Implementation of a Low Cost Demonstrator HF Riometer on a Flexible FPGA Backend

The First Steps in Adding a Riometer Mode to the Digital SuperDARN Radar at SANAE IV

Presented by:
Thomas Düsterwald

Supervisor:
D. W. O'Hagan

Dissertation submitted to the Department of Electrical Engineering at the University of Cape Town in partial fulfilment of the academic requirements for an MSc in Radar and Electronic Defence

January 7, 2018

Keywords: Riometer, SuperDARN, FPGA, Red Pitaya

Declaration

I know the meaning of plagiarism and declare that all the work in the document, save for that which is properly acknowledged, is my own. This thesis/dissertation has been submitted to the Turnitin module (or equivalent similarity and originality checking software) and I confirm that my supervisor has seen my report and any concerns revealed by such have been resolved with my supervisor.

Signature:.. Signed by candidate

Thomas Düsterwald

Date:.. 25 January 2018

Acknowledgments

SANSA for funding me in this project.

Professor Daniel O'Hagan for his advice and guidance in this project.

Professor Michael Kosch for his stellar introduction to the ionosphere and riometry.

Jon Ward for his many hours helping me set up and take down my riometer and providing useful insight into the front-end of my system.

Tighe for his help with programming the Red Pitaya and always being willing to lend a hand.

Skippy for the many hours he spent helping me with filters, antennas and all things hardware related.

My family for supporting me throughout this project and always providing sound advice.

Golesworthy for providing his proof reading services and lending a thoroughly scientific eye to my work.

Finally, to my girlfriend Caitlyn, for not only finding the time to proof read my entire dissertation, but also sharing endless cups of tea, feeding me too many meals to count and always being there to support me.

Abstract

SuperDARN is an international network of 35 HF radars located near the poles of the Earth dedicated to determining the state of the ionosphere at high latitudes. One of the SuperDARN radars is located at South Africa's base in Antarctica (SANAE IV) and is administered by the South African National Space Association (SANSA). The radar at SANAE IV was recently upgraded to a fully digital transceiver, with the addition of a Field Programmable Gate Array (FPGA) at the core of this upgrade. FPGAs allow for easy hardware reconfiguration and high performance computing.

The aim of this project is to determine the feasibility of using the FPGA on board the radar at SANAE IV to implement a riometer mode to run simultaneously with its main mode of operation, adding a new tool to the radar's set of abilities without any investment in new hardware. The riometer function could easily be ported to other radars in the SuperDARN network, allowing for a significant increase in riometer coverage of the polar regions. As a first step towards achieving this goal, a demonstrator riometer is developed using the Red Pitaya FPGA platform as its backend, and tested at the University of Cape Town, at SANSA in Hermanus and at Fish Hoek.

A riometer measures the opacity of the ionosphere with respect to cosmic radiation. Doing this over a wide band of frequencies results in a spectral riometer. This dissertation describes the design and implementation of both a single frequency and a spectral riometer, both implemented on the Red Pitaya, and the results of testing these implementations. Experimentation alongside a La Jolla $38\,MHz$ riometer revealed very similar performance for the low cost demonstrator riometer. It is thereby shown that low cost HF riometry is possible and that it is feasible to implement a riometer on the radar at SANAE IV.*

*However, an additional FPGA is required.

Contents

List of Figures

List of Tables

Glossary

ADC	Analogue to Digital Converter
AXI	Advanced eXtensible Interface
CIC	Cascaded Integrator-Comb
CME	Coronal Mass Ejection
DAC	Digital to Analogue Converter
DSP	Digital Signal Processing
EM	Electromagnetic
FFT	Fast Fourier Transform
FIFO	First In, First Out
FIR	Finite Impulse Response
FOV	Field of View
FPGA	Field Programmable Gate Array
HF	High Frequency
HDL	Hardware Description Language
IRIS	Imaging Riometer for Ionospheric Studies
I/Q	In-phase/Quadrature
LPDA	Log Periodic Dipole Array
MHD	Magneto Hydrodynamic
OTH	Over-the-Horizon
PCA	Polar Cap Absorption
PSD	Power Spectral Density
QDC	Quiet Day Curve
RBW	Resolution Bandwidth
Riometer	Relative Ionospheric Opacity Metre
RF	Radio Frequency
ROS	Radar Operating System
SANAE	South African National Antarctic Expedition
SANSA	South African National Space Agency
SDR	Software Defined Radio
SOC	System on a Chip
SuperDARN	Super Dual Auroral Radar Network
ST	Sidereal Time
TIGER	Tasman International Geospace Environment Radar
TTFD	Twin Terminated Folded Dipole
UHF	Ultra High Frequency
VBW	Video Bandwidth
VHF	Very High Frequency

Chapter 1

Introduction

In 1997 the South African National Space Agency (SANSA) completed the construction of an HF radar as part of the Super Dual Auroral Radar Network (SuperDARN) at the South African research base, SANAE IV, in Antarctica [1]. Over the years the radar has been upgraded with new technology, including, in 2013, an upgrade to a fully digital transceiver, sampling at the RF frequency (125 MHz) and equipped with a reconfigurable Field Programmable Gate Array (FPGA).

This project is concerned with the viability of using the reconfigurable FPGA backend to allow the existing radar hardware to be used as a relative ionospheric opacity meter, or riometer. The main benefit of the reconfigurable FPGA-based upgrade is that it permits SuperDARN hardware to perform different scientific function (eg. riometry) with minimal system changes required. Therefore, it is both cost effective and allows for the possibility of implementing the changes on other SuperDARN radars in the network to greatly increase riometry coverage of the high latitudes.

Since the project was conducted in Cape Town and not Antarctica, the work focusses on the design and implementation of a demonstrator system on the low cost Red Pitaya development platform [2]. The Red Pitaya is a device featuring an integrated processor and FPGA, along with ADC and DAC capabilities. The successful implementation of a riometry system on the Red Pitaya would provide a strong indication that it is possible on the SANAE IV radar system. It also demonstrates the ability to make cost effective HF devices using ever cheaper FPGA and System on a Chip (SOC) technology.

The introduction starts by giving a background to the study, including details about SuperDARN and the radar at SANAE IV. Thereafter the objectives of the study are evaluated and the hypothesis presented. Scope and limitations follow this before concluding with the apparatus and tools used. Finally the plan of development for the rest of the dissertation is presented.

1.1 Background to the Study

The areas covered in the background to this study are an overview of the SuperDARN network and then a more in-depth study of the SuperDARN radar at the South African Antarctic research base SANAE IV. The section concludes by introducing the demonstrator system that has been developed for this project.

1.1.1 Introduction to SuperDARN

The Super Dual Auroral Radar Network (SuperDARN) is an international network of HF radars capable of sensing backscatter from ionospheric irregularities. These radars are located at high latitudes, near the poles, and are able to detect irregularities in the E and F regions of the ionosphere. The network, at the time of writing, consists of 35 radars, with the majority of the radars residing in the Northern hemisphere [3].

SuperDARN radars are designed to detect coherent backscatter due to Bragg scattering (constructive reflections from regular, periodic structures), from small scale electron density irregularities in the ionosphere. SuperDARN radars are placed at high latitudes since ionospheric irregularities tend to align with the magnetic field of the Earth and thus backscatter only occurs when the incident wave is orthogonal to the magnetic field direction of the Earth. At mid-latitudes the orthogonality constraint can be met with VHF and UHF radars. At high latitudes, however, the orthogonality constraint requires that the transmitted radio wave be refracted enough by the ionosphere to be orthogonal to the almost vertical magnetic field. Only HF waves can be refracted enough to meet this orthogonality condition, as Figure 1.1 demonstrates.

In principle, SuperDARN radars operate in pairs with each pair covering an overlapping field of view (FOV). The area covered by the network of radars is shown in Figure 1.2. Overlapping the FOV allows for 2D Doppler determination which is important in producing accurate plasma convection maps of the ionosphere, the primary output product that SuperDARN generates. SuperDARN radars operate as monostatic, pulsed radars with a FOV of 52 degrees azimuth over 16 beams [1]. They operate over a wide range of HF frequencies, depending on the prevailing conditions. Usually this covers a frequency range of 8-20 MHz. A sequence of 7 pulses at varying PRFs is sent out in order to address the range-doppler ambiguity problem and give unambiguous Doppler velocities upwards of 300 m/s [1].

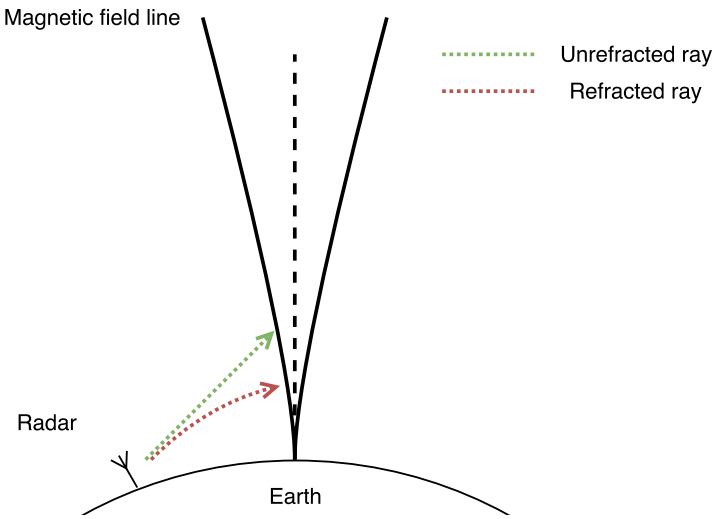


Figure 1.1: The HF ray (red) is refracted by a sufficient amount to be orthogonal to the near vertical magnetic field line at the poles, allowing for backscatter from ionospheric irregularities. At higher frequencies (green) the ray does not refract and cannot meet the orthogonality condition.

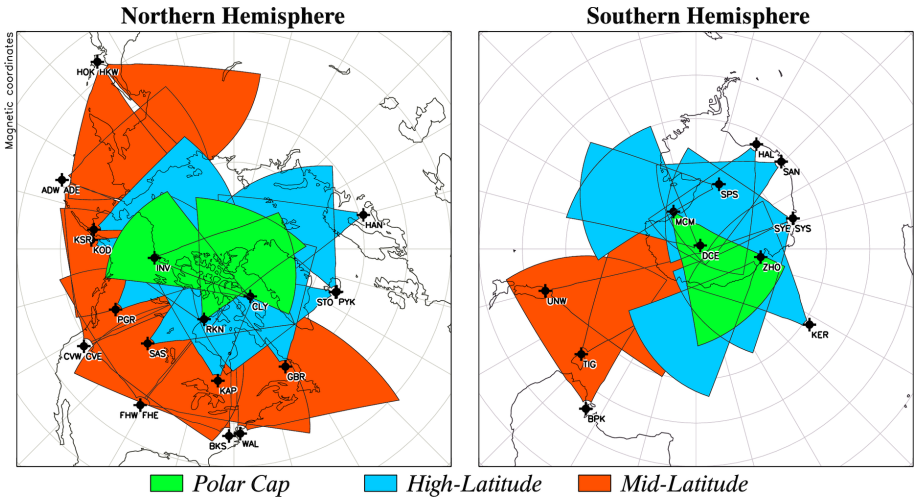


Figure 1.2: Area covered by SuperDARN radars [3].

An example of the type of plasma convection map generated by SuperDARN is shown in Figure 1.3. These convection maps can help researchers study large scale dynamical processes in the magnetosphere-ionosphere system [4].

