chirp rate. When simulating the ranging a smaller chirp time causes the transmitted vector to be shorter due to the bandwidth being a set value. When a too small chirp time was used the vector could no longer represent this due to having too few elements. For the transmitted wave to be able to represent a linear frequency increase in a sine wave, the chirp time needs to be at least 8 μs for the increase to be visible as figure 20.

6.8 Continuation of the Project

Although this project achieved subgoals 1 through 6 as seen in section 3.1, there is still room for improvement and continuation to complete subgoals 7 and 8.

The first category that could be improved is the hardware. If it instead of the ADALM Pluto SDR had been another module with the capacity for a higher sample frequency alongside a larger bandwidth, the range and resolution could have been greatly improved. This could lead to better estimates of the measured distances and smaller targets could be detected.

The second category of improvement is signal processing, both in simulation and testing. In simulation, the offset for each signal is currently an integer for more accurate simulations this could be changed to float values. Noise could also be added to better simulate real-world phenomenons. Currently, the only signal processing conducted is the usage of FFT. Future improvements that could be made are noise reduction for testing and subsample analysis. Noise reduction would make weak signals easier to differentiate from background noise, meaning further distances and smaller targets could be detected. Subsample analysis, using regression, would make it possible to get a better range approximation than the current resolution allows.

For completion of subgoal 7, the triangulation could be simulated and tested using 4 modules, 2 receivers, and 2 transmitters, by using the same principles as done earlier with two modules but two different center frequencies for each pair of receivers and transmitters. However, with the current sample rate, this setup's x-value according to figure 12 would need to be at least 40 meters wide. It would require 20 meters between the first Tx and Rx pair, and another 20 meters for the second pair. This makes it impractical for short-range object detection due to the nature of these systems often being placed on vehicles.

Lastly, there was not enough time to complete subgoal 8, create a UI. This could automatically calculate the range to the targets instead of making it manually which would save time. If subgoal 7 is completed the UI could also automatically, by using triangulation, visualize the object not only with range but also angle in relation to the RADAR system.

7 Conclusion

The possibility of measuring direct and echoed signals using FMCW RADAR with SDR modules has been proven successful in a controlled environment. However, the minimum range of the RADAR system was measured to 50 meters while the resolution was 10.1 meters. This originates from the relatively low sample rate at 30 MHz that the hardware operated at.

Another limiting factor was the leakage inside the SDR module which made it impossible for the receiver and transmitter antennas to be on the same module. This leads to a system where the receiver and transmitter are placed on different modules fitted with tin-cans over their respective antenna. This makes the system sub-optimal for short-range object detection due to the need for a large distance between the receiver and transmitter antenna.

In summary, target detection and ranging are possible by using FMCW RADAR technology with ADALM PLUTO SDRs that are controlled by using MATLAB, although not at a short range.

A Appendix

```
clc, clear, beep off
%tx part

%creating a vector with 100 ones and 100 negative ones
tx_waveform = [ones(1,100)*-1,ones(1,100)+eps*1i ];

%define a radio object to transmit from
radio = sdrtx('Pluto');
radio.CenterFrequency = 2.415e9;
radio.BasebandSampleRate = 1e7;
radio.Gain = 0;

%tx repededly to create a continous square wave
transmitRepeat(radio,tx_waveform');
%visualizing the sent wave
plot(tx_waveform)

%findPlutoRadio
%release(radio)

%%
%rx part

fs=1e6;

%creating our receiving radio object and defining some parameters
rxPluto = sdrrx('Pluto',...
           'RadioID','usb:0',...
           'CenterFrequency',2.415e9,...
           'BasebandSampleRate',fs);
rxLogNoOverflow = dsp.SignalSink;
rxLogDataValid = dsp.SignalSink;

%creating an infinite loop meaning the receiver is constantly listening
%we have commented out when it wants to display if samples are droped in
%order to have a faster code
for counter = 1:inf
    [data,datavalid,overflow] = rxPluto();      %store received signal

% Check for overflow of received samples.
    if (overflow) % dropped samples
        %disp('samples dropped');
    else
        rxLogNoOverflow(data);
    end
% Check for overflow and validity of received data.
    if ~(overflow) % no dropped samples
        if ~(datavalid) % received desired data
             rxLogDataValid(data);
        end
    else

        %disp('no valid data received');
    end
    %plot the received data in time domain
    if mod(counter,1)==0
        plot(real(data))
        title('Received signal in time domain')
        ylabel('Amplitude')
```

```
        xlabel('Time')
        drawnow         %continuously draws the plot meaning we have real time updates
    end

end


%findPlutoRadio
%release(radio)
```

