Design and Implementation of a Narrowband 1GHz Phased Array & Adaptive Digital Beamforming System

ELEN90081/ELEN90082 Electrical Engineering Capstone Project
Final Report

Group CSP6:
Benjamin Ding
Wilhelm Schuetz
Erfan Shooraj

Department of Electrical and Electronic Engineering
The University of Melbourne

Abstract

This report details the theoretical underpinnings and practical implementation of two types of beamforming technologies - phase-shift and LMS digital beamforming. First, the mathematical foundations of beamforming are examined and then demonstrated through simulation. Using this theoretical foundation, the report evaluates the design, implementation, and results obtained from two custom beamformer implementations for analog transmit phase-shift beamforming, and receive digital beamforming.

The phase-shift beamforming system is shown to be able to steer a beam to both 0° and 20° through the use of analog phase-shift operations performed on the transmitted RF signal, closely matching computer simulated radiation patterns.

Furthermore, the Least-Mean-Squares algorithm is used in conjunction with the constructed receiver array to form a complete digital receive beamforming system, which is demonstrated to increase signal-to-noise ratio of evaluation signals up to 10dB after using trained array weights. The digital receive beamformer is able to increase signal-to-noise ratio by up to 5dB even after repositioning of the transmitter from 0° to -30°.

Through this, a holistic view is given of both conventional (phase-shift) and digital (LMS) beamforming. System performance is verified to match simulations to a reasonable extent and sources of deviation are examined and discussed.

The detailed design considerations and development of such a beamforming system are also covered, including all aspects such as high speed/RF PCB design, achieving a phase shift/gain adjustment with low-cost electronic components, practical concerns in receiving/sampling, and single-element antenna/multi-element array simulation & design for an appropriate (approximately 1GHz) frequency.

Contents

1 Introduction

In this report the design, implementation and evaluation of a transmit and a receive beamforming system is discussed. The design goal of the project is to construct these two systems such that the radiation pattern of the transmitter and spatial sensitivity pattern of the receiver are able to be manipulated as desired. These systems are also required to be operational within the Industrial Scientific and Medical (ISM) band with a narrowband message.

It is demonstrated that these two different types of beamformers, a conventional transmit (phase-shift) and digital receive beamformer (using the Least Mean Squares algorithm) can be implemented using typical off-the-shelf electronics components in a custom hardware and software design.

Achieved performance is tested experimentally for these two systems in which the radiation or spatial sensitivity pattern of the system is electronically steered using a controller. These results are then compared with theoretical simulations and validated to match the expected outcome to a good extent.

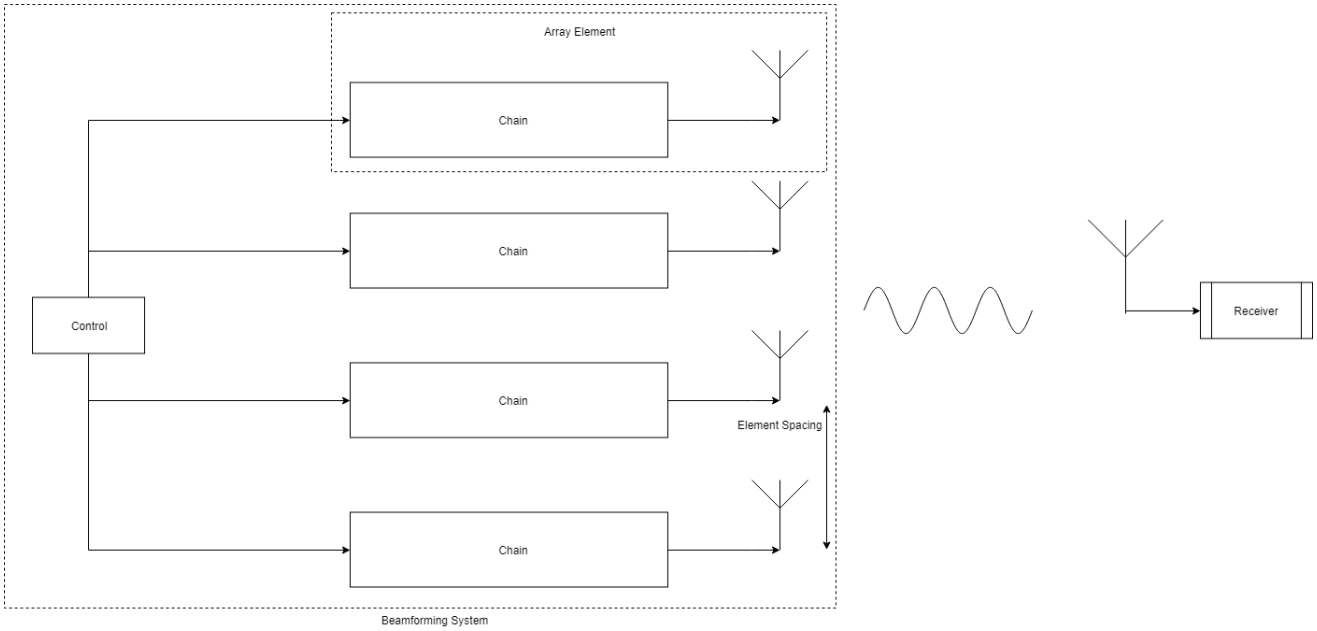


Figure 1: High level diagram of the transmit beamforming system, demonstrating four RF chains along with their respective antenna, each comprising an element

A high-level diagram of the transmit beamforming system can be seen in Figure 1. As is shown in the diagram, the transmitter consists of an array of multiple elements[1]. Each element is physically identical, and comprises of an RF chain which performs all necessary transforms on a signal prior to transmission at the antenna. The system allows for different configuration of individual elements in the chain via the control module in order to manipulate the signals such that a desired pattern of constructive and destructive interference is achieved.

[1]In the specific case of this project, four elements are used - more elements yield better performance but higher cost and development time. Four elements was determined to be the minimum amount of elements to achieve a reasonable main lobe width. More details can be found in Section 4.

Figure 2: A depiction of chain internals for analog phase-shift beamforming

Figure 2 further details how the transmitted signal is transformed for each individual element. The same baseband signal is sent to each RF chain in the antenna array, which modulates the signal onto a carrier before performing a phase shift and amplification on the signal; the phase shifts and gains used by each element in conjunction with the physical spacing of the elements is what ultimately causes the resultant signals from each antenna to constructively and destructively interfere in a way that produces a desired radiation pattern as a whole. This allows for spatial steering of transmitted power in a desired direction.

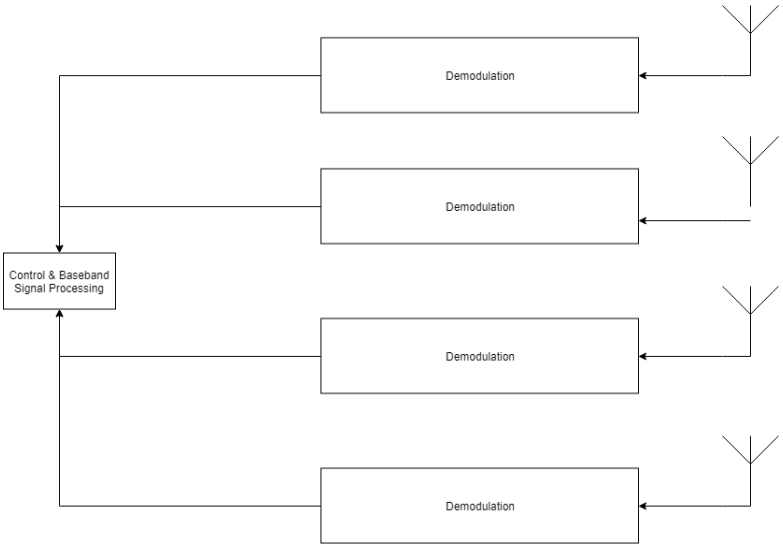


Figure 3: A high level depiction of the array elements for digital beamforming, note the array is receiving

The high level architecture of the receive beamforming system is essentially the same as that which is shown in Figure 1 but with a reversal of direction: the previously depicted receiver is now a transmitter, and sends a signal to an array of four receiving elements that feed into RF chains which demodulate the signal before the output of each receiving element is separately sampled by the control module. This can be seen in Figure 3.

Although the high level system architecture is very similar between the transmit and receive cases, digital receive beamforming does not steer the radiation pattern of the array, but rather the spatial sensitivity pattern. This is done through manipulation of the sampled baseband signals in the digital domain, allowing for a greater flexibility of beam patterns and number of beams in addition to a simpler hardware design[2].

In the case of the type of digital receive beamforming explored in this project, the output signal of the entire

[2]There is no need for phase shifting operations at RF frequencies, which typically contributes to a large part of the monetary cost in an analog beamforming system.

receiver array is the sum of the products of each element's output with a complex weight[3]. Furthermore, adaptive beamforming algorithms can be used to detect the desired signal and optimise array characteristics (the aforementioned weights) to receive such a signal even in a time-varying system. These algorithms can be viewed as a closed-loop control system or optimization problem where the power of the desired signal is maximised, and the algorithm used for weight calculation (Least-Mean Squares) will be further explored in Section 2.

As such, the two described beamforming processes effectively allow for improvement in the directivity of both the transmitter and receiver. These techniques (beamsteering/spatial filtering) have been applied to many fields such as radar, wireless communications, and acoustics.

1.1 Report Roadmap

The report is organized in the following manner (seen in Figure 4). Refer to the table of contents for a full list.

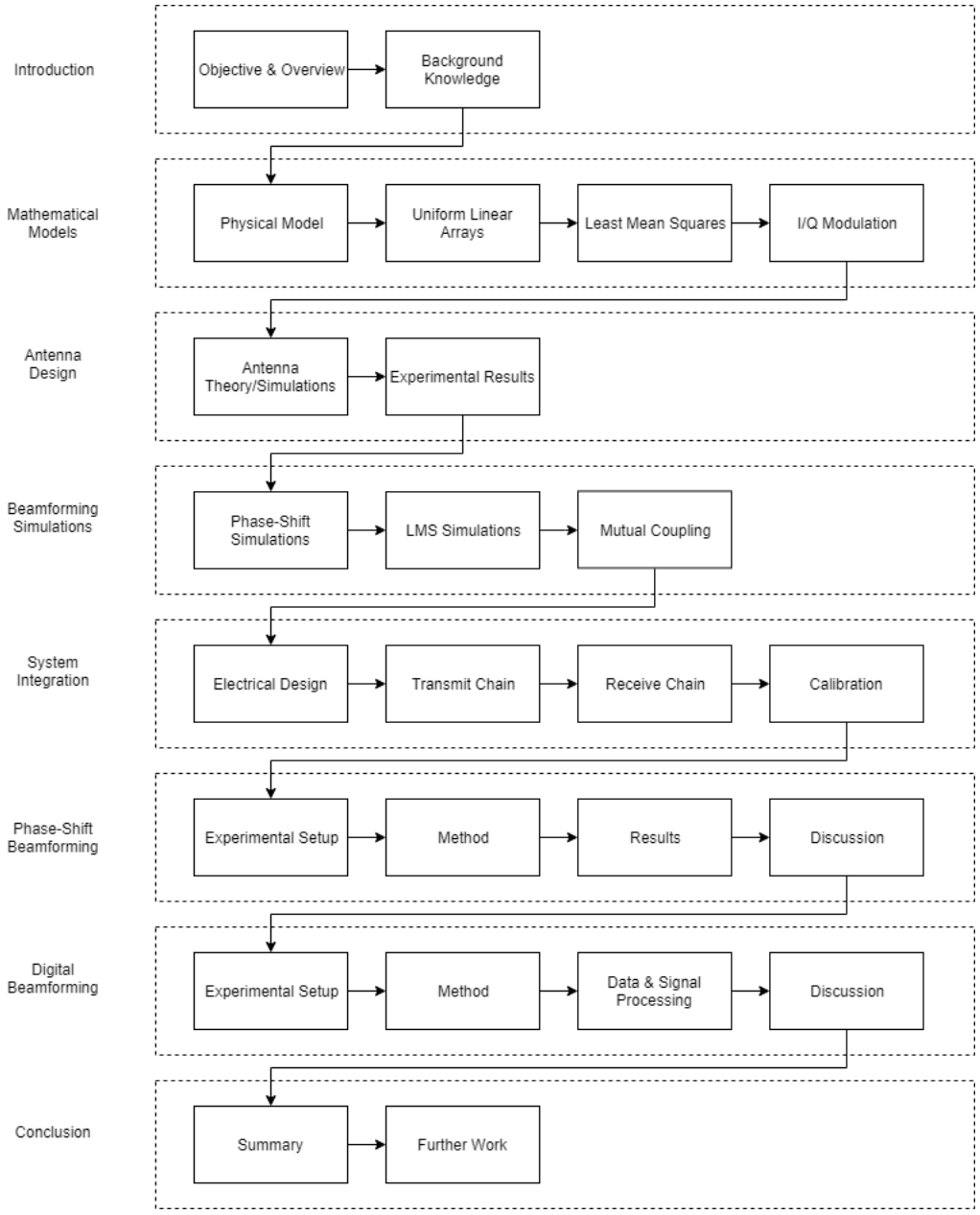


Figure 4: Report Roadmap

In this section (Introduction), we discuss the overall objective and purpose of this report as well as providing a

[3]A complex weight allows for manipulation of both phase and gain

high-level overview of the designed and implemented beamforming system, along with a conceptual overview of the two types of beamforming (phase-shift and digital) explored in the project.

In Section 2, the mathematical underpinnings of classical beamforming theory is explained. The basic physical model is discussed before specialising into the case of narrowband signals being transmitted from a uniform linear array. Theory on the Least Mean Squares algorithm for adaptive digital receive beamforming is also covered.

In Section 3, the design of an antenna suitable for a frequency within the range of 800-1000MHz is shown which will be used later as a building block in the beamforming system. Simulation and experimental evaluation of the antenna radiation pattern is performed, along with discussion of performance measures such as half power bandwidth and return loss are discussed.

In Section 4, the simulated design of an antenna array using results from the previous section is covered - an appropriate spacing is decided on for the array to support both transmit and receive beamforming, accounting for tradeoffs such as widening of the main lobe and steering capability. Simulations of the LMS algorithm are also examined in order to explore the relationship between parameters and beamformer output. Lastly, mutual coupling between adjacent elements is also considered and simulated.

In Section 5, the detailed design considerations of the two beamforming system realizations are described. The electrical component/schematic level design of individual circuit board modules comprising the RF chains are shown along with methodology on how individual modules are tested and calibrated.

In Section 6, an experiment in which the entire system is used to perform phase-shift beamforming is performed, along with analysis of results collected that demonstrate the successful steering of the array, matching the expected theoretical simulations. Sources of error and discrepancy are also discussed.

In Section 7, a receive beamforming experiment using the elements constructed in Section 5 is performed, and an analysis of the LMS beamformer output and performance is shown.

Finally, in Section 8, a summary of the report is provided along with suggestions for further work in the field.

2 Mathematical Models

The following section is an introduction to the theory behind the transmit and receive beamforming techniques used in this project, namely phase-shift beamforming and Least Mean Squares (LMS) beamforming. The mathematical derivations examined in this section are presented for the sake of completeness, and covered in full detail by Van Trees[1] in the text *Optimum Array Processing*.

The transmit and receive beamforming techniques are conceptually different from each other. The goal of the phase-shift beamforming in the transmit case is to electrically steer the radiation pattern of an antenna array by adjusting the phase of the carrier signal in a way that creates constructive/destructive interference at desired locations in the far field of the array. In the receive case, LMS beamforming processes the baseband signal in the digital domain and combines these signals such that the received message power is increased/decreased from desired locations in the far field of the array.

This section will begin by examining the physical model in the general case and for the uniform linear array before delving into the theory behind phase-shift and LMS beamforming.

2.1 Physical Model

2.1.1 General Case

Consider the model shown in Figure 5 where the plane wave $f(t, x, y, z)$ is transmitted by N isotropic point sources which are located at $\mathbf{p}_i = [p_{x_i}, p_{y_i}, p_{z_i}]^T$ with $i \in \{0, ..., N-1\}$.

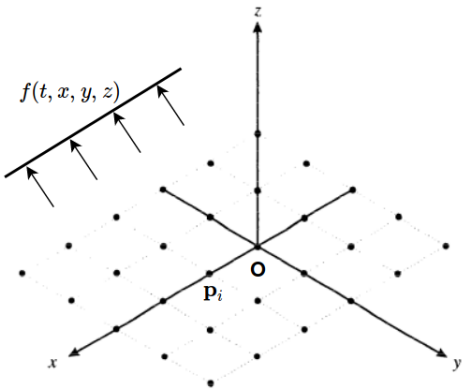


Figure 5: Plane wave arriving at an arbitrary array of sensors

Plane waves are typically described in terms of their temporal and spatial frequencies. The plane wave emitted by the isotropic point source located at $\mathbf{p}_i$ will have the form:

$$s_i(t, x, y, z) = e^{j\omega t - \mathbf{k}^T \mathbf{p_i}}. \tag{1}$$

Here $\mathbf{k}$ is defined as the wavenumber of the plane wave with $\mathbf{k}$ being related to ω via the relationship:

$$\mathbf{k} = \frac{\omega}{c}\mathbf{a} = \frac{2\pi}{\lambda}\mathbf{a} \tag{2}$$

with λ being the wavelength of the signal with frequency ω. The vector $\mathbf{a}$ represents the plane wave's direction of travel and is given by:

$$\mathbf{a} = \begin{bmatrix} -\sin\theta\cos\phi \\ -\sin\theta\sin\phi \\ -\cos\theta \end{bmatrix} \tag{3}$$

with θ and ϕ representing the azimuth and elevation angle related to the direction $\mathbf{a}$. It is worth noting that $|\mathbf{a}| = 1$.

Referring back to Figure 5, the plane wave $f(t, x, y, z)$ can be constructed by summing the plane waves emitted by each point source $s_i(t, x, y, z)$. The plane waves described by $s_i(t, x, y, z)$ will have the same direction in the far field hence their direction can be described by the same $\mathbf{a}$. This assumption holds because the length of the array is small compared to the distance to the far field which means that the waves appear as parallel fronts.

Now assume that the same signal is emitted from each point source without any coupling between the sources. Taking the origin as the reference, it can be said that $f(t, x, y, z)$ will be made up of delayed versions of the same signal. The delay τ_i can be determined by finding the projection of $\mathbf{p}_i$ in the direction of $\mathbf{a}$. The plane wave is assumed to travel at the speed of light in a vacuum c, and exist in a locally homogenous and infinite medium (this assumption avoids the modal behaviour displayed in non-homogenous spaces). Specifically:

$$\tau_i = \frac{\mathbf{a}^T \mathbf{p}_i}{c}, \tag{4}$$

This delay term is present in (1) by recognising that:

$$\mathbf{k}^T \mathbf{p}_i = \frac{\omega}{c} \mathbf{a}^T \mathbf{p}_i = \omega \tau_i \tag{5}$$

A useful approach to compactly write the plane wave $f(t, x, y, z)$ is to take the Fourier Transform of (1):

$$s_i(t, x, y, z) \xrightarrow{\mathcal{F}} S(\omega) e^{-j\mathbf{k}^T \mathbf{p}_i} \tag{6}$$

and then define the array manifold vector $\mathbf{v_k}(\mathbf{k})$:

$$\mathbf{v_k}(\mathbf{k}) = \begin{bmatrix} e^{-j\mathbf{k}^T \mathbf{p}_0} \\ e^{-j\mathbf{k}^T \mathbf{p}_1} \\ \vdots \\ e^{-j\mathbf{k}^T \mathbf{p}_{N-1}} \end{bmatrix} \tag{7}$$

where the subscript $\mathbf{k}$ denotes that the argument is in $\mathbf{k}$ space. The array manifold vector can be understood as containing information about the physical structure of the array. Finally, the plane wave transmitted by the array is:

$$\mathbf{F}(\omega) = S(\omega) \mathbf{v_k}(\mathbf{k}) \tag{8}$$

2.1.2 Uniform Linear Array

A uniform linear array (ULA) is an array with the radiating elements equally spaced along a line, centered at the origin. The elements are spaced d meters apart along the $\hat{\mathbf{y}}$ direction with norm of the elements facing the $\hat{\mathbf{x}}$ direction. This is shown in Figure 6.

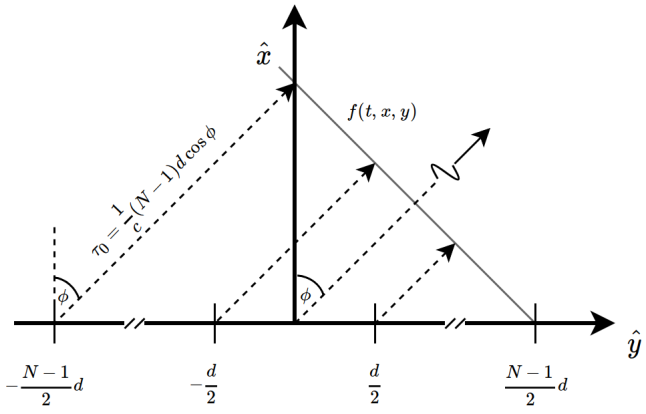


Figure 6: Plane wave transmitted by a Uniform Linear Array

Throughout this project, only plane waves arriving in the $x-y$ plane (elevation angle = 0°) were considered due to the linear nature of implemented arrays preventing any steering in other elevation angles. This decision, as well as positioning the sensors along the $\hat{\mathbf{y}}$ direction results in:

$$\mathbf{p}_i = \begin{bmatrix} 0 \\ p_{y_i} \\ 0 \end{bmatrix} \tag{9}$$

$$\mathbf{a} = \begin{bmatrix} -\sin\phi \\ -\cos\phi \\ 0 \end{bmatrix} \tag{10}$$

$$\tau_i = -\frac{p_{y_i}\cos\phi}{c} \tag{11}$$

The resulting array manifold vector is:

$$\mathbf{v_k}(\phi) = \begin{bmatrix} e^{-j\frac{2\pi}{\lambda}p_{y_0}\cos\phi} \\ e^{-j\frac{2\pi}{\lambda}p_{y_1}\cos\phi} \\ \vdots \\ e^{-j\frac{2\pi}{\lambda}p_{y_{N-1}}\cos\phi} \end{bmatrix} \tag{12}$$

The result in (12) can be substituted into (8) to give:

$$\mathbf{F}(\omega) = S(\omega)\mathbf{v_k}(\phi) \tag{13}$$

2.2 Phase-Shift Beamforming

Conventional beamforming (otherwise known as delay-and-sum or phase-shift beamforming) involves delaying each element of $\mathbf{F}(\omega)$ described in (8) such that they are weighted equally with reference to the original signal before adding each the components from each element together. Such a system is shown in Figure 7.

First, to show that the delays τ_i can be approximated by a phase shift in the case of a narrowband signal, it is necessary to consider the input signal $s_i(t)$. If $s_i(t)$ is a bandpass signal then it can be written as:

$$s_i(t) = \sqrt{2}\mathcal{R}e\{\tilde{s}_i(t)e^{j\omega_c t}\}, \tag{14}$$

where ω_c is the carrier frequency and $\tilde{s}_i(t)$ is the complex envelope. For the delayed signal it follows that:

$$s_i(t) = \sqrt{2}\mathcal{R}e\{\tilde{s}_i(t-\tau_i)e^{j\omega_c(t-\tau_i)}\}, \tag{15}$$

If the bandwidth of $\tilde{s}_i(t)$ is much smaller than the time taken for the plane wave to travel across the array (the maximum time in the case of a linear array is the side-fire scenario from one end of the array to the other), it can approximated that:

$$\tilde{s}_i(t-\tau_i) \approx \tilde{s}_i(t) \tag{16}$$

This result can be substituted into (15) to give:

$$s_i(t) = \sqrt{2}\mathcal{R}e\{\tilde{s}_i(t)e^{-j\omega_c\tau_i}e^{j\omega_c t}\}, \tag{17}$$

which shows that delays τ_i can be approximated by phase shifts.

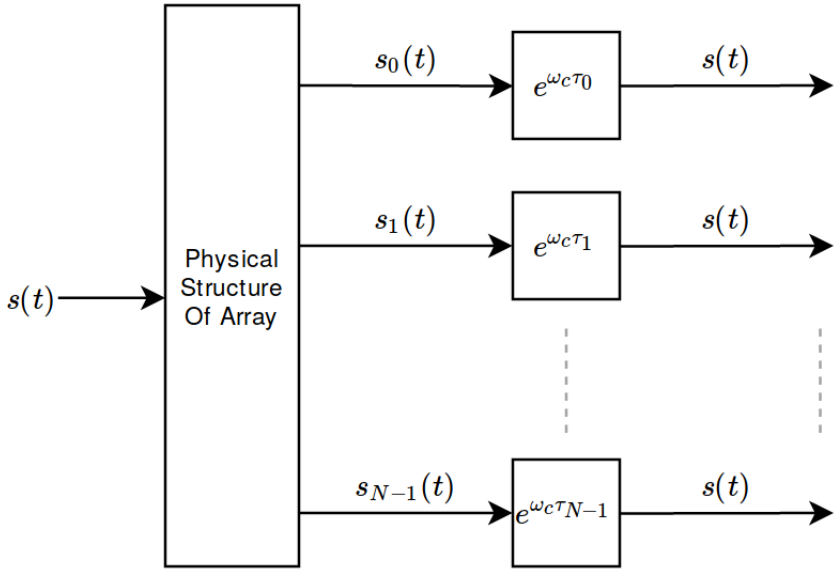


Figure 7: Phase-shift beamformer

The operation described in Figure 7 can be written as:

$$Y(\omega) = \mathbf{H}^T(\omega)\mathbf{F}(\omega), \tag{18}$$

where $Y(\omega)$ is the output, and,

$$\mathbf{H}^T(\omega) = \mathbf{w}^H = \mathbf{v}_\mathbf{k}^H(\phi_s), \tag{19}$$

for the ULA case with with $\mathbf{w}$ denoting the complex weight vector. Note that the H superscript denotes the complex conjugate transpose of a vector. The term ϕ_s in (19) refers to the steering angle measured as the angle of arrival of the plane wave. By substituting (13) and (19) into (18), it follows that:

$$Y(\omega) = \mathbf{w}^H \mathbf{v}_\mathbf{k}(\mathbf{k})S(\omega) = NS(\omega) \tag{20}$$

The output of the beamformer in (18) can be written in terms of the frequency-wavenumber response function $\Upsilon(\omega, \mathbf{k})$ as shown in (21). This function describes the complex gain of the array.

$$Y(\omega) = \Upsilon(\omega, \mathbf{k})S(\omega) \tag{21}$$

The term $\Upsilon(\omega, \mathbf{k})$ can be understood to be the transfer function of the array. A useful approach is to rewrite (21) in terms of the complex variable z. This can be achieved by:

$$\begin{aligned} \Upsilon(\omega, \phi_s) &= \mathbf{w}^H \mathbf{v}_\mathbf{k}(\phi_s) \\ &= \sum_{i=0}^{N-1} w_i^* e^{-j(n-\frac{N-1}{2})\frac{2\pi}{\lambda} d\cos\phi} \end{aligned} \tag{22}$$

Making the substitution:

$$\psi = \frac{2\pi}{\lambda} d\cos\phi, \tag{23}$$

and:

$$z = e^{j\psi} \tag{24}$$

(22) can be written as:

$$\Upsilon(z) = z^{-\frac{N-1}{2}} \sum_{i=0}^{N-1} w_i^* z^n \tag{25}$$

Viewing the frequency-wavenumber response function in the z domain provides some insight into how the array operates and how to choose array design parameters. For instance, the beam pattern of the array can be found by evaluating (25) over the domain of $z = e^{j\psi}$. An example of the magnitude response of a 10 element array with uniform weights is shown in Figure 8.

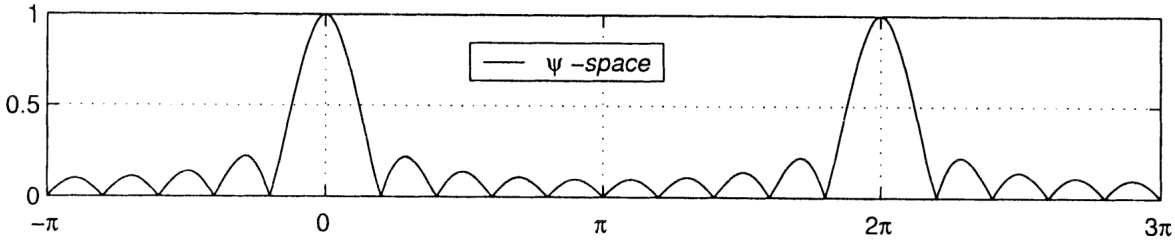


Figure 8: Magnitude response of a 10 element array with $d = \frac{\lambda}{2}$.

The plot in Figure 8 shows where the power of the array will be directed. The visible region of the array is defined for the region $-\frac{2\pi}{\lambda} d \cos\phi \leq \psi \leq \frac{2\pi}{\lambda} d \cos\phi$ with the region outside these bounds defined as the virtual region. Performance parameters of the array are main lobe width and sidelobe attenuation. A narrow main lobe and low power in the sidelobes is ideal as it results in most of the transmitted power being directed towards the intended target.

From (25) it can be seen that the elements of the array act as zeros in $\Upsilon(z)$. Due to this, when more elements are added to the array the main lobe width and sidelobe power of $|\Upsilon(z)|$ is reduced. It is therefore desirable to have more elements in the array. This comment neglects the effects of coupling between the elements which can perturb the beam pattern.

Steering of the array can be considered as shifting the magnitude response in the z domain. To steer the array in the direction of ϕ_s the weights should be chosen such that:

$$\mathbf{w} = \begin{bmatrix} e^{-j\frac{2\pi}{\lambda} p_{y_0} \cos\phi_s} \\ e^{-j\frac{2\pi}{\lambda} p_{y_1} \cos\phi_s} \\ \vdots \\ e^{-j\frac{2\pi}{\lambda} p_{y_{N-1}} \cos\phi_s} \end{bmatrix} \tag{26}$$

The complex weights shown in (26) can also include the magnitude component which can be used to shape $|\Upsilon(z)|$. This problem is similar to choosing the weights of a windowing function in filter design. Depending on the number of elements in the array, applying different windows to the array may provide some benefit by reducing the power in the sidelobes. This is typically at the expense of the main lobe width and so there is a trade-off between mainlobe width and sidelobe power.

The spacing term d in (23) will expand or compress the spectrum of the magnitude response. Ideally, the spacing of the elements is as far apart as possible such as to reduce the effects of coupling between the elements. For spacings above $d = \frac{\lambda}{2}$, the spectrum becomes compressed to the point which main lobes in the virtual region are steered into the visible region. These lobes are referred to as grating lobes and are to be avoided when phase-shift beamforming.

In general, the pattern of the array is the multiplication of the beam pattern by the field $S(\omega)$. This is true because the array performs a linear operation on the inputs and the output is a linear combination of these inputs. The assumption here is that the signal transmitted by the array element has not been affected by adjacent antenna elements. This effect is known as mutual coupling and is explored in Section 4.

2.3 Least Mean Squares (LMS) Algorithm Beamforming

Receive beamforming can be performed on the digitised baseband signals received by the elements in an array. An example of such a system is shown in Figure 9. Here, the samples $x_i(n)$ are filtered by the complex weight w_i before being combined with the filtered signals from the other elements in the array. The complex weights $\mathbf{w} = [w_1 \ w_2 \ \dots \ w_n]^T$ are adjusted by an algorithm that uses the error between the output $y(n)$ and some desired signal $d(n)$ to meet some criteria. An algorithm that is commonly used in receive beamforming to adjust the weights is the LMS algorithm.

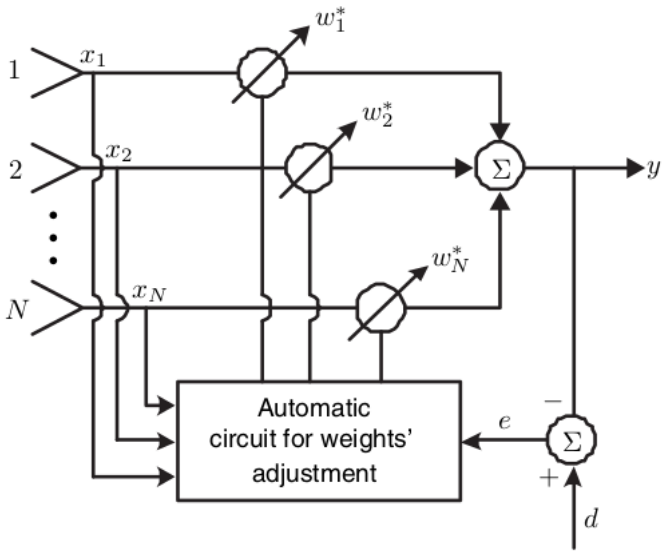


Figure 9: Block diagram of adaptive beamforming system[2].

The Least Mean Squares algorithm[3] is an adaptive beamforming algorithm that uses gradient descent optimization to calculate a weight vector $\mathbf{w}$ which minimises the mean squared error between the output of the beamformer and a known calibration signal. The received signals from the array elements are then digitally filtered using the calculated weights once the calibration has been completed.

Consider a ULA with N isotropic elements, in which the output of the array $y(n)$ is given by:

$$y(n) = \mathbf{w}^H(n)\mathbf{X}(n) \tag{27}$$

where $\mathbf{X}(n) \in \mathbb{R}^N$ is the sample of the baseband signal from each element in the array at time instance n. In the literature, $\mathbf{X}(n)$ is referred to as a snapshot. The weights $\mathbf{w}(n)$ are written as a function of n because they are adjusted after each iteration of the algorithm.

An iterative gradient descent algorithm can be described as follows:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{1}{2}\mu[-\nabla(\mathbf{E}[e^2(n)])] \tag{28}$$

where $e^2(n) = [d(n) - \mathbf{w}^H\mathbf{X}(n)]^2$, the mean squared error between the calibration signal and the weighted input. Here, $\mathbf{E}$ is defined as the expectation operator.