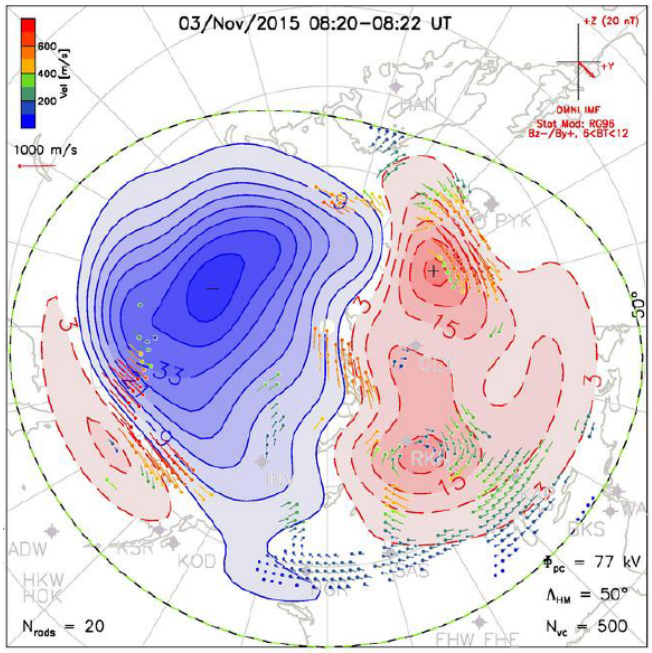


Figure 1.3: Example of a convection map produced by SuperDARN [5].

The main scientific objectives of SuperDARN are to study [4]:

- Convection structure: provide a global-scale view of the configuration of plasma convection in the high-latitude ionosphere.
- Convection dynamics: provide a global-scale view of the dynamics of plasma convection in the high-latitude ionosphere.
- Remote sensing of magnetic reconnection.
- Substorms: test the various theories of polar cap expansion and contraction under changing IMF conditions and observe the large-scale response of the nightside convection pattern to substorms. Substorms are a major contributor to large-scale magnetosphere-ionosphere dynamics. They comprise a cycle of energy storage and release. The substorm cycle consists of growth, expansion and recovery.
- Magneto hydrodynamic (MHD) waves: measure the energy influx from MHD waves on a larger spatial scale than previously possible.

Studies on gravity waves, planetary waves, and meteor scatter and mesospheric winds are also performed using SuperDARN data. The addition of a riometer function to SuperDARN radars would help researchers gain further insight into the aforementioned scientific objectives and expand into other previously unexplored areas.

1.1.2 The SuperDARN Radar at SANAE IV

South Africa's SuperDARN radar came online in 1997 [1]. It was built at the South African research base in Antarctica, SANAE IV. The antenna consists of 16 Twin Terminated Folded Dipoles (TTFD) in the main antenna array and 4 further dipoles 100 m further back to allow for interferometry applications [6]. The antennas feature a log-periodic design for wide band operation, nominally from 8-20 MHz. The two sets of antennas can be seen in Figure 1.4 in front of the SANAE IV base. Recently it was upgraded to allow for fully digital operation with sampling taking place at each dipole. The majority of the front-end processing takes place on a Virtex-5 FPGA. The main reason for this upgrade is the extra flexibility that a digital front-end provides, but it also features increased range as compared to its analogue predecessor [7].

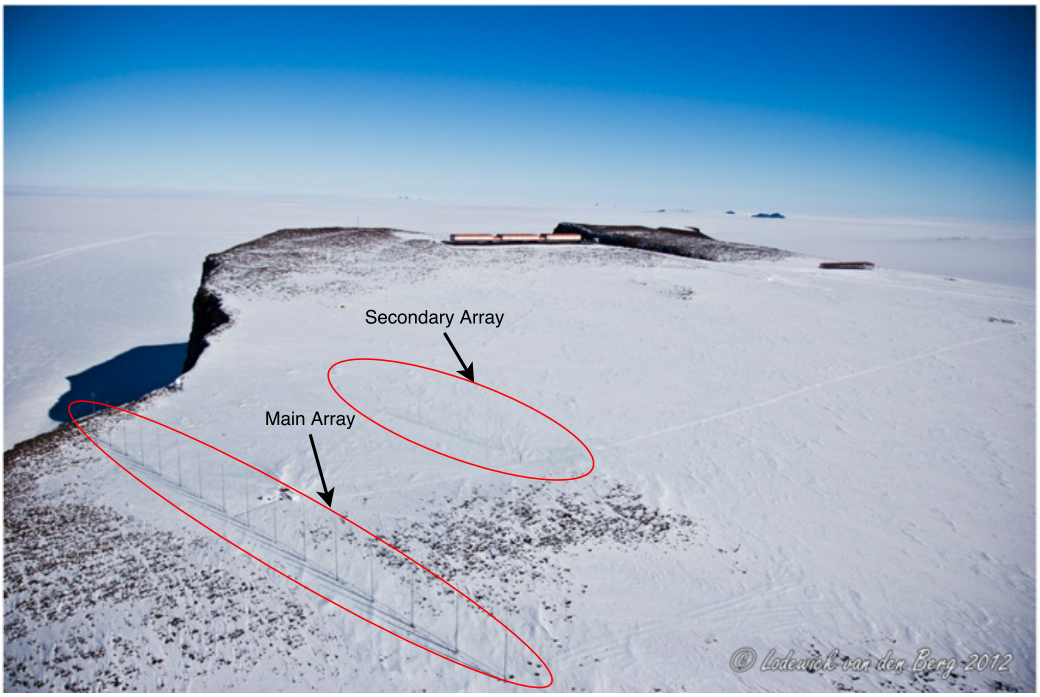


Figure 1.4: The SuperDARN radar in the foreground with SANAE IV in the distance [6].

A digital SuperDARN, or 'digidarn' had already been motivated for in 2006 by Parkinson *et al.* as it could provide consistent output data products across the network, extended coverage through time and frequency multiplexing and adaptive beam-forming [8]. This in addition to the aforementioned flexibility that comes with using an FPGA. La Trobe University in Australia took up this challenge and upgraded their SuperDARN radar at Buckland park to a digital transceiver, the TIGER-3 [9]. The South African upgrade was based on La Trobe University's work. Each of the 16 antennas has an FPGA based transceiver sampling at 125 MHz, to record the RF frequency. I/Q down-conversion takes place digitally inside the FPGA. Figure 1.5 shows an overview of the TIGER-3. The antennas are sampled directly by the transceivers after initially filtering to the frequency of interest: 8-30 MHz, and providing some amplification. After I/Q down-conversion the data from each transceiver is transferred to a central server running the Radar Operating System (ROS), which produces the standard outputs

of a SuperDARN radar.

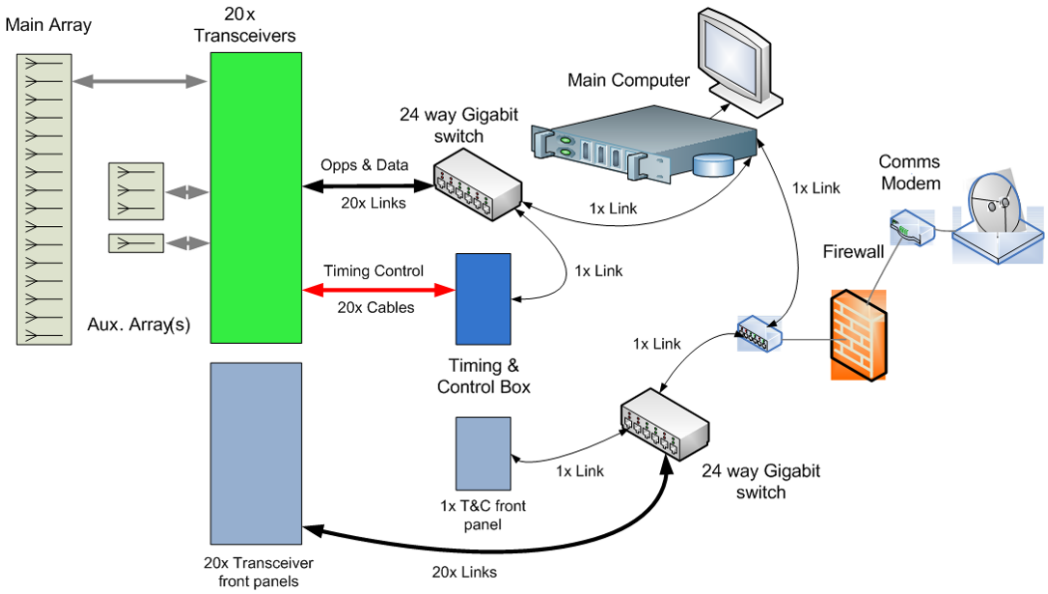


Figure 1.5: An overview of the TIGER-3 SuperDARN radar system [7].

Each FPGA can be reconfigured via Ethernet in a matter of minutes allowing for extremely flexible operation. In fact, La Trobe University prefers to refer to the TIGER-3 as a general purpose HF system, rather than a dedicated SuperDARN radar.

1.1.3 Demonstrator System

While a working riometer would be implemented on the radar at SANAE IV, in practice this is not possible due to time and distance constraints. Instead a demonstration riometer has been assembled at the University of Cape Town (UCT) with the aim of showing that digital HF riometry is possible. The Red Pitaya was chosen as the ideal prototyping device as it is both an FPGA and processing System on a Chip (SOC). The front-end and dipole antennas used for testing were also designed and built at UCT.

1.2 Objectives of this Study

The objectives of this study are presented in two parts: the areas that are investigated in this study, including the hypothesis, and the motivation for the study.

1.2.1 Areas to be Investigated

The main hypothesis in this research is that it is possible to create a digital HF riometer that samples at the RF. An addendum to this hypothesis is that it is also possible to simultaneously operate a digital HF radar system and riometer, as will be required for the SuperDARN radar at SANAE IV.

In order to explore fully this hypothesis and its addendum, the following are key objectives:

- A review of literature critically evaluating the work done on riometry.
- A developed understanding of the principles of riometry and how the ionosphere affects HF propagation, with a focus on absorption.
- The implementation of both a single frequency and spectral riometer using the Red Pitaya as a backend.
- The objective evaluation of the performance of these systems.
- An evaluation of the feasibility of a radar system operating simultaneously with the riometer and the trade-offs that might need to be made.
- An evaluation of what changes need to be made to the SuperDARN radar to allow it to operate as a riometer.

1.2.2 Motivation for the Study

With the SANAE IV radar's upgrade to a fully digital system and the flexibility the FPGA provides, the possibility of expanding the range of operations offered by he SANAE IV radar is attractive, especially considering how cost effective such a solution would be.

In addition it is hoped to be able to extend the implementation to other digital SuperDARN radars in the network, greatly improving riometer coverage of the polar regions without significant extra cost. Spectral riometry also opens up the possibility of determining the height profile of the ion density in the ionosphere at a particular location using a passive system.

Riometry can be used by scientists to track high energy electron and proton precipitation events in the ionosphere and provides a different way of looking at space weather in general. It can also indicate to HF Over-The-Horizon (OTH) radars and HF communication systems the extent to which their signals will be attenuated due to ionospheric absorption. Expanding riometry coverage can only lead to better predictions.

1.3 Scope and Limitations

This project is limited to the construction and testing of a demonstrator riometer at UCT, and does not concern the specifics of implementation on the actual SuperDARN radar at SANAE IV. It does, however, consider some of the implications that implementation on the SANAE IV system would carry and how one might go about adapting the SuperDARN radar to act as a riometer.

The two types of riometers being considered are scanning (single frequency) and spectral riometers. The imaging riometer not being considered for implementation in this project. However, given the multiple antennas and receivers that comprise the SANAE IV radar, it would be possible to implement an imaging riometer on the SANAE IV SuperDARN radar. Therefore when considering how the SANAE IV SuperDARN radar would have to be adapted for riometry, the possibility for imaging riometry is taken into account.

1.4 Apparatus - Software and Tools

In terms of software, MATLAB was used as the backend of the system and for post-processing operations. C++ was used to program the processor side of the Red Pitaya with the Red Pitaya running a Linux operating system. Verilog in the Vivado software package was used to program the FPGA.