Listing 3: MATLAB script to transmit and receive square wave

```
%% Configuration
%cleans up workspace and command window,
clear, clc, beep off
findPlutoRadio %ConnectPluto

Fc=2.415e9; %Center frequency
Fs=1e6; %SampleRate

%%Choices 0 for off 1 for on
PlotWave=0;
LiveFFT=1;

%creating the radio object and defining some parameters for it
radio = sdrtx('Pluto');
radio.CenterFrequency = Fc;
radio.BasebandSampleRate = Fs;
radio.Gain = 0;

rxPluto = sdrrx('Pluto',... %Defining RX
           'RadioID','usb:0',...
           'CenterFrequency',Fc,...
           'BasebandSampleRate',Fs);
rxLogNoOverflow = dsp.SignalSink;
rxLogDataValid = dsp.SignalSink;

%% Transmitter



%Creating sinuswave
sw = dsp.SineWave;
sw.Amplitude = 0.5;
sw.Frequency = 200e3;
sw.ComplexOutput = true;
sw.SampleRate = Fs;
sw.SamplesPerFrame = 5000; % to meet waveform size requirements
wave = sw();

%Plotting wave
if PlotWave==1
    plot(wave);
    xlabel('Tid (s)');
    ylabel('Amplitud');
title('Sine Wave');
end

%Continously transmitting the wave vector
transmitRepeat(radio, wave)

```

```
%% Receiver

%How long Pluto should receive.
for counter = 1:5000
    [data,datavalid,overflow] = rxPluto();

% Check for overflow of received samples.
    if (overflow) % dropped samples
        %disp('samples dropped');
    else
        rxLogNoOverflow(data);
    end
% Check for overflow and validity of received data.
    if ~(overflow) % no dropped samples
        if ~(datavalid) % received desired data
             rxLogDataValid(data);
        end
    else
        %disp('no valid data received');
    end
    if LiveFFT ==1
        %Draw FFT
        L=length(data);
        Y = fft(data);
        plot(Fs/L*(0:L-1),abs(Y),"LineWidth",3)
        title("Received signal in time domain")
        title("FFT plot")
        xlabel("f [Hz]")
        ylabel("|fft(X)|")
        drawnow
    end
end
release(radio)
```

Listing 4: MATLAB script to transmit sine wave and visualize received signals FFT

```
%% Configuration
%cleans up workspace and command window,
clear, clc, beep off
findPlutoRadio %ConnectPluto

Fc=2.415e9; %Center frequency
Fs=1e6; %SampleRate

%%Choices 0 for off 1 for on
PlotWave=0;
LiveFFT=1;

%creating the radio object and defining some parameters for it
radio = sdrtx('Pluto');
radio.CenterFrequency = Fc;
radio.BasebandSampleRate = Fs;
radio.Gain = 0;

rxPluto = sdrrx('Pluto',... %Defining RX
           'RadioID','usb:0',...
           'CenterFrequency',Fc,...
           'BasebandSampleRate',Fs);
rxLogNoOverflow = dsp.SignalSink;
rxLogDataValid = dsp.SignalSink;

%% Transmitter
```

```



%Creating the frequency increase
T = 2; %Time for one chirp
f_max = 8e5;
f_min = 0;
B = f_max - f_min;      %deciding the bandwidth
t = (0:1/Fs:T)';        %this is the timevector
mu = B/T;               %definiing the chirp rate
f = Fc.*t + 0.5*mu.*t.^2;
A = 1;

%creating the wave function
wave = A*exp(1i*2*pi*f);

%Plotting wave
if PlotWave==1
    plot(t, wave);
    xlabel('Time (s)');
    ylabel('Amplitude');
    title('Sine wave with increased frequency');
end

%Continously transmitting the wave vector
transmitRepeat(radio, wave)

%% Receiver

%How long Pluto should receive.
for counter = 1:5000
    [data,datavalid,overflow] = rxPluto();

% Check for overflow of received samples.
    if (overflow) % dropped samples
        %disp('samples dropped');
    else
        rxLogNoOverflow(data);
    end
% Check for overflow and validity of received data.
    if ~(overflow) % no dropped samples
        if ~(datavalid) % received desired data
             rxLogDataValid(data);
        end
    else
        %disp('no valid data received');
    end
    if LiveFFT ==1
        %Draw FFT
        L=length(data);
        Y = fft(data);
        plot(Fs/L*(0:L-1),abs(Y),"LineWidth",3)
        title("Received signal in time domain")
        title("FFT plot")
        xlabel("f [Hz]")
        ylabel("|fft(X)|")
        drawnow
    end
end
release(radio)
```