The step size parameter μ must be chosen such that the tradeoff between convergence time and stability of convergence is optimized. A small step size will converge slowly but the estimated weights will be stable. A large step size will converge quickly however there is a risk that the weights will oscillate about the optimal point and may result in an unstable system.

The gradient vector is then computed as such

$$\nabla_{\mathbf{w}^H}(\mathbf{E}[e^2(n)]) = -2r + 2R\mathbf{w}(n) \tag{29}$$

where the r and R are covariance matrices. The computation is simplified by using instantaneous covariance as opposed to actual covariance.

$$R(n) = \mathbf{X}(n)\mathbf{X}^H(n) \tag{30}$$

$$r(n) = d(n)\mathbf{X}(n) \tag{31}$$

Substituting (30) and (31) into the update step (28), the following equation is obtained

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu\mathbf{X}(n)[d(n) - \mathbf{X}^H(n)\mathbf{w}(n)] \tag{32}$$

A single iteration of the algorithm is as follows

1. Generate Output: $y(n) = \mathbf{w}^H\mathbf{x}(n)$
2. Calculate Error: $e(n) = d(n) - y(n)$
3. Update Weights: $\mathbf{w}(n+1) = \mathbf{w}(n) + \mu\mathbf{X}(n)e^*(n)$

It is possible to choose the step size μ such that stability is guaranteed in the mean squared error sense. This condition is:

$$0 \leq \mu \leq \frac{2}{\lambda_{max}}, \tag{33}$$

where λ_{max} is the largest eigenvalue of $R(n)$. Assuming that the noise is additive white Gaussian, the signals produced by the array elements will be uncorrelated and independent. This results in $R(n)$ being a diagonal matrix with it's eigenvalues along the diagonal. Given that $R(n)$ is only an estimate, another upper bound on μ that is more likely to be stable is usually used.

$$0 \leq \mu \leq \frac{2}{\mathrm{Tr}\,(R(n))} \tag{34}$$

Given that the statistics are updated after every iteration of the algorithm, it follows that the step size changes on every iteration. This step size will eventually converge to a constant value as more data is seen and the statistics become better known.

The convergence characteristics of the LMS algorithm are dependent on the eigenvalues of $R(n)$ and convergence can be slow when the eigenvalues are widely spread [2]. Other algorithms exist which converge faster such as the recursive least squares algorithm. However the LMS algorithm was used in this project due to it's low complexity and robustness. As the goal of the project is only to demonstrate basic beamforming, it is possible to create an experimental setup with low variance by fixing the location of both the calibration source and the array before adjusting for any multi-path issues that arise.

3 Antenna Design

The antenna element is a key component in the array design for both the transmit and receive array. In the ideal case, an array will be capable of steering the main lobe (of either the radiation or spatial sensitivity pattern) in a way that it radiates and receives equal amount of power in any steered direction. As Section 2.2 describes, the far field pattern of an array is the multiplication of the array factor and the element factor. This idea along with the requirement to steer the main lobe in any direction with equal power means that the antenna element should ideally be omnidirectional.

Additionally, the decision to perform array element design at approximately 1GHz is due to the availability of RF electronics at the required frequencies within the cost budget of the project. The desired frequency for antenna design is 920MHz to match simulated results, although frequencies within the range 800-1000MHz are supported by the electronic design (see Section 5 for more details) and therefore acceptable.

Considering that project will be using a ULA orientated with the norm directed at an azimuth angle of 0°, the antenna element only needs to be approximately omnidirectional for azimuth angles between -90°to +90°. Furthermore, the antenna element should be simple to manufacture and integrate into the array.

An antenna element that meets this requirement is a patch antenna. Volakis et al. [4] shows that a circular patch antenna has no more than 5dB reduction in power over an azimuth angles -70°to 70°. Disadvantages to using a patch antenna are generally lower radiation efficiency and small bandwidth [4]. These disadvantages were considered irrelevant in terms of demonstrating array steering.

The general structure of a circular patch antenna as shown in Figure 10 consists of a ground plane, substrate, patch and feed. The radius of of the circular patch can be calculated using Equations 35 and 36 [4].

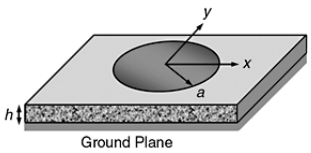


Figure 10: Structure of circular patch antenna [4].

$$f_{resonance} = \frac{c}{2\pi a_e \sqrt{\epsilon_r}} 1.8418 \tag{35}$$

$$a_e = a\sqrt{1 + \frac{2h}{\pi a \epsilon_r}(\ln \frac{\pi a}{2h} + 1.7726)} \tag{36}$$

Where:

- $f_{resonance}$ is the resonant frequency of the antenna
- c is the speed of light
- ϵ_r is the dielectric constant of the substrate
- c is the speed of light
- a is the actual radius of the patch antenna
- a_e is the effective radius of the patch antenna which accounts for the fringing effect[4]
- h is the thickness of the substrate

Since the antenna will be used in an array, it is desirable to reduce the physical size of the element while maintaining the same resonant frequency. If the antenna is physically smaller, coupling between elements will be reduced and the array will more closely approximate the ideal case. Drabowitch et al.(2005, p. 153) [5] indicates that miniaturisation of microstrip antennas can be achieved by modifying the geometry of the antenna through the introduction of slots. One such example is found in a paper presented by Ariff et al. [6]. The proposed design added slots and notches to a standard circular patch as shown in Figure 11. These modifications to the standard geometry reduce the radius of the patch and cause the antenna to be circularly polarised. A

[4]The fringing effect results from discontinuities in the boundary conditions of the antenna

circularly polarised antenna has the advantage of being less susceptible to multipath issues due to the fact that the direction of polarisation flips when the signal reflects off a surface, and that a signal with opposite direction of polarization to a receiver is severely attenuated.

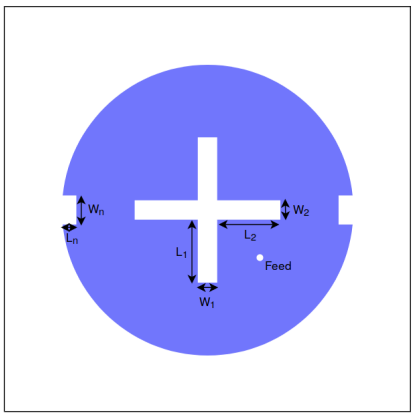


Figure 11: Circular patch antenna geometry proposed by Ariff et al. [6].

The feed method used in this design is a SMA coaxial connector. Coaxial connectors are easily impedance matched and also reduce spurious feed radiation when compared to other methods. However, the connectors tend to have a narrow bandwidth [7]. This disadvantage of coaxial connectors does not affect array performance because of the narrow band nature of the project.

The antenna is connected to the electronics through 50Ω coaxial cable and SMA connectors meaning that the antenna needs to be impedance matched to the connector. At the resonant frequency the antenna will ideally have an impedance of $50 + j0\Omega$. This will ensure that most energy is abosrbed by the antenna and radiated away whilst minimal energy is reflected back to sensitive RF chain components (mainly amplifiers) which can be damaged by such reflections. Return loss can also be used to measure the amount of energy reflected and can be measured using a vector network analyser. A return loss of greater than 15dB is considered sufficient for a functioning antenna[5].

A challenge in the design of the antenna was determining the dielectric constant of the substrate. Substrates used in the manufacture of PCBs have dielectric constants that are frequency dependent and materials with tightly controlled tolerances such as Rogers RO4000 Series laminates [8] are significantly more expensive than standard FR-4. The manufacturer of the antenna used in the project provides specifications for the ϵ_r of their standard FR-4 as $4.5Fm^{-1}$. This value was used in the initial design despite not reflecting the frequency dependent nature of ϵ_r. Because $a_e \propto \frac{1}{\sqrt{\epsilon}}$ approximately holds, an error in ϵ_r would result in a resonant frequency shift which could be measured during testing and accounted for in subsequent designs.

An iterative approach was used in designing the antenna element. MATLAB's Antenna Designer toolbox was used to simulate the antenna and to generate the Gerber files needed for manufacturing the design. The design proposed by Ariff et al. was used as a starting point for the design and used the ϵ_r provided by the manufacturer. The design was then tuned in MATLAB to produce an antenna with the required resonant frequency and suitable return loss. The design was manufactured and then tested using a mRS minVNA Tiny [9]. Parameters of the final design are listed in Table 1.

3.1 Simulation

The parameters listed in Table 1 were used to simulate the performance of the antenna in MATLAB using the Antenna Designer toolbox. The design can be seen in Figure 12. Figure 13a shows that the antenna is close to hemispherical radiation pattern with Figure 13b showing that there is approximately no more than 5dB loss for azimuth angles of -70°to +70°. These results indicate that the antenna will be suitable in an array.

[5]Common consumer grade antennae as of 2019 typically range from VSWR of 2.0-1.5, which translates to a 10-14dB of return loss

Table 1: Final antenna design parameters.

Parameter	Value (mm)
a	40.7
W_1	3.0
W_2	3.0
L_1	13.0
L_2	13.0
W_n	3.0
L_n	7.0
$L_{groundplane}$	120.0
$W_{groundplane}$	120.0

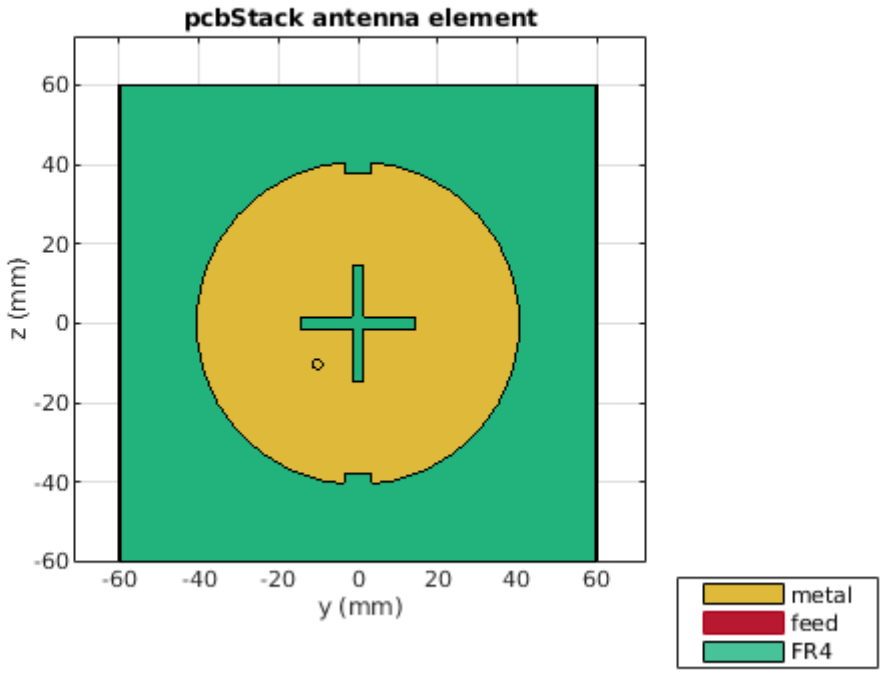
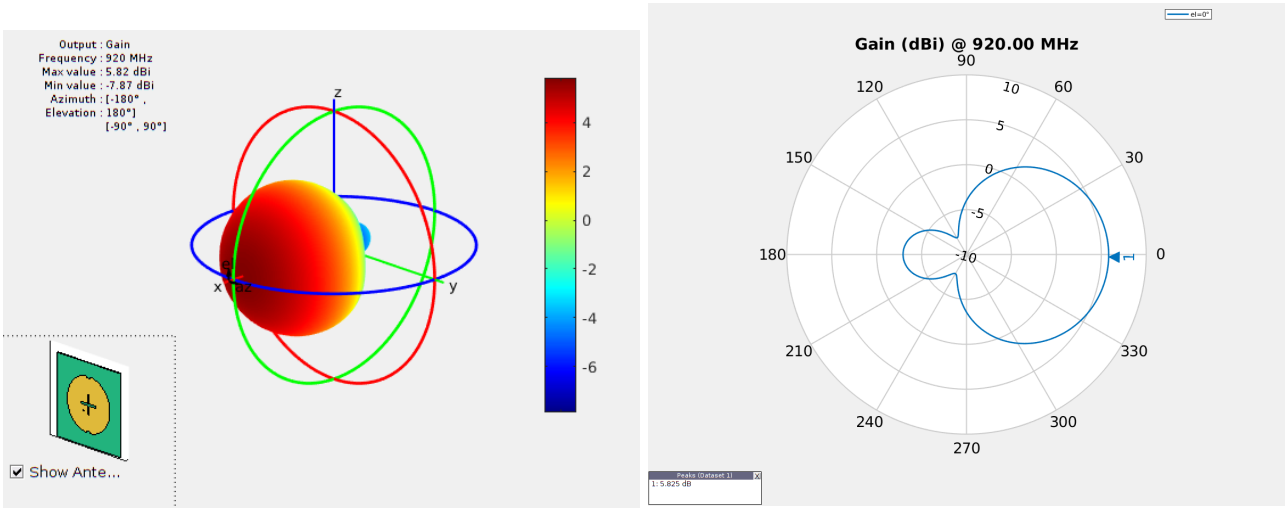


Figure 12: Final design of the circular patch antenna.



(a) 3D directivity of the circular patch antenna.

(b) Directivity plot of circular patch antenna in the $x-y$ plane.

Figure 13: Simulated directivity plots of the final circular patch antenna.

The return loss of the antenna was simulated using a 50Ω connector and the result is shown in Figure 14. Figure 14 indicates that the antenna is resonant at approximately 930MHz. The simulation was not expected to produce an antenna resonant at 920MHz because the radius of the patch was interpolated from the resonant frequency of two iterations of antennas. The details of the interpolation method are discussed in Section 3.2. It was found during testing however that there was a reasonably good match between the shape of the simulated and measured return loss plots. The simulated antenna is also shown to have sufficient bandwidth with Figure 14 showing a bandwidth of 15MHz for a return loss greater than 15dB.

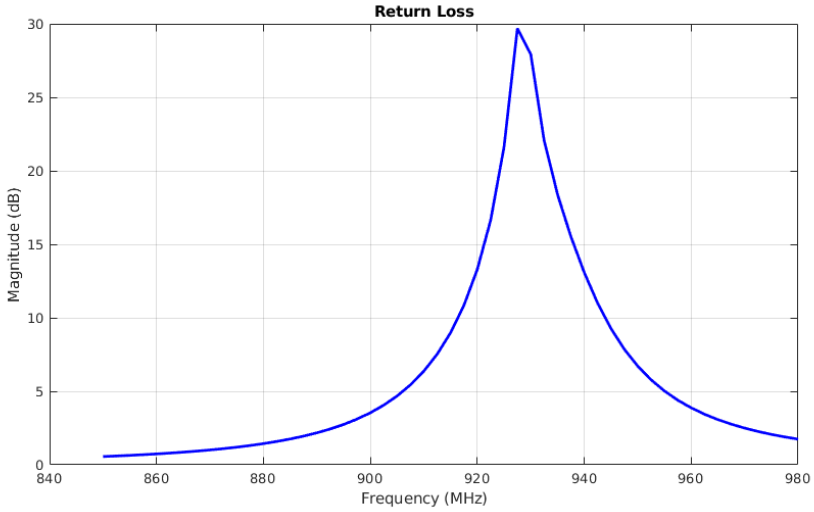


Figure 14: Return loss of antenna with modelled with 50Ω connector.

3.2 Evaluation

There were three iterations of the antenna design. The first design used the parameters listed in the paper presented by Ariff et al.[6] with a, L_1 and L_2 adjusted until a suitable antenna was simulated. The return loss of the antenna was measured using a mRS minVNA Tiny[9] with the antenna facing an open space. It was found that the resonant frequency of the first antenna was 876MHz. This result was not surprising given that the dielectric constant was practically unknown. It was also found that the shape of the measured return loss versus frequency plot was similar to that of the simulated antenna. These results indicated that the MATLAB Antenna Designer toolbox could simulate antennas sufficiently well given the correct dielectric constant.

An assumption was made using (35) and (36) that over a small interval, there was an approximately linear

relationship between $f_{resonance}$ and radius $\frac{1}{7}a$. This assumption was verified in Figure 15 by plotting the two equations with the known parameters.

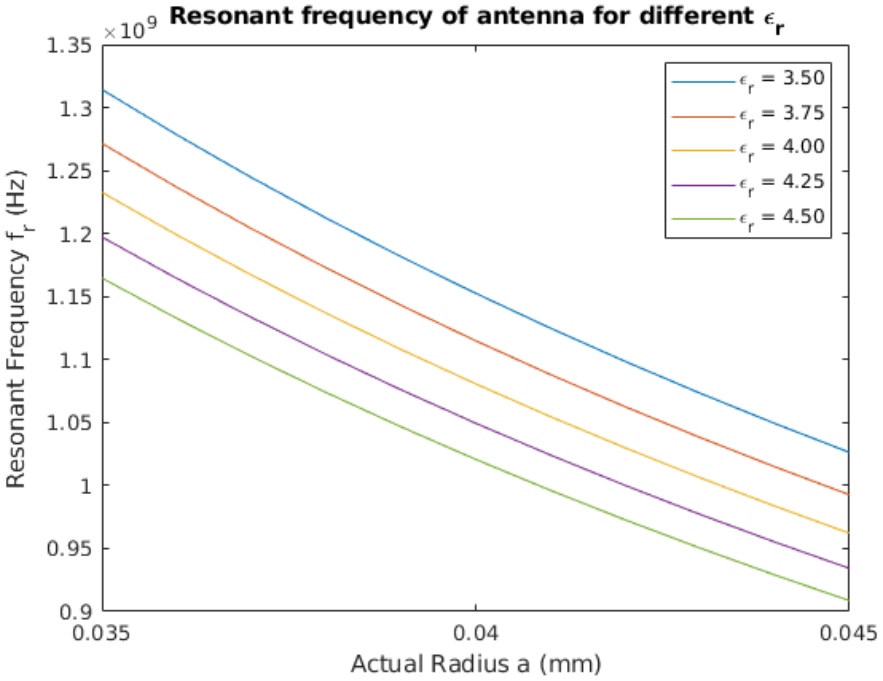


Figure 15: Calculated resonant frequency f_r for known parameters and different ϵ_r.

Using this assumption, the radius of the second antenna was decreased by a factor corresponding to the ratio of the desired and measured resonant frequency. All other parameters were unchanged for the second design. The resonant frequency of the second design was determined from the return loss versus frequency plot and was found to be 966MHz. The radius of the third antenna was determined by interpolating the radius using the data points from the two subsequent designs. The resonant frequency and radius of all designs is shown in Figure 16.

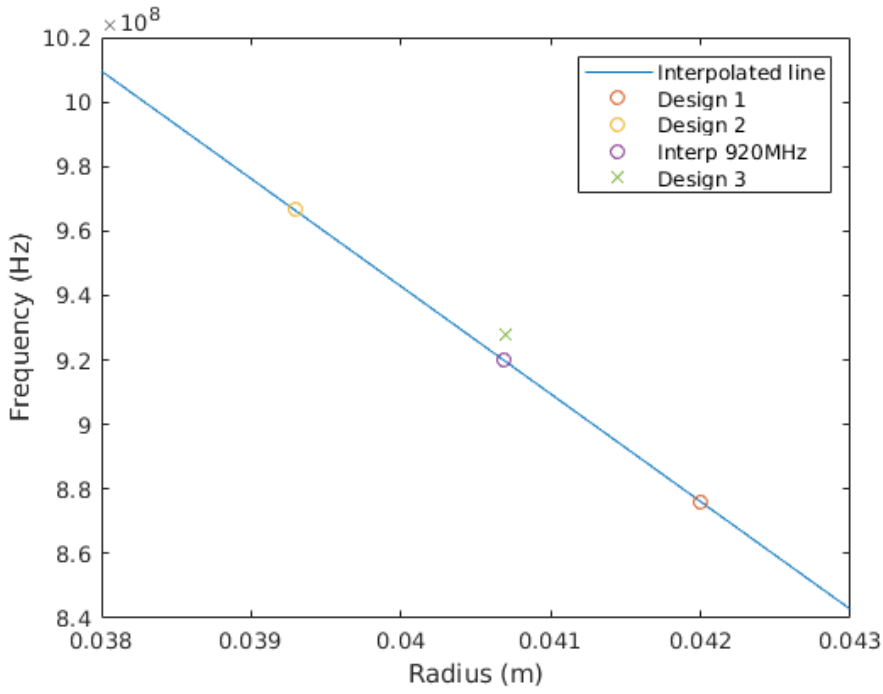


Figure 16: Resonant frequency and radius of all three designs.

The return loss of the third iteration of the antenna was measured and compared to simulation. The results are shown in Figure 17. It can be seen that the measured return loss is a close match to simulation. This was unexpected since the radius of the antenna was interpolated from measurements. An explanation for this could be that the batch of FR-4 used in the first and second iterations was different to the one used in the third iteration, resulting in a varying ϵ_r. Another reason could be that fixing the dimensions of the slots and notches while changing the radius had an unexpected effect on the resonant frequency. Regardless, there is more than 15dB return loss for the upper half of the targeted ISM band, meaning the antenna can operate sufficiently well within the desired frequencies. Operating within this region, the antenna has a bandwidth of 4MHz.

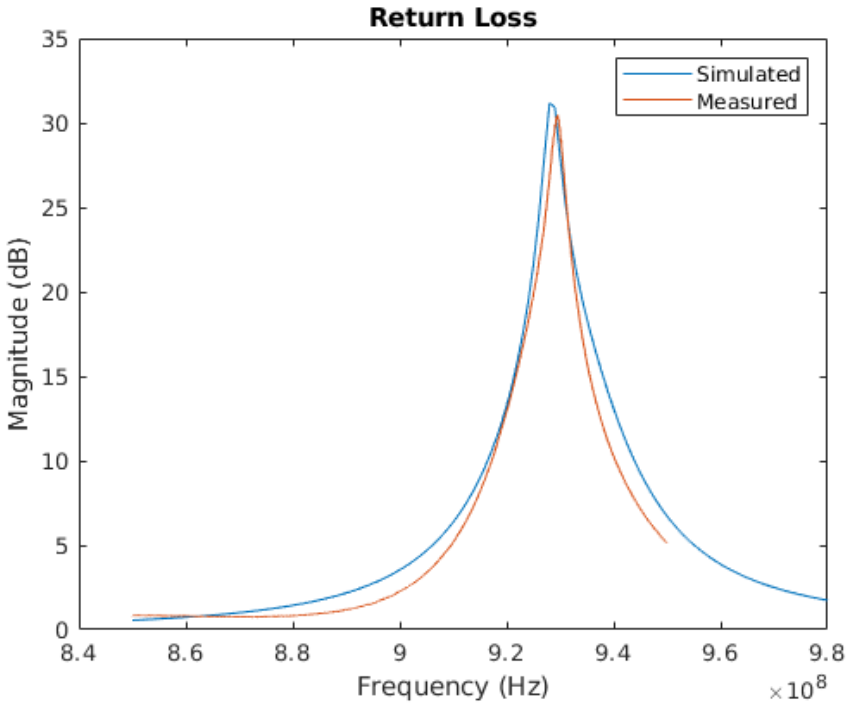


Figure 17: Measured and simulated return loss of third antenna design.

The directivity of the antenna was measured using the setup shown in Figure 18. Power measurements were taken at 5° intervals at points along a hemisphere of a 4m radius using an identical antenna.

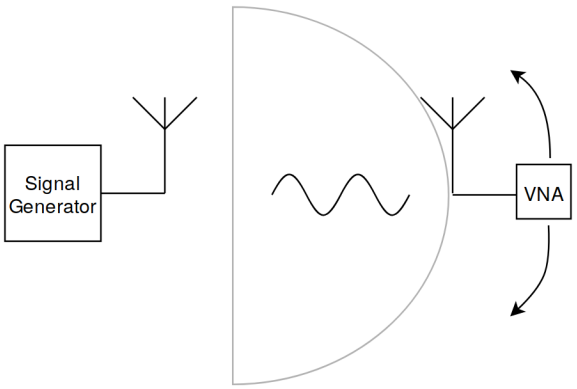


Figure 18: Setup for measuring antenna directivity.

The measured directivity of the single element was compared to simulation. The results are shown in Figure 19. It can be seen from the figure that there is a close match between the measured and simulated directivity.

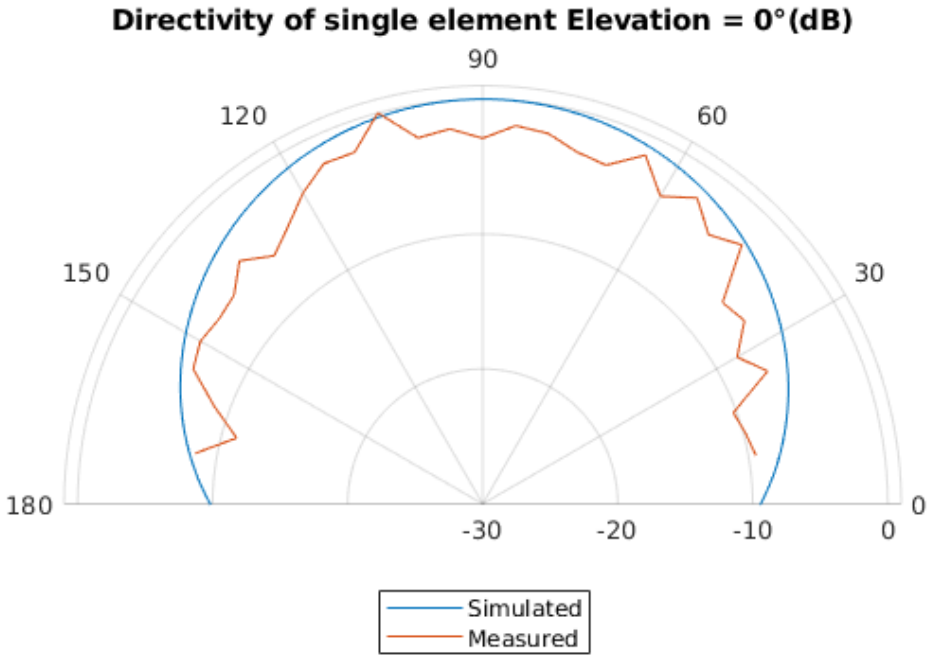


Figure 19: Measured and simulated directivity of third antenna design.

The results from testing show that the radiation pattern of the designed antenna is approximately omnidirectional for the angles of interest and that it operates within the ISM band with a bandwidth of 4MHz at 924MHz. This demonstrates that the antenna has met the design requirements.

4 Beamforming Simulations

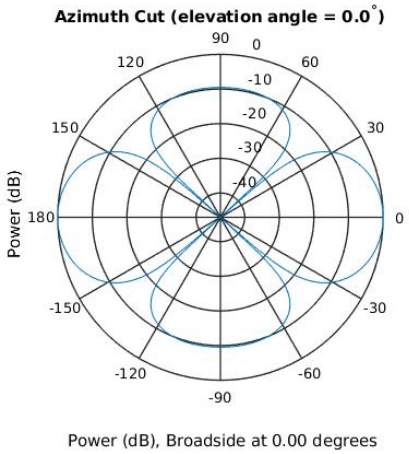
The operation of the transmit and receive beamforming systems were simulated using MATLAB's Phased Array toolbox. Using this toolbox, it was possible to gain a deeper understanding of beamforming, verify understanding of the theory presented in Section 2 and to verify design parameters before committing to manufacturing hardware. All simulations were performed with a carrier frequency of 920MHz.

The phase-shift simulations to be presented looked at how the beam pattern was effected by the number of elements, element spacing and weights. These simulations were then used to finalise the transmit system design. The LMS beamformer was also simulated in order to gain an understanding in how to integrate the analog and digital electronics as well as verifying the algorithm implementation. Mutual coupling was also simulated to verify that such effects on the system were minimal.

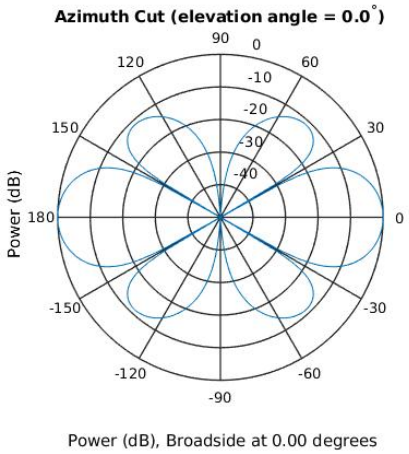
4.1 Phase-Shift Beamforming

4.1.1 Number of Elements

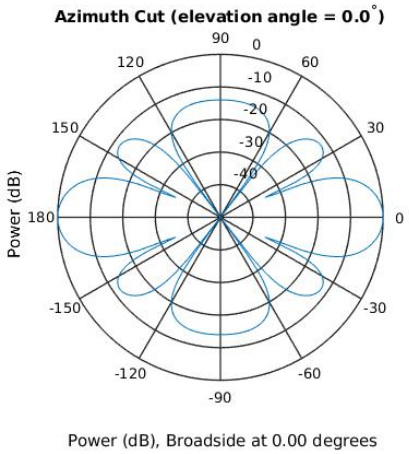
The effect of the number of elements in the array was assessed by using uniform weights at a fixed spacing of $\frac{\lambda}{2}$ with a perfect isotropic radiator as the antenna element. The arrays were then steered to an azimuth angle of 0°and 30°to assess the effects on steering. The results are shown in Figures 20 and 21.



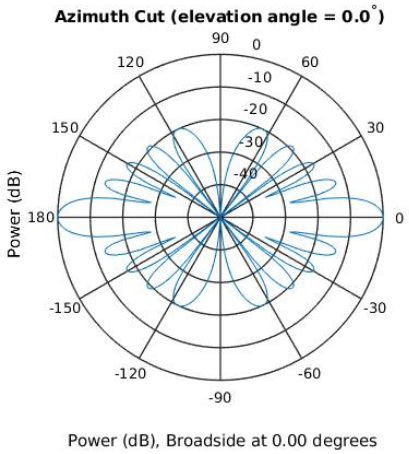
(a) N = 3



(b) N = 4



(c) N = 5



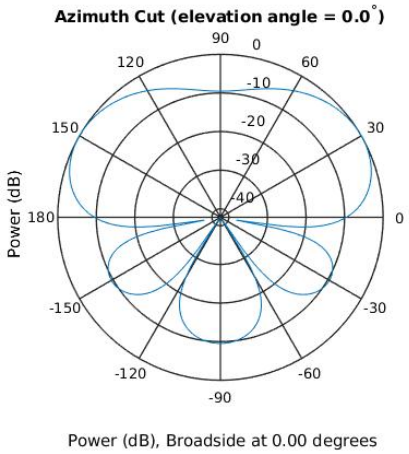
(d) N = 10

Figure 20: Beam pattern steered to 0° for a varied number of elements N. 0° is normal to the array.

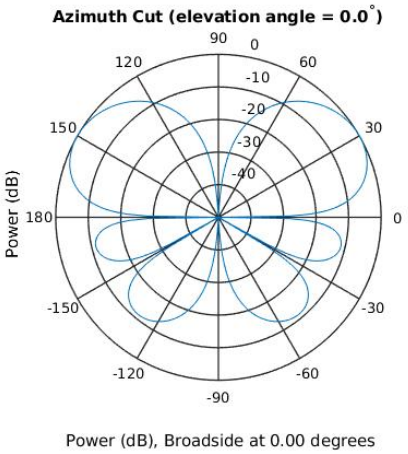
It is clear from Figure 20 that as the number of elements increases, the main lobe width decreases. Furthermore, the power in the sidelobes decrease as the number of elements increases. This illustrates that it is desirable to

have more elements in the array. The trade-off is that more elements results in a larger amount of electronics, increasing project costs, build times, and controller complexity.

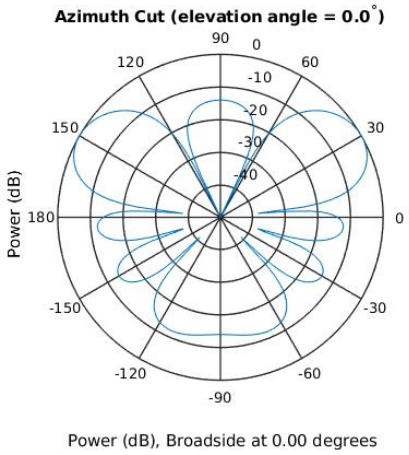
Note that the sidelobes radiate and receive energy from unintended directions. This leads to power wastage in these directions when transmitting, or increased noise being received. Ideally the total area under such sidelobes is minimized.



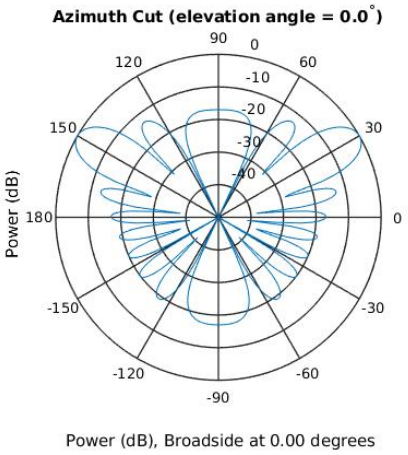
(a) N = 3



(b) N = 4



(c) N = 5



(d) N = 10

Figure 21: Beam pattern steered to 30°for a varied number of elements N. 0°is normal to the array.

The results in Figure 21 further illustrate that more elements in the array is desirable. When steered, the main lobe width increases, decreasing directionality.