The main piece of hardware used was the Red Pitaya, featuring a Zynq 7010 SOC and two ADCs and DACs. At the front-end of the demonstrator system some RF pre-select filters were used to prevent aliasing and limit the noise bandwidth and two identical Mini-Circuits amplifiers were used to boost the incoming signal.

Additionally various signal generators, oscilloscopes, spectrum and network analysers were used to test the demonstrator riometer in UCT's radar lab.

1.5 Plan of Development

Chapter 2 presents an overview of the ionosphere and how it affects electromagnetic (EM) radiation. Specific emphasis is placed on EM absorption due to the ionosphere.

Chapter 3 then presents a review of riometry – its history, principle of operation and modern digital riometers. Over and above a simple single frequency riometer, imaging and spectral riometers are also examined, as well as the post-processing that allows absorption measurements to be made.

This is followed by the design of the demonstrator riometers for this project in Chapter 4. Two different types of riometers were designed: a single frequency (sometime called 'scanning' in this dissertation) riometer and a multi-frequency riometer. These were also combined into a single implementation that allowed for the selection of one mode or the other. The different front-ends and antennas used by the demonstrator system are also explained.

In Chapter 5 the testing methodology, and results of those tests, of the demonstrator system is covered. Various tests were designed and completed to show that the digital design of the riometer worked, the front-end performed as expected and some field trials were run to test the demonstrator system in the real world.

Finally Chapter 6 covers the discussion of the results and how the SuperDARN radar could be adapted to work as a riometer, before Chapter 7 concludes the dissertation.

Chapter 2

An Introduction to the Ionosphere

The ionosphere is a layer of the Earth's atmosphere which is partially ionised (less than 1 % in reality) and, as a result, electrically conductive. It forms between 60 and 1000 km altitude as the result of neutral particle ionisation due to two different mechanisms: solar radiation and high energy particle collisions. The ionisation forms a neutral plasma with approximately equal quantities of ions and electrons. The positive ions are too heavy to interact much with radiowaves, and thus the electron density determines the properties of the ionosphere with respect to HF radiation. Solar radiation and high energy particle collisions are the mechanisms that help to form ions in the ionosphere, however, there are also competing mechanisms, recombination (the formation of neutral electrons or atoms) and electron attachment (the creation of negative ions), that constantly deplete ionisation in the ionosphere. These competing factors result in peaks in the electron density of the ionosphere at particular altitudes, which have been named the D-, E- and F- (further separated into F1 and F2) layers.

This chapter will discuss the effects that produce and deplete ionisation and will emphasise the D-layer as the main driver of HF absorption. Much of the insight in this chapter is drawn from Giuseppe Fabrizio's book, *High Frequency Over-the-Horizon Radar* [10], with additional insight from Davie's, *Ionospheric Radar* book [11], Ratcliff's paper, *The Ionosphere and the Engineer* [12] and Stuart Andersen and James Meadrick's Chapter on HF Over-The-Horizon radar in *The Radar Handbook* [13].

Section 2.1 details the formation of layers in the ionosphere by describing the factors that produce and deplete ions and how together these result in ion density peaking at certain altitudes, forming the different layers of the ionosphere. Section 2.2 then covers how the D-layer of the ionosphere, in particular, absorbs EM radiation.

2.1 The Formation of Layers in the Ionosphere

Peaks in electron density are the result of competing processes that produce and deplete ions. Processes that produce ions are discussed first before examining processes that deplete ions.

2.1.1 Ion Production

There are two main drivers of ion production in the ionosphere: photo-ionisation and particle ionisation, each of which is discussed in this section.

Photo-ionisation

Ionisation due to solar energy predominantly occurs during the day. The extent of ionisation varies as a function of altitude, atmospheric composition and intensity of ionising radiation [10]. At the edge of the ionosphere the atmospheric density is very low, resulting in limited ionisation. The density increases at lower altitudes, however, more of the ionising energy is absorbed which means less of it reaches lower altitudes. Thus ion production due to photo-ionisation peaks at a certain altitude. A distribution of altitude versus photo-ionisation potential of an ideal atmosphere is known as a Chapman layer[14]. Photo-ionisation is the dominant form of ionisation in the E- and F-layers of the ionosphere.

Chapman described the rate of ionisation one would expect in an atmosphere composed of a single species (a single type of atom or molecule) due to monochromatic light [14]. Photo-ionisation can be described as the process of a photon transferring it's energy, hv (h is Planck's constant and v is the frequency of the photon), to an atom or molecule which causes it to release an electron as shown in Equation 2.1.

$$P + hv \rightarrow P^{+} + e^{-}. \tag{2.1}$$

Different molecules and atoms ionise at different photonic frequencies, with the most important ones being O_2, N_2 and NO, with atomic Oxygen and Nitrogen becoming more important at higher altitudes [10]. Predominantly Extreme-Ultra Violet (EUV) and X-Ray radiation from the Sun cause ionisation in these species.

The rate of ionisation at a particular altitude is proportional to the intensity of radiation at that altitude and the density of neutral species. The radiation intensity decreases with decreasing altitude due to absorption, whereas the neutral density increases with decreasing altitude due to gravity. Thus the rate of production of ions peaks where the neutral density is sufficiently high, and the radiation intensity

is not too low. The Chapman equation describes the rate of ionisation in a horizontal-plane stratified atmosphere made up of a single isothermal gas by absorption of a parallel beam of monochromatic solar radiation[14]. While this is a greatly simplified model, it serves its purpose in showing the formation of layers in the ionosphere. The Chapman equation is derived in Chapman's paper [14], but the final result can be seen in Equation 2.2 where q_0 is a constant, z' is the relative altitude and X is the zenith angle that the Sun's rays make to the vertical.

$$q(z', X) = q_0 e^{1-z'-\sec(X)e^{-z'}}. \tag{2.2}$$

As Figure 2.1 shows, the zenith angle influences the peak height of electron production, with angles closer to the 90 degrees resulting in greater ion production rates at lower altitudes.

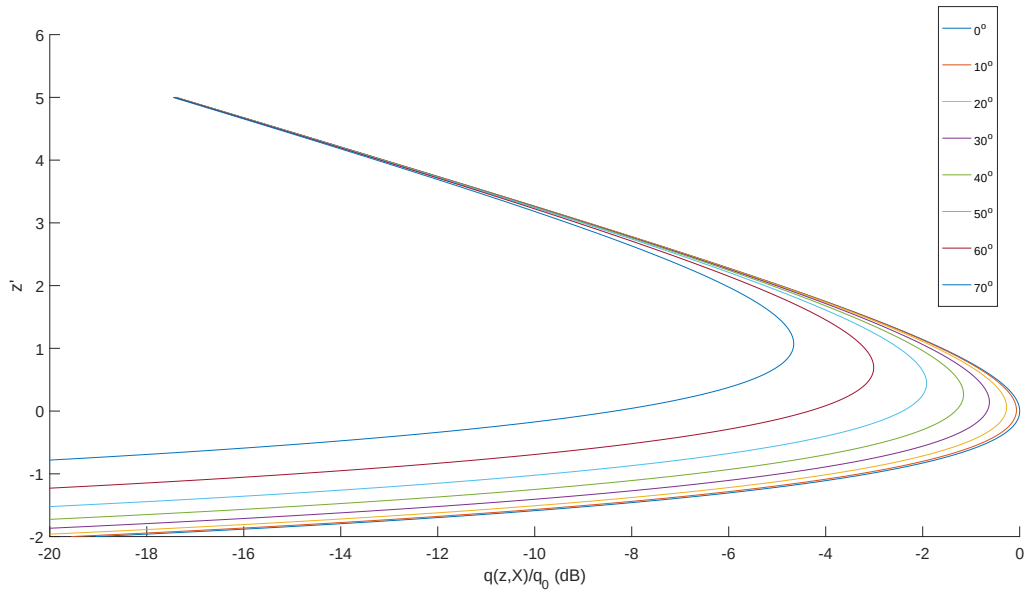


Figure 2.1: Normalised ion production rate as a function of zenith angle and relative altitude.

Particle Ionisation

Particles (electrons or protons) with sufficiently high energy are capable of entering the atmosphere and ionising it. They release most of their energy at altitudes of less than 100 km, therefore having the greatest effect on the D-layer of the ionosphere. Charged particles get trapped on the magnetic field lines of the Earth, only being able to travel along the lines on which they are trapped. Thus they are far more likely to enter the atmosphere near the magnetic poles of the Earth, where the magnetic field lines come towards the Earth vertically. When trapped on the magnetic field lines of the Earth, charged particles experience the following motions [15]:

- **Cyclotron** – a circular motion around the magnetic field line.
- **Bounce** – the particle "bounces" between the Southern and Northern magnetic poles as they are reflected by the strong magnetic fields at the poles.
- **Drift** – particles can drift across field lines.

At high latitudes energetic particles have a much greater impact on the ionisation of the atmosphere than at lower latitudes. In part, when a riometer measures changes in absorption it is trying to help identify particle ionisation events.

Energetic particles mainly enter the atmosphere through two mechanisms. The first is solar wind from the Sun, allowing charged particles to cross the interplanetary medium and impact with the Earth's atmosphere. The second is high energy electrons entering into the ionosphere[10].

2.1.2 Ion Depletion

Production processes tend to be driven by solar energy and are thus largely dependent on the daily solar cycle. Depletion processes, however, are constantly acting to reduce ionisation in the ionosphere. Despite this, they are never able to completely deplete the ionisation of the ionosphere, particularly in the higher layers, and ionisation tends to be high enough in these layers to permit HF radar and communication [10]. There are two main depletion processes: recombination and electron attachment.

Recombination

Recombination happens constantly as the result of molecular interactions as ions seek to gain lower energy states. In the simplest case an ionised particle, P^+, recombines with an electron, e^-, to reach a lower energy state, releasing energy, hv, in the process:

$$e^- + P^+ \rightarrow P + hv. \tag{2.3}$$

It is important to note that the recombination rate of ionised molecules is about five orders of magnitude larger than that of ionised atoms [10]. The reason for this is that atomic ions are first converted to molecular ions before recombination takes place. The process of converting atomic ions to molecular ions is known as a charge transfer reaction.

The rate of electron depletion is dependent on the density of molecular ions at different altitudes. At lower altitudes the density of molecular ions tends to be quite high, resulting in a greater electron

depletion rate. The density of molecular ions is also dependent on the density of atomic ions, as these are then converted to molecular ions via the charge transfer reaction. Below about 140 km, the region of interest for absorption measurements, molecular recombination tends to dominate [10]. In this region the loss, L, is proportional to the product of the density of free electrons, N_e, and the density of ionised molecules which has the same density as the number of free electrons in an electrically neutral plasma. Thus the ion depletion rate follows a square law where α is the mean molecular dissociative recombination rate given by:

$$L = \alpha N_e^2. \tag{2.4}$$

Atomic recombination is the dominant form of ion depletion in the upper atmosphere, above about 210 km [10]. Due to reduced molecular ion density, the charge transfer step takes much longer. Lighter atomic species are more prevalent in the upper ionosphere and tend to have a longer lifetime, thus the charge transfer step dominates in this region. Therefore, the ion depletion rate depends linearly on the electron density, and is given in Equation 2.5 where β is the linear loss coefficient.

$$L = \beta N_e. \tag{2.5}$$

Between 140 km and 210 km altitude the ion depletion rate is a combination of Equations 2.5 and 2.4, slowly changing between molecular recombination being dominant and atomic recombination being dominant. Overall, ion depletion happens much faster at lower altitudes, accounting for the D-layer of the ionosphere rapidly depleting at night, while higher layers are never fully depleted.

Electron Attachment

Another loss process is electron attachment, where electrons attach to neutral molecules, M, to form negative ions, as shown in Equation 2.6. The loss rate due to electron attachment is related to the density of neutrals at a particular altitude, and is therefore much more prevalent at lower altitudes.

$$e^- + M \rightarrow M^-. \tag{2.6}$$

The loss rate is linearly related to the neutral density and is given by Equation 2.5.