Listing 5: MATLAB script to transmit FMCW and visualize received signals FFT

```matlab
clear, clc, beep off

Fc=4.2e9;       %Center frequency
Fs=3e7;         %SampleRate



%Creating the frequency increase

T = 8e-6;                       %Time for one chirp
f_max = 14.8e6;
f_min = 0;
B = f_max - f_min;              %deciding the bandwidth
t = (0:1/Fs:T)';                %this is the timevector
mu = B/T;                       %definiing the chirp rate
f = Fc.*t + 0.5*mu.*t.^2;
A = 1;                          %amplitude

%creating the wave function
wave = A*exp(1i*2*pi*f);


% Replicating wave
n = 20000;                  %decidiing the length based
repetitions = ceil(n / numel(wave));        %deciding repetitions
new_wave = repmat(wave, 1, repetitions);    %repeating the wave
new_wave = new_wave(1:n);                   %making the wave n units long

%% Echowave

shiftecho=zeros(1,10);          %deciding delay in echoed signal
shiftdirect=zeros(1,2);         %deciding delay in the direct signal
echowave=[shiftecho new_wave];          %adding the padding to the waves
directwave=[shiftdirect new_wave];

conj_echo=conj(echowave(1:n)');         %taking the conjugate of the waves
conj_direct=conj(directwave(1:n)');

rxwave=conj_echo + conj_direct;           %creating the received signal

window_offset=zeros(1,200);              %adding theoretical offset to sample window
txwave = [window_offset new_wave];
txwave = txwave(1:n)';

%% FFT

matrix_data = rxwave.*txwave;  %multiplying conjugate of received signal with the transmitted wave

le = length(matrix_data);       %creating length units for the fft

figure(1)
Y = fft(matrix_data');          %creating the fft and plotting it
plot(Fs/le*(0:le-1), abs(Y))
title("FFT conjugate plot")
xlim([0 f_max])
xlabel("f [Hz]")
ylabel("|fft(X)|")
```

Listing 6: MATLAB script of simulation

```
%% Configuration
%cleans up workspace and command window,
clear, clc, beep off
findPlutoRadio %ConnectPluto
%%

Fc=4.2e9; %Center frequency
Fs=3e7; %SampleRate


%creating the radio object and defining some parameters for it
radio = sdrtx('Pluto');
radio.CenterFrequency = Fc;
radio.BasebandSampleRate = Fs;
radio.Gain = 0;

rxPluto = sdrrx('Pluto',... %Defining RX
           'RadioID','usb:0',...
           'CenterFrequency',Fc,...
           'BasebandSampleRate',Fs,...
           'GainSource','Manual',...
           'ShowAdvancedProperties', 1);
rxLogNoOverflow = dsp.SignalSink;
rxLogDataValid = dsp.SignalSink;

%% Transmitter

%Creating the frequency increase
T = 8e-6; %Time for one chirp
f_max = 14.8e6;
f_min = 0;
B = f_max - f_min;      %deciding the bandwidth
t = (0:1/Fs:T)';        %this is the timevector
mu = B/T;               %definiing the chirp rate
f = Fc.*t + 0.5*mu.*t.^2;
A = 1;

%creating the wave function
wave = A*exp(1i*2*pi*f);

%repeating the wave so that it is the same length as the received signals vector
n = 20000;
repetitions = ceil(n / numel(wave));
new_wave = repmat(wave, 1, repetitions);
new_wave = new_wave(1:n)';

%Continously transmitting the wave vector
transmitRepeat(radio, new_wave)


%% Receiver
%storing the received data in a vector
[data] = rxPluto();

%taking the conjugate and multiplying received and transmitted data
data = double(data);
con_data = conj(data);
matrix_data = con_data.*new_wave;

%visualizing in an FFT
le = length(matrix_data);
ti = (0:le-1)* 1/Fs;
```

```
Y = fft(matrix_data');
figure(1)
plot(Fs/le*(0:le-1), abs(Y))
title("FFT Plot")
xlim([0 f_max])
xlabel("f [Hz]")
ylabel("|fft(X)|")
```