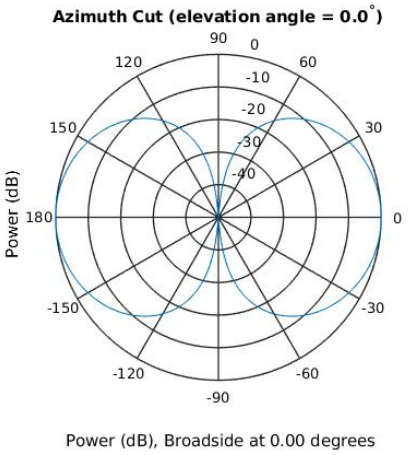
The project budget and time constraints were ultimately the deciding factor in determining the number of elements in the array. For each element added to the array an additional set of electronics for the transmit and receive chains was required. This additional cost along with the additional time to manufacture and test as well as the increased complexity required in the control software meant that the number of elements needed to be as little as possible while still being able to demonstrate beam steering. The results in Figure 21b showed that with $N = 4$ elements, reasonable results for beam steering could be achieved. As such, the project implemented four elements in the array design.

4.1.2 Element Spacing

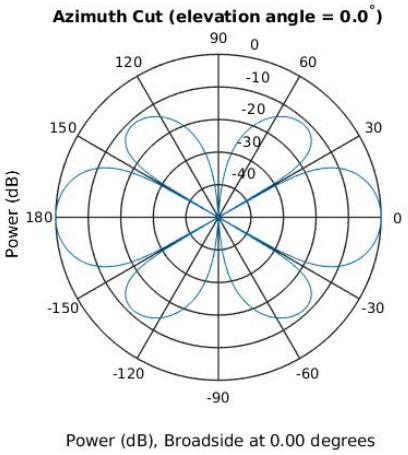
The impact of inter-element spacing was assessed by simulating beam patterns for spacings of 0.25λ, 0.5λ and 0.75λ. The number of elements was fixed at $N = 4$. The results are shown in Figure 22.

It can be seen that as the spacing increases, the main lobe width decreases. This is due to the fact that that the array is physically longer. The presence of a grating lobe can be seen in Figure 22c and is a direct result of the shift in the beam pattern shifting grating lobes into the visible region.

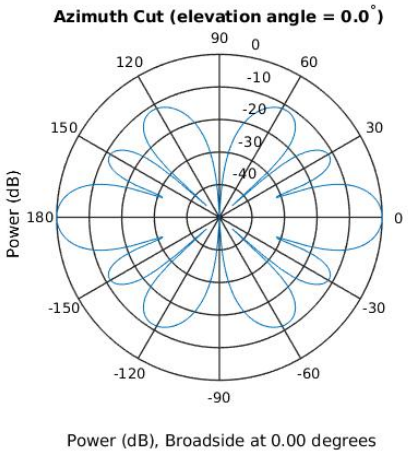
These results show the that the array spacing should be as large as possible, up to $d = \frac{\lambda}{2}$ so as to avoid grating lobes.



(a) d = $\frac{\lambda}{4}$



(b) d = $\frac{\lambda}{2}$

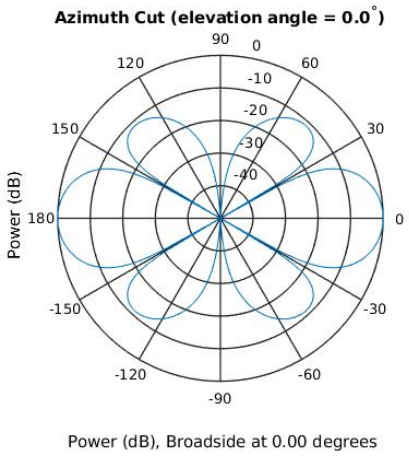


(c) d = $\frac{3\lambda}{4}$

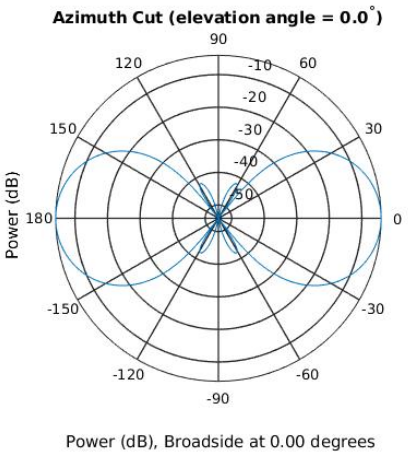
Figure 22: Beam pattern steered to 0° for various spacings

4.1.3 Array Weights

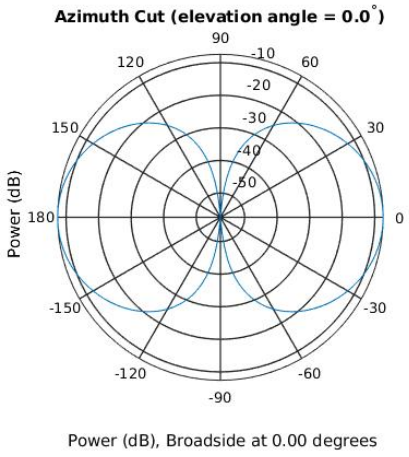
Several windowing schemes were assessed for their suitability for the array design. For the simulation, the number of elements was set to $N = 4$ and array spacing was set to $d = \frac{\lambda}{2}$. The results are shown in Figure 23 with the corresponding weights shown in Figure 24.



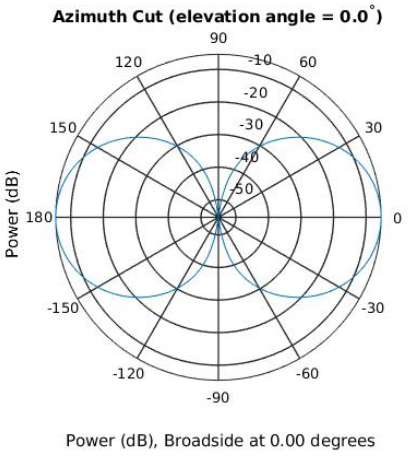
(a) Rectangular Window



(b) Hanning Window



(c) Hamming Window



(d) Kaiser Window

Figure 23: Beam pattern for various window weights for $N = 4$.

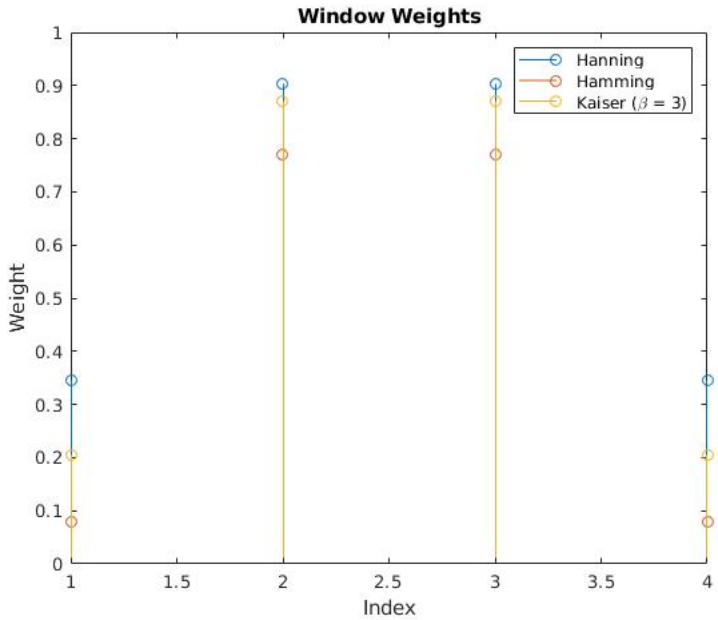
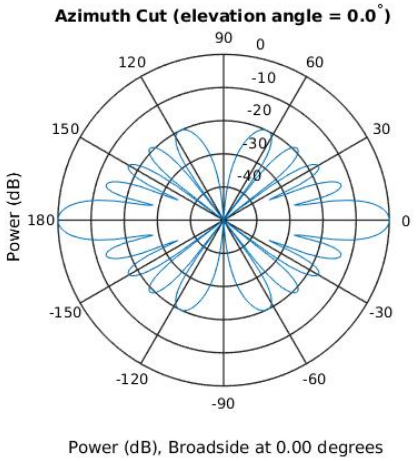


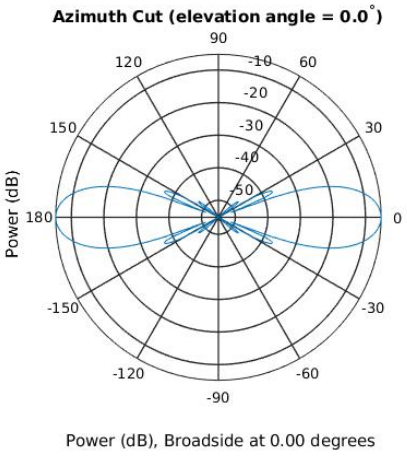
Figure 24: The window weights used to generate Figure 23

As expected, all window weights other than the rectangular window significantly increased the main lobe width

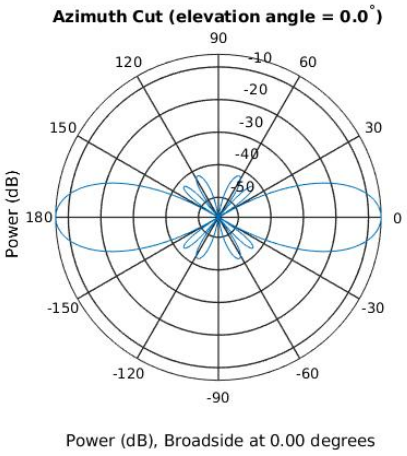
while nearly removing the sidelobes. For the $N = 4$ case, weights other than a rectangular window result in a nearly omnidirectional array which fails the design goal. This problem occurs only when there are few elements in the array. To demonstrate this, the simulation was carried out again with $N = 10$ using the same windowing schemes. The results are shown in Figure 25 with the corresponding weights shown in Figure 26.



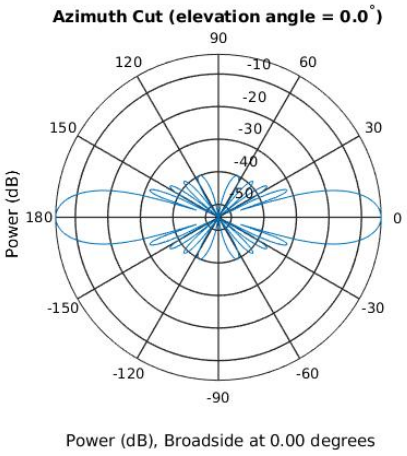
(a) Rectangular Window



(b) Hanning Window



(c) Hamming Window



(d) Kaiser Window

Figure 25: Beam pattern for various window weights for $N = 10$.

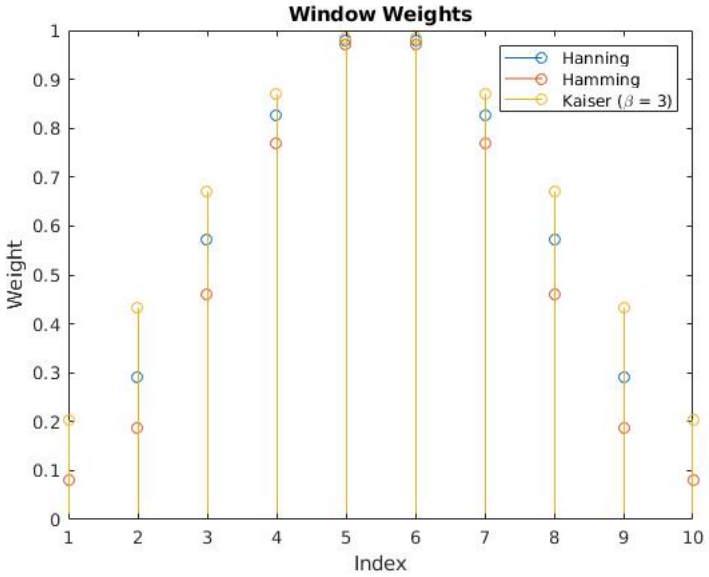


Figure 26: The window weights used to generate Figure 25

For larger N, windowing significantly reduces sidelobes resulting in most of the energy radiated or received to be in the direction of the main lobe. The increase in the main lobe width because of windowing is acceptable considering the improvement in the sidelobe power.

Since the project is constrained to $N = 4$ elements, the rectangular window was chosen because directionality of the main lobe is required to demonstrate beam steering. The energy radiated and received in the sidelobes was considered irrelevant in terms of this goal.

4.1.4 Simulation using Antenna Design

The final array design parameters were decided based on the results from the previous subsections. These parameters are listed in Table 2. This array design was simulated with the antenna designed in Section 3. The 3D and 2D antenna patterns are shown in Figures 27 and 28.

Table 2: Final array design parameters.

Parameter	Value
Carrier frequency f_c	920MHz
Number of elements N	4
Element spacing d	$\frac{\lambda}{2}$
Windowing weights	Uniform

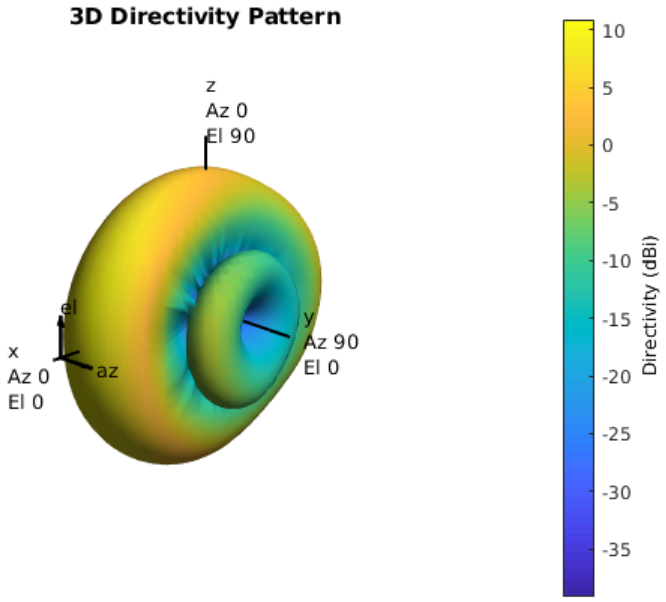


Figure 27: 3D antenna pattern for the final array and antenna design

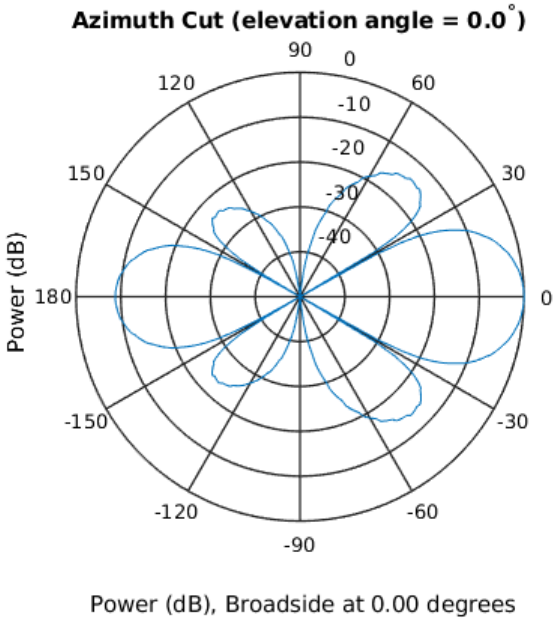


Figure 28: 2D antenna pattern for the final array and antenna design

Figures 27 and 28 indicate that the array and antenna design will function as required.

4.2 LMS Algorithm Simulation

The LMS algorithm described in Section 2.3 was simulated in MATLAB using the Phased Array toolbox. The simulation uses the function *collectPlaneWave* to modulate an input signal with a phase corresponding to the delay caused by the direction of arrival at the N element array. The output of the function is a N column matrix with each column corresponding to the received signal of an element in the array. For the purposes of this section, the output of the MATLAB function *collectPlaneWave*[10] will be referred to as the signal received by the array.

The parameters used in the simulation are listed in Table 3. Additive white Gaussian noise was added to the received signal in order to better approximate a real channel. The infinite bandwidth channel model was also used in the simulation since the calibration signal was narrowband. The Signal to Noise Ratio (SNR) at each element of the array was set to be approximately 15dB, an example of the spectrum of the baseband received signal is shown in Figure 29.

Table 3: LMS simulation parameters.

Parameter	Value
N	4
Carrier frequency f_c	920MHz
Calibration Signal	$0.1\cos(2\pi 10 \times 10^3 t)$
Signal length	1 second
Direction of arrival (azimuth)	-20°
Sampling frequency F_s	50kHz
Initial weight vector	$[0,0,0,0]^T$

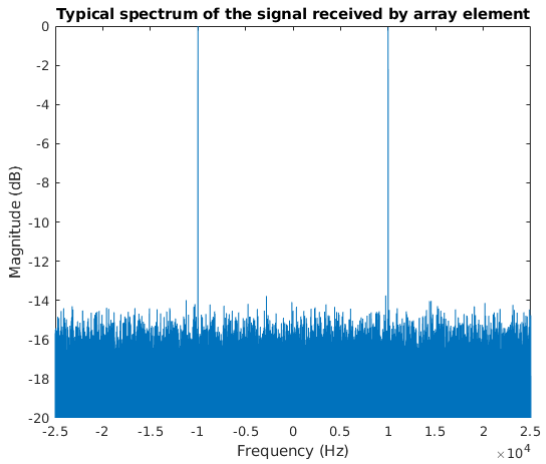


Figure 29: Typical spectrum of the baseband signal received by an element in the array.

The LMS algorithm was performed over the full length of the received signal for various step sizes. Figure 30 shows the mean squared error (MSE) of the entire estimated and actual calibration signal. It can be seen that a step size of 0.5 provided the best performance in the MSE sense. The system was found to be unstable for step sizes above 1000.

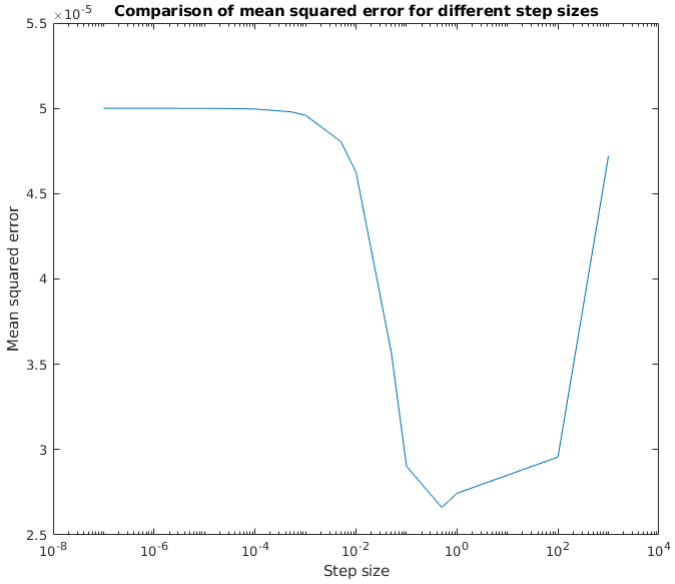


Figure 30: The mean squared error for different step sizes.

The beam pattern was calculated from the weights determined by the LMS algorithm in order to verify correct operation. As can be seen in Figure 31, the algorithm correctly detected the angle of arrival of the signal. Further evidence of the correct operation of the algorithm can be seen in Figure 32. The figure shows that the calibration signal was correctly recovered and that there was approximately a 2dB increase in the SNR.

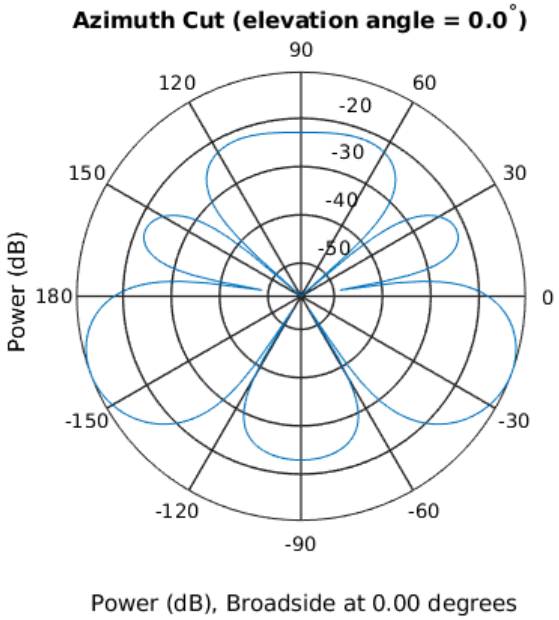


Figure 31: The beam pattern for a signal arriving from $-20°$ azimuth using a step size of 0.5.

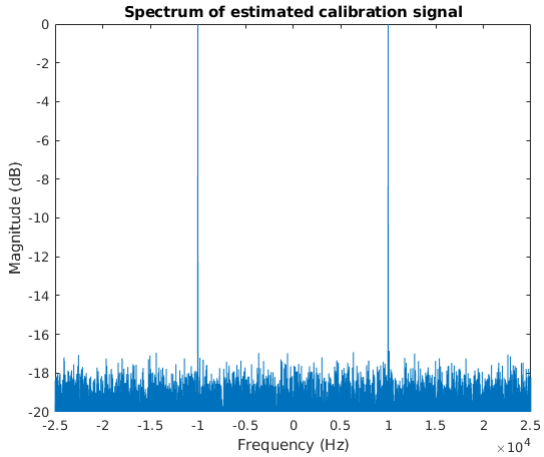
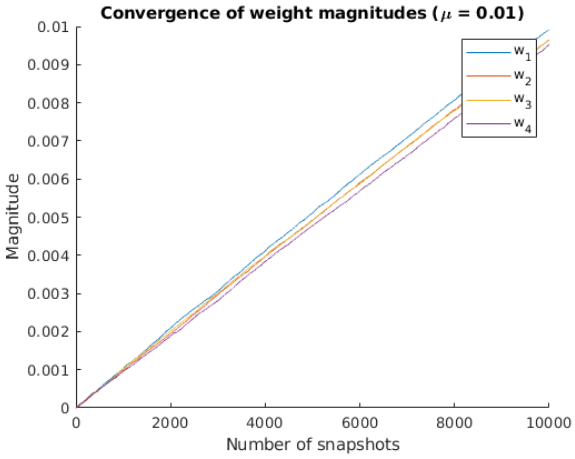


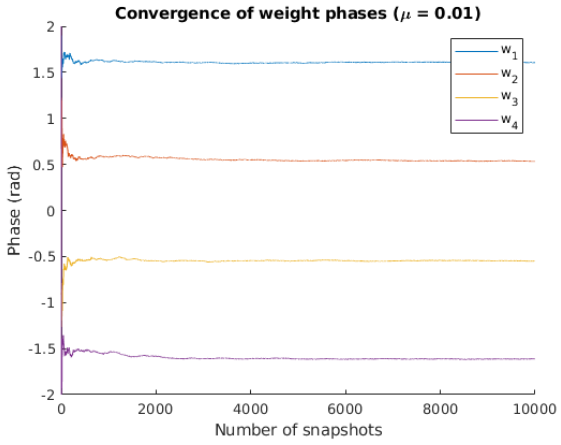
Figure 32: The spectrum of the estimated calibration signal.

Analysis of the convergence of the estimated weight vector was conducted with the results shown in Figure 33. what is apparent is that it takes a longer time for the magnitude component of the weight vector to converge when compared to the phase component. as can be see in Figure 33a, the magnitude of the weights does not converge after 10000 snapshots. this is in contrast to Figure 33b which shows that the phase converges after approximately 2000 snapshots. interestingly, the phase appears to converge within 2000 snapshots for figures 33d and 33f as well. a noticeable feature of Figures 33e and 33f is that there is a significant amount of noise in the steady state. this noise can be attributed to the large step size resulting in oscillation of the estimated weights around the true weight vector. Figure 33d shows that the magnitude of the weights converges at approximately 10000 snapshots.

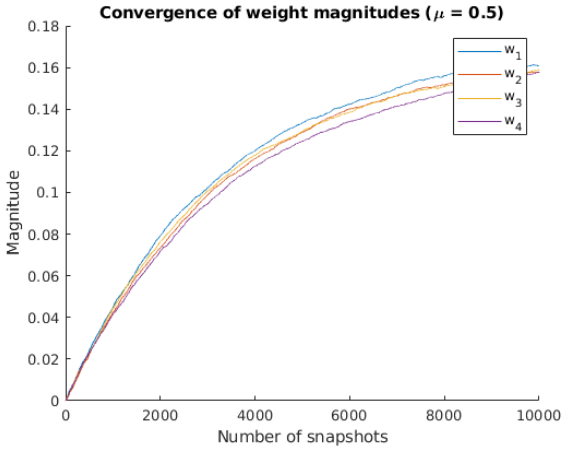
Considering Figure 33 in terms of the MSE corresponding to the step size, the step size of 0.5 has the lowest MSE because the weights have converged and are not oscillating about the true weight vector. This is in comparison to Figure 33a where the weights have not converged and Figure 33f where the weights are oscillating.



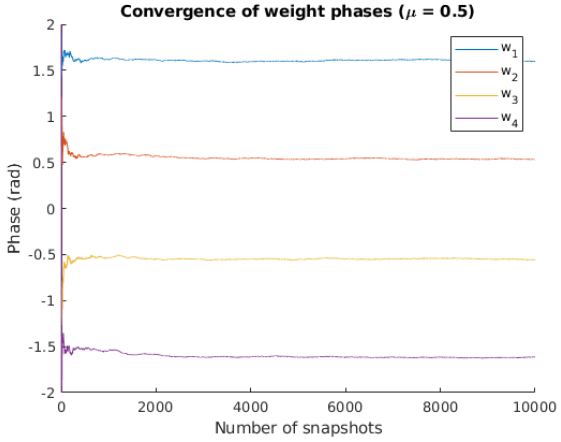
(a) $\mu = 0.01$



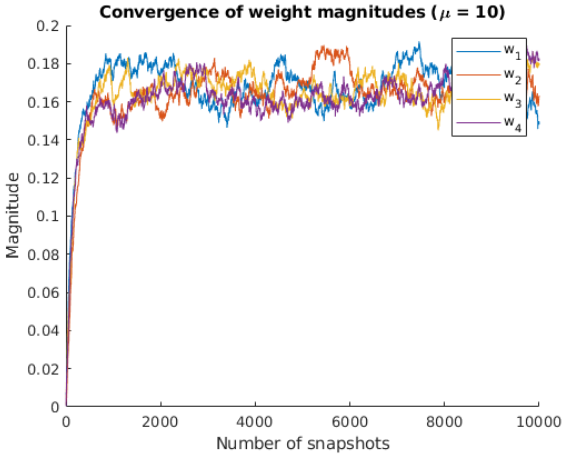
(b) $\mu = 0.01$



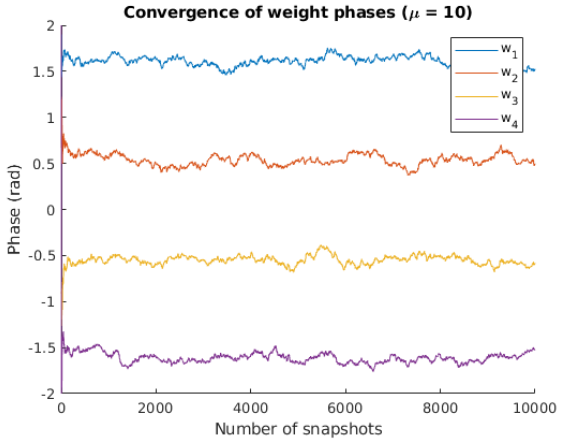
(c) $\mu = 0.5$



(d) $\mu = 0.5$



(e) $\mu = 10$



(f) $\mu = 10$

Figure 33: Convergence of the estimated weight vectors for different step sizes. 10000 snapshots corresponds to a signal length of one second.

The SNR of the received signal was varied in order to gain an understanding of the relationship between SNR and the beam pattern. It was found that at approximately 10dB the beam pattern began to distort with the side lobes slightly increasing in power. At an SNR of less than 5db there was 10dB more power in the sidelobes than in the main lobe. At an SNR of less than 2dB, the beam pattern was unrecognisable. Based on these results, it is expected that the algorithm can successfully implemented on the system outlined in Section 5 provided the received signal has a sufficient SNR.

4.3 Mutual Coupling

An important consideration in array design is the effect mutual coupling has on the radiation pattern of the array. Multiple mechanisms have been cited as sources of mutual coupling by Stutzman and Thiele [11] in the text *Antenna Theory and Design*, which most of the derivations in this section are sourced from. Examples of objects which may couple together are array elements, nearby objects, and the feed network for the elements of the array, as can be seen in Figure 34 [11].

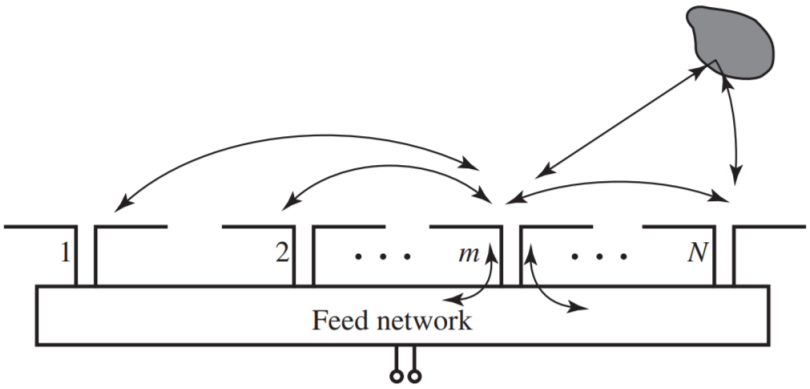


Figure 34: Sources of Mutual Coupling [11]

For this project, it is assumed that there are no nearby objects with which the array will couple as the plywood mounting structure used for the array has minimal reflective properties. In addition, the mutual coupling induced by the feed network is assumed to be minimal due to the coaxial shielding present on feed cables. The main source of coupling is assumed to be between neighbouring elements. This effect is similar to the coupling between the two plates of a capacitor, and can be modelled by Thevenin or Norton equivalent circuits as shown in Figure 35, where V_m^g is applied generator voltage and Z_m^g is terminal impedance.

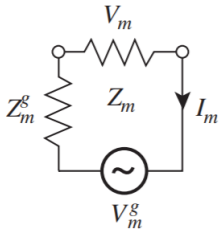


Figure 35: Thevenin Equivalent Circuit [11]

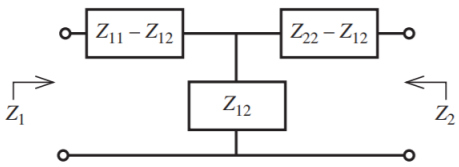


Figure 36: 2-Port Network Representation of 2 Element Array [11]

This model forms an N-port network as shown in Figure 36 for the case of a 2 element array. The impressed current and voltage in each element N can be expressed by:

$$V_N = Z_{1N}I_1 + Z_{2N}I_2 + ... + Z_{NN}I_N \tag{37}$$

where Z_{nn} is self impedance when all elements are at open circuit, and Z_{mn} is mutual impedance:

$$Z_{mn} = V_m/I_n \tag{38}$$

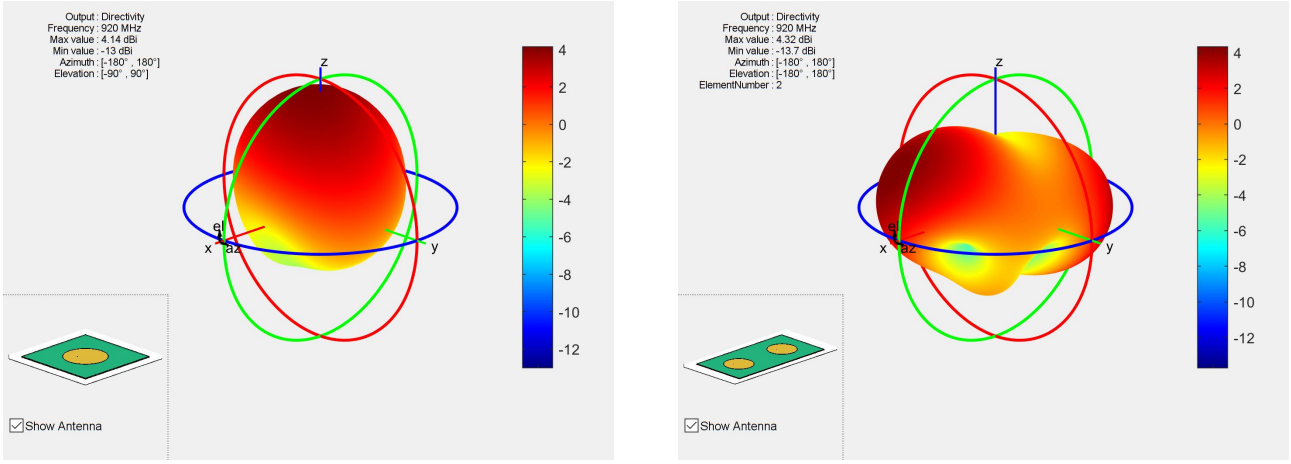
where $I_i = 0$ for all i except $i = n$. As such, the mutual impedance is difficult to compute or measure [11]. Simulation tools such as MATLAB, compute an estimate of this using the method of moments [12]. Better approximations have been suggested in [13].

Despite the fact that four simultaneous equations, for each terminated element in the array as well as the one excited element, need to be solved to find the mutual impedance relationship between the pair of elements in the array, for the simple case of a two element array it is possible to use the mathematical model to demonstrate the effect.

In order to do this, the self impedance of a single isolated element is written using this model as:

$$Z_{11} = V_1/I_1 \tag{39}$$

Due to the limitations of MATLAB to perform this kind of analysis on custom antennas (such as the antenna designed in section 3 which uses additional notches and slots)[6], this effect is shown by analysis of a standard circular patch antenna with the same dimensions and materials as the designed antenna. The radiation pattern of this isolated element can be seen in Figure 37a.