2.1.3 The Layers of the Ionosphere

The combination of ion production and depletion mechanisms results in several different peaks in the electron density of the ionosphere being formed at different altitudes. The overall rate of ion depletion/production is found by subtracting the ion depletion rate from the ion production rate.

Photo-ionisation is the dominant and most predictable form of ion production and so is assumed to be the main form of ion production across layers. At lower altitudes the dominant form of ion depletion is molecular recombination which obeys the square law. Thus the rate of change of electron density is given by the continuity equation as:

$$\frac{dN_e}{dt} = Production - Depletion - Transport. \tag{2.7}$$

The transport term is ignored for the purposes of this analysis since it tends to become more dominant above about $210\,km$, and absorption mainly takes place below about $100\,km$ altitude [10]. Thus the equation becomes:

$$\frac{dN_e}{dt} = Q(z', X) - \alpha N_e^2. \tag{2.8}$$

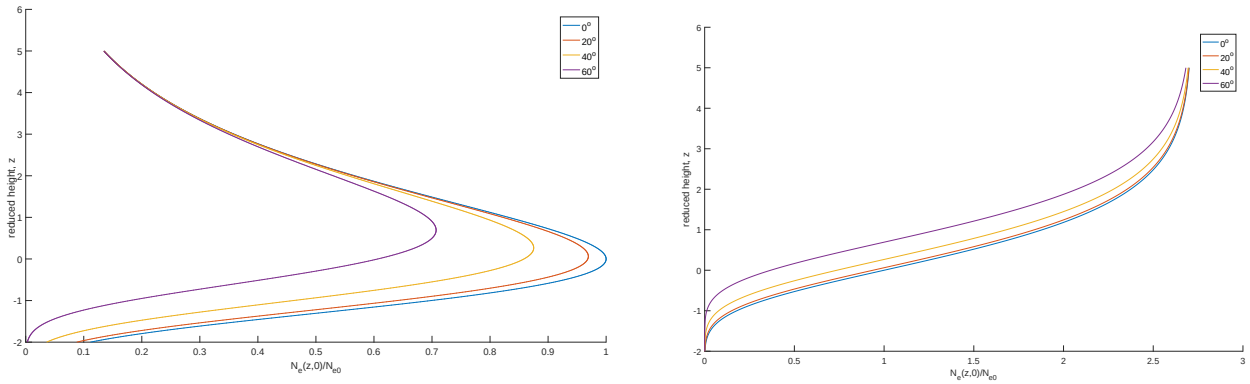
An equilibrium will be reached when the rate of change of electron density is equal to zero. Substituting in Chapman's equation for $Q(z', X)$, setting $\frac{dN_e}{dt}$ to zero and rearranging gives the steady state electron density in terms of reduced height and zenith angle:

$$N_e(z, X) = \sqrt{\frac{q_0}{\alpha}} e^{\frac{1}{2}(1 - z' - \sec(x) e^{-} z')}. \tag{2.9}$$

This is known as the α-Chapman layer. The β-Chapman layer is formed in the region where the linear law loss equation dominates, above 140 km. It can be found in an equivalent manner to the α-layer to be:

$$N_e = \frac{q_0}{\beta} e^{1 - \sec(x) e^{-} z'}. \tag{2.10}$$

Figure 2.2 compares the α- and β-layers for different zenith angles. An interesting point to note is that the α-layer electron density varies with $\cos X$, where X is the zenith angle. Thus, at mid-latitudes, the electron density follows a $\cos X$ function as the day progresses.



(a) α-layer **(b)** β-layer

Figure 2.2: Comparison between the α- and β-layer electron densities at different zenith angles against reduced height. The α-layer has a much steeper roll-off on either side of its peak value, as compared to the β-layer.

All of these competing ion production and depleting mechanisms lead to the formation of several layers in the ionosphere. Layers are characterised by local peaks in the electron density of the ionosphere at different heights. The layers formed are the D-, E- and F-layers. The F-layer is further subdivided into the F1- and F2-layer. Figure 2.3 shows the electron density of a representative ionosphere as a function of height, along with the layers that are typically present on a mid-latitude Summer's day.

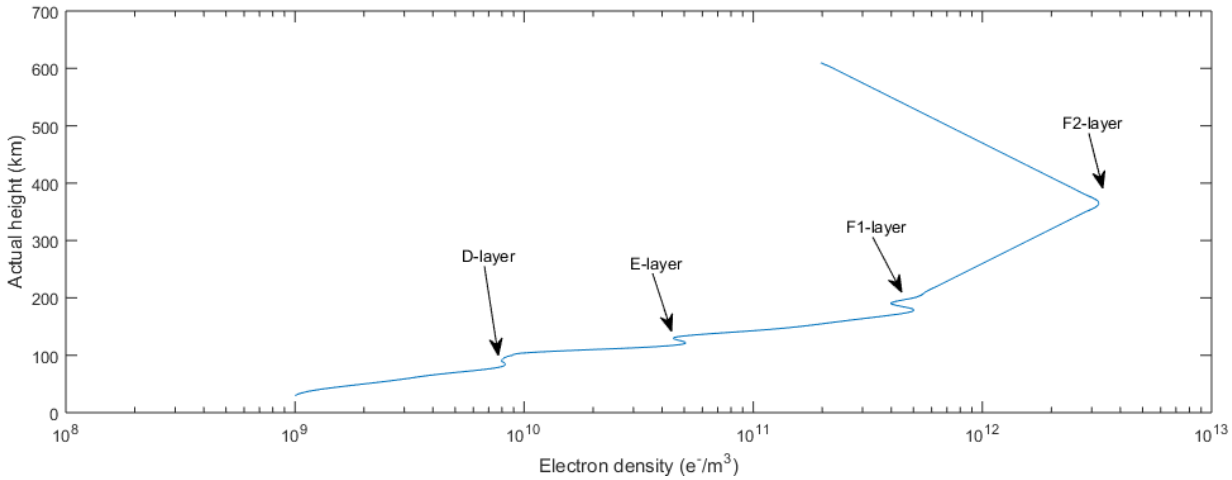


Figure 2.3: Representative ionospheric density as a function of actual height during a mid-latitude Summer's day [10].

At night the strong depletion factors at lower altitudes cause the D-layer to almost completely disappear [10]. The other layers have a lower electron density but tend to persist overnight, which is useful for the purposes of HF over-the-horizon (OTH) radar and HF communications. The D-layer, on the other hand, mainly absorbs radiation at HF frequencies.

2.2 Absorption

The D-layer of the ionosphere can strongly attenuate radio waves, especially in the HF band. The reason for this is that electrons oscillating in the wave field collide with neutrals, resulting in heating. The absorption experienced is mostly non-deviative and increases with both electron and neutral density and decreases with radiowave frequency [16]. Thus, while the D-layer has lower electron density than the higher layers in the ionosphere, it also has even greater neutral density, allowing it to absorb more radiation. Absorption shows strong diurnal variation, with much more absorption occurring during the day as photo-ionisation causes increased electron density. At night the fast ion depletion action at low altitudes quickly depletes the D-layer [13]. The riometer, of course, attempts to detect the amount of absorption in the D-layer. There are two events that cause increased absorption which are of interest:

- **Auroral Absorption** – occurs as the result of particle ionisation due to high energy electrons ($> 10\,KeV$) precipitating into the ionosphere and is associated with the aurorae.
- **Polar Cap Absorption (PCA)** – occurs as the result of particle ionisation due to high energy protons ($< 1\,MeV$) as the result of Coronal Mass Ejection (CME) or solar flares. PCA results in strong absorption ($> 3\,dB$) which can wipe out HF communications and radar for a period of time.

These events are mostly observed at higher latitudes, where the Earth's magnetic field lines are almost vertical, allowing for charged particles to enter the ionosphere more easily. One such consequence of particle precipitation into the atmosphere is the aurorae which can be seen near the poles.

The amount of radiowave absorption, in decibels, is approximately proportional to Equation 2.11.

$$A \propto \frac{N_e v_{en}}{v_{en}^2 + (\omega \pm \omega_L)^2}, \tag{2.11}$$

where N_e is the electron density, v_{en} is the electron to neutral collision frequency, ω is the radiowave angular frequency and ω_L is the angular gyrofrequency of electrons. The absorption is largely governed by the product of electron density and electron to neutral density frequency. This accounts for the greater absorption in the D-layer of the ionosphere as it has a low enough altitude to allow for both a relatively high (compared to the other layers) neutral density and a high enough electron density. In addition, the absorption is inversely proportional to the square of the radiowave frequency, resulting in decreased absorption at higher frequencies. Thus, D-layer absorption is only really an issue at HF frequencies and lower.

Chapter 3

The Riometer

The relative ionospheric opacity meter (riometer) is a device for measuring the opacity of the ionosphere with respect to radiowaves in the HF and VHF frequency bands. This chapter starts by giving a brief history of the development of the riometer in Section 3.1 before diving into its principle of operation in Section 3.2. Two alternative versions of the basic riometer, the imaging riometer and the scanning, or single frequency, riometer, are then discussed in Sections 3.3 and 3.4 respectively. Section 3.5 covers the construction of the quiet day curve (QDC), which is a key element in determining absorption from power readings. Finally the different applications of riometers are covered in Section 3.6.

3.1 A Brief History of Riometers

Little and Leinbach published a paper in 1959 which detailed the design of a riometer as a means to determine the opacity of the ionosphere with respect to EM radiation [17], ie. how much EM radiation is absorbed in the ionosphere at a particular frequency. Their device was based off the work of Mitra and Shain in 1953 [18] with their method for using cosmic radiation as a means to measure absorption in the ionosphere. They determined the signal strength of the extra-terrestrial waves received at a fixed receiving system at one point in a sidereal day (every 23 hours, 56 minutes and 4 seconds) and the signal strength at the same point in the sidereal day under conditions of negligible ionospheric absorption. By dividing the power received on the normal day from that on the ionospherically quiet day they were able to determine how much of the signal was being absorbed as compared to the quiet day. Figure 3.1 shows the block diagram of a riometer system with an analogue front-end, as an example of the type of design that Little and Leinbach designed. Their system is operated by matching a noise source to the signal coming in from the antenna at a chosen frequency and integrating over the energy of the resulting signal.

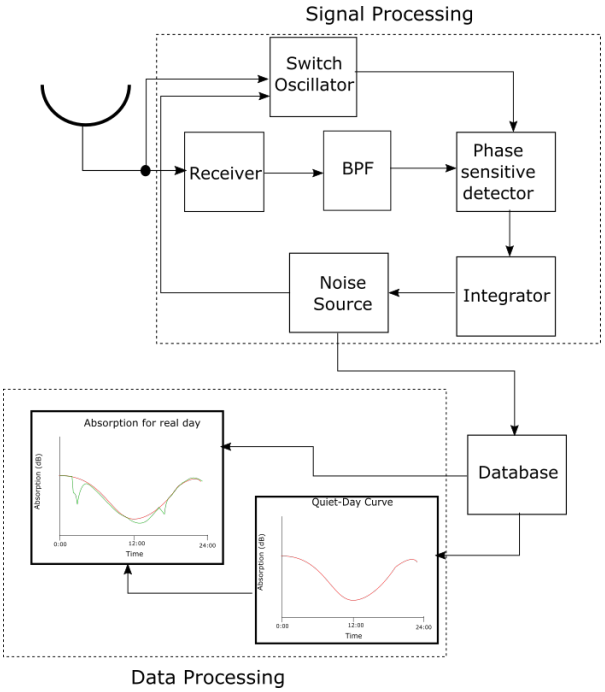


Figure 3.1: Block diagram of a basic riometer with an analogue front-end.

The integration time is typically of the order of seconds. The most common frequency chosen is $38.2\,MHz$, a quiet-band reserved for astronomical observations and a measurement bandwidth of between $5\,kHz$ and $250\,kHz$ can be used depending on the conditions, application and type of riometer. Measurements after integration are stored and used to build up a quiet day curve. The level of cosmic noise radiation varies as a function of the portion of the galactic sky which is currently being viewed, as well as the bandwidth, frequency and beamwidth of the receiver.