Listing 7: MATLAB script of the FMCW RADAR system

B Appendix

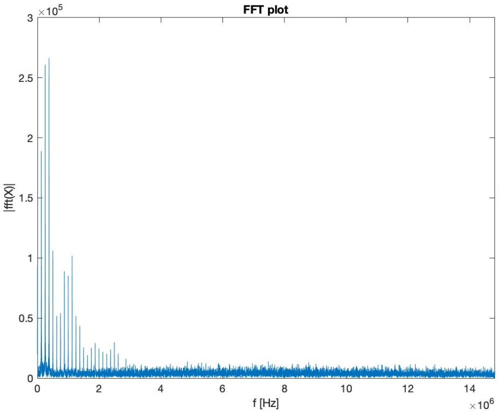


Figure 29: Test 1 measurement 1

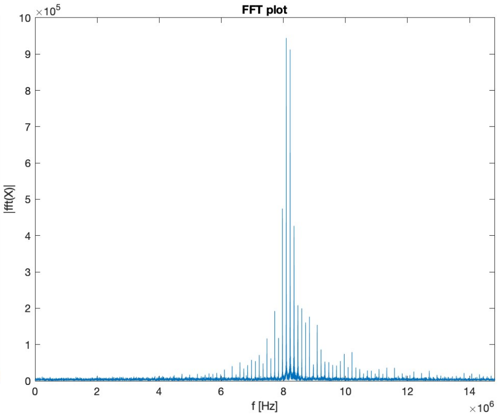


Figure 30: Test 1 measurement 2

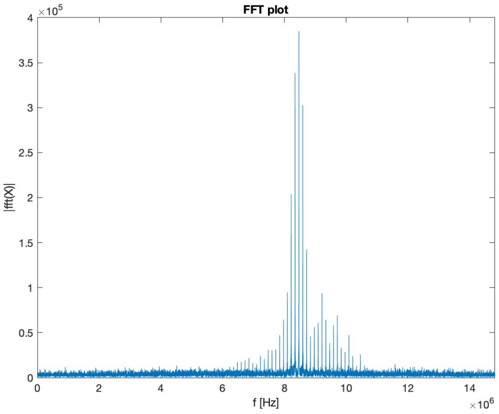


Figure 31: Test 1 measurement 3

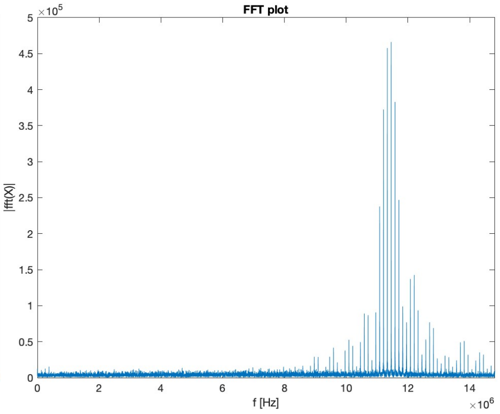


Figure 32: Test 1 measurement 4

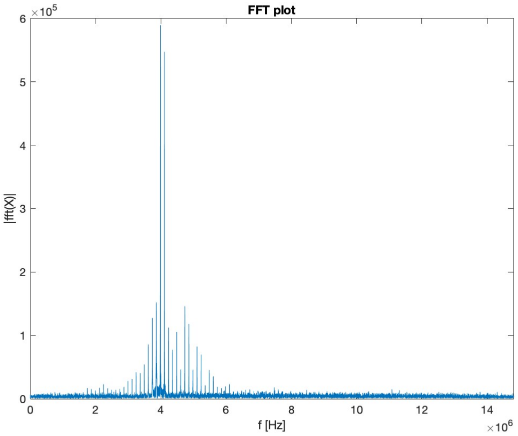


Figure 33: Test 1 measurement 5

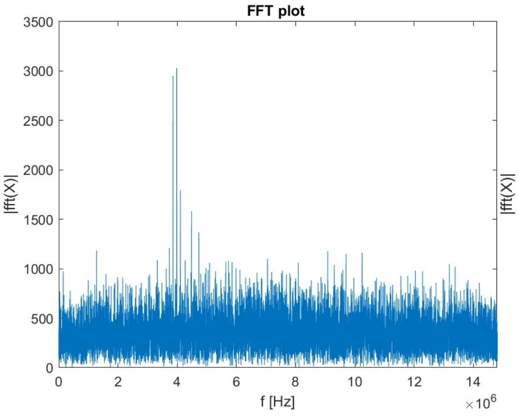


Figure 34: Test 2 measurement 1

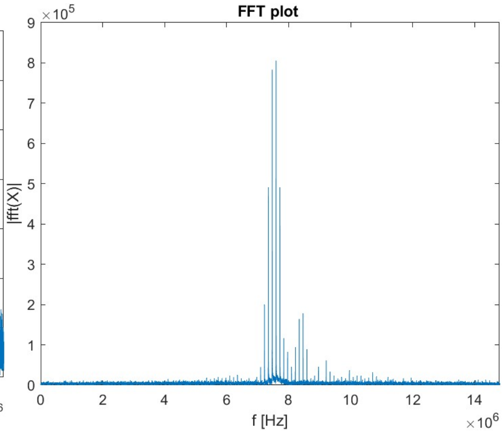


Figure 35: Test 2 measurement 2

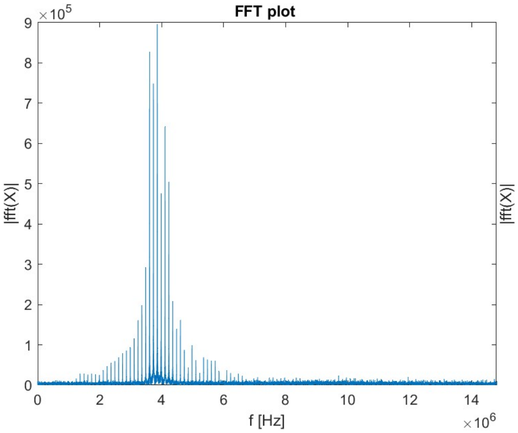


Figure 36: Test 2 measurement 3

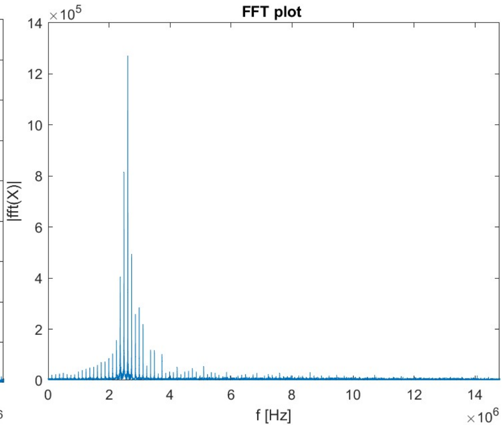


Figure 37: Test 2 measurement 4

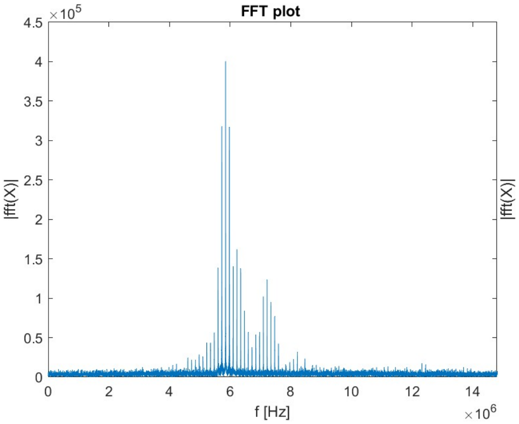


Figure 38: Test 2 measurement 5

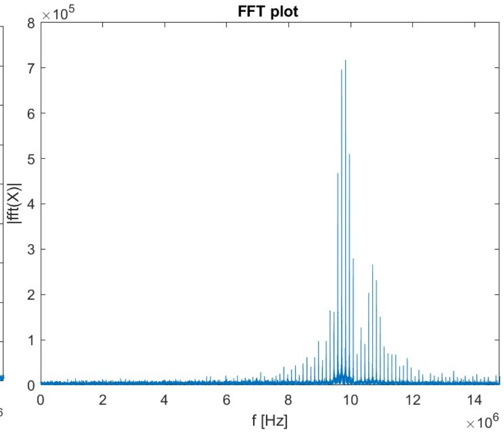


Figure 39: Test 2 measurement 6

Figure 40: Test 2, view of targets

References

[1] S. Campbell, N. O'Mahony, L. Krpalcova, *et al.*, "Sensor Technology in Autonomous Vehicles : A review," in *29th Irish Signals and Systems Conference, ISSC 2018*, 2018, ISBN: 9781538660461. DOI: `10.1109/ISSC.2018.8585340`.