(a) Isolated Antenna

(b) N = 2 (Right Element Terminated)

Figure 37: Radiation Pattern of a Single Antenna and Terminated 2 Element Array

Stutzman and Thiele (8.7, p305)[11] have noted that "*If a second antenna is brought into proximity with the first, then radiation from the first antenna induces currents on the second, which in turn radiates and also influences the current on the first antenna.*". Using the derived model, this can be expressed as:

$$V_1 = Z_{11}I_1 + Z_{12}I_2 \tag{40}$$

$$V_2 = Z_{21}I_1 + Z_{22}I_2 \tag{41}$$

In the analysis each neighbouring antenna element is treated as a termination impedance of 50Ω (i.e. $Z_n^g = 50\Omega$), which is the input impedance of the SMA connector at the feed of the antenna, while the active element is excited. Therefore, since for the terminated element $V_2^g = 0$, equation 41 can be expressed as:

$$-Z_2^g I_2 = Z_{21}I_1 + Z_{22}I_2 \tag{42}$$

By solving this equation for I_2, and substituting it into equation 40, the first equation can be rewritten as:

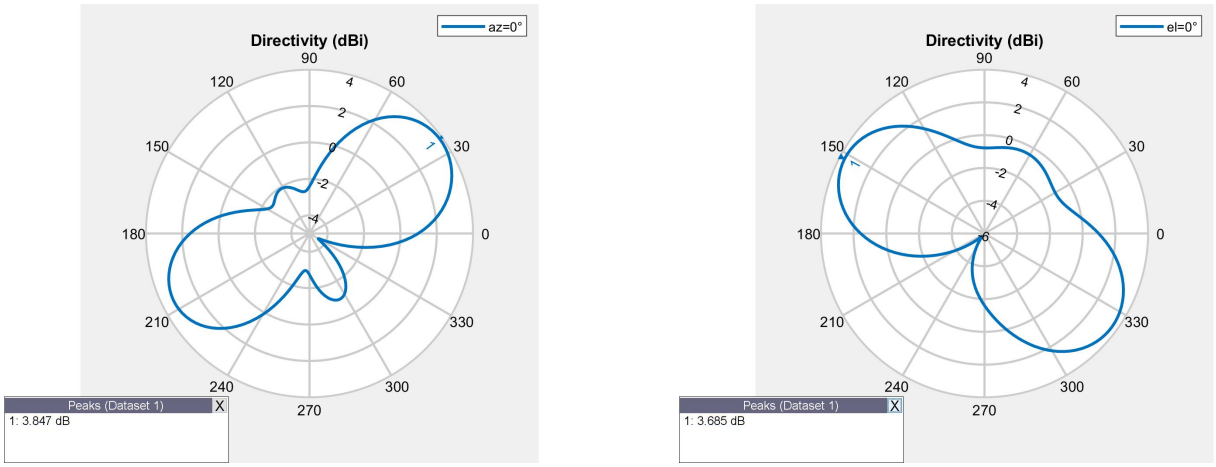
$$V_1/I_1 = Z_1 = Z_{11} - Z_{12}^2/(Z_{22} + Z_2^g) \tag{43}$$

[6]This is due to incompatibility of data types

In this special case, there is a drop in value of Z_1, as seen through the first element, assuming Z_{12} is real and positive. However, it should be noted that the sign of both real and imaginary part of Z_{12} are a function of spacing and can be predicted using method of moments. Since the voltage of the excited element V_1 is set to a reference value (1V in MATLAB simulations) an increase in current and therefore, directivity is expected [14]. The MATLAB simulation for this case of a 2 element array in which one element is excited whilst the other is terminated can be seen in Figure 37b.

As it can be seen in Figure 37a, the maximum value of the directivity of the antenna element is 4.14 dBi. However, due to the effect of mutual coupling of the second unexcited antenna with spacing of $\lambda/2$, the maximum directivity has now increased to 4.32 dBi in Figure 37b.

The envelope graphs of the two element terminated array (2D slices of the 3D pattern), can be seen in Figure 38.



(a) E-Plane Azimuth = 0°

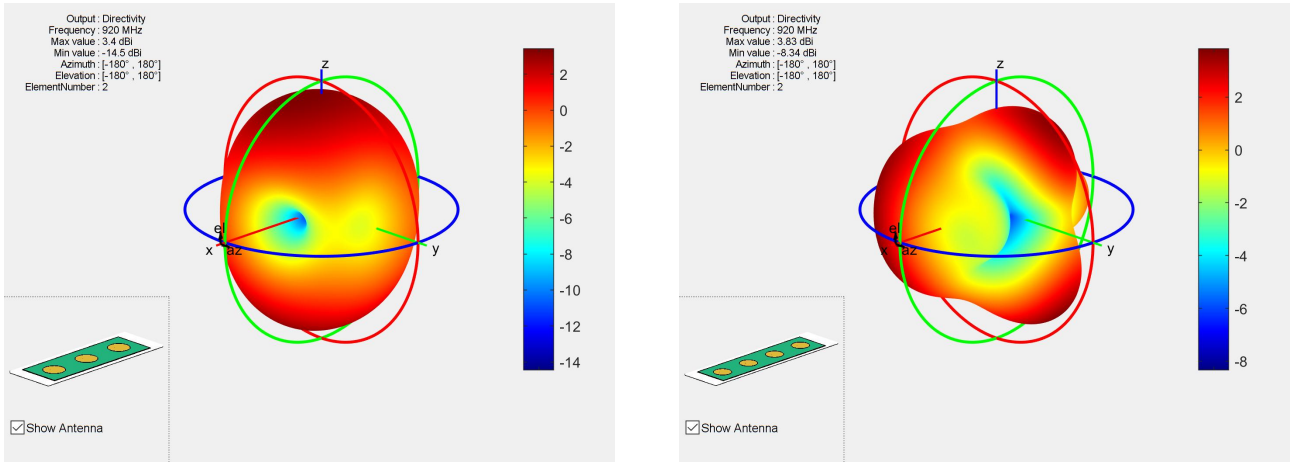
(b) H-Plane Elevation = 0°

Figure 38: E-Plane and H-Plane Array Directivity N = 2

As more elements are added to the array whilst the centre element is kept active, the effect of mutual coupling becomes more unpredictable. This is due the equation expressing Z_n of the active element, which is taken to be the second element in subsequent simulations, being a factor of multiplications of mutual impedances of the different elements in the array. Since each of these mutual impedance values can have positive or negative values dependent on their phase, and combine together in higher order terms, the directivity of the terminated array can decrease or increase.

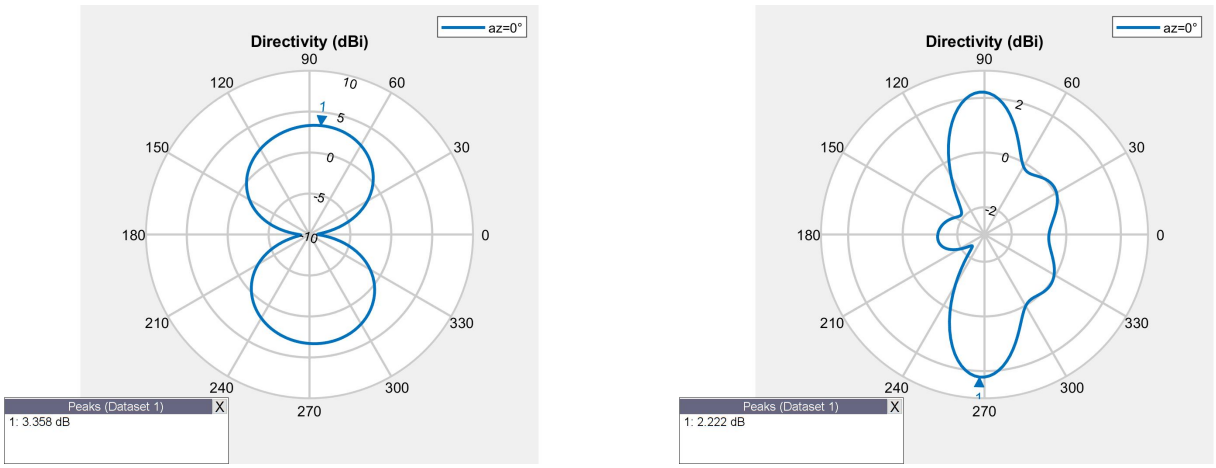
In the case of three elements as can be seen in Figure 39a, the maximum directivity has decreased significantly to 3.4 dBi. However, in the case of four element array in Figure 39b, the maximum value at peak is now at 3.83 dBi, demonstrating the unpredictability of coupling effects. Regardless, the effect on directivity of individual elements from coupling is small enough that it is not considered in the project.

2D plots of the radiation patterns in E-plane, which is the direcion of transmission, can be seen in Figure 40a and Figure 40b.



(a) N = 3 (b) N = 4

Figure 39: Radiation Pattern of terminated 3 and 4 Element Arrays

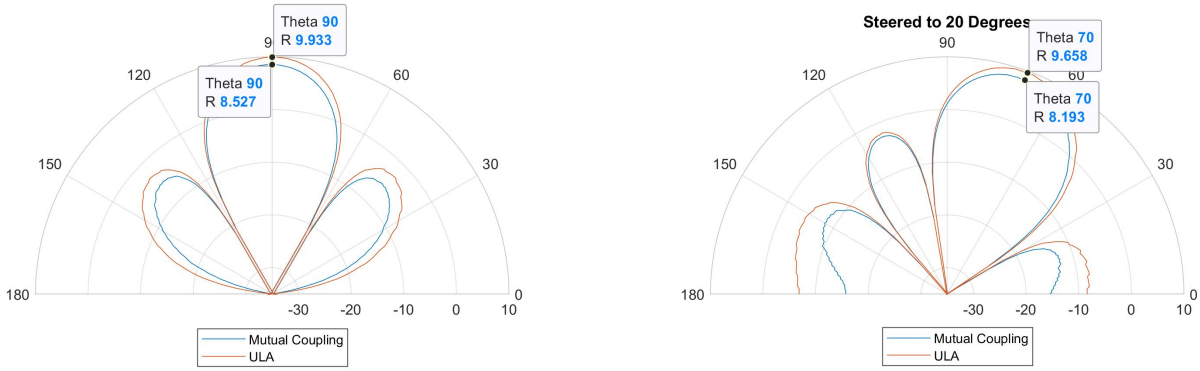


(a) N = 3 E-Plane (Azimuth = 0°) (b) N = 4 E-Plane (Azimuth = 0°)

Figure 40: E-Plane Array Directivity Comparison

It should be noted that as the equation for input impedance Z_n shows, the input impedance is dependent not only on mutual impedances, but also on the phase of the current of each element. The practical significance of this has been noted by Stutzman and Thiele(8.7, p308)[11]: "*The input impedance is affected by the phase of the terminal currents as it is adjusted to phase scan the pattern. Thus, the input impedance of all elements in an array will change with scan angle*". Therefore, as the array is steered towards different directions, mutual coupling can play a changing role in the directivity of the pattern.

To demonstrate this effect, the radiation pattern of the 3 element array (side elements terminated, center excited) in Figure 39a was used to create a custom antenna element representing an element affected by coupling from the two neighbouring elements. This pattern only gives a rough approximation of the actual effect due to mutual coupling, as it does not consider elements beyond the immediate neighbours and ignores edge effects. This custom antenna was then used to form a 4 element array. The radiation pattern of this array can be seen in Figure 41a and in Figure 41b where it has been steered to 20°, along with the radiation pattern of a uniform linear array without considering the effect of mutual coupling.



(a) Steering Angle = 0°

(b) Steering Angle = 20°

Figure 41: Radiation Pattern Comparison With and Without Mutual Coupling

As it can be seen in the unsteered case of Figure 41a, the peak directivity of the array has slightly decreased from 9.933 to 8.527 dBi. When the array is steered to 20° as shown in Figure 41b, while slight decrease in directivity even without mutual coupling should be expected, a larger decrease should be expected when mutual coupling effects are modelled. In this case, while the ULA without mutual coupling has lost 0.275 dBi as it is steered, the array with the terminated antennas as its elements shows a 0.334 dBi loss in directivity.

Considering the results from these simulations, it has been deemed that the effect of mutual coupling on the system should be minimal. However, if quantification of mutual coupling is of interest, another approach besides the mutual impedance approach is the scattering matrix. This approach is more popular due to the fact that Vector Network Analysers (VNA) are capable of measuring the s-parameter matrix. The mathematical relationship between the entries of the scattering matrix and mutual impedances has been described in Stutzman and Thiele(8.7.1, 307-308)[11].

Another phenomenon of interest is the edge effect within an array. This can be seen in Figure 42 [11].

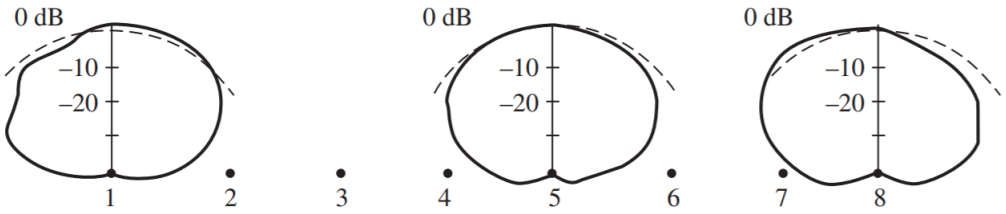


Figure 42: Radiation pattern of elements within an array, demonstrating the edge effect [11]

As the number of elements in the array increase to infinity, the radiation pattern of the centre elements becomes symmetric and almost identical. As shown in Figure 39a and 40a, the radiation pattern of the center active element in a 3 element terminated array shows symmetric properties. In the case of the 4 antenna terminated array of Figure 39b, due to the active element lying closer to one edge of the array than the other, these symmetric properties are not present. These properties will also be present in the third element, in an equal and opposite manner.

This asymmetric effect is seen further in Figure 43 where the excited element is the fourth element.

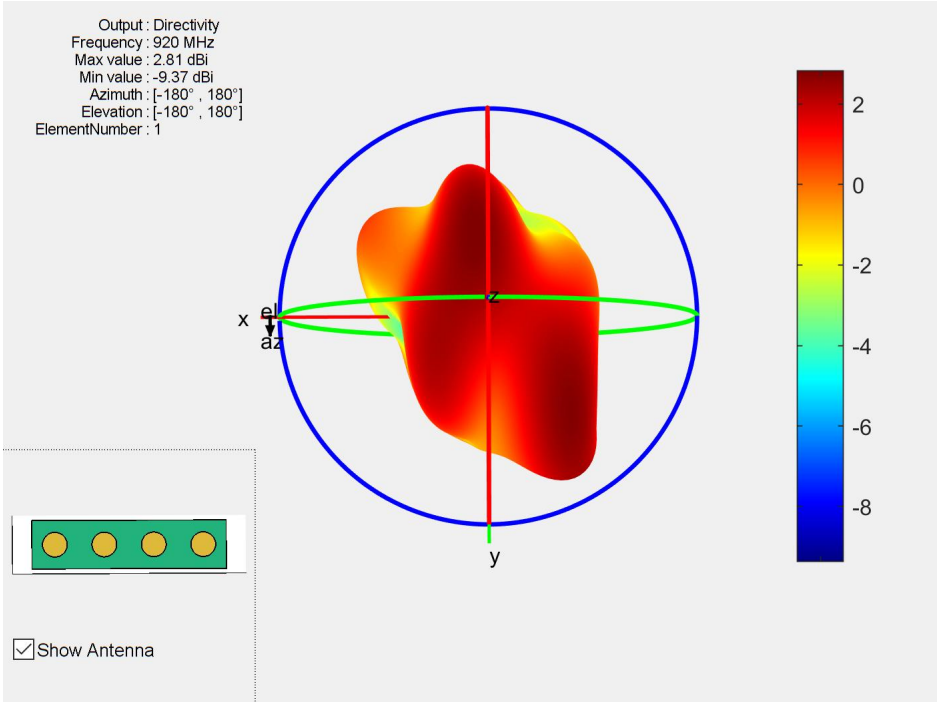


Figure 43: N = 4 Radiation Pattern with the fourth (right edge) element excited

Ultimately, the analysis in this section shows that although mutual coupling typically negatively affects the array performance by introducing unpredictable behaviour and by typically reducing element directivity, the effects on the system designed for the project should be minimal and can be largely ignored.

5 System Implementation & Integration

The following parameters of a phase-shift beamforming system are defined as such:

- Phase resolution - The minimum step in which phase can be changed
- Phase variance/jitter - The maximum variance in phase when input signals are constant
- Phase accuracy - The difference in actual phase from expected phase of the I/Q vector
- Gain variance - The variance in gain when input signals are constant
- Relative Gain Accuracy - The difference in end-end gain between each chain with the same relative phase

The beamforming system designed for this project is capable of meeting the specifications specified in the phase-shift Section 6 and receive beamforming Section 7 sections. The specifications achieved are as follows:

- Transmitter, and receiver all capable of operating within the frequency range 800-1000MHz
- Maximum transmit power of more than 5dBm
- Transmit Chain
  - Phase shift angle resolution of approximately 0.1°
  - Phase shift angle jitter of less than $\pm 0.5^\circ$
  - Phase shift accuracy of $\pm 0.2^\circ$ to intended when calibrated, $\pm 0.5^\circ$ when uncalibrated
  - Gain of individual transmit chains able to be controlled within ± 0.25dB of desired power
- Receive Chain
  - Effective array sample rate of 4.28kHz (8 channels)
  - Data transfer rate of 921600 baud to PC for baseband processing

5.1 Logical Overview

As previously described in Section 1, two systems are implemented in this project: a receiver and transmitter array.

For the purposes of this project, the transmit array is used to demonstrate analog (phase shift) beamforming, and the receive array is used for adaptive digital beamforming using the LMS algorithm.

The basic functions of a transmit chain include synthesizing the desired baseband signal, modulating it onto a carrier frequency, and then allowing for phase and gain adjustment (the main logical function required in analog beamforming) at the output of each array element before passing the signal onto the antenna.

A receive chain acts as the functional reverse of the transmit chain. It contains a pre-demodulation (low-noise) amplifier, an IQ demodulator, any required filtering of the demodulated signal (for anti-aliasing purposes), and a means of sampling the demodulated baseband signal.

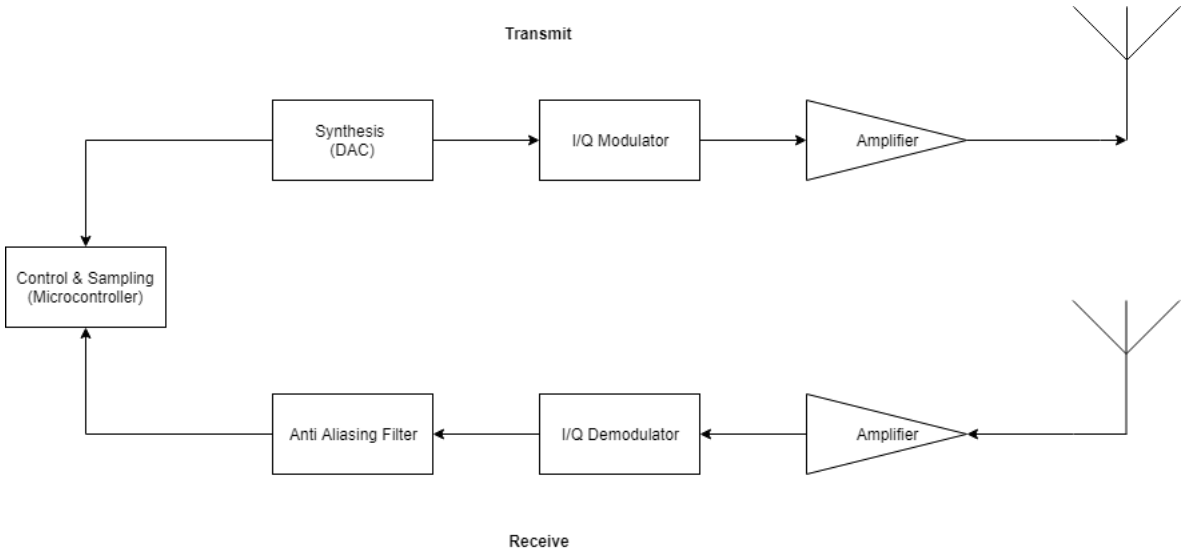


Figure 44: Logical flow diagram of both beamforming systems connected to a shared control module

Lastly, the transmit and receive chains both require to be interfaced to a control module that can coordinate the operation of multiple chains to achieve beamforming.

5.2 Physical Overview

The physical realisation of the previously described beamforming system consists of a series of chained PCBs, each containing a logical module which forms part of either a receive or transmit chain.

Only one of the two halves (transmit or receive) is used at any one time. As such, only one set of four antennas has been assembled and mounted, with the antennas being exclusively connected to either the receive chains or transmit chains at a given time. All the electronics are mounted on the same physical structure, and coaxial cables can be used to connect/disconnect an antenna from any chain as required.

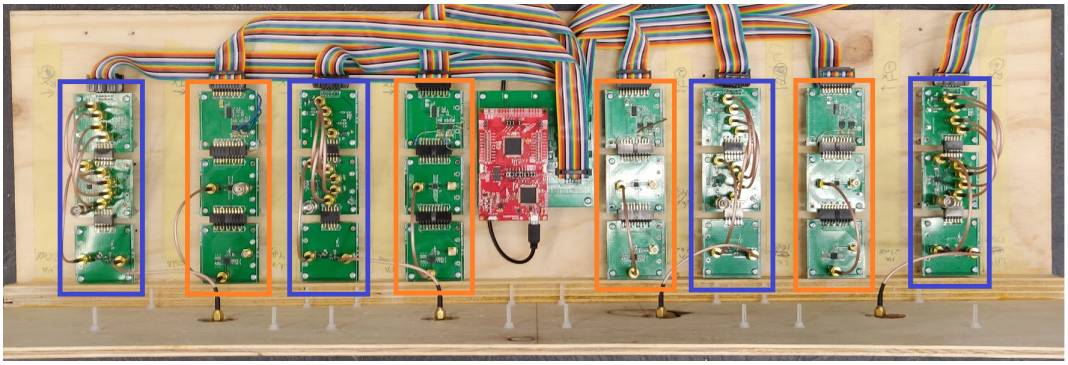


Figure 45: Top-down view of the full system mounted on a plywood structure. Connections to the antennae can be seen at the bottom of the photograph, the orange and blue outlines show the transmit and receive chains respectively, the control module can be seen in the center

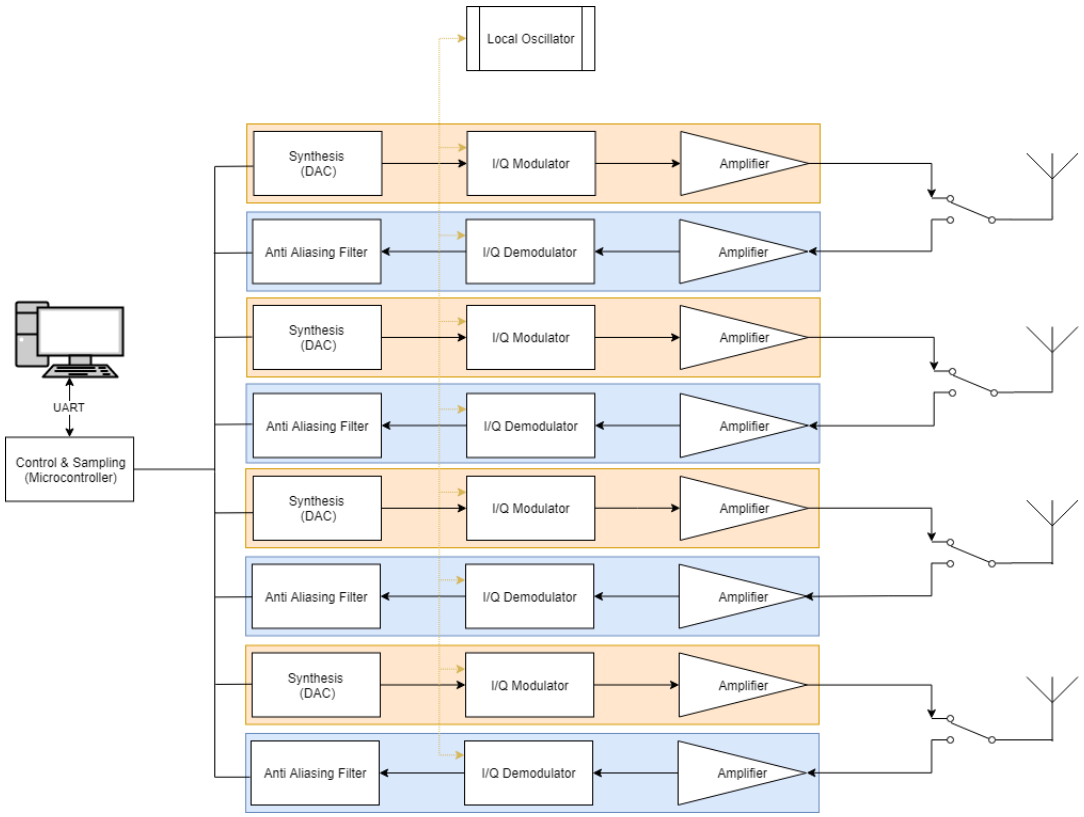


Figure 46: Full system diagram, the orange and blue rectangles show the transmit and receive chains respectively

All modules and antennae are mounted on a right angle plywood structure (Figure 46) which allows the antenna to transmit horizontally when the structure sits upright on an elevated surface. Plastic screws are used near sensitive components which addition to the wooden mounting structure minimise sources of coupling. The purpose of sitting the structure on an elevated surface is to minimize any RF signal reflection from the ground which may introduce error in experimental results.

Both the transmit and receive systems share a single set of antennae, as the systems are never used simultaneously. This simply involves moving the coaxial cable from each antenna antenna to the relevant chain when needed.

Each module is a 50mm x 50mm printed circuit board designed in a modular fashion, allowing each logical function to be evaluated in isolation, whilst allowing for the final implementation to require minimal additional wiring. In addition, this specific size of board allowed for cheap costs due to easy panelization during manufacturing. Each module "chains" onto the next with a right-angled header, passing through any low frequency signals required for operation.

The control module consists of a TI MSP432P401R Microcontroller Launchpad development board[15] mounted on a custom bus board which interfaces via IDC ribbon cables to the controllable modules of the transmit/receive chains. This microcontroller is used to perform tasks such as sampling the received signal, synthesizing the baseband signal, and controlling the phase/gain of the transmit array using SPI. The microcontroller is also responsible for moving sampled data to a PC connected via UART whilst satisfying the real-time constraints of the system.

Additionally, an external RF signal generator is used to produce a local oscillator signal which is split (using simple BNC T-Junction adapters) and fed to the modulators and demodulators.

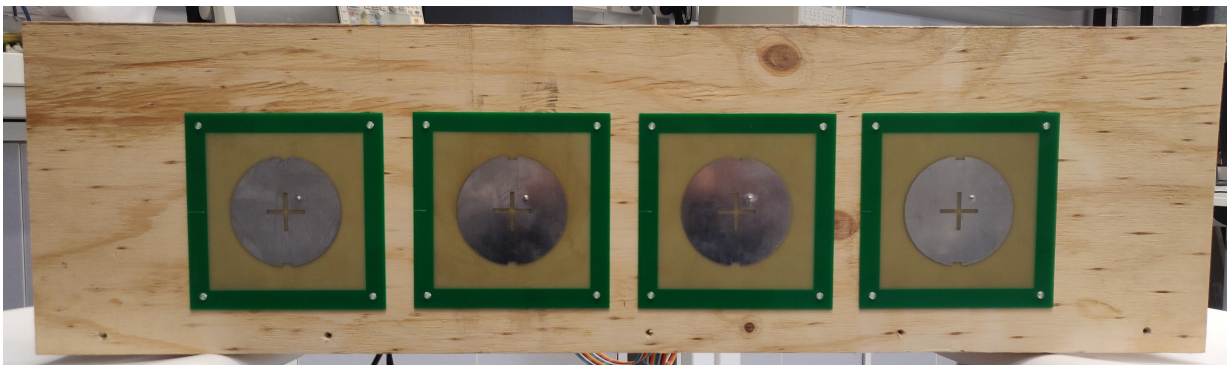


Figure 47: Front (broadside) view of the four antennae

5.2.1 Printed Circuit Board Design & Layout

Ideally the boards would use a 4 layer configuration for the advantageous RF characteristics of internal ground and power planes. Internal ground/power planes simplify routing and reduce the amount of inductive loops created by ground and power traces.

Another significant advantage of a 4 layer board is that due to the nature of the layer stackup, the top/bottom copper layers sit much closer to internal AC ground planes than in a two layer board.

High frequency signals are routed via microstrip lines, where the distance between the microstrip and any surrounding adjacent ground pour is greater than twice the width of the trace. In the case where surrounding AC ground is closer than twice the width of the trace, the contribution of parasitic capacitance to this ground must be included in the impedance calculation and the line can be considered a coplanar waveguide [16], [17].

The equation for microstrip impedance is as follows[18]:

$$\frac{W}{H} < 1$$

$$\epsilon_{eff} = \frac{\epsilon_r + 1}{2} + \frac{\epsilon_r - 1}{2}\left[\frac{1}{\sqrt{1 + 12\frac{H}{W}}} + 0.04\Big(1 - \frac{W}{H}\Big)^2\right] \tag{44}$$

$$Z_0 = \frac{60}{\epsilon_{eff}} \ln\big(\frac{8H}{W} + \frac{W}{4H}\big) \tag{45}$$

Therefore, a trace of 0.035mm thickness and 13mil width on a typical 4 layer board with 0.2mm between the top layer and inner ground plane (dieletric constant of 4.6) will have an impedance of 50.9Ω, whereas the same specifications on a two layer board with 0.6mm of dieletric between the top and bottom layers will yield an impedance of 86.8Ω. This impedance can be reduced by increasing the trace width to around 40mil, but in many cases this is not feasible due to board space constraints.

Given that one can assume a transmission line characteristics when a given trace is longer than a quarter of the signal wavelength, and that the wavelength of a 800-1000MHz signal is approximately 30cm, 50Ω impedance matching on intra-board connections is not strictly necessary, but still advantageous.

As such, for the purposes of minimising costs a 2 layer board with 0.6mm thickness[7] has been used in the final design.

Layers	2
Dieletric Constant	4.5
Substrate Thickness	0.6mm
Copper Thickness	0.035mm

[7] the minimum thickness at standard manufacturing cost

5.3 Transmit Chain

5.3.1 Design

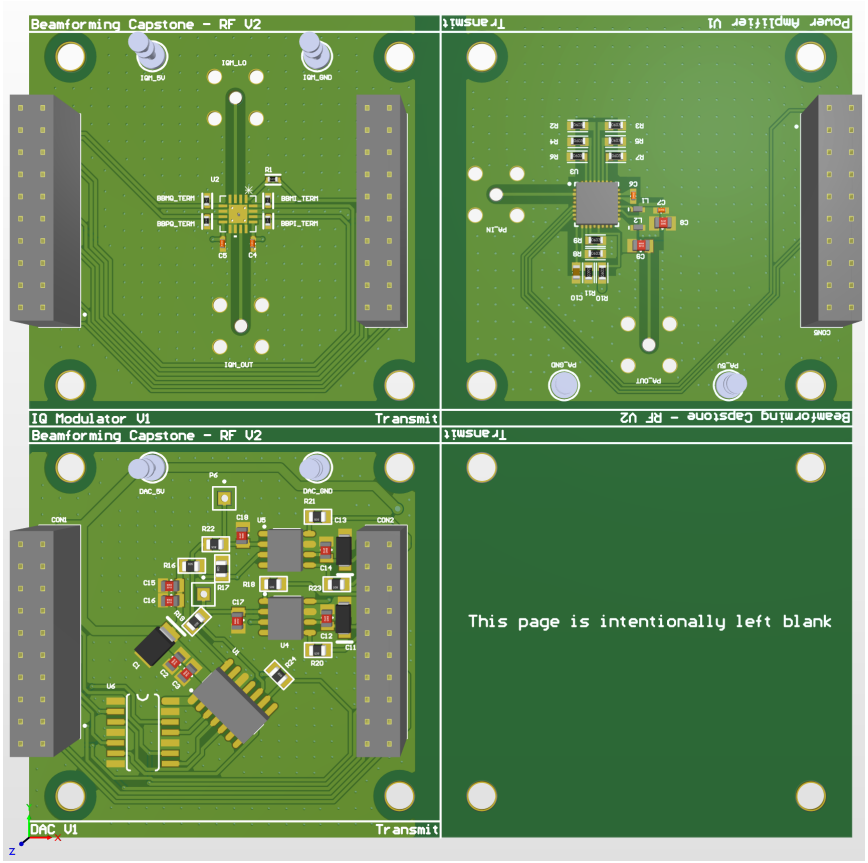


Figure 48: 3D Render of the transmit modules on a single panel

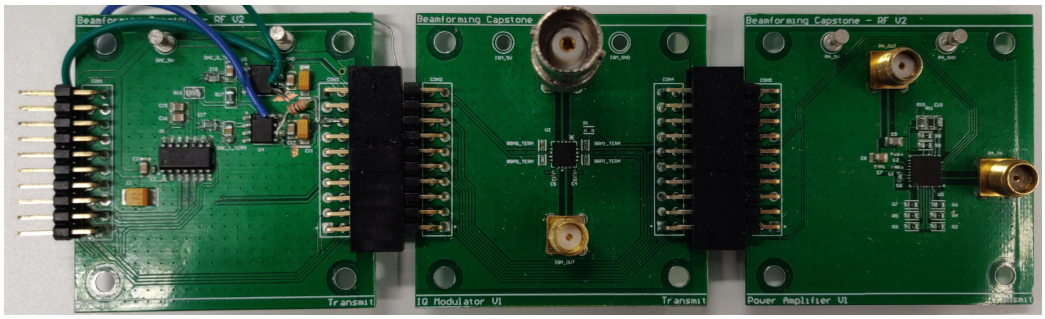


Figure 49: Photograph of a transmit chain, from left to right: DAC/Synthesis, I/Q Modulator, Power Amplifier

The transmit chain consists of three modules

- Digital-to-Analog Converter (DAC) (Synthesizes the baseband signal)
- IQ Modulator (Performs the modulation onto a carrier and any phase shifting)
- Variable Gain Power Amplifier (Controls the gain of the transmitted signal)

See appendix for circuit schematics and board design files of the modules described below

5.3.1.1 Synthesis - MCP4912/EL5171ISZ

The MCP4912 is a 10-Bit Voltage Output DAC with two channel latched output[19]. This allows it to synthesize both the I and Q baseband inputs to the IQ modulator simultaneously with no phase delay. This is an important

requirement as phase shifts between the I and Q baseband inputs can cause undesired behaviour after modulation (0.5V common mode)

With 10 bits of accuracy referenced to a 5V voltage rail, the effective resolution of the DAC is 4.88mV. In order to synthesize a sine wave at 10kHz, the outputs of the DAC are further filtered with a coupling capacitor to smooth the baseband inputs.