The receiver is continuously calibrated by rapidly switching the receiver input between a known noise source and the antenna. Differences between the noise source and the antenna noise power levels produce a square wave signal at the switching frequency. The difference signal is isolated using a narrowband filter detected in the receiver, integrated and fed back to drive the noise diode so that the noise output matches the antenna's. The current through the diode is proportional to the noise power at the antenna. The main advantage of this technique is that since the noise source and the antenna are both fed through the same analogue front-end, changes in front-end gain and bandwidth are naturally calibrated out through the switching process.

One of the issues that a riometer faces is that man-made noise can easily interfere with the noise floor of the riometer operating at a fixed frequency. Therefore Little and Leinbach included a scanning mechanism whereby the riometer would scan across a range of frequencies with relatively narrow bandwidth and detect the minimum power in the frequency span as the power it would record [17]. The idea is that the noise floor will then be detected and narrow narrowband interference avoided. Initially these circuits were completely analogue, but advances in the 1970s led to the development of a digital minimum detection circuit [19].

With the advent of solid state computers and ADCs capable of sampling in the MHz range, a new generation of digital riometers is being produced. Low power devices for remote operation, such as on Mars [20], have also been produced with the help of digital technology. Figure 3.2 shows the block diagram for the digital riometer that was proposed for operation on Mars. Sampling at an intermediate frequency and performing integration and power detection in the digital FPGA shortens the analogue front-end, which means a lower chance of gain or bandwidth drift and a more stable, more linear system.

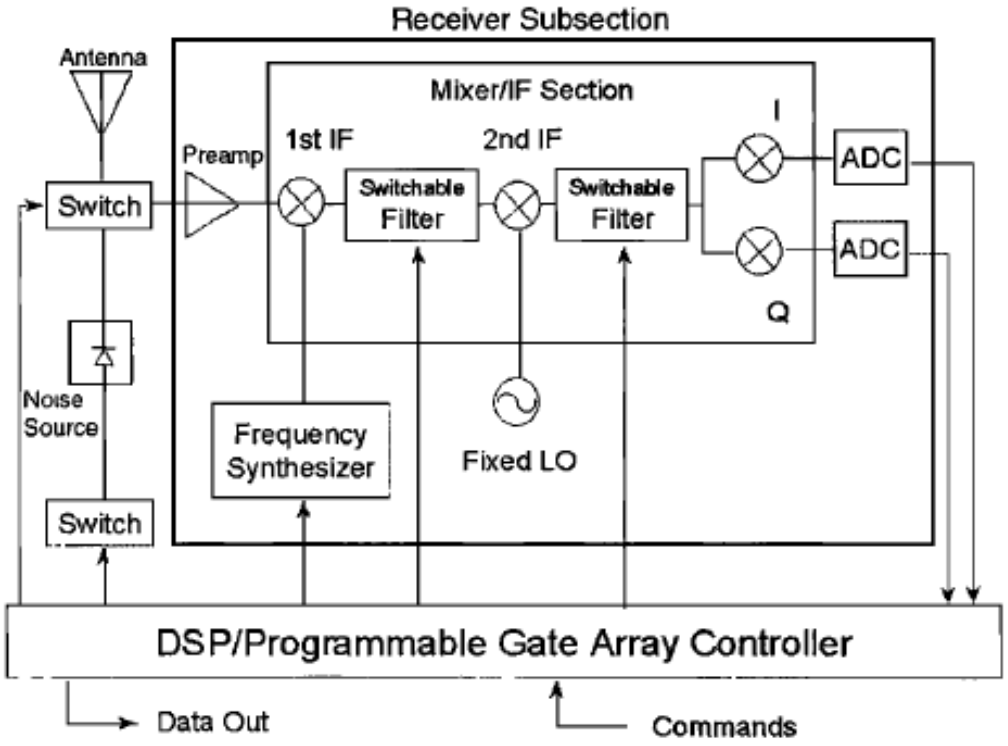


Figure 3.2: Block diagram of a digital riometer with an analogue front-end, including an intermediate frequency stage, for operation on Mars [20].

However, the riometer for Martian operation was produced in 2000 and with even faster ADCs available now, digital riometers can sample the entire HF and VHF band, completely eliminating the need for analogue mixers. A digital riometer usually features a short analogue front-end, comprising amplifiers and analogue filters (at the very least an anti-aliasing filter before the ADC is required), but sometimes bandpass filters are used too. Thereafter the signal is processed digitally. In a basic single-frequency riometer this would mean providing a digital filter to limit the incoming signal to the bandwidth and frequency of interest and determining the power. The power of the signal is then integrated over a period of time, usually several seconds. If a Dicke switched or Ryle-Vonberg system is used (see Section 3.2.1), then the receiver chain is switched between a known noise source and the antenna with the noise source being used to calibrate out changes in the analogue receiver chain.

Sampling directly at the RF has been the trend in recent times [21, 22], especially with the development of cheap and easy to use Software Defined Radios (SDR). This also allows for up to 45 dB less front-end gain required than for a traditional analogue riometer [22] and comes with increased flexibility and a

cheaper product that can easily be adapted for other purposes. For instance, the riometer designed in this project can also easily be used as an HF spectrum analyser.

3.2 Principle of Operation

Before describing how a riometer actually works, one requires a basic understanding of cosmic noise radiation. As the name suggests, cosmic noise radiation is extra-terrestrial noise originating, for the most part, from our own galaxy. Its intensity decreases with frequency, with noise power being approximately proportional to the square of the frequency, and thus it causes the most interference to radar systems in the HF band and lower. In fact it usually exceeds the noise figure of the front-end. For instance, at about $30\,MHz$ the galactic noise temperature will be in the region of $2.9 \times 10^4\,K$. Figure 3.3 shows the noise floor due to different phenomena across the HF and VHF frequency spectrum. At frequencies below about $50\,MHz$ galactic noise, man-made noise and atmospheric noise all play a part in determining the noise floor.

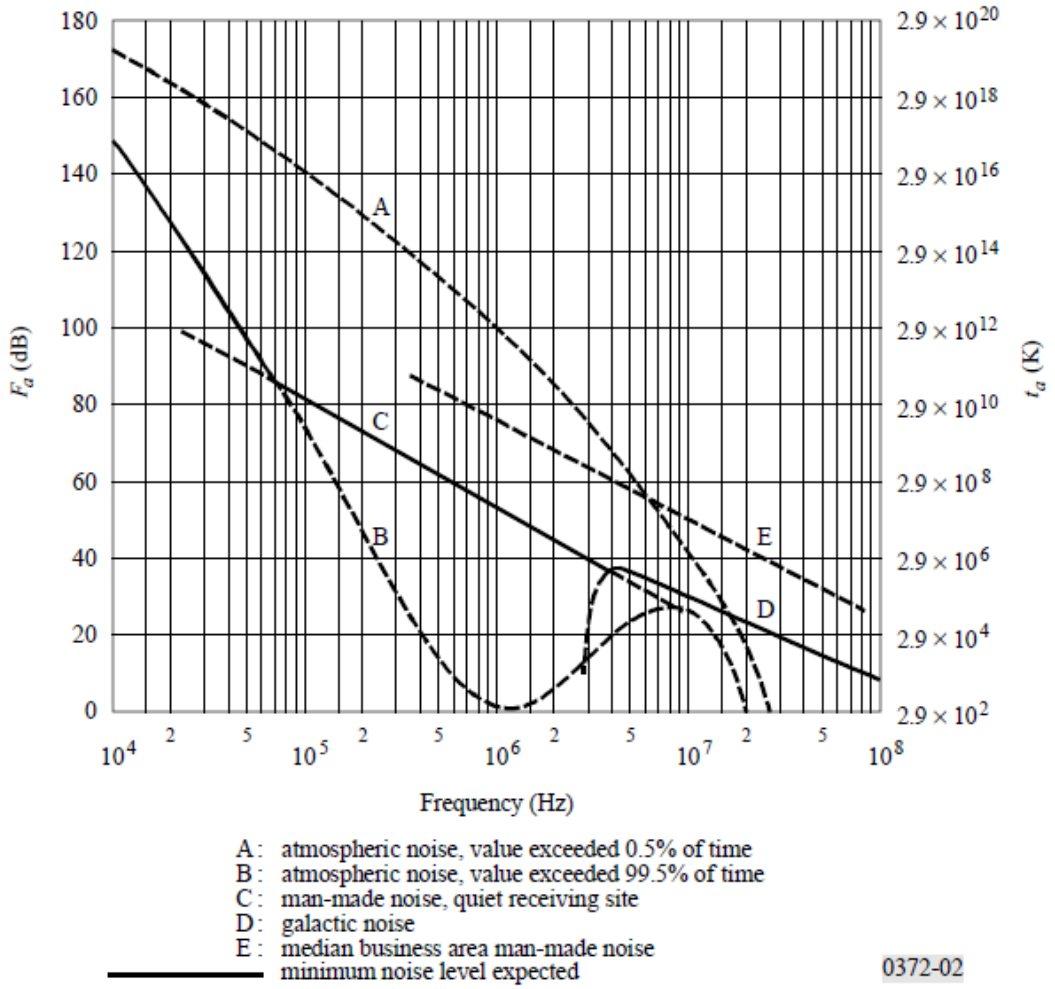


Figure 3.3: Noise temperature from $10\,kHz$ to $100\,MHz$ as per the International Telecommunication Union's (ITU) recommendations [23].

A riometer uses cosmic noise radiation as its signal source. However, if one considers a directional antenna pointing at the sky, as the Earth turns the antenna will point at different portions of the universe. Clearly the received galactic noise will then change as the Earth is rotating. In addition the Earth moves around the Sun. Thus the antenna will not point at the same place in the galaxy once every 24 hours, but rather once every 23 hours 56 minutes and 4 seconds. This is known as a sidereal day and will be discussed further in the section on quiet day curves.

So with a basic understanding of cosmic noise radiation, a riometer is fundamentally a device that measures the noise power received through the antenna. The difficult part is distinguishing between noise due to cosmic radiation, and other noise sources or man-made signals. If the noise temperature due to cosmic radiation through an antenna is given by T_{source} and this is the noise power of interest, then the antenna power, in watts, is given by:

$$P_{ant} = kBG(T_{sys} + T_{source} + T_{ground} + T_{sky}), \tag{3.1}$$

where T_{sys}, T_{ground} and T_{sky} are interfering noise sources. In Equation 3.1, BG is the bandwidth-gain product, where B is the bandwidth and G is the system gain and k is Boltzmann's constant. The problem is that BG, and particularly the gain component, varies over time. Since the contribution from cosmic noise sources is relatively small compared to the contributions from other sources, finding the variation in noise level due to absorption becomes impossible when a changing BG amplifies the large noise sources more than the small ones. Ironing out changes in the time-gain product is a large part of getting an accurate power reading.

Another way this can be expressed is through the radiometer sensitivity equation. The radiometer sensitivity equation is derived in Appendix B and is found to be:

$$T_{min} = \frac{T_{sys}}{\sqrt{BT}}. \tag{3.2}$$

Here T_{min} is the minimum detectable noise temperature change, measured in kelvin. Clearly increasing the bandwidth, B, and integration time, T, both allow for better sensitivity. For a riometer a typical integration time is of the order of 0.1 to 10 s, with a bandwidth of 10 kHz being fairly typical. Thus, assuming a system noise temperature of about 250 K, a bandwidth of 10 kHz and an integration time of 0.5 s, the minimum detectable change is 3.5 K. If the incident noise temperature on the antenna due to cosmic noise is 5000 K, then the sensitivity allows us to detect a 0.003 dB change.