[2] L. Bossi, P. Falorni, and L. Capineri, "Versatile Electronics for Microwave Holographic RADAR Based on Software Defined Radio Technology," *Electronics (Switzerland)*, vol. 11, no. 18, 2022. DOI: `10.3390/electronics11182883`.

[3] J. A. Scheer and W. A. Holm, "Introduction and radar overview," in *Principles of Modern Radar: Basic Principles*, Atlanta, GA, United States: Institution of Engineering and Technology, Nov. 2010, pp. 3–58.

[4] R. J. James, "A history of radar," *IEE Review*, vol. 35, no. 9, pp. 343–349, 1989. DOI: `10.1049/ir:19890152`.

[5] M. I. Skolnik, "Introduction to radar," *Radar handbook*, vol. 2, 1962.

[6] C. Wolff, *Continuous Wave Radar*. [Online]. Available: `https://www.radartutorial.eu/02.basics/Continuous%20Wave%20Radar.en.html`.

[7] N. Bhatta and P. M Gheeta, "RADAR and its applications," vol. 10, pp. 1–9, Jan. 2017.

[8] A. Stove, "Linear FMCW radar techniques," *IEE Proceedings, Part F: Radar and Signal Processing*, vol. 139, no. 5, pp. 343–350, 1992. DOI: `10.1049/ip-f-2.1992.0048`.

[9] J. Banar and T. Eriksson, "Estimation of relative timing drift in coherent distributed radars," in *European Radar Conference (EuRAD)*, Paris, France, 2024.

[10] R. Getz, *ADALM Pluto Overview*, Oct. 2021. [Online]. Available: `https://wiki.analog.com/university/tools/pluto`.

[11] MathWorks, *configurePlutoRadio*. [Online]. Available: `https://se.mathworks.com/help/comm/plutoradio/ref/configureplutoradio.html`.

[12] Y. Kou and Y. Morton, "Oscillator frequency offset impact on software GPS receivers and correction algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 4, pp. 2158–2178, 2013. DOI: `10.1109/TAES.2013.6621808`.

[13] T. Eriksson, private communication, Feb. 2024.

[14] D. Cheng, "Antennas And Radiating Systems," in *Field and Wave Electromagnetics*, Pearson Education, 2014, ch. 11.

[15] R. Getz, *Self Reception on the ADALM-PLUTO*, Jul. 2019. [Online]. Available: `https://wiki.analog.com/university/tools/pluto/hacking/listening_to_yourself?fbclid=IwAR1H-RA6NOIQTAIJk3YabLfKFgmyIkJjZNB_UeNbrEljK9eYlUJ8Fx1oZSg`.