In order to convert the single-ended output of DAC to the required differential input of the next stage, an the EL5171ISZ op-amp[20] is used after an anti-imaging RC filter. Through experimental results, it has been determined that the usable differential output of the op-amp with only a single-ended supply voltage (5V) is from 0-3.5V when provided with a 2.5V external common mode reference.

Using 200Ω output resistors in series with the 50Ω input termination resistors of the next stage, the differential output of the amplifier is attenuated to 0.5V common mode, which serves the dual purpose of increasing the DAC resolution by a factor of 5, and also reducing the voltages down to within the acceptable input range of the modulator.

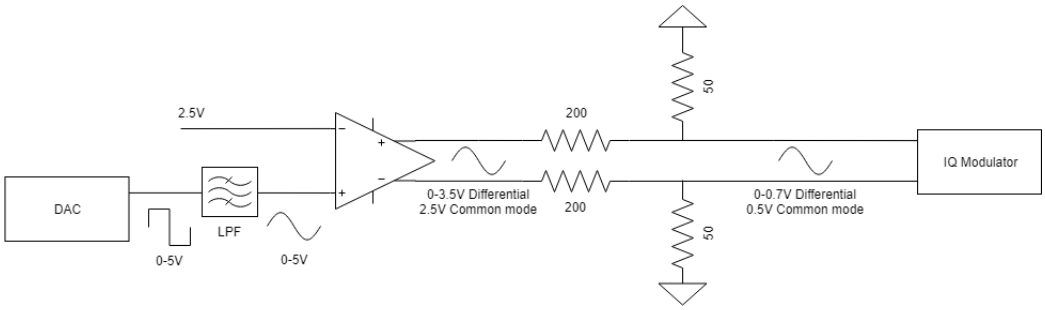


Figure 50: Signal conditioning circuitry for the synthesized baseband waveform

The module was tested by sweeping the DAC I/Q channels across 0-3.5V, measuring the output differential of the module and confirming that it had the expected 0.5V common mode voltage and appropriate differential voltage.

Additionally, control code and functionality was developed to drive a sinusoid at the output of the synthesis module. Experimentally, a frequency of up to 10kHz was able to be obtained, which was limited by the SPI communications with the DAC. The sinusoid is additionally smoothed by an RC filter at the input of the amplifier circuitry, which assists in achieving higher frequencies.

5.3.1.2 Phase Shifter (I/Q Modulator) - LT5571

I/Q Modulation

I/Q Modulation is an important concept used in the project to perform arbitrary phase shifts in the implementation of a phase-shift beamformer.

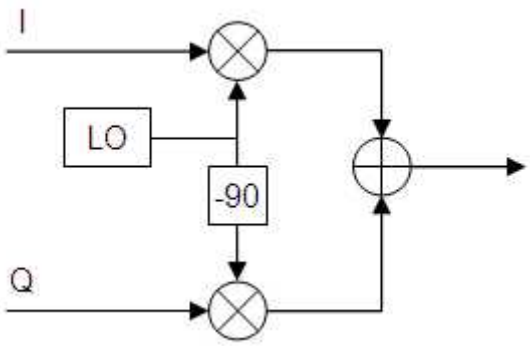


Figure 51: I/Q Modulator Block Diagram - A local oscillator is mixed with I and Q signals, after being shifted 0° and 90° respectively. The final output is the summation of the two mixed signals

Given a modulation scheme described in Figure 51, we observe two baseband input signals $I(t)$ and $Q(t)$. In the case that both inputs are sinusoidal $I(t) = \cos(\phi(t))$ and $Q(t) = \sin(\phi(t))$, the output $Y(t)$ of the system with a carrier frequency of f_c can be described as

$$Y(t) = \sin(2\pi f_c(t) + \phi(t)) = \sin(2\pi f_c t)I(t) + \sin(2\pi f_c t + \frac{\pi}{2})Q(t) \tag{46}$$

$$= \sin(2\pi f_c(t) + \phi(t)) = \sin(2\pi f_c t)\cos(\phi(t)) + \sin(2\pi f_c t + \frac{\pi}{2})\sin(\phi(t)) \tag{47}$$

Given the trigonometric identity

$$\sin(\alpha + \beta) = \sin(\alpha) + \cos(\beta) + \cos(\alpha) + \sin(\beta) \tag{48}$$

$Y(t)$ can be expressed as the following

$$Y(t) = \sin(2\pi f_c t + \phi(t)) = \sin(2\pi f_c t)\cos(\phi(t)) + \cos(2\pi f_c t)\sin(\phi(t)) \tag{49}$$

By removing time variance from $\phi(t)$, it can be seen that a phase shift can be applied to the carrier frequency by simply manipulating the signals $I(t), Q(t)$ as seen in Figure 52.

$$Y(t) = \sin\left(2\pi f_c t + \arctan\left(\frac{Q(t)}{I(t)}\right)\right) \tag{50}$$

This relationship is used in order to arbitrarily phase shift a carrier wave by controlling DC input voltages into an I/Q modulator.

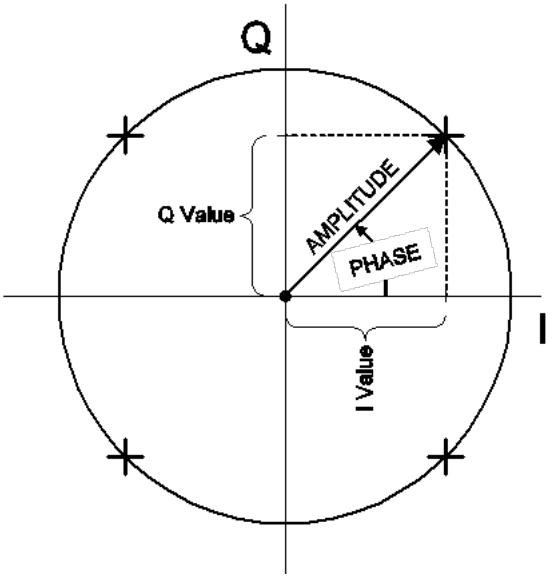


Figure 52: A visual demonstration of the I/Q vector forming a phase angle on a unit circle

LT5571

The LT5571 is a direct I/Q modulator from baseband frequencies to RF (800MHz - 1.5GHz) This reduces the number of stages required and any need for IF (intermediate-frequency) conversion [21].

Internally, the I/Q modulator consists of I and Q input differential voltage-to-current converters feeding into I and Q up-conversion mixers which use a local oscillator (externally supplied) phase generator. The output is fed into a balun which provides a 50Ω matched single ended RF output.

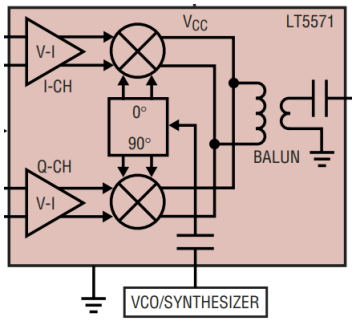


Figure 53: LT5571 IQ Modulator Block Diagram

The module was tested by providing a 920MHz carrier wave to the powered module, using a function generator to supply either the I or Q input with a low frequency sine wave while keeping the other at a constant DC voltage, and observing a DSB-SC signal centered at the carrier using a spectrum analyzer.

5.3.1.3 Power Amplifier - DVGA1-242A+

The DVGA1-242A+ is a digital variable gain RF power amplifier that allows up to 30dB of gain in 0.5dB steps for 450-2400MHz. It is controlled through a serial interface to the control module.[22]

The serial interface includes a latch-enable input pin which allows for precision coordination of gain changes simultaneously across multiple transmit chains.

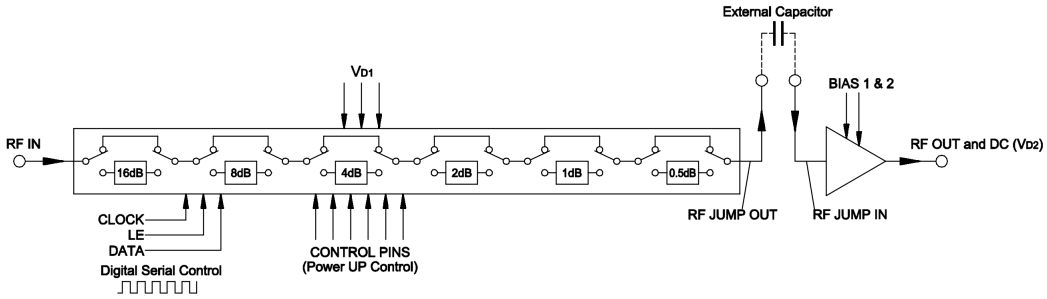


Figure 54: DVGA-242A+ Block Diagram

The module is tested by providing a 920MHz carrier wave input, varying the gain across the 31.5dB of range and observe changing signal magnitude in correct increments.

5.3.2 Evaluation

After the individual modules are verified, the entire chain is tested together for its ability to phase and gain shift, which will provide the essential functionality for phase-shift beamforming.

In order to test for the phase characteristics, 36 constant magnitude I/Q vectors corresponding to 360° of a circle in 10° increments are synthesized and input into the I/Q modulator module. A vector network analyzer is used to measure the phase difference between the local oscillator input to the I/Q modulator and the chain output of the power amplifier at the carrier frequency.

The magnitude of the I/Q vector was chosen so that it maximised usage of the synthesis module range, this range is calculated to be 1.5-3.5V output from the DAC, translating to a 0.3-0.7V input to the I/Q modulator (See 4.3.1 for details on signal conditioning circuitry).

Full results can be found in the appendix, a summary is provided below:

- An expected 10° phase shift typically results in a phase shift between 9.5°and 10.5°
- Phase jitter is measured to be approximately ±0.5° at most

- Gain appears to be mostly independent of phase, there was an observed 1dB drift in gain across the experiment, but the cause may not necessarily be attributed to change in phase
- All 360 degrees of phase are successfully able to be synthesized with the hardware
- The hardware is capable of generating phase shifts of less than 1°, with the limiting factor being phase jitter as opposed to DAC resolution.
- Due to varying cable lengths and assembly tolerances, each chain will need to be individually calibrated to find the required I/Q vector for 0°phase.
  - After calibrating for the 0° point the remaining phases can be synthesized to within ±5° accuracy by simply calculating the theoretical vector relative to the calibrated point
  - Manual calibration for each desired phase yields best results with accuracy equivalent to the phase jitter at ±0.5°
- Approximately 10dB of maximum gain difference was measured between the chain with the most gain, and the chain with the least gain. This difference was compensated using the variable gain power amplifier such that each chain had identical gain of 0 ± 0.25dB from the input local oscillator to output
  - A one-way relationship between gain and phase was observed, when the gain is varied through the power amplifier, the relative phase of the entire system changes. However, changing the phase does not affect the gain.
  - As such, the gain of each chain was first calibrated to 0dB before then calibrating for the 0° phase points.

5.3.3 Beamsteering Calibration

Despite verifying that each chain is able to shift 360° of phase to a sufficient resolution and accuracy compared to the theoretical I/Q vector angle, the best performance was observed when desired phase shifts were directly calibrated for their corresponding I/Q voltages driven from the synthesis board. In doing so, phases within ±0.25° of the desired angle could be achieved consistently, along with gain within ±0.25dB. The limiting factor in achieving better gain and phase accuracy was in both cases the jitter, or noise inherent in the system, as opposed to the actual resolution of the DAC.

Therefore, in order to achieve the best setup for phase shift beamforming, the required phase shifts for each array element were determined respective to 0°, and the corresponding voltages were then found through a manual calibration process.

Calibrations for steering angles of 0°, 10°, 20°, and 30° were found for each chain in the 4 element transmit array. Each steering angle requires an I and Q voltage along with a power amplifier gain setting, resulting in 3 calibration points for each chain, for each steering angle.

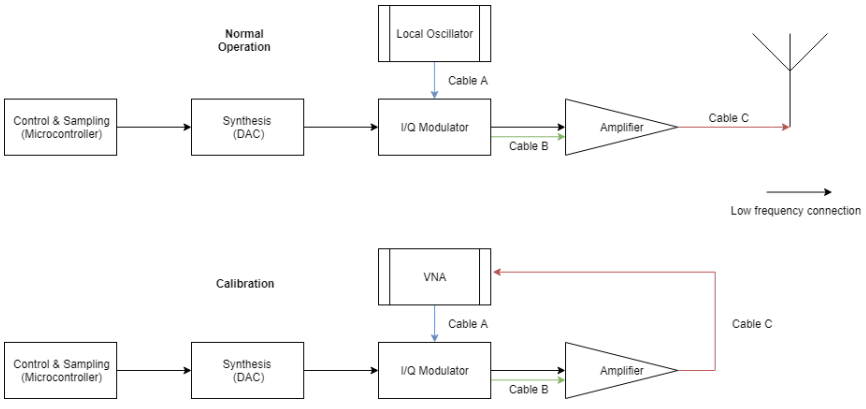


Figure 55: Diagram of VNA insertion points into system for calibration

Calibration was performed over as much of each chain as possible in order to be accurate. Only the local oscillator and antenna were removed from the calibration with all cables used identical to that of in actual operation to minimise deviation in phase between the calibration and operational setup.

Full calibration data can be found in the appendix.

5.4 Receive Chain

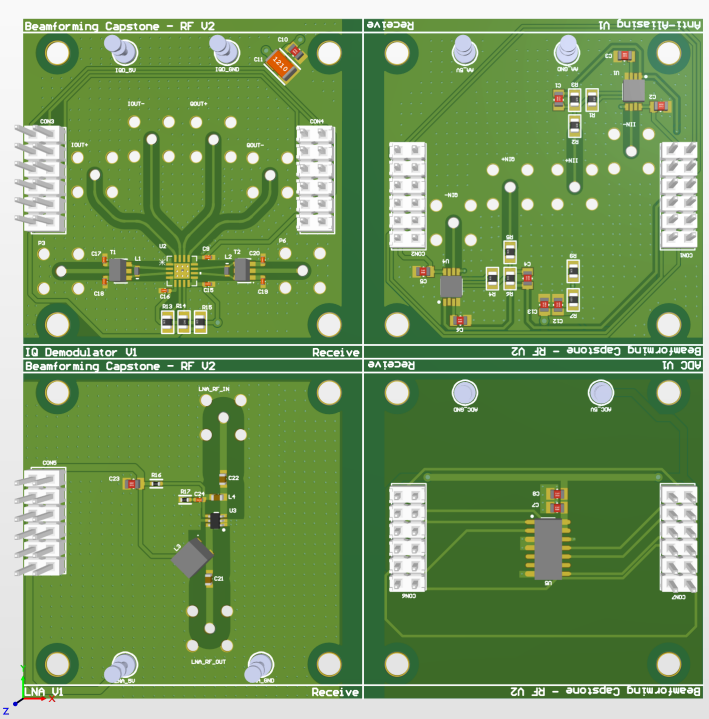


Figure 56: 3D Render of the receive modules on a single panel (The bottom left module pictures an external ADC which was not used in the final design due to the sufficient sampling rate of the MSP432P401 microcontroller built-in ADC).

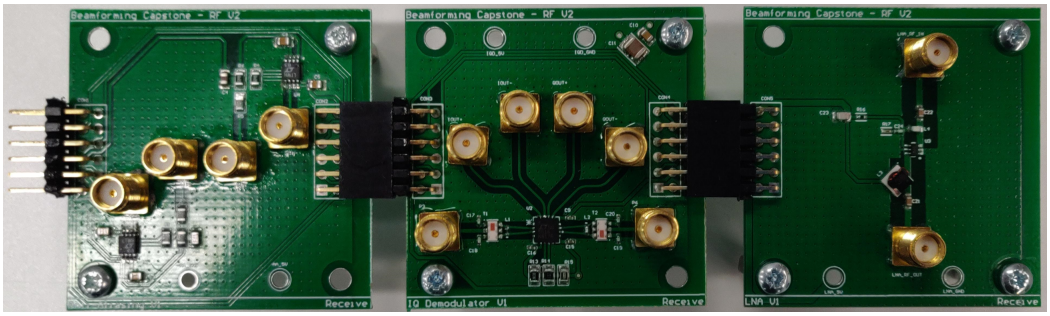


Figure 57: Photograph of a receive chain, from left to right: Anti-Aliasing Filter, I/Q Demodulator, Low-Noise Amplifier

The following section details the design and evaluation of the receive chain electronic hardware.

The receive chain consists of three modules

- Anti-Aliasing Filter (Filters the undesired bandwidth from the demodulated signal to prevent aliasing)
- IQ Demodulator (Performs demodulation of the received RF signal)
- Low Noise Amplifier (LNA) (Amplifies the received signal to above the demodulator's noise threshold)

See appendix for circuit schematics of the modules described below

5.4.1 Design

5.4.1.1 Anti-Aliasing Filter - LT1567

The LT1567 is an analog building block optimized for very low noise high frequency filter applications. In the receive chain, it is used as an anti-aliasing filter with a -3dB frequency of 25.6kHz in addition to converting

the differential output of the I/Q demodulator to a DC biased single ended voltage which maximises the swing usage of the sampling ADC [23].

Although the LT1567 is also capable of providing gain, for the purposes of this project it is configured with unity gain for simplicity.

Each anti-aliasing module contains two LT1567 chips in order to filter two pairs of differential signals for the I and Q channel outputs of the I/Q demodulator.

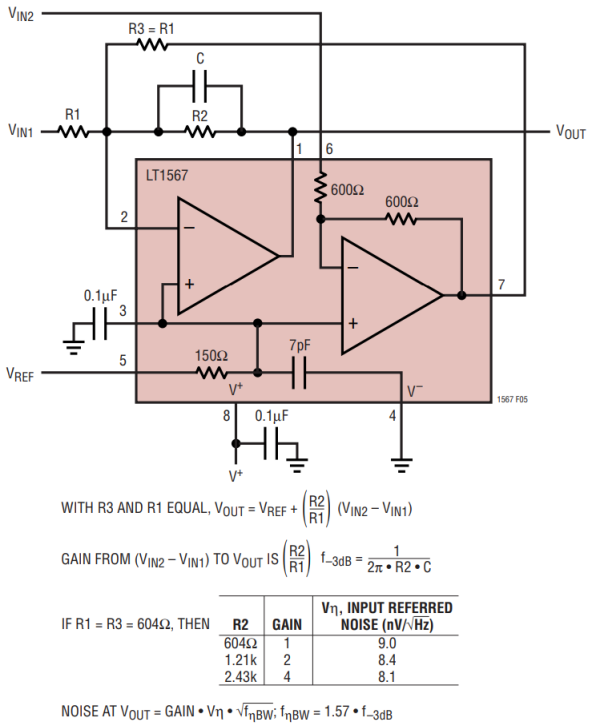


Figure 58: A differential to single-ended filter using the LT1567[23]

Additional simulations were ran using the manufacturer provided LTSPICE model to check for correct filtering of the signal at low frequencies and for correct DC biasing. The simulations yielded expected results, with the desired -3dB cutoff achieved at 25.6kHz with a positive DC bias.

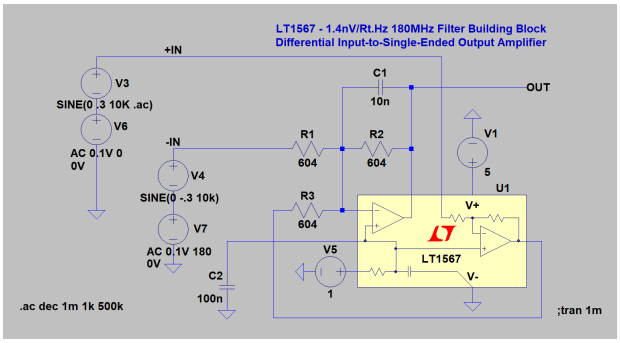


Figure 59: Simulation schematic and parameters for LT1567 as a differential - single ended anti-aliasing filter

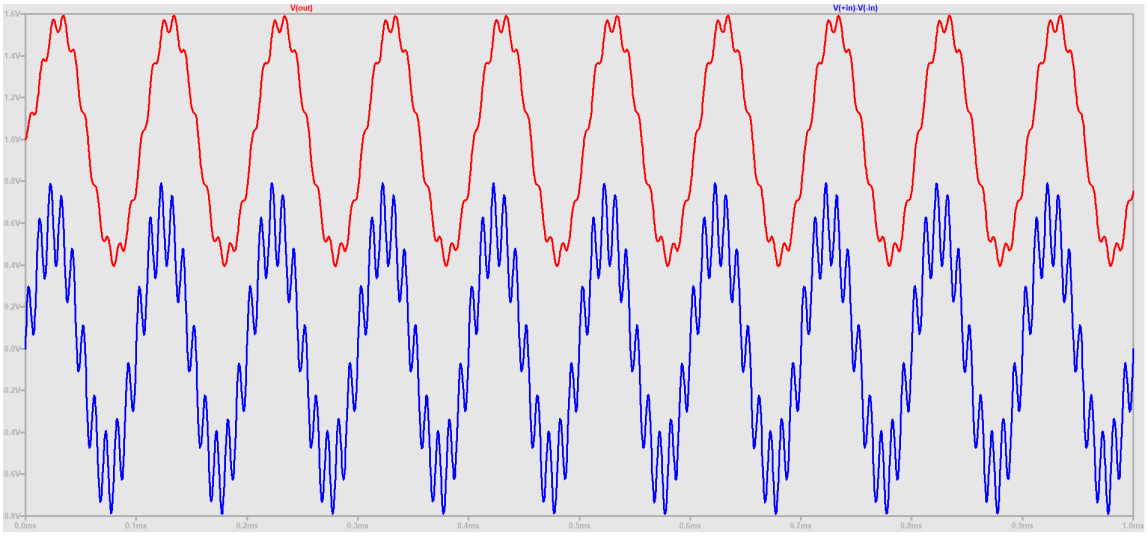


Figure 60: Filtering of sinusoidal 100kHz noise modulated onto the desired 10kHz signal

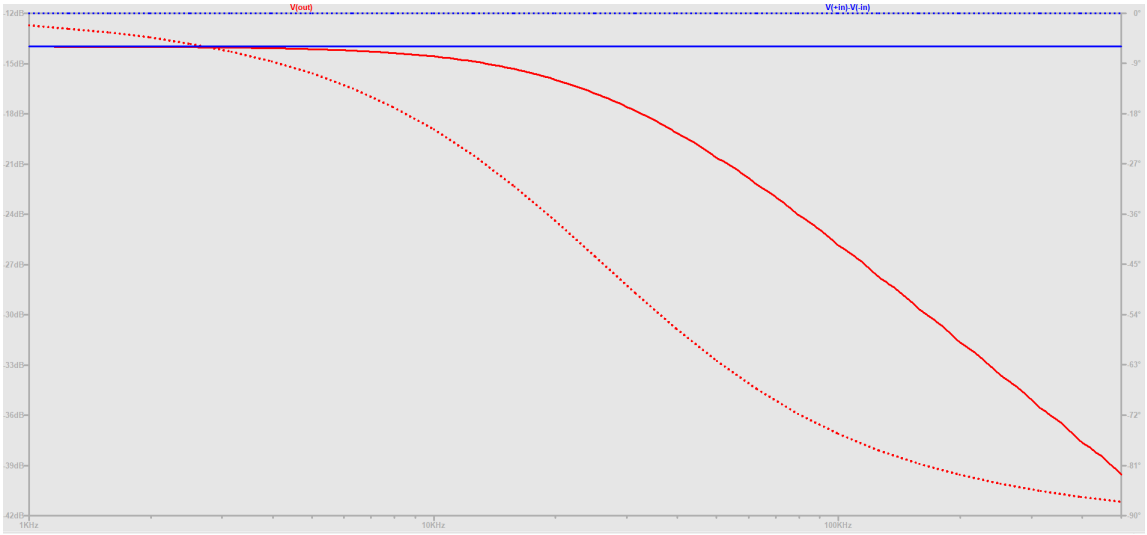


Figure 61: Frequency sweep of filtering circuit demonstrating -3dB cutoff at 25.6kHz

5.4.1.2 I/Q Demodulator - LT5516

The LT5516 is a 800MHz - 1.5GHz direct conversion quadrature demodulator. It functionally performs the inverse operation of the I/Q modulator described in the transmit chain section [24].

One minor implementation difference is that the LT5516 uses both differential input and outputs. As such, a balun transformer is used to produce the required differential local oscillator and RF signals at the input ports. The output ports are converted to single ended at the anti-aliasing filter stage (using differential amplifiers) as opposed to using a transformer. Using a balun for this operation would not be suitable due to the high-pass filter characteristics of transformers which would attenuate the desired low frequency baseband signal.

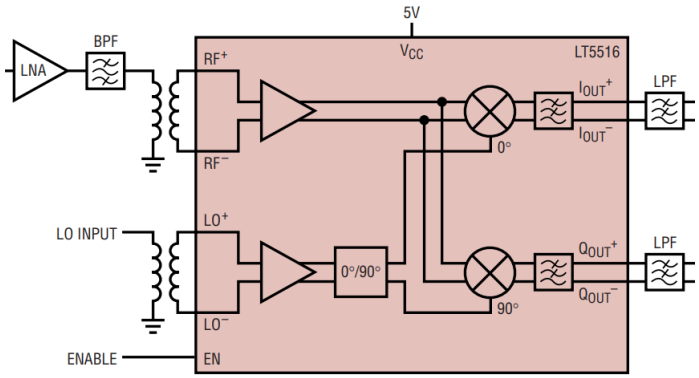


Figure 62: Block diagram of the LT5516 I/Q demodulator, note that the BPF is not included in the hardware for this experiment

5.4.1.3 Low-noise Amplifier - PSA-5454+

The PSA-5454+ is a high dynamic range (50MHz to 4GHz) low-noise amplifier suitable for sensitive receiver applications [25]. The purpose of this module is to perform necessary amplification of the potentially heavily attenuated signal received from the antenna to an appropriate power level for correct operation of the I/Q demodulator.

The module typically produces 18.3dB of gain at 1GHz. Variations in gain between multiple modules are not particularly relevant as they will be compensated for using the LMS algorithm (shown in Section 2) during the signal processing stage detailed later in this report.

5.4.2 Evaluation

The receive chain modules are difficult to evaluate in isolation due to the large number of differential inputs/outputs and lack of access to test equipment with a sufficient amount of ports. The anti-aliasing and demodulation parts of the chain are evaluated together by providing an amplitude modulated signal (generated using the transmit chain) into the coaxial input of the demodulator and observing the output of the filter stage using an oscilloscope for the correct baseband frequency.

In the test performed, a 10kHz sinusoidal baseband signal was input to the I/Q modulator directly using a function generator, with the un-amplified output connected directly to the receiving input of the I/Q demodulator. Amplification stages were removed such that the modules under testing would not be subjected to excessive input power (as there was no path loss in this testing), but were tested separately using an RF signal generator and a spectrum analyzer. The test was able to yield correctly DC biased demodulated 10kHz I/Q signals at the output of the anti-aliasing filter stage, validating the receive chain design.

5.5 Utilities

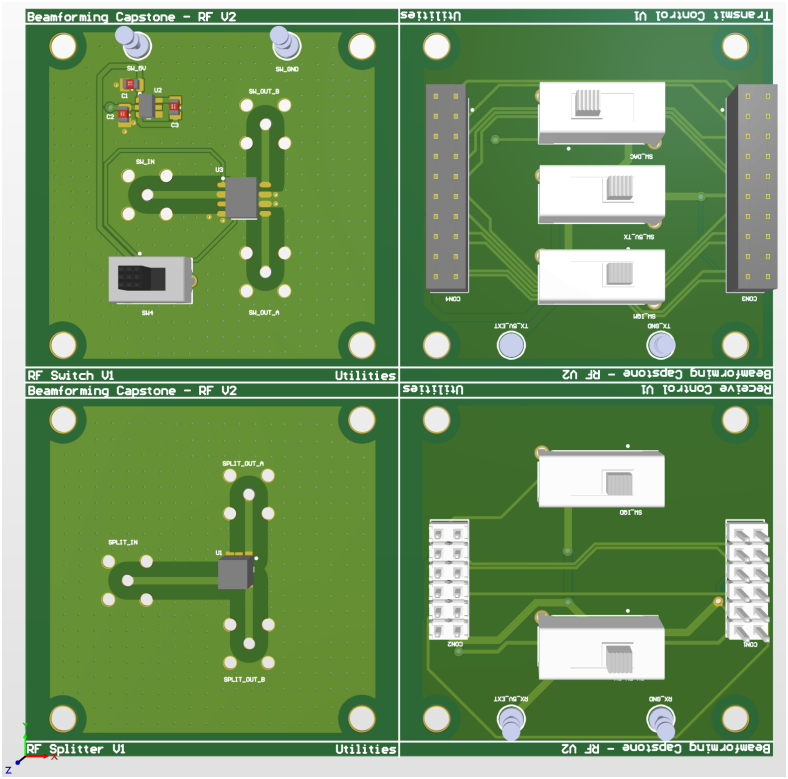


Figure 63: 3D Render of the utilities modules on a single panel (The two modules on the left side of the panel picture an RF switch and splitter module which were ultimately not used in the final system)

Additional PCBs have also been designed to physically interface to the transmit/receive chains and provide support functionality for ease of testing.

5.5.0.1 Switch Boards

Two basic switch boards (one for receive, and one for transmit) have been designed in order to provide physical switches for the enabling and disabling of individual modules via enable/shutdown pins. The boards pass through all other signals from the control module and are capable of switching enable signals between high/low for individual modules.

The switch boards are useful for testing the power draw and basic operation of each individual module.

5.6 Software/Control

5.6.1 Hardware Design

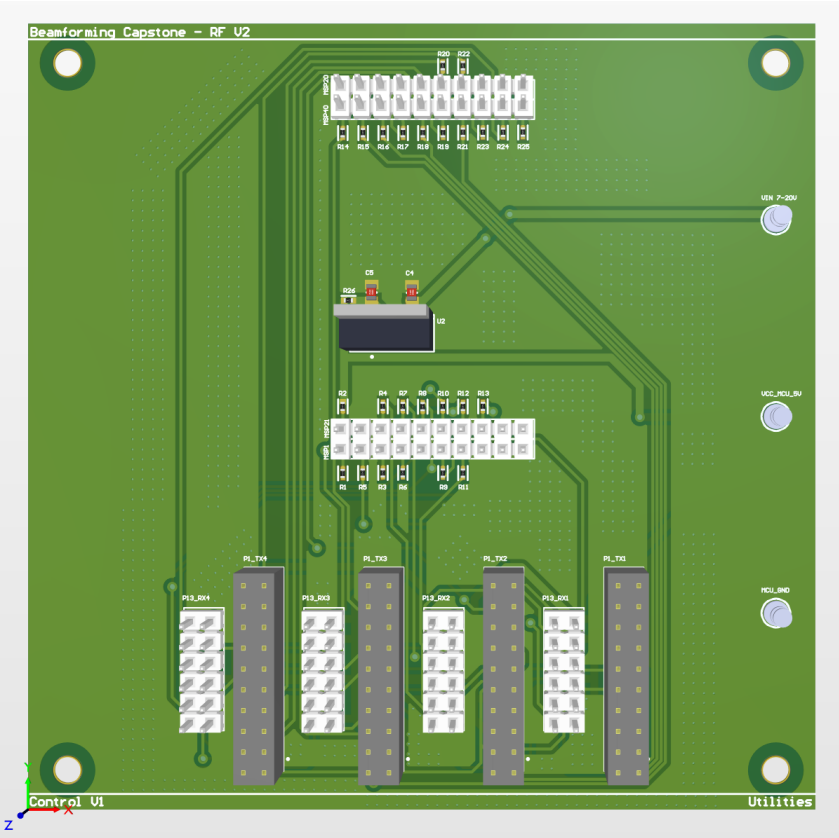


Figure 64: 3D Render of the control module

The control module is designed to break out the pinout from an MSP432P401R Launchpad development board to four transmit and receive headers via ribbon cables, allowing for four of either chain to be controlled simultaneously over a single SPI bus as an array.

It also supplies regulated 5V power, ground, and other DC utility signals to each connected chain

Regulated 5V power is important for several of the modules which use voltage dividers from the power rail to supply signal biasing voltages.

5.6.2 Microcontroller & Software Design

Controller software for the project is written for the TI MSP432P401R microcontroller[15]. The MCU has the following features which are used for the project.

- SPI peripherals and GPIO for sending control commands to the transmit chain in order to vary phase and gain
- UART peripherals for communicating with a Python client on a PC which performs user-interface and signal processing duties
- 16-bit precision ADC with up to 1MSPS and more than 8 channels for sampling of the received baseband I and Q channel signals across all four receive chains

All code was written using C++ using the following setup

- TI Code Composer Studio IDE version 9.0.1
- Simplelink MSP432P4 SDK version 3.10.00.08
- XDCTools version 3.55.0.11

- TI ARM Code Generation Tools (Compiler) version 18.12.2.LTS

The software is split into two different build configurations to suit the needs of controlling either the receive or transmit array. All control code and drivers can be found in the appendix as well as the public Git repository: `https://github.com/dingbenjamin/beamformer`

5.6.2.1 Transmit Software

The control module serves to reliably change the phase and gain of each individual antenna element in the transmit array. In order to do this, drivers have been written which take a keyboard input (in a Python client program) which is sent as an encoded message to a UART receiver on the control module[8]. The microcontroller is able to decode this message and subsequently perform the appropriate operation, whether it be changing the baseband voltages in order to manipulate the desired phase shift or adjust the gain of the power amplifier. There is also the added functionality of synthesizing low frequency baseband signals.