However, this is in the absence of gain changes in the receive chain. If gain, G, is unstable then the sensitivity equation becomes:

$$T_{min} = T_{sys}(\frac{1}{\sqrt{BT}} + \frac{\Delta G}{G}). \tag{3.3}$$

Clearly even moderate changes in the receiver gain can dominate the minimum detectable signal. It therefore becomes desirable to calibrate out changes in the receiver gain.

There are various solutions to this problem. The oldest and most common one is Dicke switching, but there are other methods too. Two advances in modern electronics have also helped to mitigate this problem. The first is high speed ADCs which have allowed for digital down-conversion and processing, which means fewer components over which gain can vary and non-linearity can occur. The second is the increased gain stability of amplifiers and filters in the front-end.

3.2.1 Dicke and Ryle-Vonberg Switching

Dicke switching for riometry is a method of calibrating out changes in gain of the analogue front-end of a riometer. It was initially developed during World War Two by Henry Dicke [24]. Figure 3.4 shows the basic block diagram for a Dicke riometer.

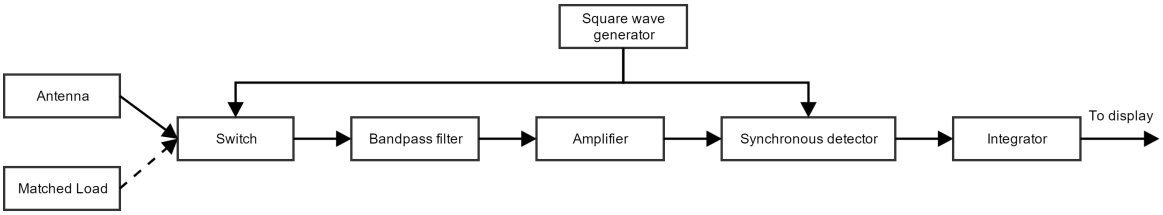


Figure 3.4: Dicke switched riometer block diagram.

The receiver and detector circuitry is switched between the antenna and a matched load at a 50% duty cycle by the square wave generator. The square wave must be at a higher frequency than the gain of the components that are changing. This signal is then bandpass filtered to a fixed bandwidth and amplified before being fed to a synchronous detector. All the synchronous detector does is phase shift the signal by 180° when the matched load is connected. Thus, after integration, the antenna signal and matched load are subtracted from each other. The expression for the noise power through the load is given by:

$$P_{load} = kBG(T_{sys} + T_{load}). \tag{3.4}$$

After subtracting the antenna and load power the result is:

$$\Delta P = kBG(T_{source} + T_{ground} + T_{sky} - T_{load}) \tag{3.5}$$

If system noise is the largest contribution to the noise that is not from T_{source}, which it usually is, then the changing gain's effect on it can largely be mitigated by Dicke switching. Changes in T_{ground} and T_{sky} are much more difficult to eliminate as they are external to the riometer system. In this analysis it is assumed that T_{load} remains constant.

Another similar, but different, mechanism is the Ryle-Vonberg riometer [25]. A block diagram of the Ryle-Vonberg riometer is shown in Figure 3.5. The main difference between the Dicke and Ryle-Vonberg riometer is that the Ryle-Vonberg version feeds back the output of the synchronous detector and tries to match the variable noise source to the difference signal. Thus it is a closed loop system, as opposed to the open loop Dicke riometer.

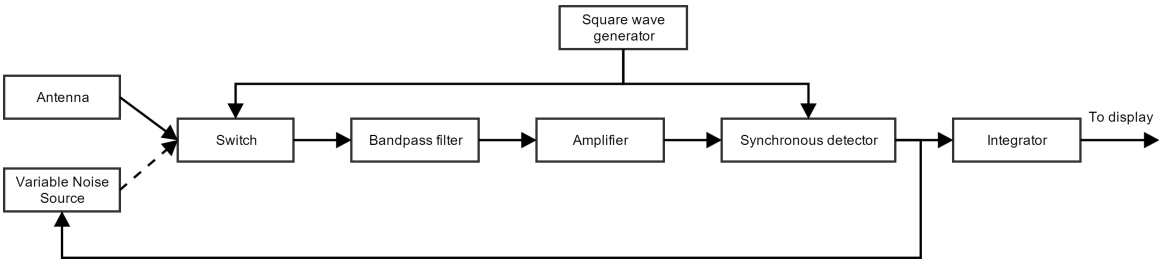


Figure 3.5: Ryle-Vonberg riometer block diagram.

The additional noise contributions from the sky and ground have relatively little impact at HF, but can become significant factors towards the centimetre wavelength. This can be improved upon by using a method known as beam-switching. In this method an auxiliary antenna points at almost the same piece of sky and ground as the main antenna but crucially not at the cosmic noise source. Thus, the first antenna will pick up:

$$P_1 = kBG(T_{sys} + T_{source} + T_{ground} + T_{sky}), \tag{3.6}$$

and the second antenna will pick up:

$$P_2 = kBG(T_{sys} + T_{ground} + T_{sky}). \tag{3.7}$$

After passing through the synchronous detection stage the ground, sky and system noises will be cancelled leaving:

$$\Delta P = kBGT_{source}. \tag{3.8}$$

At centimetre wavelengths this makes it a very powerful technique, however, it is hardly used at HF and is more useful for a dedicated radiometer, which stares at individual radiation sources in the galaxy, than a riometer.

3.3 Imaging Riometers

A normal riometer stares at a single point in the sky when taking measurements. An imaging riometer uses multiple beams to simultaneously take measurements at multiple points in the sky, building up an image of the absorption across a region. One such example of an imaging riometer is the Imaging Riometer for Ionospheric Studies (IRIS) which was first developed in the late 1980s [26]. The beam pattern for the IRIS is shown in Figure 3.6. One of the peculiarities of imaging riometers is that they require a correction factor for off-centre beams due to the angle they make with the ionosphere. A radiowave coming through the ionosphere non-perpendicularly will be more attenuated than one that enters perpendicularly since it travels through more of the ionosphere. Thus the power received through the outer beams are multipled by an obliquity factor to correct for this.

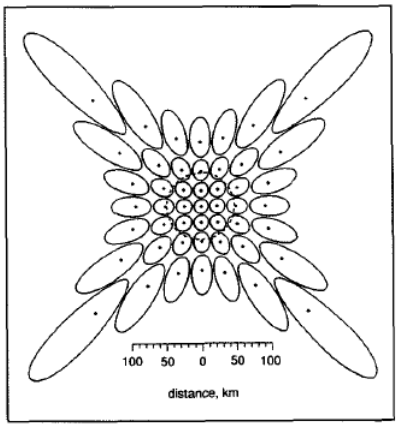


Figure 3.6: Projection of the IRIS beam pattern at 90 km altitude [27].

Subsequent imaging riometer systems have usually attempted to achieve higher resolution than the original IRIS, such as a system in Alaska with 204 beams spread over a 200 × 200 km^2 square of radio sky at 90 km altitude [28]. This as compared to the 49 beams also over about 200 × 200 km^2 square of radio sky at 90 km for IRIS. Imaging riometers also usually use a single riometer between multiple beams and switch between the antennas for each reading. This cuts into the integration time for each beam, but reduces the hardware requirements.

IRIS was one of the first imaging riometers produced and used a Butler matrix phasing assembly to perform its beam-forming. Since then advancements in digital signal processing have allowed

for digital beam-forming to take place. Honary *et al.* created an imaging riometer that uses an FPGA to do its beam-forming [29]. They showed successfully that their system was capable of having better measurement precision for filled arrays and higher spatial resolution when using a Mills cross configuration for their antennas. Thus while analogue beam-forming has been the norm until recently, digital imaging riometers are set to become more and more prevalent.

The SuperDARN radar at SANAE IV has beam-forming capabilities allowing for it to be used as an imaging riometer. It does not point straight up, however, and the angle it makes to the horizon must be taken into account when doing riometry calculations. Additionally, being a planar array, it only has beam-forming capabilities in one dimension and can only form a 1D image, rather than a 2D one.

3.4 Spectral Riometers

Kero *et al.* pioneered the first implementation of the spectral riometer, a device to measure absorption across a wide range of frequencies, and therefore perform electron density profile estimation [30]. It uses the Kilpisjärvi Atmospheric Imaging Receiver Array (KAIRA) as a riometer to measure the cosmic radio noise at 244 frequencies between 10 and 80 MHz [31]. A simple parametrised particle precipitation model is used to take the absorption readings, now in the form of an absorption spectrum, and invert them to obtain an electron density profile estimation. These results were compared against results obtained from the nearby European Incoherent Scatter VHF radar obtained at the same time to show that the spectral riometer method could produce comparable results under the conditions of substorm related electron precipitation.

The SuperDARN radar at SANAE IV is also capable of operating across a range of frequencies in the HF range, opening up the possibility of spectral riometry. Detailed measurements of absorption at different frequencies also carries the advantage of allowing HF communication and radar platforms to judiciously select their operating frequency.

While using multiple frequency riometer measurements to determine the electron density profile was first proposed in 1963 [32], and has since been proposed in alternative formats by a number of other papers, it has not become a routine part of riometry, with most riometers, including imaging, operating at a fixed frequency. Those earlier studies, however, only considered a limited number of frequencies, typically two or three, for which the electron density inversion problem is too unstable to give a reliable electron density profile estimate. Modern day digital technology has allowed for the measurement of absorption at hundreds of different frequencies simultaneously using a single riometer.

Coming back to the paper by Kero *et al.* on electron density estimation, the inversion process requires strong regularisation in order to obtain stable results. They solved this problem by using a parametrised electron density model, instead of calculating the electron density at each height independently. This results in a smooth electron density profile.

3.5 Constructing the Quiet Day Curve

One builds up a Quiet Day Curve (QDC) as an engineering solution to cancel the effects of the changing cosmic radiation received and the changing ionospheric opacity due to photo-ionisation on a daily basis. A quiet day curve is the amount of absorption experienced on a typical sidereal day where no additional absorption effects have been introduced. The QDC varies seasonally and so must be updated every few weeks. Any deviations away from the QDC indicate additional absorption events, either auroral or polar cap absorption, and riometers can help to quantify and identify these events. Once the QDC has been identified, the absorption at a particular time and frequency is defined as:

$$A_{dB} = 10\log_{10}(\frac{P_{qdc}}{P_r}), \tag{3.9}$$

where P_r is the power measured and P_{qdc} is the quiet day curve power. The absorption is measured in decibels.

There is no way of measuring the quiet day curve directly, but by looking at the cosmic noise measurements, it is possible to identify periods of quiet ionospheric conditions and use these to determine the quiet day curve. The methodology of doing this has been discussed since the advent of riometry in the 1950s by a variety of authors. In the early days, the QDC was determined by hand without the aid of computers ,the resulting QDCs were inaccurate and subjective [33, 18].

More recent work has looked at using statistics to iteratively identify and remove outliers before determining the QDC. One such technique was proposed by Tanaka *et al.* [34]. Tanaka's method follows the method described below.

1. Convert binary data from riometer to power over a sidereal day for each month of operation.
2. All levels of voltage for each beam corresponding to the same sidereal minute are averaged.
3. This leads to the sidereal time variation of the monthly averaged level of voltage for each beam (μ).
4. A similar procedure is applied to determine the standard deviation (σ).
5. For each beam the valid voltage levels measured. They are constrained to be in the range of $\mu \pm 3\sigma$. All data points outside of this range are discarded.
6. A new average and standard deviation is calculated and the discarding process repeated until no more points are discarded.
7. The pre-QDC value is obtained by adding 80% of the average to three times the standard deviation for each minute, over a month. The final QDC value for each beam is then obtained by finding the 30 point running average of the pre-QDC values.

A 2012 paper offered some improvements over Tanaka's technique [35]. Specifically, they cited that Tanaka's technique is unduly influenced by geomagnetic disturbances, but that these can be corrected for by identifying days of high geomagnetic activity using the Kp index, specifically when $Kp > 3$, and not using these days in the estimation [36]. Additionally, days before or after days of heightened geomagnetic activity are also discarded since these could still be influenced by the activity.