5.6.2.2 Receive Software

The main purpose of the control module in the receive array is to sample the output of the anti-aliasing filters on each RF chain. There are no controllable modules in the receive chain so there is no need to send SPI signals as with the transmit array software.

Sampling the entire array requires 8 ADC samples, one for each of the I/Q channels of each element; at the onset of the project the original desired array sampling frequency for receive beamforming was 45kHz (the Nyquist frequency of the anti-aliasing filter cutoff) before downsampling the baseband signal using a decimation filter and transmitting to a PC. Unfortunately this could not be achieved due to the large amounts of time taken to sample across multiple channels of the ADC and then transmit data to a PC over UART.

As such, there was no signal processing on the microcontroller and the highest achieved array sampling rate was at 4.28kHz by using a timer interrupt to perform ADC samples before offloading the data at 921600 baud. Each timer interrupt fires only when 8 samples (I/Q channels of all 4 elements) are finished and loaded into a buffer, and the time taken between the 8 successive samples was determined to be several orders of magnitude less than the signals of interest due to the ADC clock running in the MHz range.

Computations were performed offline after transmitting data to a receiving Python client on the connected PC. Due to the significantly lowered sample rate there was the addition of unfiltered aliased noise to all sampled signals, but in practice this did not noticeably degrade the quality of the received signal.

[8]The Google Protobuf format is used

6 Phase-Shift (Transmit) Beamforming Evaluation

A custom 4-element phase-shift beamforming system has been designed and implemented for this project. It is capable of meeting the following specifications (as is shown in Section 5):

- Antenna and transmitter both capable of operating within the frequency range of 800-1000MHz
- Phase shift angle resolution, accuracy, and jitter of better than 1°
- Maximum transmit power of at least 0dBm in order for sufficient SNR in the far field during test measurement
- Gain variance and accuracy within 1dB

As is explained previously in Section 4, the number of elements is chosen as 4 due to it being the minimum number of array elements that can demonstrate a clear and measurable difference when steering an array. A larger number of elements was not used due to the considerable amount of time and cost taken to design, manufacture and assemble each element. A downside of using an array with less elements is the inability to perform any further optimization in array weight synthesis due to the tendency of such algorithms to trade off the improvement of various parameters for main lobe width, which is already comparatively very wide compared to arrays with more elements. Instead, only basic conventional beamforming with a rectangular window is used for this experiment.

Ideally the experiment would be performed in an anechoic chamber such that none of the transmitted power is reflected back from nearby objects and walls but this was not feasible. The experiment was performed in a room with other electrical equipment present and with the transmitter and receiver situated approximately 1m off the floor. This will likely contribute as a significant source of error to the results

6.1 Equipment

- Four element transmit array comprising of four mounted antennae and assembled transmit chains as described in Section 5.
- TGR-1040 1GHz RF Function Generator, as a local oscillator
- 3 × BNC two-way T-junction splitters to split the local oscillator to each of the four transmit chains
- BNC/SMA Cables and appropriate adapters to connect RF components together (Local oscillator, receiving antenna)
- MiniVNA Tiny as a receiver[9]
- Antenna mounting block (wood) attached to a string tied to the array, for the purpose of maintaining the receiving antenna at an equidistant arc
- Receiver antenna (an identical antenna to the ones in the array is used) mounted in the aforementioned block

6.2 Parameters

- A measurement arc encompassing angles from -70° to 70° with 5° spacing between measurements
- Antenna with optimal return loss at 876MHz used for both the transmit array and the receiver[10]
- Transmit array with elements spaced 17.11 cm apart (half the carrier wavelength)
- Local oscillator power of 0dBm
  - This is further equally split 4-ways such that each chain approximately receives -6dBm of local oscillator power

[9]The MiniVNA Tiny is not able to directly measure RSS, instead the return loss (S21) measurement mode was used which measures the difference between the output and input ports. The output port was driven into a 50Ω dummy load and the receive port connected to the antenna. All RSS results are hence relative, and the absolute value was not measured. This only works in a narrowband case, as the S21 output power varies with frequency.

[10]This is the first iteration of the designed antenna. The third iteration of the antenna in Section 3 was not used due as it had not been manufactured in time for this experiment

- Receiver distance of 3.15m from the center of the transmitter array, which is clearly in the far field (greater than two wavelengths of approximately 30cm) of the array

Figure 65: Photograph of phase-shift beamforming experiment, with array pictured in the upper left and the receiving element mounted in block in the lower right

6.3 Method

1. A receiver is moved in a far-field equidistant arc from the center of the transmitter array, with the face of the receiver antenna always pointing towards the center of the transmitter array. This ensures the measurements are not affected by the radiation pattern of the receiving element.
2. The transmitter array is used to transmit an unmodulated carrier wave steered in a variety of angles, accomplished through phase-shifting individual elements appropriately.[11]
3. For a single steering vector, the arc is swept and relative received signal strength (RSS) measurements are taken at constant angular increments along the arc.
4. Repeat for different steering vectors, plot the resulting normalized radiation pattern and compare to the theoretical pattern generated through simulations.

[11] Required phase-shifts for a steering angle are calculated using the MATLAB Phased Array Toolbox

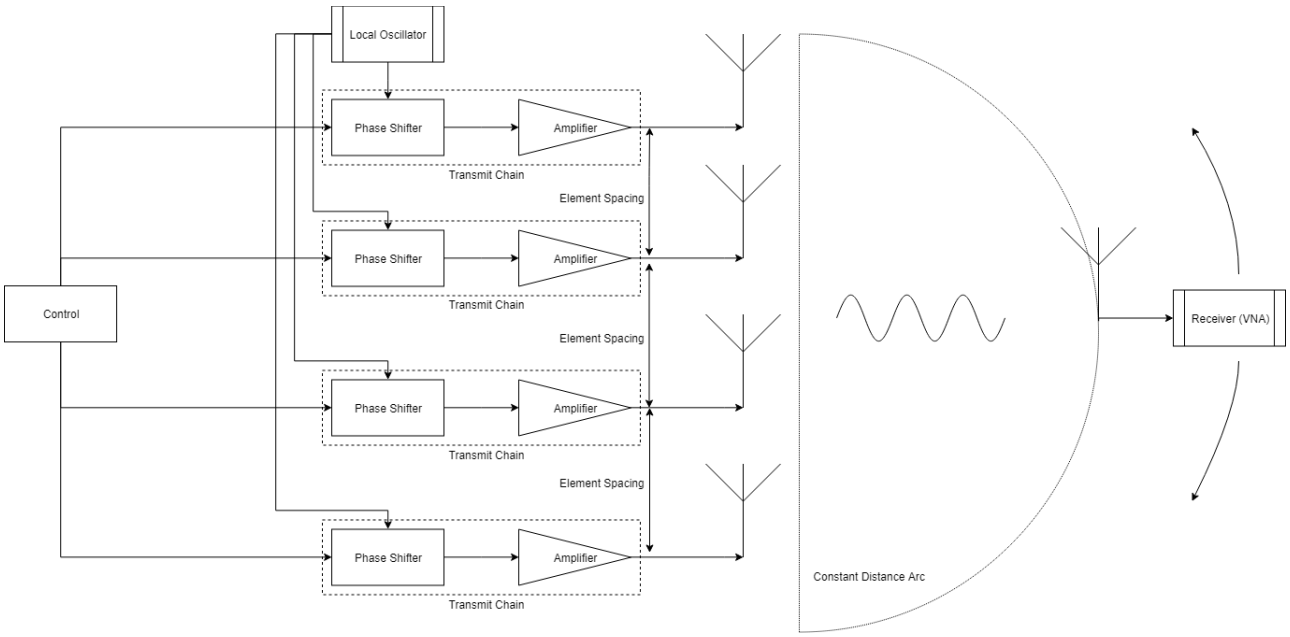


Figure 66: High-level diagram of phase-shift beamforming experiment

6.4 Safety Analysis

The ARPANSA[12] standard[26] enforces exposure limits to certain types of radiation at varying frequencies.

The limits used in the subsequent analysis have been taken as the most restrictive of the limits in the ARPANSA standard for time averaged general public exposure, and are designed to protect all members of the public (including children, the elderly, and pregnant women).

For the carrier frequency of 876MHz used in the described experiment, a limit of $4250\,\mathrm{mW/m^2}$ applies.

Given the antenna radius described in Section 3 is approximately 40 mm, the area of an individual antenna is $0.005\,02\,\mathrm{m^2}$.

Each individual chain is calibrated using a variable gain power amplifier to provide 0dB of gain from the local oscillator input to antenna output given a 150 mW magnitude baseband I/Q vector.

The maximum power used throughout the entirety of the experiment at the input of each antenna element is 5dBm (3.2mW). As such, the maximum power per squared meter at the point of a single antenna in the array is:

$$\begin{aligned} &3.2\,\mathrm{mW}/0.005\,02\,\mathrm{m^2} \\ &= 636.62\,\mathrm{mW/m^2} \end{aligned}$$

Adding more elements to this analysis also scales the effective area, such that the total power per area is the same, and below the limit.

Furthermore, if we consider a hypothetical point in the local proximity of the array at which the total RF power per area exceeds the power per area at the surface of a single element, we can loosely upper bound the maximum RF power per area due to constructive interference at four times the maximum single element power which is $2546.48\,\mathrm{mW/m^2}$, and is still below the ARPANSA limit. Realistically, the actual point (if it exists) will exhibit a much smaller power per area, as RF power will attenuate through the local medium.

6.5 Results

Due to restrictions with the large amount of time taken to calibrate each chain for a steering angle (The calibration process is explained in Section 5, only two steering patterns (0° and 20°) have been demonstrated

[12] Australian Radiation Protection and Nuclear Safety Agency

in this project. The steered patterns appear to adequately match the theoretical simulations, fulfilling the goal of the project.

The plotted radiation patterns can be seen in Figures 67, and 68, full results can be found in the appendix.

Note that in some cases the variation in RSS for measurements was up to 20dB. In order to filter the results as best as possible, the multiple readings were taken and averaged (in the mW domain, as dB is a logarithmic scale). The results have been converted back to decibel for plotting.

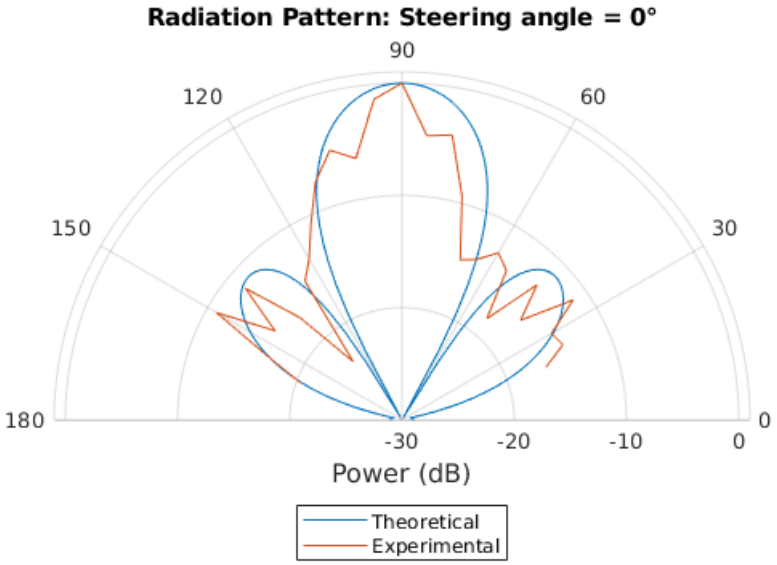


Figure 67: Experimental vs Theoretical Pattern for 0° steering

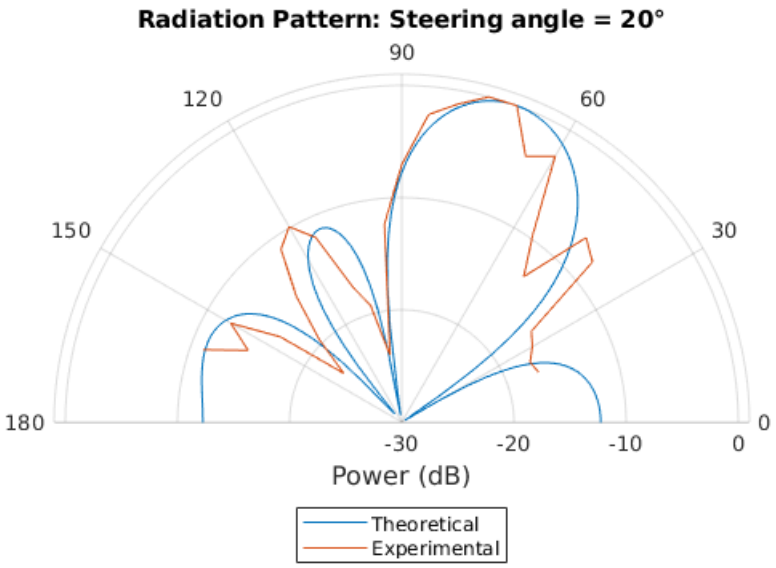


Figure 68: Experimental vs Theoretical Pattern for 20° steering

6.6 Discussion

From the plotted radiation patterns in Figures 67 and 68, it can be seen that the main lobe is clearly steered in the correct direction after applying the relevant steering weights between 0° and 20°.

The side lobes can also be made out in both plots, although there was a large amount of variance present in the received signal strength (RSS) measurement (consecutive measurements sometimes yielded differences of up to 20dB) which can also be seen in the plot. A source of this variance can be attributed to portable, USB powered VNA that was used. This VNA was unable to take power measurements in real time. If possible, the

experiment should be repeated in an anechoic chamber with minimal surrounding coupling and high performance test equipment in order to achieve an accurate and stable characterisation of radiation pattern.

Despite the large amounts of error in the measurement, clear peaks at the main lobe (0° and 20°) were seen for both steering patterns, and deep consistent nulls (in which all five measurements yielded RSS close to or at the noise floor) were also observed in areas expected between the main and side lobes.

During the sweep for 20° steering, a smaller overall RSS was observed and as a consequence a large amount of measurements fell below the noise floor. The plot seen in the results is a repeat of the same sweep but with the receiver distance approximately halved to 1.65m and local oscillator power increased from -5dBm to 5dBm which yielded an overall gain of around 10dBm. Subsequent measurements were sufficiently above the noise floor and all plots are of normalized RSS to eliminate this difference.

Smaller peak powers (typically 10dB less) were also observed when the array was steered to 30° despite the normalized plots showing a similar radiation pattern to theory. A hypothesis for the attenuation in peak power is due to the multiplication factor of the non-isotropic individual elements exhibiting a smaller gain at the particular angle steered to. However, this apparent power loss cannot be quantified without characterizing the complete three dimensional radiation pattern (a proposed method is detailed in Section 8: Further Work), and evaluating the total power transmitted.

7 Digital (Receive) Beamforming Evaluation

For evaluation of the digital receive beamforming system, the same 4 element array is used as in Section 6 but instead with each antenna connected to a receive chain detailed in Section 5 as opposed to a transmit chain.

The physical system is capable of the following specifications (as is shown in Section 5):

- Antenna and receiver capable of operating within the frequency range of 800-1000MHz
- Baseband bandwidth (anti-aliasing filter cutoff) of 25kHz
- Array sampling frequency of 4.28kHz[13]

The receiver hardware characteristics between each receive chain varied significantly, the effect of which will be explored further in later parts of this section.

7.1 Equipment

- Four element receive array comprising of four mounted antennae[14] and assembled receive chains as described in Section 5
- Transmit chain, with the synthesis module removed (only the I/Q modulator and power amplifier)
- Analog Discovery 2 [27], as a baseband signal generator for the I/Q channels of the transmit chain
- TGR-1040 1GHz RF Function Generator, as a local oscillator
- 4 × BNC two-way T-Junction splitters to split the local oscillator to each of the four receive chains in the array in addition to the single transmit chain.
  - Due to the limited amount of equipment available, the same local oscillator was shared between both the receive and transmit chains. To achieve this, 4 BNC T-Junction splitters were used to divide the local oscillator between five modules as shown in Figure 69. This resulted in -6dB loss to three modules, and -9dB loss to five modules and was compensated for by increasing the local oscillator power.
  - Relative difference in element power was not expected to be an issue as the LMS algorithm was expected to account for it during weight training.
- Antenna mounting block (wood) attached to a string tied to the array, for the purpose of maintaining the transmitting antenna at an equidistant arc
- Transmitter antenna (an identical antenna to the ones in the array is used) mounted in the aforementioned block

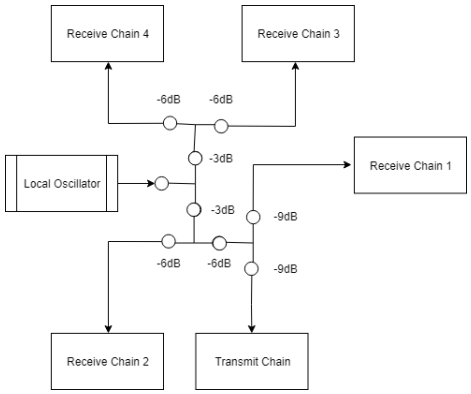


Figure 69: Division of local oscillator between transmit/receive chains

[13]This is below the Nyquist frequency of the baseband bandwidth, but in practice little noise is aliased onto the sampled band, see Section 5:Receive Software for more details

[14]Similar to the transmit beamforming evaluation, the antennae used in this evaluation are the first iteration of the design described in Section 3 for the same reason as described in Section 6

7.2 Parameters

- Local oscillator power of 7dBm at 876MHz, split 5 ways as is described in the Equipment section
- A transmitter distance of 1m
  - This was mainly limited due to the coaxial cable lengths available
  - 1m is approximately three wavelengths from the array, which is sufficient to be considered far field
- Transmitter angle of -0° and -30°
- Transmit chain power amplifier gain of 31.5dB
- I/Q modulator input signals:
  - I-: 0.58V
  - I+: 0.58V
  - Q-: 0.58V
  - Q+: Time varying with an amplitude of 75mV, centered at 0.5V
- Signal Types:
  - 200Hz Sine Wave
  - 75Hz Triangle Wave
  - 10Hz Square Wave

The specific values of the differential Q+ channel signal voltages are chosen such that there is a DC offset in order to create an AM signal. A DSB-SC signal is not desired in this case, as the baseband signal crossing across the 0 point causes a phase reversal after demodulation. Instead, an appropriate amplitude and DC offset was chosen such that the message power does not cause the modulated signal to encounter phase reversal. In this case, the maximum local oscillator power limits maximum DC offset (which is ideally at half the amplitude of the carrier), which in turn limits the baseband message amplitude.

7.3 Method

The general procedure involves collecting several sets of data, one of which is used for training (sinusoid) whilst the remainder are used for evaluation (triangle and square).

1. A single transmitter is placed at a point in the far field of of the receiving array with the face of the transmitting antenna pointing towards the center of the receiving array.
2. The transmitter is used to transmit a simple AM modulated signal as described by the parameters to the receiving array
3. Using the written software, the receiver array will sample and send acquired data to a client running on the connected PC
4. The acquired data is then input into a MATLAB script, which iterates across the data using the LMS algorithm to train a set of weights
5. The signal is then changed to a variety of different types and steps 1-3 are repeated
6. The trained weights from step 4 are then used to determine performance on the acquired testing data

7.4 Safety Analysis

RF power measurements at the output of the transmit chain yielded -5dBm, which are below that of the power levels used in the phase-shift transmit beamforming experiment in Section 6. The large reduction in transmitter power is due to the decrease of input local oscillator power resulting from the additional splitters which in practice decreased power by more than 3dB.

As such, the maximum power per area generated in this experiment is less than that which is seen in the safety analysis of Section 6, and therefore compliant with the ARPANSA standard.

7.5 Data

Six data sets were collected from the receiver array, each comprising of 62000 array samples collected at a sample rate of 4.28kHz over 15 seconds. The baseband signals used were a 200Hz sine, 75Hz triangle, and 10Hz square wave, all of which were collected at both 0° and 30°.

Figures 70 and 71 below show the data collected from each element (combined magnitude of I/Q) for the 200Hz sinusoid at 0° and -30° and illustrates the general performance of the array. Full data for the triangle and square waves can be found in the appendix.

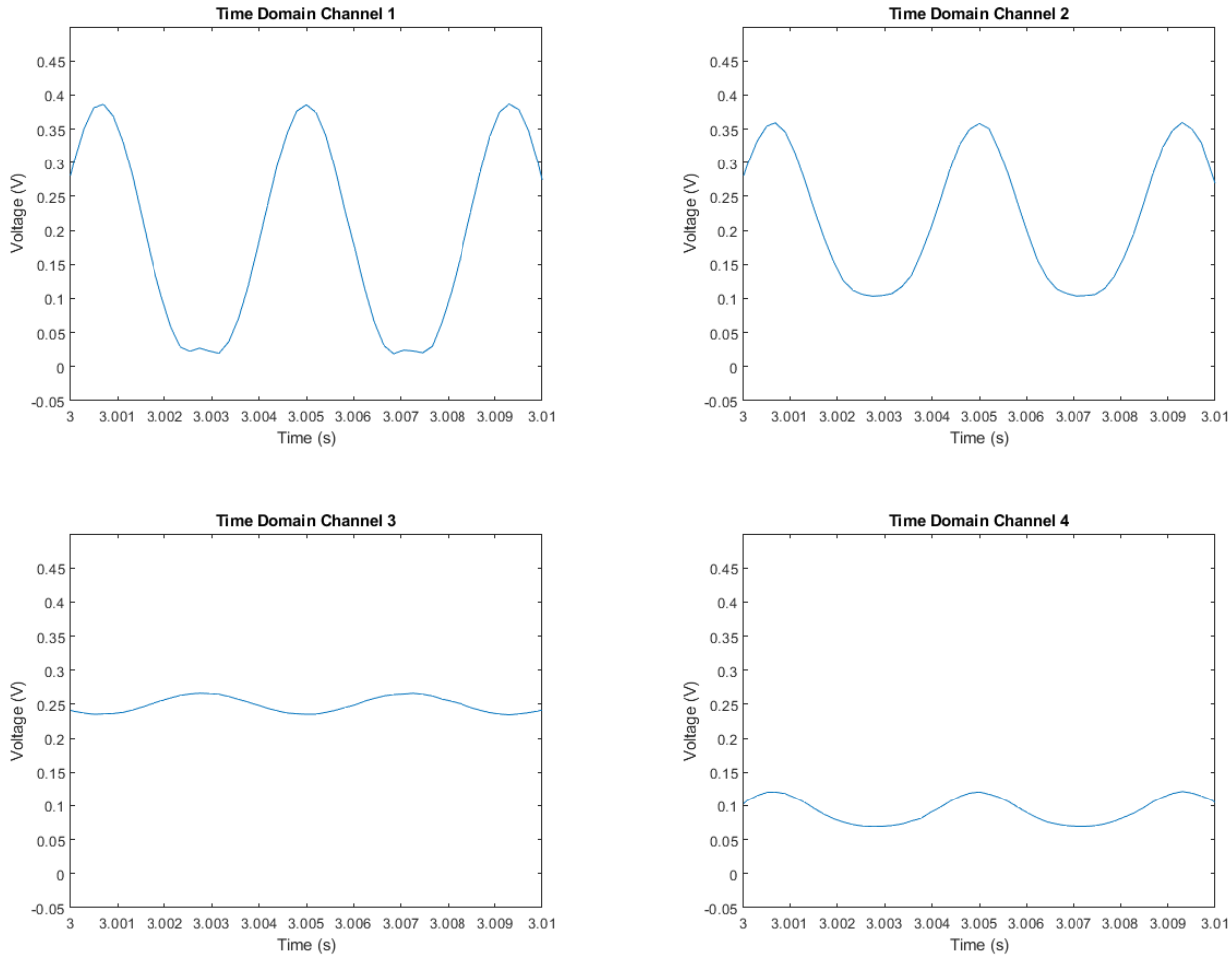


Figure 70: Time slice of received 200Hz sinusoidal signal across four elements at 0° transmitter placement

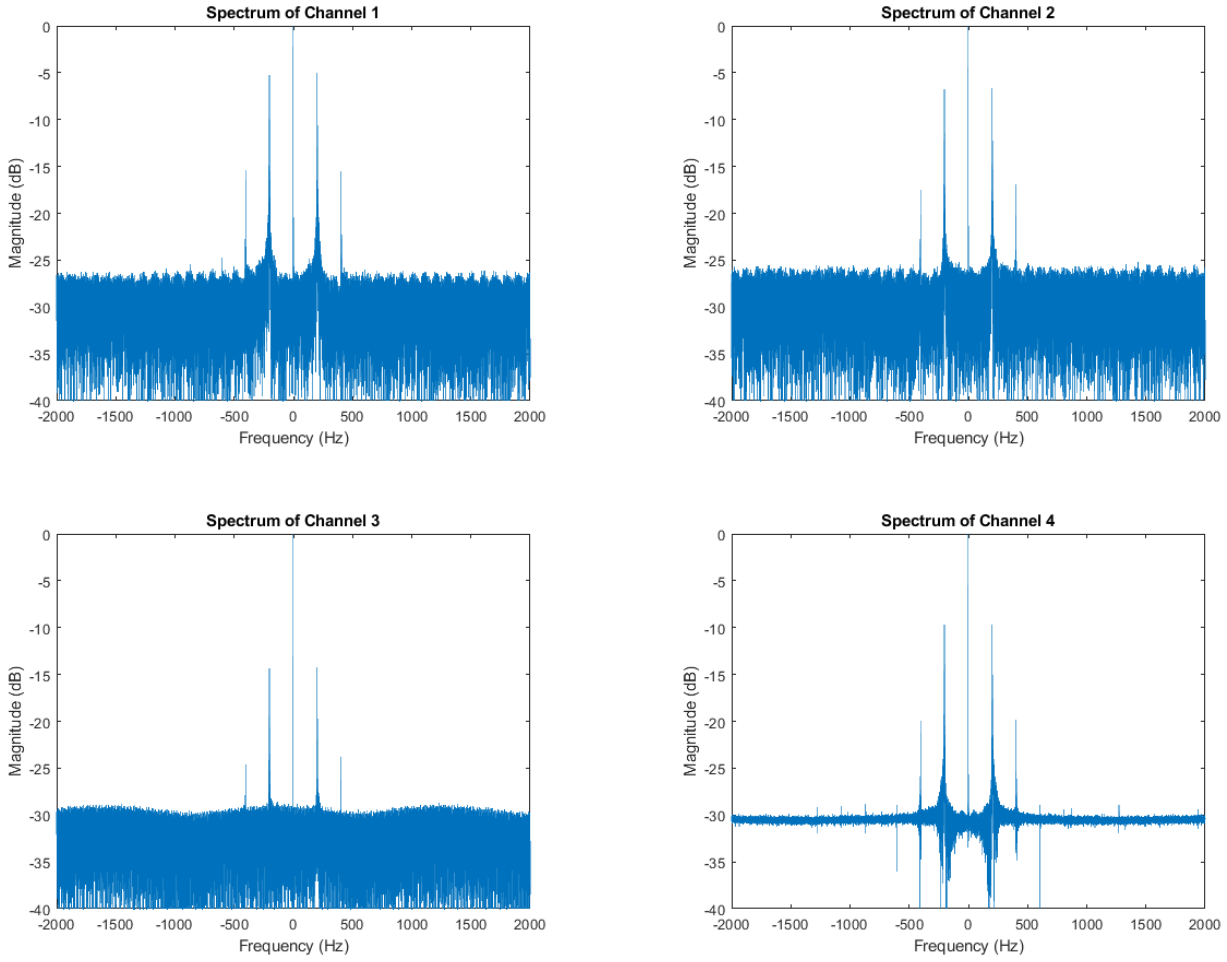


Figure 71: Spectrum of received 200Hz sinusoidal signal across four elements at 0° transmitter placement

From Figure 70, it can be seen that the message amplitude varies significantly across the different elements, with elements 3 and 4 receiving a significantly smaller signal compared to elements 1 and 2. A small clipping at the bottom of the element 1 time domain signal can also be observed, where the time domain signal approaches the zero point.

The frequency domain content of each element is also presented in Figure 71 - all received show sufficient SNR in order to continue with using the LMS algorithm. Additionally, harmonics of the 200Hz signal can be seen at 400Hz and 600Hz, the presence of these harmonics is ignored for the purposes of the experiment.

The frequency domain plots also show the large DC components of each signal - these will be filtered out in a subsequent stage and will be discussed in the next section.

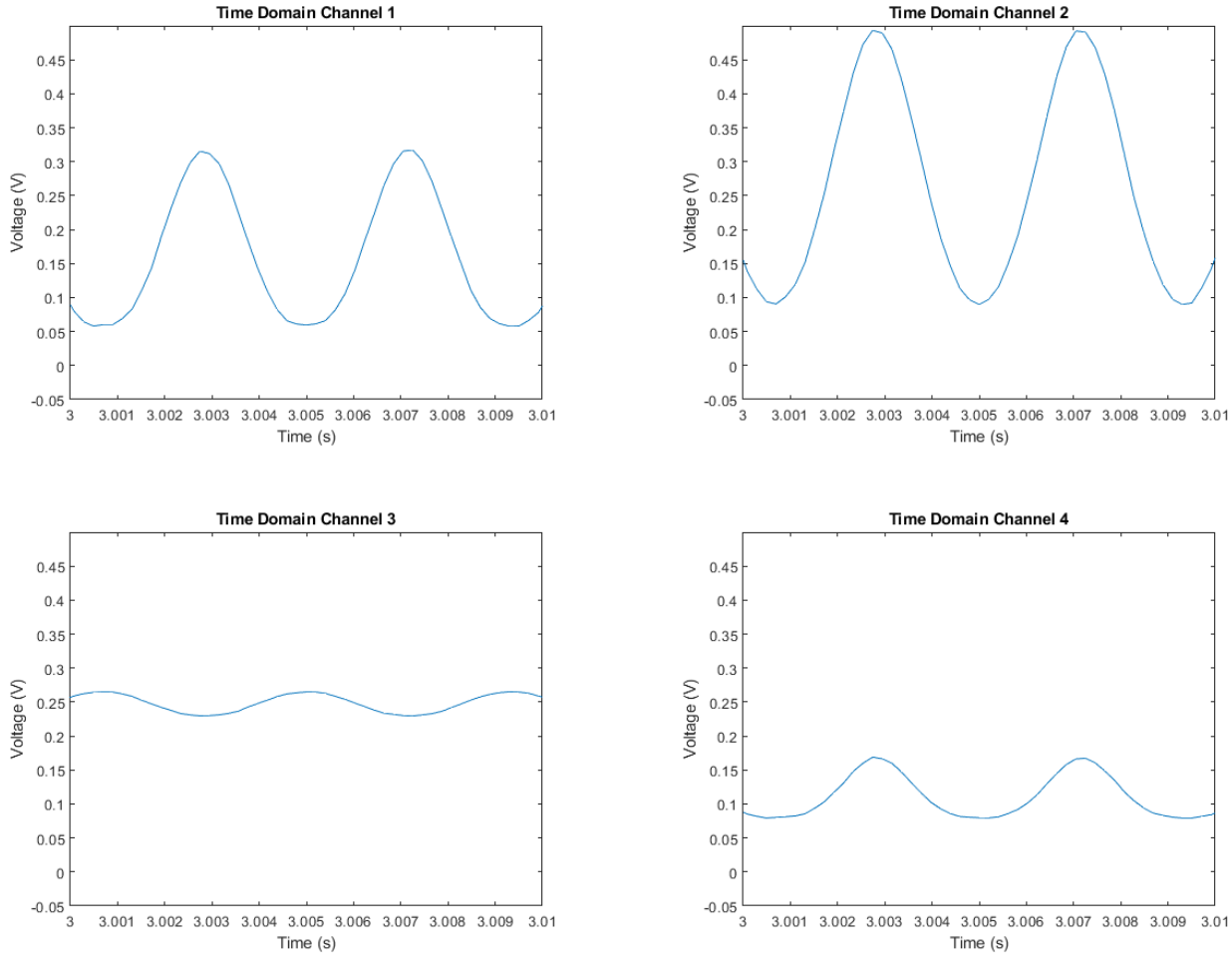


Figure 72: Time slice of received 200Hz sinusoidal signal across four elements at -30° transmitter placement

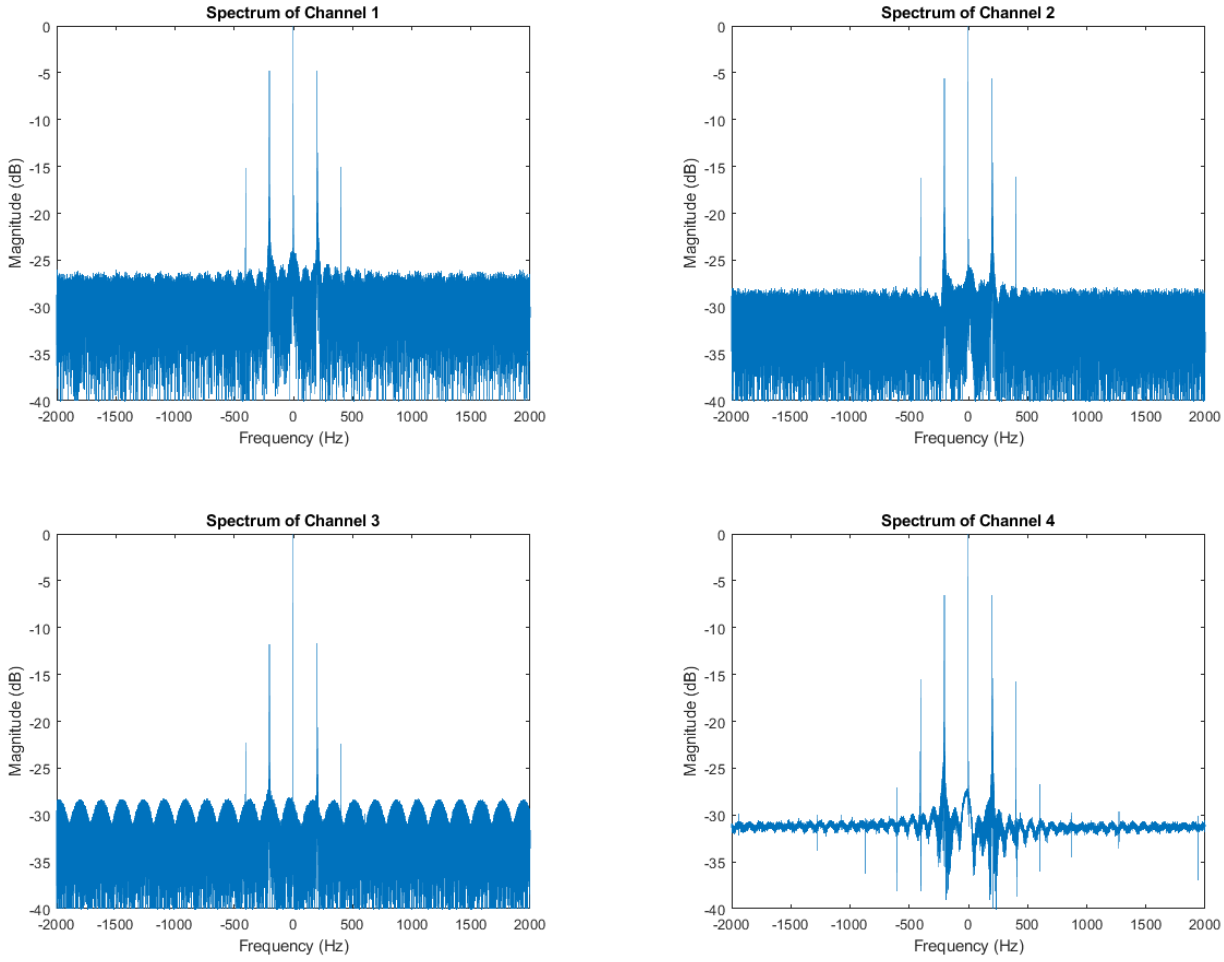


Figure 73: Spectrum of received 200Hz sinusoidal signal across four elements at 30° transmitter placement

Figures 72 and 73 show the same element measurements but with the transmitter positioned at the -30 degree point on the equidistant arc.