3.6 Riometer Applications in Research

Riometers are grouped amongst a category of devices that observe the ionosphere. These include ionosondes (radars that actively probe the ionosphere), satellites and even SuperDARN. Its main advantage over other observational sources is that it is a passive device and thus requires much less power than ionosondes and HF radars. Riometers can also still function during polar blackouts when HF radars and ionosondes fail [37]. However, it comes at the cost of not being able to directly measure the ionosphere's properties and relying on intelligent post-processing to obtain absorption information.

Absorption, as measured by a riometer, is an indicator of enhanced ionisation of the ionosphere. Thus it can be used to study events that enhance ionisation in the atmosphere outside of solar radiation. These events are broadly classed into two categories, PCA and auroral absorption, as discussed in Section 2.2.

Early ionospheric research using the riometer were forced to operate with a single beam, single frequency system. Nonetheless they were used to study both PCA and auroral absorption events. Early studies focussed on large scale polar blackouts and absorption trends, sometimes using multiple riometer stations to expand the region of interest [38]. Some work from that time was also done on auroral absorption events, even at sub-auroral latitudes [39].

After the development of the imaging riometer in the 1990s, riometer research has shifted to using these tools more as a means to study the way absorption changes across a region. For instance Bezrodny *et al.* did a study on scintillation of discrete cosmic sources using an imaging riometer [40]. Stauning has also done some work on examining, and explaining, the asymmetric absorption patterns of PCA events [41].

There have also been suggestions that the E-region plays a role in affecting riometer absorption readings through the mechanism of backscattering cosmic radio noise back into space off E-region plasma waves [42]. This results in absorption readings that include both actual absorption and the contribution from backscattered radiation that is 'seen' as absorption.

Chapter 4

Design of the Demonstrator Digital Riometer

This chapter details the design of the digital riometer for both the scanning and spectral implementations. It starts by laying out the design of the front-end, including the antenna, that was used in the demonstrator riometer in Section 4.1. Thereafter the characteristics of the Red Pitaya are given in Section 4.2 before proceeding to how the FPGA on the Red Pitaya was programmed in Section 4.3. The actual riometer designs are covered in Section 4.4 for both the scanning and spectral riometers. The chapter is completed by detailing some of the additional features of the design in Section 4.5 and going over the instruction set in Section 4.6.

Before venturing further the reader should be aware that four field trials were conducted with the riometer. While the testing methodology and results of these trials are only detailed in Chapter 5, the front-end, antenna and FPGA processing was adapted between trials and the details of these adaptations are presented in this chapter with reference to the trial being conducted in chronological order.

4.1 Front-end and Antenna

The analogue part of the riometer is limited to the antenna and a number of amplifiers and filters that are collectively grouped as the 'front-end'. These filters and amplifiers are examined first before taking a look at the two different antenna-balun systems that were used in this project. The first antenna was a dipole designed to operate at $42\,MHz$ with a 1:1 balun design. The second antenna was a dipole with a reflector designed to work at $38.2\,MHz$ and utilising a sleeve balun to also provide matching at $38.2\,MHz$.

4.1.1 Front-end

The front-end refers to all analogue circuitry before sampling takes place, but after the antenna. Two different front-end configurations were used. For the first three trials only a low-pass filter was incorporated, while for trial four an additional high-pass filter was included. Figure 4.1 shows the front-end components and their order for the trials.

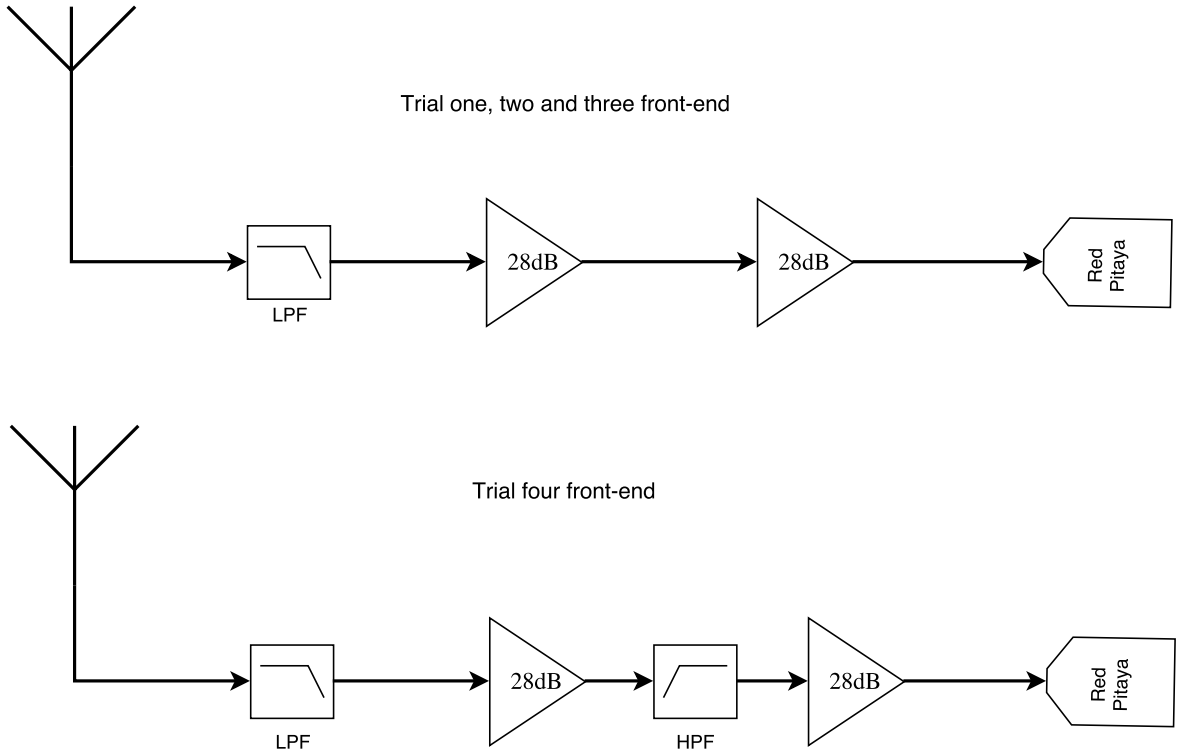


Figure 4.1: Antenna and front-end receiver chain for trials one, two and three (top) and trial four (bottom).

A sixth order Chebyshev low-pass filter is immediately after the antenna, with a -3 dB frequency of 42.5 MHz to limit noise into the amplifier and prevent it from saturating. The Chebyshev was chosen as it easy to implement and provides steep roll off than a Butterworth filter. The less smooth gain response of the passband is deemed to be acceptable for the application at hand as a particular frequency is only compared itself for riometry purposes. It had an insertion loss of 2 dB across the passband. Its response is shown in Figure 4.2 in terms of its S11 and S21 parameters. S11 is the ratio of power reflected to input power, while S21 is the ratio of output power to input power. For filters it is generally important for S21 to be as high as possible across the passband and as low as possible across the stopband. Likewise the S11 figure should be minimised across the passband and is less important over the stopband. The low-pass filter has a good cut-off and rejection outside the passband, and the S11 value is reasonably low, between -10 and -15 dB between about 20 and 40 MHz which is the region in which the demonstrator riometer should be operating. The filters were made by hand using custom-made Printed Circuit Boards (PCBs) and some of the inductors were hand wound.

Thereafter, for the first three trials, two Mini-Circuits ZFL-500LN+ HF amplifiers provide a maximum of 28 dB gain each for 56 dB total gain at the operating frequency [43]. For the last trial an additional high-pass filter was added between the two filters, as it was identified that an abundance of low frequency interference was present.

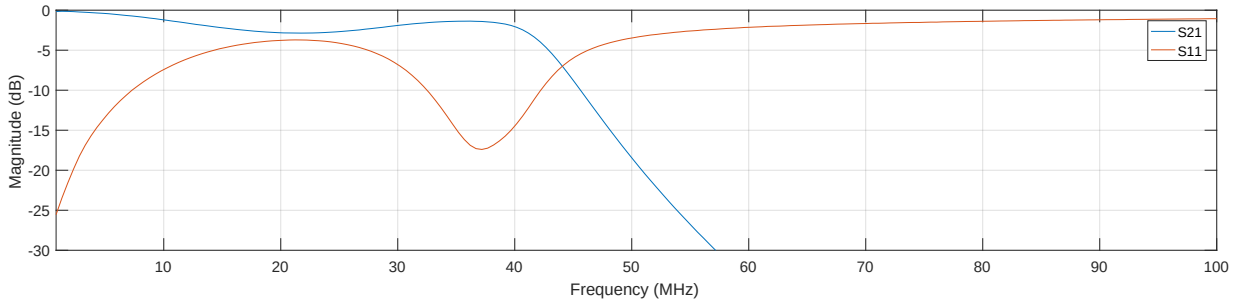


Figure 4.2: S21 and S11 of the low-pass filter.

Figure 4.3 shows the response of the high-pass filter in terms of S12 and S11. Over the passband about 2 dB of attenuation is seen, which is why the filter is shifted further along the chain, to reduce losses that might impact the sensitivity of the instrument. While the magnitude of the S21 of the filter reveals a sharp cut-off and good passband characteristics, the S11 of the filter is relatively high over the passband. At 38.2 MHz, nominally where the riometer should be operating, the S11 sits at -7.5 dB, which means that about 18 % of the power is being reflected. Ideally this should be below -10 dB.

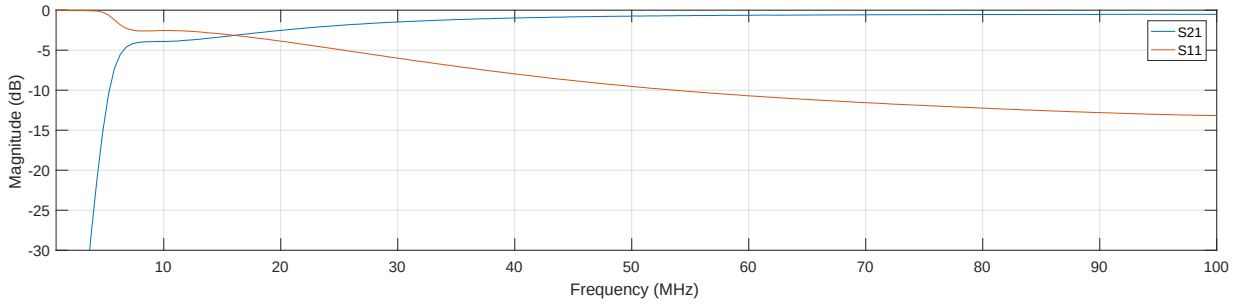


Figure 4.3: S21 and S11 of the high-pass filter.

Finally the combined response of the high-pass and low-pass filters when in series, without any intervening amplifiers, is shown in Figure 4.4. Looking at the S21 parameter one can see that the combined filter has a -3 dB cut-off between 16 and 41 MHz. The S11 parameter is overall a bit too high near the operating frequency, 38.2 MHz, hovering at around -7.5 dB here. At 24.7 MHz there is a strong dip in the reflected power down to -16.85 dB which corresponds with a very high S21 and indicates that this is the best frequency at which to operate these combined filters.

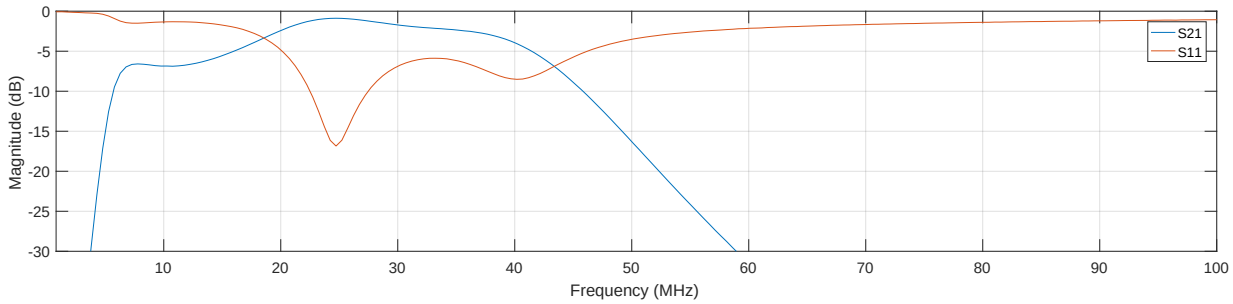


Figure 4.4: S21 and S11 of the combined low-pass, high-pass filter system.