Interestingly, the noise floor of element 4 appears to cover a smaller range of magnitudes compared to that of the other elements. In addition, the frequency spectrum of element 3 exhibits periodic oscillations. This behaviour is not investigated further, as the main signal of interest is the 200Hz sinusoid.

7.6 Signal Processing

This section will discuss how the experimental data is processed in order to manipulate the array spatial sensitivity pattern using the LMS algorithm. All operations are performed in MATLAB and full scripts can be found in the appendix.

7.6.1 Preprocessing

First, the received training data is filtered to remove anomalous readings, an example of anomalous readings generated by the sampling module can be seen in Figure 74. These anomalous readings effectively decrease the SNR of the signal and are present in all elements with varying frequency. This preprocessing is not performed on evaluation data as the nature of the filtering requires prior information to be known about the signal, which can only be assumed for the training signal (200Hz sinusoid).

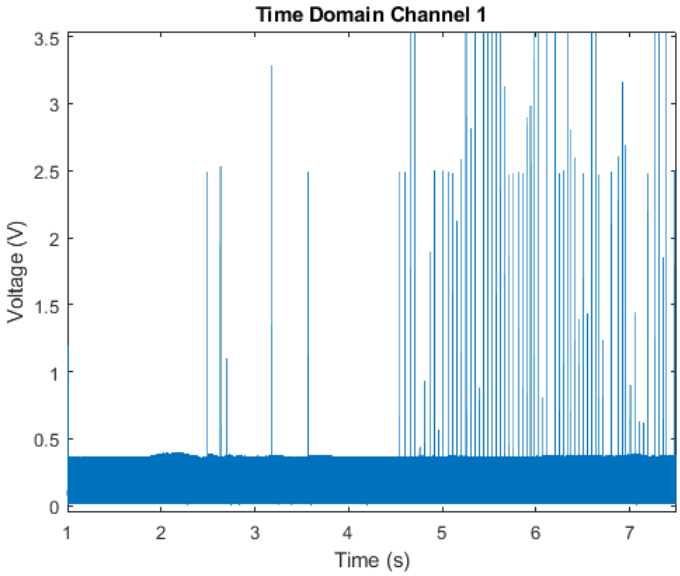


Figure 74: Example of anomalous readings on Element 1

Three approaches are considered to filtering the data

- No filtering
- Low pass filter with at 250Hz cutoff
- Set all samples above a threshold (determined by the mean of the data) to the mean

The results obtained from each method can be seen below in Figures 75, 76, 77.

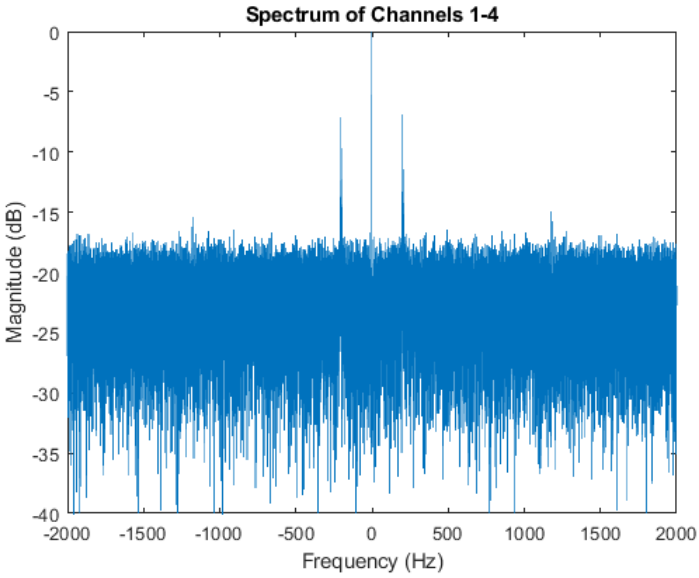


Figure 75: Summation of Elements 1-4 at 0° steering with no filtering

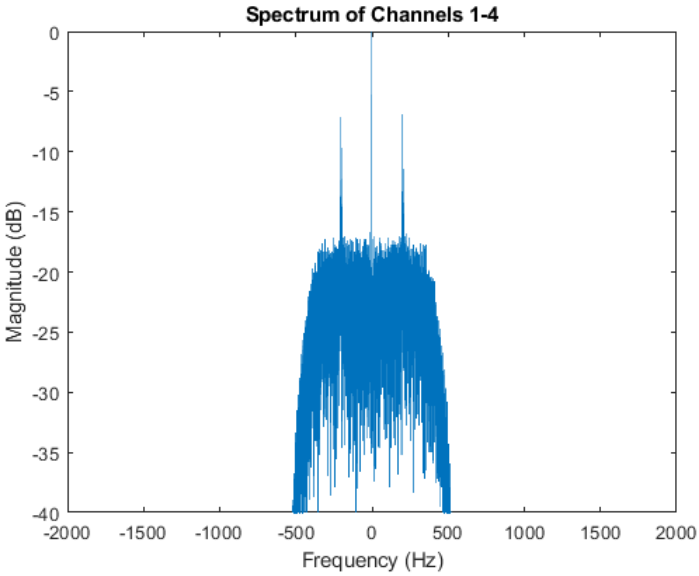


Figure 76: Summation of Elements 1-4 at 0° steering with a low-pass filter at 250Hz cutoff

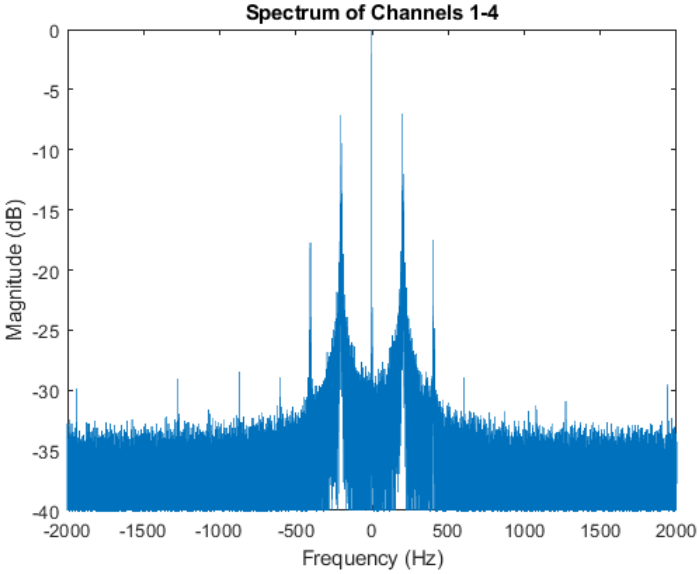


Figure 77: Summation of Elements 1-4 at 0° steering with a thresholding filter

The thresholding filter is able to reduce the noise floor of the data the most out of all the options, but requires a precalculated mean across the dataset for thresholding as opposed to the low-pass filter. The low-pass filter however, is also able to remove noise from unwanted bands above the cutoff.

In subsequent runs of the LMS algorithm, the data sets collected are processed with both no filter, and the thresholding filter. The low pass filter is not used as it changes the spectral shape of the noise floor and is more difficult to compare.

In addition, all datasets are DC filtered to remove frequencies below 10Hz. This brings the bias of all the time domain signals to 0 such that they may be compared against the training signal during the LMS algorithm.

7.6.2 LMS Algorithm

Using differently preprocessed data from the previous section, the LMS algorithm is run using the 200Hz sinusoid as the training signal. The converged weights are then used to evaluate performance on the test data set of a 75kHz triangle wave and 10kHz square wave against a control weight vector of $[1, 1, 1, 1]^T$ which simply sums all elements.

The step size used to generate data in Figures 79 and 78 is set to 0.00001. This was experimentally determined to be a suitable step size to achieve convergence without oscillation. Although the benefits of a variable step size are considered in the simulations for the LMS algorithm in Section 4, processing of data in this section will only use a fixed step size.

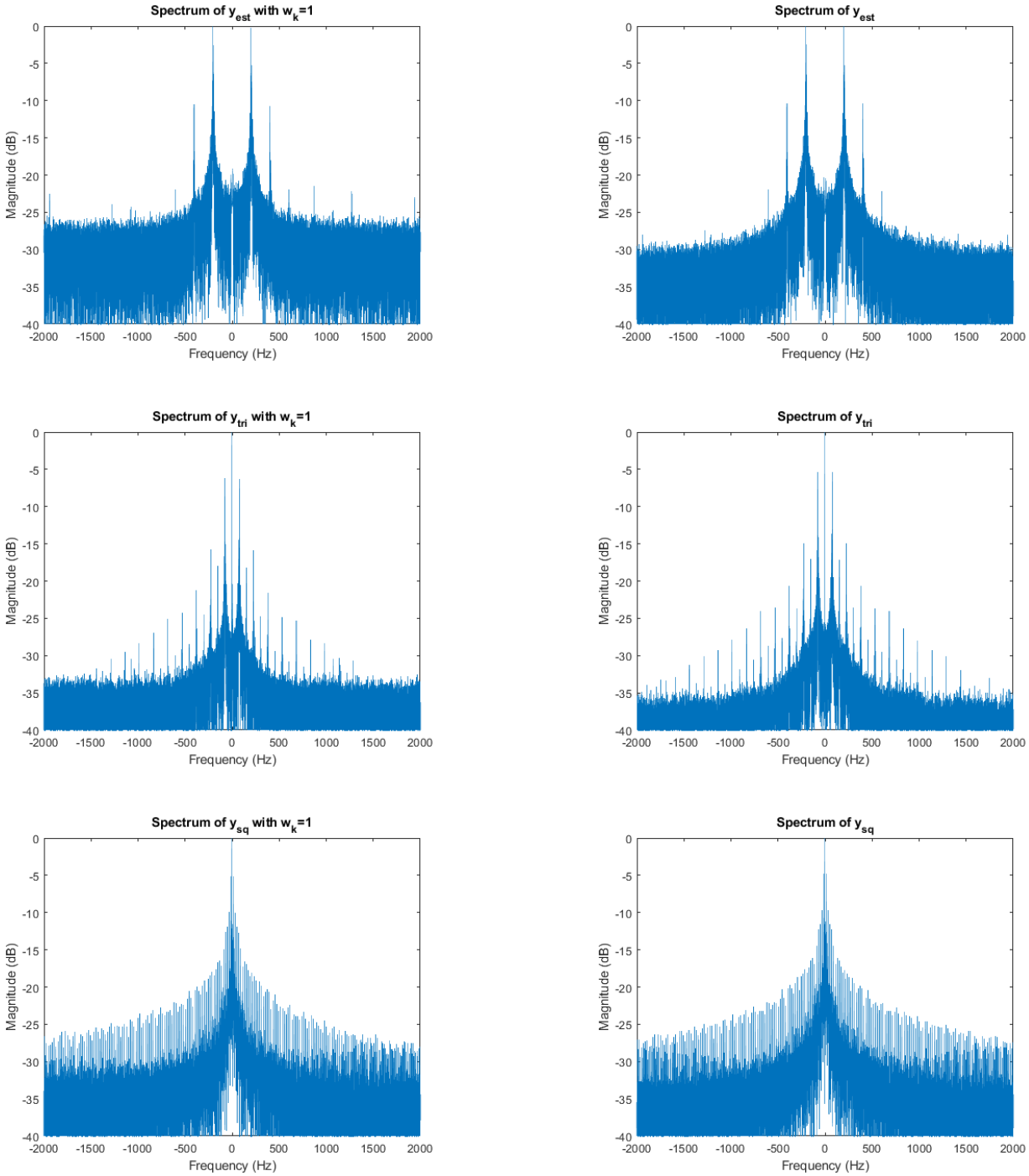


Figure 78: A comparison of the LMS algorithm performed on data preprocessed using the thresholding filter

On data preprocessed using the thresholding filter, aproximately 5dB of increased SNR can be seen on the training sinusoid as well as the triangle wave after using the LMS algorithm Figure 78. However, there is no observed SNR increased on the 10Hz square wave.

However, when the amount of noise in the system is increased by removing the thresholding filter and instead using no filter, the performance gain from the LMS algorithm can be seen to drastically increase in Figure 79. The SNR increase from using the LMS algorithm is observed to be approximately 10dB across all three signals, demonstrating the effectiveness of correctly weighting the appropriate elements.

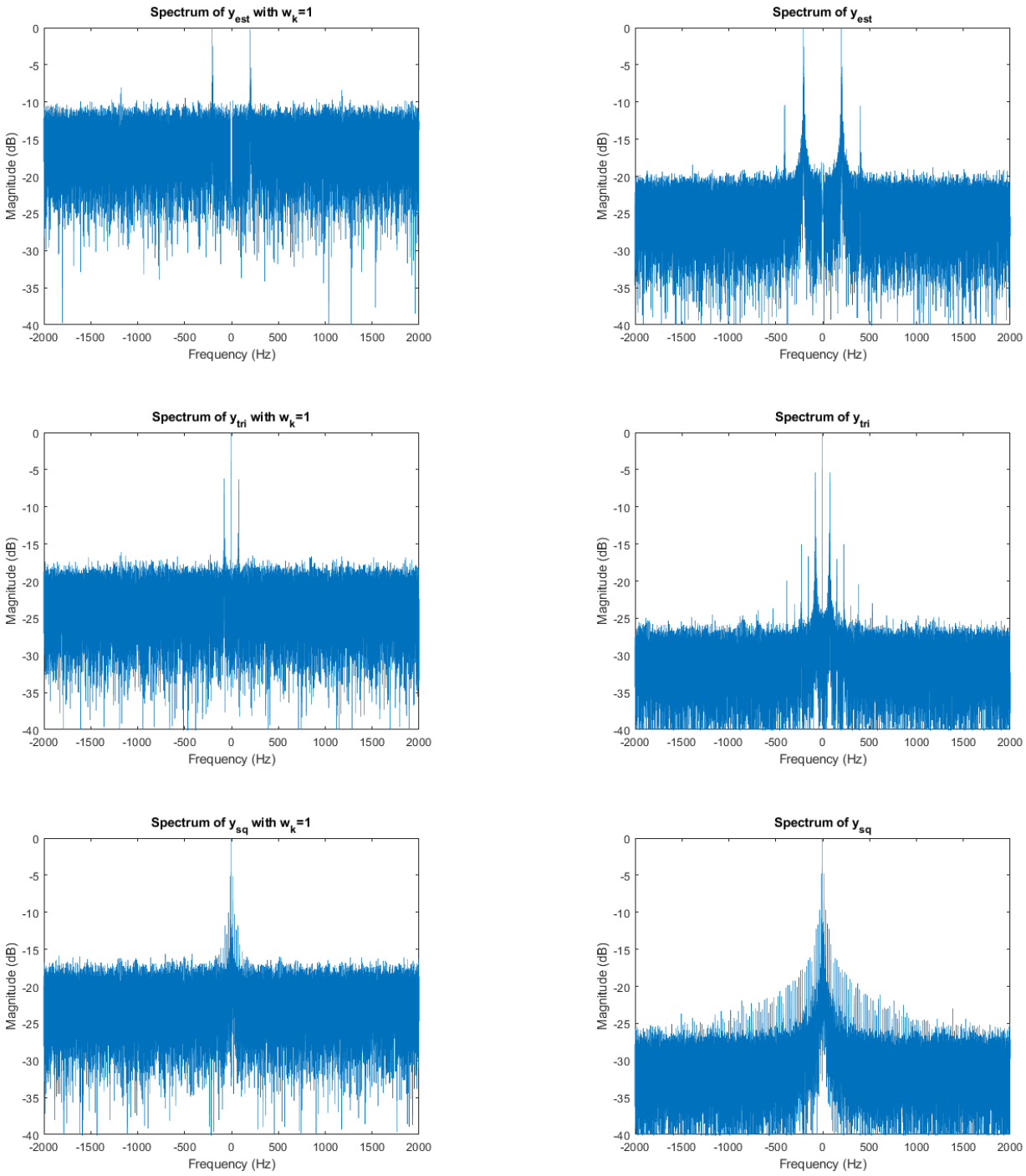


Figure 79: A comparison of the LMS algorithm performed on data preprocessed with only a DC filter

The LMS algorithm is also able to adapt the array weights in the case of moving the transmitter to a -30° angle. Results can be seen in Figure 80, and demonstrate approximately 4-5dB of SNR increase across all three signals. Although the SNR increase from LMS beamforming is less for the case of -30° steering compared to 0° steering, this mimics the loss in total signal power experienced in the phase-shift beamforming evaluation when the array is steered, and can potentially be attributed to the non-omnidirectional individual element pattern of each antenna in the array.

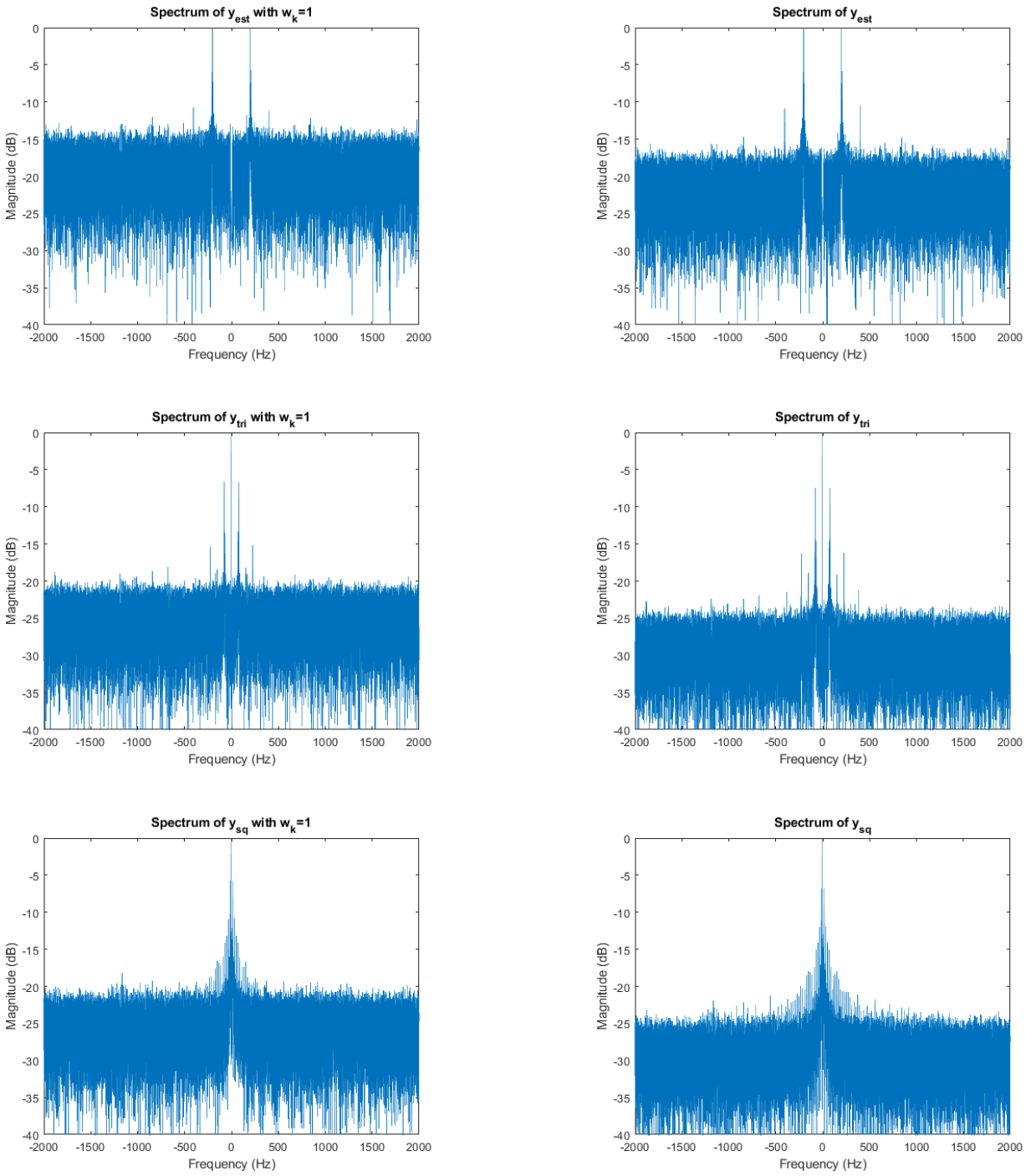


Figure 80: A comparison of the LMS algorithm performed on data preprocessed with only a DC filter

Lastly, the radiation pattern generated by the LMS optimized weights can be seen in Figure 81. Unfortunately, it is difficult to see any significant steering between the radiation patterns as the pattern at 0° is already skewed due to the variances in each element of the array, and further changes in weights in the -30° data set only distort the radiation pattern even more. Instead, performance of the algorithm is evaluated using relative SNR gain measurements.

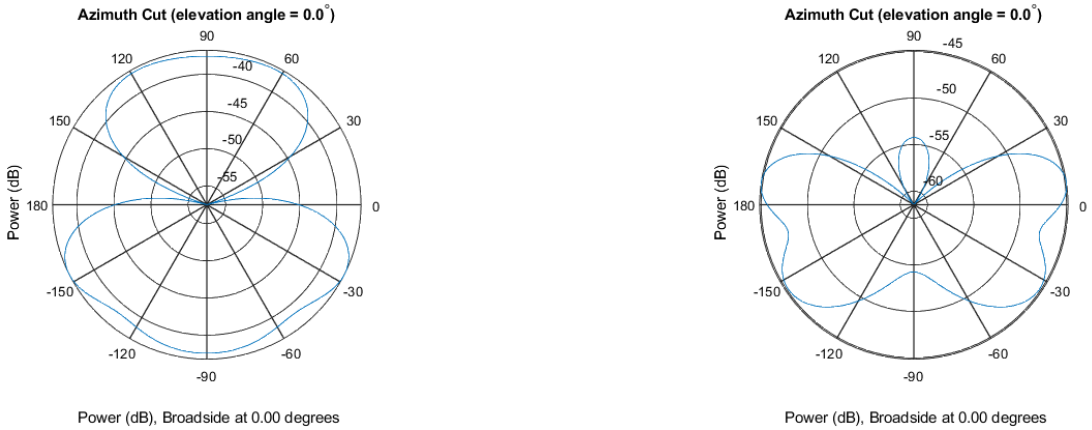


Figure 81: Radiation pattern of weights generated from 0° (left) and -30° (right) transmitter position data

7.6.3 Weight convergence

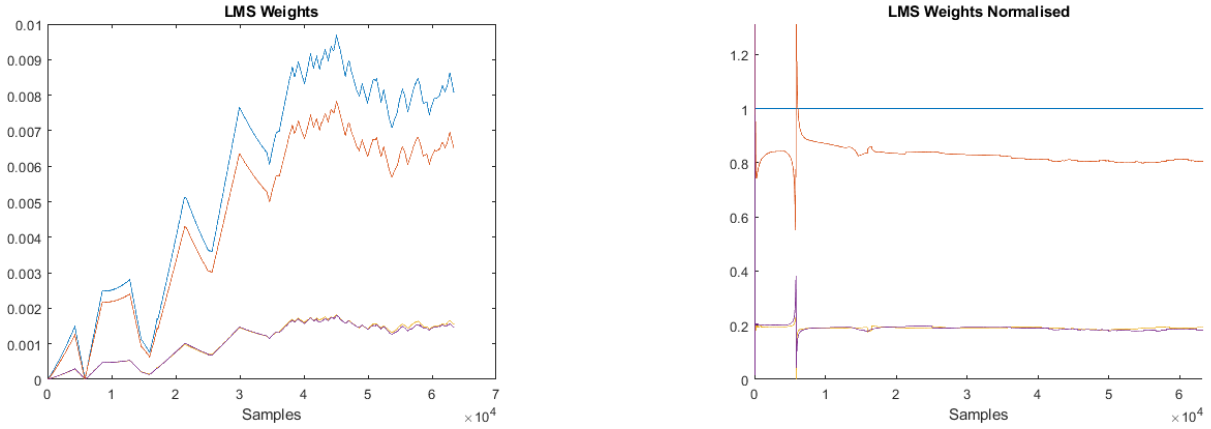


Figure 82: Weight convergence of the LMS algorithm at 0° with only a DC filter

The convergence of weights throughout each iteration of the algorithm can be seen in Figures 82 and 83. The plots show the weight values divided by the magnitude of the first weight (normalised)[15], as well as the raw weights. Although the raw weight values appear to oscillate whilst training, the ratio between weights converges very quickly and maintains a stable value after convergence.

One exception to this is sharp spikes seen in the normalised weight plot, which correspond to snapshots where the raw weight magnitudes are very small. This can be seen around snapshot 5000 in the plots. A hypothesis for the generation of these unstable spikes is that when the weight magnitudes are smaller than the order of the relative step size, the ratios between weights change very quickly, hence causing a spike in the relative magnitude.

However, the algorithm appears to be able to recover from these spikes and weights quickly return to the same stable ratio.

[15]Note this sets the weight of the first element to a constant 1 in all plots

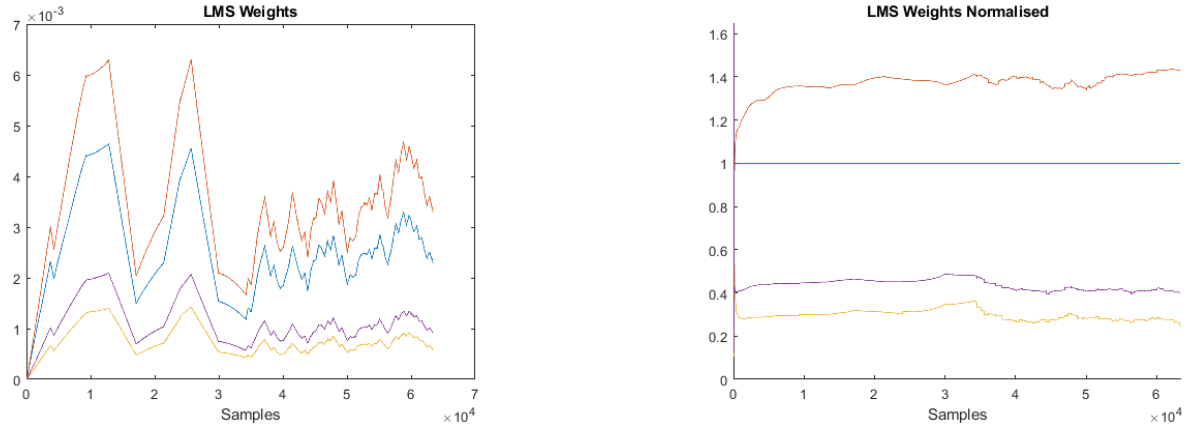


Figure 83: Weight convergence of the LMS algorithm at 30° with only a DC filter

The effect of a frequency deviation between the prerecorded training signal and the received signal can have a large effect on the convergence stability of the LMS weights. In the case of this project, the sampling frequency was only known to a tolerance of a few Hz. Because of this, the effective frequency of the sampled signal was not exactly 200Hz, resulting in a need to manually tune the prerecorded training signal for a better frequency match.

For the data sets seen in this section, the training signal frequency was experimentally tuned to 202.3kHz after data collection for best performance. A real-world implementation of the LMS algorithm will require this calibration to be done prior to usage.

In order to examine the effects of a frequency deviation, weight convergence with an identical dataset but using a training signal frequency of 200Hz (as opposed to 202.3Hz) is shown below in Figure 84.

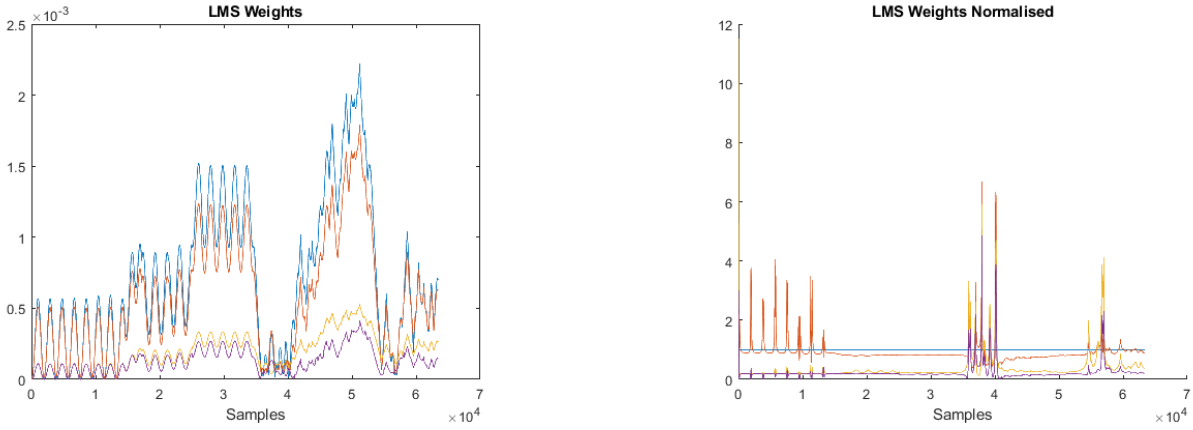


Figure 84: Weight convergence of the LMS algorithm at 0° with only a DC filter, using a frequency mismatched training signal of 200Hz

Higher frequency oscillations can be observed in the absolute weight magnitudes of the 200Hz trained LMS algorithm whilst a fairly constant ratio can again be observed in the normalised weight ratios. This suggests the cause of lower frequency oscillations in the optimal dataset may also be a result of (but to a lesser extent) a frequency mismatch.

Given the same accuracy or tolerance of training frequency, a potential suggestion for reducing the effects of a frequency deviation would be to increase the frequency of the training signal such that a constant deviation forms a smaller proportion of the frequency.

7.7 Discussion

The results of the receive beamforming evaluation show that in all cases usage of LMS algorithm calculated weights performs at least equally (in certain cases with very low amounts of noise) but typically better by

3-10dB over using a one-initialised weight vector.

Although there is a definite measured performance gain, the magnitude of the weights (and hence the estimated output training signal) does not appear to converge properly to recreate the recorded training signal. Despite this, the ratios between the weights do converge, and it is seen that this ratio converges to improved values (in terms of SNR) over the basic $[1, 1, 1, 1]^T$ weight vector.

The data collected suggests that in the presence of high noise on certain elements, the calculated weights will prefer elements with a higher SNR over lower. This can also be inferred from an intuitive understanding of the theory behind the LMS algorithm, where elements with higher noise will be attenuated to reduce the mean squared error to the training signal.

This is further seen in Figures 82, 83, 84 where two elements with higher SNR (the red and blue weights) consistently maintain a higher relative magnitude over the other two elements after convergence.

Aside from oscillation of weight magnitudes during training, the aforementioned figures all show ratio convergence within 5000 snapshots, which translates to around 1 second - demonstrating that the algorithm implementation is suitable for adaptive beamforming in real-time.

8 Conclusion

8.1 Future Work

Several array performance measures[16] are detailed by Van Trees[1] in Chapter 2 of *Optimum Array Processing* and can be used to quantitatively compare theoretical and experimental results for the phased array experiment. However, this approach is not taken in this report due to the labour intensive method of measurement resulting in data with low angular resolution, as well as the large amounts of noise present in the system. If a more isolated setup with better means of measurement were used, it may be possible to collect data with higher resolution and less noise which could be quantitatively analyzed. Additionally, this data could be used to quantify the total power dissipated by the array at various steering angles through a numerical integration of the entire un-normalized radiation pattern, confirming or denying the hypothesis regarding peak power loss in Section 6.

Similarly, the effect of filter perturbations and array location perturbations can be modeled and analyzed mathematically to compare the behaviour of the beamforming system designed in this report. Assembly and functional tolerances of individual modules and the array mounting are fairly well known and can be treated as Gaussian random variables in an analysis that determines the sensitivity of array performance with respect to small changes in spacing, gain, and phase for an element.

In the area of digital receive beamforming, the performance of algorithms other than LMS can be investigated on the same datasets used in this report. This can include exploration into the RLS (Recursive Least Squares) and MVDR (Minimum Variance Distortionless Response) algorithms, considering the hardware structure is already complete, validated, and should support the sample rates required.

As opposed to processing data offline in MATLAB, performing beamforming live in the control module can be considered in future work. In order to meet the real time constraints of processing the data stream as it is sampled, an FPGA can be used to perform hardware acceleration of computations as well as support higher sample rates allowing for a wider baseband.

Lastly, the experiments detailed in this report were originally intended to be performed at 920MHz, but due to the limited amount of time and high amount of unknown factors in antenna design the first iteration antenna was used (optimal at 876MHz) as opposed to the third (920MHz). Although the relationships investigated hold regardless of frequency, it would be useful to perform measurements again on an antenna supporting 920MHz. This is fairly labourious however, and would require remounting and positioning of the antenna elements in the physical structure of the array to optimally support the new frequency.

8.2 Summary

This report documents the successful design, development and evaluation of a phase-shift beamforming and digital beamforming system. Beamforming theory is first discussed in a strictly mathematical sense, followed by simulations which verify the expected behaviour, before finally using the aforementioned system to evaluate the performance of phase-shift beamforming in an imperfect, real world scenario. The four-element array is shown to be able to steer the main lobe of the desired radiation pattern to 0° and 20° with an expected slight widening of half power bandwidth in the latter case. Analysis shows the system is able to steer to an angular accuracy of less than 1°, although due to a sub-optimal testing environment this was not able to be accurately measured.

Furthermore, mathematical derivation and simulation of receive beamforming using the LMS algorithm are examined and compared to results obtained using the implemented receiver array in an experimental setting. Although the experiment was not optimal and showed degradation from the simulated case in weight convergence, the SNR of received signals was still able to be improved by 10dB at 0° steering, and 5dB at -30° steering.

The system itself is described in detail, including the modelling and design of a circular patch antenna operational at 876MHz with a return loss of approximately 40dB, the design of RF electronic circuit board modules to perform RF/baseband operations, and the effect of mutual coupling on array elements. The phased-array (transmit) portion of the beamforming system is able to support phase shifts of accuracy within 0.5° and a gain error of within 0.5dB after calibration about a desired point, although uncalibrated operation is possible and yields an angular accuracy of within 5°. Additionally, the system is able to digitally synthesize baseband

[16] Directivity, Array Gain vs Spatially White Noise, Sensitivity and the Tolerance Factor

sinusoids for modulation at frequencies of up to 10kHz and sample four receiver I/Q channels effectively simultaneously at 4.28kHz. The effect of parameter variations in both the entire antenna array, and individual modules of the system is also discussed and examined. Lastly, the control and software components of the system are included to form a complete specification of the design.

9 References

[1] H. L. Van Trees, *Optimum Array Processing*, ser. Detection, Estimation, and Modulation Theory. John Wiley & Sons, 2004, ISBN: 0471463833. [Online]. Available: https://books.google.com.au/books/about/Optimum_Array_Processing.html?id=K5XJC_fMMAwC&source=kp_book_description&redir_esc=y.

[2] C. A. Balanis and P. I. Ioannides, *Introduction to smart antennas. [electronic resource]*. Ser. Synthesis lectures on antennas: #5. Morgan & Claypool Publishers, 2007, ISBN: 1598291777. [Online]. Available: https://ezp.lib.unimelb.edu.au/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=cat00006a&AN=melb.b5410656&site=eds-live&scope=site.

[3] B. Widrow, J. McCool, M. Larimore, and C. Johnson, "Stationary and nonstationary learning characteristics of the lms adaptive filter," *Proceedings of the IEEE*, vol. 64, pp. 1551–1162, 8 Aug. 1976. [Online]. Available: https://ieeexplore.ieee.org/document/1454555.

[4] J. Volakis, *Antenna engineering handbook. [electronic resource]*. Ser. McGraw-Hill's AccessEngineering. McGraw-Hill, 2007, ISBN: 0071546855. [Online]. Available: https://ezp.lib.unimelb.edu.au/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=cat00006a&AN=melb.b6455331&site=eds-live&scope=site.

[5] S. Drabowitch, *Modern antennas. [electronic resource]*. Springer, 2005, ISBN: 9780387262314. [Online]. Available: https://ezp.lib.unimelb.edu.au/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=cat00006a&AN=melb.b5615101&site=eds-live&scope=site.

[6] M. H. Ariff, M. Y. Hisyam, M. Z. Ibrahim, S. Khatun, I. Ismarani, and N. Shamsuddin, "Circular microstrip patch antenna for uhf rfid reader," *Journal of Telecommunication, Electronic and Computer Engineering*, vol. 10, no. 1-2, pp. 61–65, 2018, ISSN: 2180-1843. [Online]. Available: http://journal.utem.edu.my/index.php/jtec/article/view/3322.

[7] C. G. Christodoulou and P. F. Wahid, *Fundamentals of antennas. [electronic resource] : concepts and applications*. Ser. Tutorial texts in optical engineering: v. TT50. SPIE, 2001, ISBN: 9780819478887. [Online]. Available: https://ezp.lib.unimelb.edu.au/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=cat00006a&AN=melb.b5425022&site=eds-live&scope=site.

[8] *Ro4000 laminates*, Rogers Corporation, 2019. [Online]. Available: https://www.rogerscorp.com/acs/producttypes/9/RO4000-Laminates.aspx.

[9] *Minivna tiny*, mini Radio Solutions. [Online]. Available: http://miniradiosolutions.com/54-2/.

[10] *Simulated received plane waves*, Mathworks. [Online]. Available: https://au.mathworks.com/help/phased/ref/phased.ula.collectplanewave.html.

[11] W. L. Stutzman and G. A. Thiele, *Antenna Theory and Design*. John Wiley & Sons, 2012, ISBN: 978-0-470-57664-9. [Online]. Available: https://books.google.com.au/books/about/Antenna_Theory_and_Design.html?id=xhZRA1K57wIC&redir_esc=y.

[12] *Modeling mutual coupling in large arrays using embedded element pattern*, Mathworks. [Online]. Available: https://au.mathworks.com/help/phased/examples/modeling-mutual-coupling-in-large-arrays-using-embedded-element-pattern.html.

[13] D. Kelley, "Embedded element patterns and mutual impedance matrices in the terminated phased array environment," *IEEE Antennas and Propagation Society International Symposium*, Aug. 2005. [Online]. Available: https://ieeexplore-ieee-org.ezp.lib.unimelb.edu.au/document/1552340?arnumber=1552340&SID=EBSCO:edseee&fbclid=IwAR135CDATZklzx-dDhXCVorzW6GuNCYuqHcbauSNQQzcYOqeAeCisMsajIA.

[14] *Radiation pattern and phase of antenna or array*, Mathworks. [Online]. Available: https://au.mathworks.com/help/antenna/ref/pattern.html.

[15] *Simplelink mixed-signal microcontrollers*, MSP432P401, Texas Instruments, Sep. 2017. [Online]. Available: http://www.ti.com/lit/ds/symlink/msp432p401r.pdf.

[16] *High speed pcb layout techniques*, Texas Instruments. [Online]. Available: http://www.ti.com/lit/ml/slyp173/slyp173.pdf.

[17] A. Weller, A. Pakosta, and A. Verma, *High speed pcb layout guidelines*, Texas Instruments, Aug. 2017. [Online]. Available: http://www.ti.com/lit/an/scaa082a/scaa082a.pdf.

[18] K. Blattenberger, *Microstrip substrate equations formulas*. [Online]. Available: http://www.rfcafe.com/references/electrical/microstrip-eq.htm.

[19] *8/10/12-bit dual voltage output digital-to-analog converter with spi interface*, MCP4902, Microchip Technology, 2010. [Online]. Available: http://ww1.microchip.com/downloads/en/devicedoc/22250a.pdf.

[20] *250mhz differential twisted-pair drivers*, EL5171, Intersil Americas, 2015. [Online]. Available: https://www.renesas.com/sg/en/www/doc/datasheet/el5171-371.pdf.

[21] *620mhz – 1100mhz high linearity direct quadrature modulator*, LT5571, Linear Technology, 2006. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/5571f.pdf.

[22] *Digital controlled variable gain amplifier*, DVGA-242A+, Mini-Circuits. [Online]. Available: https://ww3.minicircuits.com/pdfs/DVGA1-242A+.pdf.

[23] *1.4nv/sqrt hz 180mhz filter building block*, LT1567, Linear Technology, 2001. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/1567fa.pdf.

[24] *800mhz to 1.5ghz direct conversion quadrature demodulator*, LT5516, Linear Technology, 2003. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/5516fa.pdf.

[25] *Monolithic amplifier*, PSA-5454+, Mini-Circuits. [Online]. Available: https://ww3.minicircuits.com/pdfs/PSA-5454+.pdf.

[26] A. Government, *Radiation protection series 3 - radiation protection standard for maximum exposure levels to radiofrequency fields - 3 khz to 300 ghz*, Australian Radiation Protection and Nuclear Safety Agency, May 2016. [Online]. Available: https://www.arpansa.gov.au/sites/default/files/legacy/pubs/rps/rps3.pdf.

[27] *Analog discovery 2: 100ms/s usb oscilloscope, logic analyzer and variable power supply*, AD2, Digilent, 2019. [Online]. Available: https://store.digilentinc.com/analog-discovery-2-100msps-usb-oscilloscope-logic-analyzer-and-variable-power-supply/.

10 Appendix

10.1 Transmit Chain DC Bias Measurements

DAC outputs at 2.5V	PSU at 5.32V for 5V on modules			
Power amplifier at default (max) gain				
	Chain 1	Chain 2	Chain 3	Chain 4
5V rail	4.98	4.99	5	5
Channel A	2.48	2.49	2.49	2.49
Channel B	2.49	2.49	2.49	2.49
Amplifier Bias	2.49	2.5	2.54	2.5
IQM I+	0.492	0.494	0.502	0.501
IQM I-	0.492	0.495	0.51	0.501
IQM Q+	0.495	0.495	0.51	0.498
IQM Q-	0.493	0.494	0.502	0.5

10.2 Transmit Chain Full-Circle Test Results (Chain 4)

Unit Circle Angle	I Voltage	Q Voltage	Observed Gain	Observed Phase	Phase Difference
0	3.50	2.50	-7.72	-5.5	
10	3.48	2.67	-7.8	4.73	10.23
20	3.44	2.84	-7.88	14.73	10
30	3.37	3.00	-7.89	24.95	10.22
40	3.27	3.14	-7.81	35.33	10.38
50	3.14	3.27	-7.75	45.35	10.02
60	3.00	3.37	-7.75	55.58	10.23
70	2.84	3.44	-7.65	65.56	9.98
80	2.67	3.48	-7.54	75.42	9.86
90	2.50	3.50	-7.39	85.39	9.97
100	2.33	3.48	-7.44	95.12	9.73
110	2.16	3.44	-7.45	104.83	9.71
120	2.00	3.37	-7.48	114.27	9.44
130	1.86	3.27	-7.42	123.95	9.68
140	1.73	3.14	-7.41	133.56	9.61
150	1.63	3.00	-7.47	143.37	9.81
160	1.56	2.84	-7.48	152.82	9.45
170	1.52	2.67	-7.52	162.35	9.53
180	1.50	2.50	-8.04	-368.36	-530.71
190	1.52	2.33	-8.12	-178.48	189.88
200	1.56	2.16	-8.06	-167.3	11.18
210	1.63	2.00	-8.04	-157.46	9.84
220	1.73	1.86	-7.94	-147.17	10.29
230	1.86	1.73	-7.99	-137.17	10
240	2.00	1.63	-8.08	-127.08	10.09
250	2.16	1.56	-8.1	-117.17	9.91
260	2.33	1.52	-8.12	-107.2	9.97
270	2.50	1.50	-8.1	-96.88	10.32
280	2.67	1.52	-8.17	-86.21	10.67
290	2.84	1.56	-8.16	-76.3	9.91
300	3.00	1.63	-8.56	-67.82	8.48
310	3.14	1.73	-8.53	-57.33	10.49
320	3.27	1.86	-8.55	-47.22	10.11
330	3.37	2.00	-8.58	-36.74	10.48
340	3.44	2.16	-8.55	-26.4	10.34
350	3.48	2.33	-8.45	-16.16	10.24
360	3.50	2.50	-8.33	-5.73	10.43

10.3 Transmit Chain Phase Shift Calibration Data

10.3.1 Chain 1

Steering Angle	Desired Phase	Observed Phase	Observed Gain	PA Attenuation	I Voltage	Q Voltage
0	0	0.02	-0.08	11.5	3.299	1.898
10	-46.885	-46.98	-0.17	11.5	2.637	1.509
20	-92.345	-92.45	0.14	11.5	1.866	1.727
30	-135	-135.00	0.46	11.5	1.866	1.727

10.3.2 Chain 2

Steering Angle	Desired Phase	Observed Phase	Observed Gain	PA Attenuation	I Voltage	Q Voltage
0	0	0.05	0.12	18.5	3.494	2.609
10	-15.628	-15.45	-0.16	18.5	3.488	2.347
20	-30.782	-30.72	-0.23	18.5	3.419	2.105
30	-45	-45.10	-0.23	18.5	3.298	1.897

10.3.3 Chain 3

Steering Angle	Desired Phase	Observed Phase	Observed Gain	PA Attenuation	I Voltage	Q Voltage
0	0	0.01	0.14	17	2.234	1.536
10	15.628	15.53	0.07	17	2.504	1.5
20	30.782	30.82	-0.03	17	2.771	1.537
30	45	45.03	-0.05	17	2.997	1.632

10.3.4 Chain 4

Steering Angle	Desired Phase	Observed Phase	Observed Gain	PA Attenuation	I Voltage	Q Voltage
0	0	0.050	0.240	21	3.485	2.67
10	46.885	46.74	0.26	21	3.064	3.326
20	92.345	92.46	0.63	21	2.29	3.478
30	135	135.02	0.58	21	1.668	3.055

10.4 Software

`https://github.com/dingbenjamin/beamformer`

10.5 Antenna Design and Simulation

```
simulate_antenna = true
generate_gerbers = false

clf
% Array design parameters
fc = 920e6;
vp = physconst('lightspeed');
lambda = vp/fc;
N = 4;
spacing = lambda/2;

% Antenna design parameters
w1_w3_tuning_param = 3e-3;

a = 0.0407;
w = 3e-3;
```

```
l = 13e-3;
Lg = 120e-3;
Wg = 120e-3;
W1 = w1_w3_tuning_param;
L1 = l;
W2 = l;
L2 = w;
W3 = w1_w3_tuning_param;
L3 = l;
W4 = l;
L4 = w;
Wn = 3e-3;
Ln = 7e-3;
centreNotchRX = a-Wn/2;
centreNotchLX = -(a-Wn/2);
FeedX = (W1/2+W2)*cos(deg2rad(45)); % x^2+y^2=W1/2+W2
FeedY = -(W1/2+W2)*sin(deg2rad(45));

% Create patch
p = antenna.Circle('Radius', a, 'Center',[0 0]);
sloty = antenna.Rectangle('Length', L3+L2+L1, 'Width', W3, ...
    'Center', [0 0]);
slotx = antenna.Rectangle('Length', L2, 'Width', W4+W1+W2, ...
    'Center', [0 0]);
notchL = antenna.Rectangle('Length', Wn, 'Width', Ln, ...
    'Center', [centreNotchLX 0]);
notchR = antenna.Rectangle('Length', Wn, 'Width', Ln, ...
    'Center', [centreNotchRX 0]);
patch = p-sloty-slotx-notchL-notchR;

% Create ground plane
gnd = antenna.Rectangle('Length',Lg,'Width',Wg, 'Center', [0 0]);

% PCB parameters
pcbThickness = 1.6e-3;
pcbMaterial = 'FR4';
pcbEpsilonR = 4.5

% Create stack
d = dielectric(pcbMaterial);
d.EpsilonR = pcbEpsilonR;
d.Thickness = pcbThickness;
circular_patch = pcbStack;
circular_patch.Name = 'Small patch';
circular_patch.BoardShape = gnd;
circular_patch.BoardThickness = pcbThickness;
circular_patch.Layers = {patch,d,gnd};
circular_patch.FeedLocations = [FeedX FeedY 1 3];
circular_patch.FeedDiameter = 1.27e-3;

circular_patch.ViaDiameter = circular_patch.FeedDiameter

% ------------------------------------------------
% Simulate antenna
% ------------------------------------------------
if simulate_antenna
    circular_patch.Tilt = 90;
    circular_patch.TiltAxis = [0 1 0];
    figure(1)
    show(circular_patch)
    p = circular_patch

    figure(2)
    pattern(p, fc)

    fRange = 850e6:0.5e6:980e6;
    connectorZ = 50;

    figure(3)
    impedance(p,fRange)

    figure(4)
    returnLoss(p,fRange,connectorZ)
end

% ------------------------------------------------
```

```
% Generate gerbers
% ------------------------------------------------------
if generate_gerbers
    circular_patch.ViaLocations = [FeedX FeedY 1 3; ...
                                   0.065 0.065 1 3; ...
                                   0.065 -0.065 1 3; ...
                                   -0.065 0.065 1 3; ...
                                   -0.065 -0.065 1 3];

    C = sma_jack_cinch
    W = PCBServices.PCBWayWriter
    W.Filename = 'antenna_design_file_v3';
    W.DefaultViaDiam = circular_patch.ViaDiameter
    PW = PCBWriter(circular_patch,W,C)
    PW.Solderpaste = 0;
    PW.Soldermask = "bottom";
    PW.PCBMargin = 0.01;
    gerberWrite(PW)
end
```

10.5.1 Beamforming simulations

10.5.1.1 Number of elements

```
fc = 920e6;
wavelength = physconst('LightSpeed')/fc;
spacing = wavelength/2;

N1 = 3;
N2 = 4;
N3 = 5;
N4 = 10;

% ------------------------------------------------------
% Antenna array creation
% ------------------------------------------------------
arr1 = phased.ULA;
arr1.NumElements = N1;
arr1.ElementSpacing = spacing;

arr2 = phased.ULA;
arr2.NumElements = N2;
arr2.ElementSpacing = spacing;

arr3 = phased.ULA;
arr3.NumElements = N3;
arr3.ElementSpacing = spacing;

arr4 = phased.ULA;
arr4.NumElements = N4;
arr4.ElementSpacing = spacing;

% Steering vectors
az = 60;
el = 0;

steervec1 = phased.SteeringVector('SensorArray',arr1);
sv1 = 1/N1*steervec1(fc, [az; el]);

steervec2 = phased.SteeringVector('SensorArray',arr2);
sv2 = 1/N2*steervec2(fc, [az; el]);

steervec3 = phased.SteeringVector('SensorArray',arr3);
sv3 = 1/N3*steervec3(fc, [az; el]);

steervec4 = phased.SteeringVector('SensorArray',arr4);
sv4 = 1/N4*steervec4(fc, [az; el]);

% Patterns
figure(1)
patternAzimuth(arr1,fc,'Weights',sv1,'Type','powerdb',...
'PropagationSpeed',physconst('LightSpeed'));

figure(2)
patternAzimuth(arr2,fc,'Weights',sv2,'Type','powerdb',...
```

```
'PropagationSpeed',physconst('LightSpeed'));

figure(3)
patternAzimuth(arr3,fc,'Weights',sv3,'Type','powerdb',...
'PropagationSpeed',physconst('LightSpeed'));

figure(4)
patternAzimuth(arr4,fc,'Weights',sv4,'Type','powerdb',...
'PropagationSpeed',physconst('LightSpeed'));
```

10.5.1.2 Spacing

```
fc = 920e6;
wavelength = physconst('LightSpeed')/fc;
N = 4;

spacing1 = wavelength/4;
spacing2 = wavelength/2;
spacing3 = 3*wavelength/4;


% ------------------------------------------------------
% Antenna array creation
% ------------------------------------------------------
arr1 = phased.ULA;
arr1.NumElements = N;
arr1.ElementSpacing = spacing1;

arr2 = phased.ULA;
arr2.NumElements = N;
arr2.ElementSpacing = spacing2;

arr3 = phased.ULA;
arr3.NumElements = N;
arr3.ElementSpacing = spacing3;

% Steering vectors
az = 30;
el = 0;

steervec1 = phased.SteeringVector('SensorArray',arr1);
sv1 = 1/N*steervec1(fc, [az; el]);

steervec2 = phased.SteeringVector('SensorArray',arr2);
sv2 = 1/N*steervec2(fc, [az; el]);

steervec3 = phased.SteeringVector('SensorArray',arr3);
sv3 = 1/N*steervec3(fc, [az; el]);

% Patterns
figure(1)
patternAzimuth(arr1,fc,'Weights',sv1,'Type','powerdb',...
'PropagationSpeed',physconst('LightSpeed'));

figure(2)
patternAzimuth(arr2,fc,'Weights',sv2,'Type','powerdb',...
'PropagationSpeed',physconst('LightSpeed'));

figure(3)
patternAzimuth(arr3,fc,'Weights',sv3,'Type','powerdb',...
'PropagationSpeed',physconst('LightSpeed'));
```

10.5.1.3 Window weights

```
close all
fc = 920e6;
wavelength = physconst('LightSpeed')/fc;
N = 10;

spacing = wavelength/2;


% ------------------------------------------------------
% Antenna array creation
% ------------------------------------------------------
```

```
arr1 = phased.ULA;
arr1.NumElements = N;
arr1.ElementSpacing = spacing;

% Steering vectors
az = 0;
el = 0;

steervec1 = phased.SteeringVector('SensorArray',arr1);
sv1 = 1/N*steervec1(fc, [az; el]);

% Weights
hann = hanning(N);
hamm = hamming(N);
kai  = kaiser(N, 3);

figure(5)
stem(hann)
hold on
stem(hamm)
hold on
stem(kai)
legend("Hanning", "Hamming", "Kaiser (\beta = 3)")
xlabel("Index")
ylabel("Weight")
title("Window Weights")

% Patterns
figure(1)
patternAzimuth(arr1,fc,'Weights',sv1,'Type','powerdb',...
'PropagationSpeed',physconst('LightSpeed'));

figure(2)
patternAzimuth(arr1,fc,'Weights',sv1.*hann,'Type','powerdb',...
'PropagationSpeed',physconst('LightSpeed'));

figure(3)
patternAzimuth(arr1,fc,'Weights',sv1.*hamm,'Type','powerdb',...
'PropagationSpeed',physconst('LightSpeed'));

figure(4)
patternAzimuth(arr1,fc,'Weights',sv1.*kai,'Type','powerdb',...
'PropagationSpeed',physconst('LightSpeed'));
```

10.5.1.4 LMS algorithm

```
% ------------------------------------------------------
% Array parameters
% ------------------------------------------------------
fc = 920e6;
vp = physconst('lightspeed');
wavelength = vp/fc;
spacing = wavelength/2;
N = 4;
c = physconst('LightSpeed');
rng('default');
array = phased.ULA('NumElements',N,'ElementSpacing',spacing);

% ------------------------------------------------------
% Signal parameters
% ------------------------------------------------------
fr = 10.0e3;
fs = 5*fr;
sample_period = 1/fs;

% Becareful with these parameters, a directional antenna may result in
% these angles point to a null.
steering_el= 0;
steering_az = -20;
incidentAngle = [steering_az;steering_el];

noise_power = 0.0001;
signal_amplitude = 0.01;
cw = cell(1,length(0:1/fs:1));
j=1;
step_size = [1e-7 1e-6 1e-5 5e-5 1e-4 5e-4 1e-3 5e-3 1e-2 5e-2 1e-1 5e-1 1 1e2 1e3 1e4 0.5];
```

```
for k = step_size
    w = zeros(N,1);
    t = 0:1/(5*fr):1;
    xm = signal_amplitude*cos(2*pi*fr*t');
    x = collectPlaneWave(array,xm,incidentAngle,fc,c);
    noise = noise_power*randn(size(x)) + noise_power*1i*randn(size(x));
    rx = x + noise;
    mu = k;
    x_k = transpose(rx); % get sample in required form
    for i = 1:length(t)
        if mod(i,100) == 0
            disp("Snapshot processed: "+num2str(i))
        end
        %--------------------------------------------------------
        % LMS Algorithm
        %--------------------------------------------------------
        y = ctranspose(x_k(:,i))*w;
        yy(i) = y;
        e = conj(100*xm(i)) - y;
        w = w + mu*x_k(:,i)*e;
        cw{i} = w;

        xk = x_k(:,i);
    end
    y_est = ctranspose(x_k)*w;
    mse(j) = mean(abs(xm-y_est).^2);
    j=j+1;
end
figure(1)
semilogx(step_size, mse)
xlabel("Step size")
ylabel("Mean squared error")
title("Comparison of mean squared error for different step sizes")

ffty = fftshift(fft(y_est));
nffty = 10*log10(abs(ffty)/max(abs(ffty)));
fs = 5*fr;
freqy=linspace(-1-2/fs,1,length(ffty))*fs/2;
figure(2)
plot(freqy, nffty)
xlabel("Frequency (Hz)")
ylabel("Magnitude (dB)")
title("Spectrum of estimated calibration signal")
axis([-25e3 25e3 -20 0])

figure;
fftrx = fftshift(fft(rx(:,1)));
nfftrx = 10*log10(abs(fftrx)/max(abs(fftrx)));
figure(3)
plot(freqy, nfftrx);
xlabel("Frequency (Hz)")
ylabel("Magnitude (dB)")
title("Typical spectrum of the signal received by an array element")
axis([-25000 25000 -20 0])

figure(4)
patternAzimuth(array, fc, 'Weights',w, 'Type', 'powerdb')

mw = cell2mat(cw);
figure(5)
figure(6)
for i = 1:N
    figure(5)
    hold on
    plot(abs(mw(i,:)))
    figure(6)
    hold on
    plot(angle(mw(i,:)))
end
figure(5)
xlabel("Number of snapshots")
ylabel("Magnitude")
title("Convergence of weight magnitudes (\mu = 10)")
legend("w_1", "w_2", "w_3", "w_4")

figure(6)
```

```
xlabel("Number of snapshots")
ylabel("Phase (rad)")
title("Convergence of weight phases (\mu = 10)")
legend("w_1", "w_2", "w_3", "w_4")
axis([0 10000 -2 2])
```

10.5.1.5 Mutual Coupling Patterns

```
%Modelling Mutual Coupling Using Terminated Arrays

clear all
close all
% Antenna Parameter
w1_w3_tuning_param = 3e-3;
a = 0.0407;
w = 3e-3;
l = 13e-3;
Lg = 120e-3;
Wg = 120e-3;
W1 = w1_w3_tuning_param;
L1 = l;
W2 = l;
L2 = w;
W3 = w1_w3_tuning_param;
L3 = l;
W4 = l;
L4 = w;
Wn = 3e-3;
Ln = 7e-3;
centreNotchRX = a-Wn/2;
centreNotchLX = -(a-Wn/2);
FeedX = (W1/2+W2)*cos(deg2rad(45)); % x^2+y^2=W1/2+W2
FeedY = -(W1/2+W2)*sin(deg2rad(45));

% PCB parameters
pcbThickness = 1.6e-3;
pcbMaterial = 'FR4';
pcbEpsilonR = 4.5;

% Circular Patch Antenna
cp = patchMicrostripCircular('Radius',a,'Height',pcbThickness,...
        'GroundPlaneLength',0.16,'GroundPlaneWidth',0.16,...
        'FeedOffset',[FeedX FeedY]);
cp.Substrate.Name = 'FR4';
cp.Substrate.EpsilonR=pcbEpsilonR;
cp.Substrate.Thickness=pcbThickness;

% Show the antenna
figure;
show(cp);

% Show pattern of the single antenna
fc = 920e6;
figure
pattern(cp,fc)

vp = physconst('lightspeed');
lambda = vp/fc;

% Terminated impedance
Zinf = 50;
% Centre element
ElemCenter = 2;
az = -180:2:180;
el = -180:2:180;

% Show terminated pattern
for n=[2,3,4]

    fullWaveArray = linearArray( ...
        'NumElements', n,...
        'ElementSpacing', lambda/2);
    fullWaveArray.Element = cp;

    figure
    pattern(fullWaveArray,fc,az,el,...
```

```
        'ElementNumber',ElemCenter,'Termination',real(Zinf));

    figure
    pattern(fullWaveArray,fc,0,el,...
        'ElementNumber',ElemCenter,'Termination',real(Zinf));


    figure
    pattern(fullWaveArray,fc,az,0,...
        'ElementNumber',ElemCenter,'Termination',real(Zinf));

end

% Show the edge effect
fullWaveArray = linearArray( ...
    'NumElements', N,...
    'ElementSpacing', lambda/2);
fullWaveArray.Element = cp;

figure
pattern(fullWaveArray,fc,az,el,...
    'ElementNumber',1,'Termination',real(Zinf));
```

10.5.1.6 Mutual Coupling Custom Array Scans

```
clear all
close all

% Antenna Parameters
w1_w3_tuning_param = 3e-3;
a = 0.0407;
w = 3e-3;
l = 13e-3;
Lg = 120e-3;
Wg = 120e-3;
W1 = w1_w3_tuning_param;
L1 = l;
W2 = l;
L2 = w;
W3 = w1_w3_tuning_param;
L3 = l;
W4 = l;
L4 = w;
Wn = 3e-3;
Ln = 7e-3;
centreNotchRX = a-Wn/2;
centreNotchLX = -(a-Wn/2);
FeedX = (W1/2+W2)*cos(deg2rad(45)); % x^2+y^2=W1/2+W2
FeedY = -(W1/2+W2)*sin(deg2rad(45));

% PCB parameters
pcbThickness = 1.6e-3;
pcbMaterial = 'FR4';
pcbEpsilonR = 4.5;

% Circular Patch Antenna
cp = patchMicrostripCircular('Radius',a,'Height',pcbThickness,...
        'GroundPlaneLength',0.16,'GroundPlaneWidth',0.16,...
        'FeedOffset',[FeedX FeedY]);
cp.Substrate.Name = 'FR4';
cp.Substrate.EpsilonR=pcbEpsilonR;
cp.Substrate.Thickness=pcbThickness;

% Show the antenna
%figure;
%show(cp);

% Show pattern of the single antenna
fc = 920e6;
%figure
%pattern(cp,fc)

vp = physconst('lightspeed');
lambda = vp/fc;

% Terminated impedance
```

```
Zinf = 50;
% Centre element
ElemCenter = 2;
az = -180:2:180;
el = -180:2:180;
% Terminated Array (3 Elements)
fullWaveArray = linearArray( ...
        'NumElements', 3,...
        'ElementSpacing', lambda/2);
fullWaveArray.Element = cp;
fullWaveArray.TiltAxis = [0 1 0;1 0 0];
fullWaveArray.Tilt = [90 90];

el = -90:2:90;
EmbElFieldPatCenter = pattern(fullWaveArray,fc,az,el,...
    'ElementNumber',ElemCenter,'Termination',real(Zinf),'Type','efield');

embpattern=mag2db(EmbElFieldPatCenter);

fRange = 850e6:2.5e6:980e6;
fmin=min(fRange);
fmax=max(fRange);
freqVector = [fmin fmax];
% Custom Antenna
embantenna = phased.CustomAntennaElement('FrequencyVector',freqVector,...
    'AzimuthAngles',az,'ElevationAngles',el,...
    'MagnitudePattern',embpattern,'PhasePattern',zeros(size(embpattern)));
% Array with Mutual Coupling
embeddedULA = phased.ULA;
embeddedULA.Element = embantenna;
embeddedULA.NumElements = 4;
embeddedULA.ElementSpacing = lambda/2;

figure
pattern(embeddedULA,fc)
eplane_indx = find(az==0);
scan_az1 = [-20 0];
scan_el1 = zeros(1,numel(scan_az1));
scanHplane = [scan_az1;scan_el1];

% compute array scanning weights
steeringvec = phased.SteeringVector('SensorArray',embeddedULA,...
    'IncludeElementResponse',true);
weights = steeringvec(fc,scanHplane);

% Steered pattern
%figure;
%pattern(embeddedULA,fc,az,0,'Weights',weights(:,1));

% array scanning
legend_string1 = cell(1,numel(scan_az1));
scanHPat = nan(numel(az/2),numel(scan_az1));
for i = 1:numel(scan_az1)
    scanHPat(:,i) = pattern(embeddedULA,fc,az/2,scan_el1(i),...
        'Weights',weights(:,i));
    legend_string1{i} = strcat('scan = ',num2str(scan_az1(i)));
end

% Normal ULA (Without MUtual Coupling) for comparison
ULA = phased.ULA;
cp.Tilt = 90;
cp.TiltAxis = [0 1 0];
ULA.Element = cp;
ULA.NumElements = 4;
ULA.ElementSpacing = lambda/2;

figure;
polarplot(deg2rad(az'/2)+pi/2,scanHPat(:,1))
rlim([-35 10])
thetalim([0 180])

p=pattern(ULA,fc,az/2,0,'Weights',weights(:,1));
hold on;
polarplot(deg2rad(az'/2)+pi/2,p)
rlim([-35 10])
legend('Mutual Coupling','ULA','Location','best')
```

```
128
129 figure;
130 polarplot(deg2rad(az'/2)+pi/2,scanHPat(:,2))
131 rlim([-35 10])
132 thetalim([0 180])
133
134 p=pattern(ULA,fc,az/2,0,'Weights',weights(:,2));
135 hold on;
136 polarplot(deg2rad(az'/2)+pi/2,p)
137 rlim([-35 10])
138 legend('Mutual Coupling','ULA','Location','best')
```