Returning to the amplifiers, the noise figure of each low noise amplifier (LNA) is given as 2.9 dB by Mini-Circuits [43]. Given the insertion losses of the filter at 2 dB, the overall system noise figure of the cascaded components can be calculated to be 4.0 dB for trial one through three. For trial four, with the addition of the high-pass filter, the noise figure can be calculated to also be 4.0 dB, with the additional gain from the low noise amplifier before the high-pass filter keeping the noise figure from rising.

4.1.2 Antenna – Trials One and Two

The antenna itself is a simple half-wave dipole designed and built at UCT. A 1:1 balun was attached to the antenna to convert between the balanced coaxial signal and the unbalanced coaxial cable. Figure 4.5 shows a representation of the antenna. The dipole is designed to work at 42 MHz with a wavelength of 7.14 m. Therefore the length of the entire dipole is about 3.57 m.

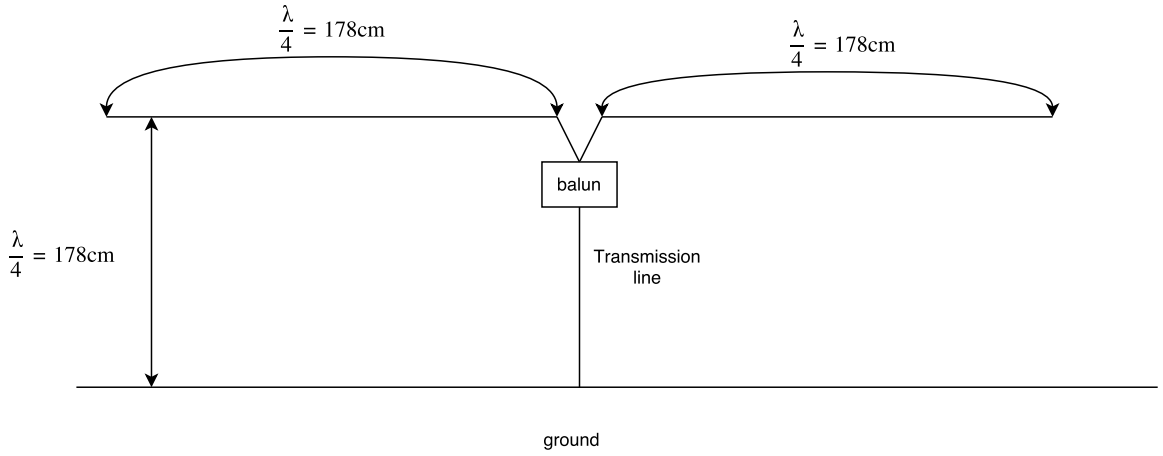


Figure 4.5: Dipole antenna setup for trials one and two – the wavelength (λ) used is 7.14 m.

The dipole is mounted a quarter of a wavelength above the ground to shape the beam pattern towards the sky. Simulations in FEKO showed that the beam pattern at the frequency of interest formed a roughly conical shape with a half power beamwidth of about 60° orthogonal to the ground.

The 1:1 balun design is shown in Figure 4.6. It features three coils wound around a core. The principle of operation is that the green and red coils are as straight feed throughs, one for the centre conductor and one for the shield conductor, each which is fed directly to one of the balanced connections. The blue coil is a reversed pick up which balances the dipole connections.

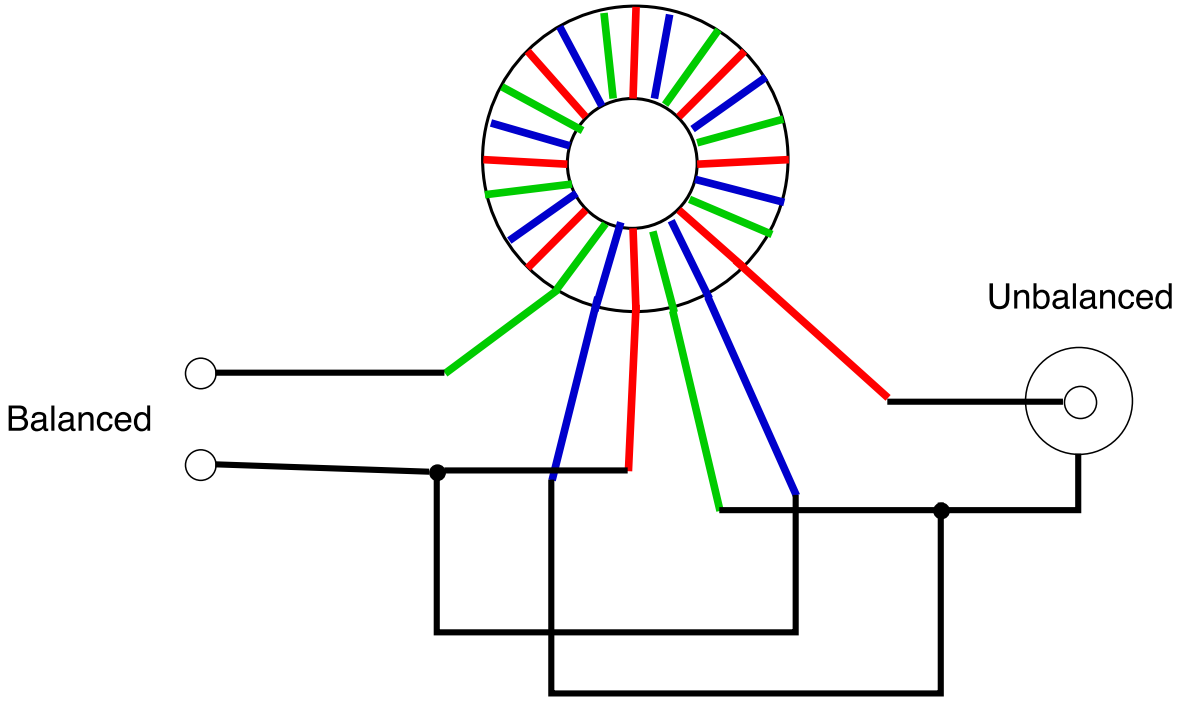


Figure 4.6: Balun used with the antenna in trials one and two. Each different colour line is a different coil of wire.

Figure 4.7 shows the S11 parameter of the antenna. The lower the S11 at a frequency the more power is transmitted to the front-end. The antenna is tuned to operate at 42 MHz where it achieves an S11 of -20 dB.

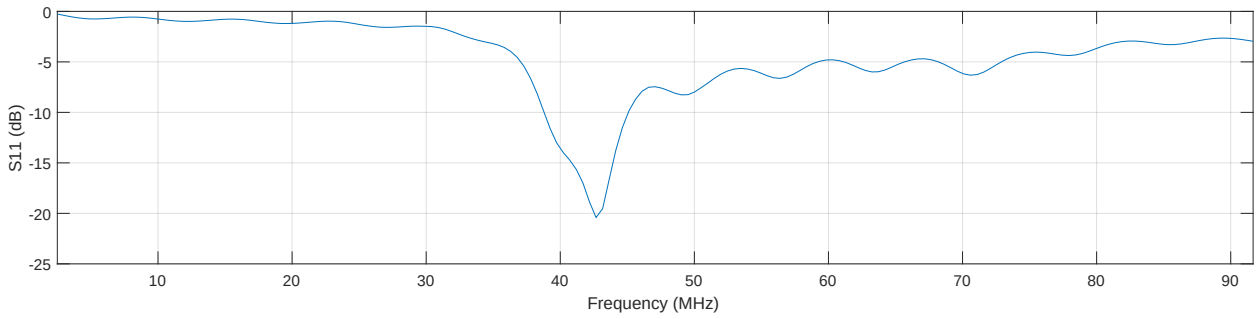


Figure 4.7: S11 parameter of the dipole antenna used in trials one and two.

4.1.3 Antenna – Trials Three and Four

For trial three the antenna detailed above was replaced with a wide-beam half-wave dipole antenna design used by La Jolla riometers, including a reflector [44]. The antenna was designed to operate at $38.2\,MHz$ to match the frequency of the riometer used by SANSA in testing (see Chapter 5). The actual dipole consists of two conductors, each $196\,cm$ long, mounted $120\,cm$ above the ground. The reflector is mounted halfway between the dipole and the ground. It consists of a single conductor slightly longer than half the wavelength. The combined result of the reflector and dipole is a beam-pattern that points largely towards the sky, with a wide beam-width of 60°. The design and dimensions of the antenna is shown in Figure 4.8.

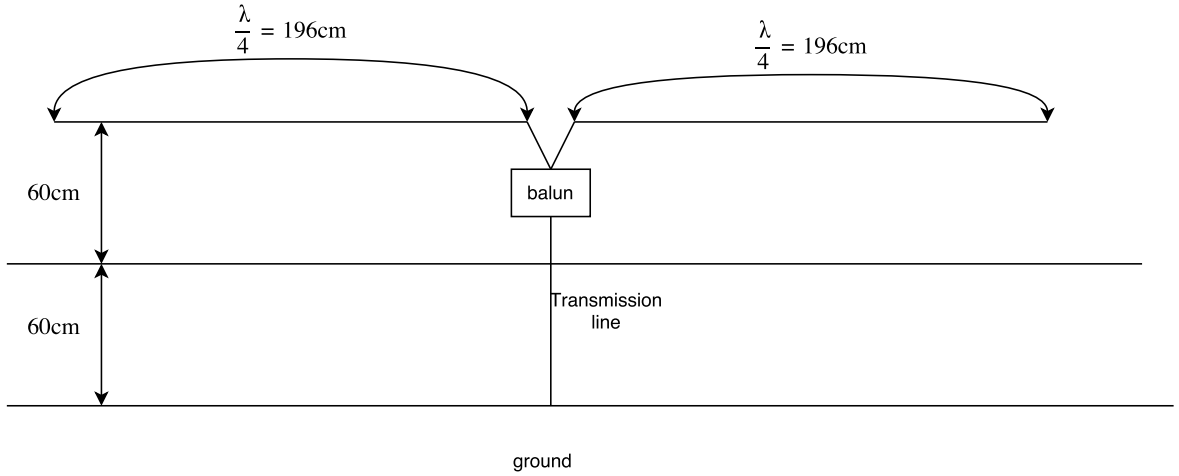


Figure 4.8: Design of the antenna used in trials three and four – the antenna is tuned to operate at $38.2\,MHz$.

Coaxial cable shielding was used for the conductors. These were tied taut at the correct height using rope between three wooden poles, with the middle pole used to support the feed. The wooden poles were hammered into the ground to provide a stable base. Additional stability was obtained by using guy ropes to support the wooden poles.

The balun performed the dual purpose of converting the balanced dipole signal to a signal carried in an unbalanced coaxial cable and providing impedance transformation between the $73\,\Omega$ dipole and the $50\,\Omega$ cable at $38.2\,MHz$. The balun of choice was a quarter wavelength sleeve balun. The balun design is shown in Figure 4.9. At the antenna side, one conductor connects to the inner conductor of a $75\,\Omega$ coaxial cable, while the other connects to the shield of that cable. A quarter wavelength of the shield of a different coaxial cable is sleeved over the plastic jacket of the other one and soldered at the coaxial end. The sleeve is not joined to anything at the dipole end. From the end of the coaxial cable of the balun, a $50\,\Omega$ cable or load can be connected.

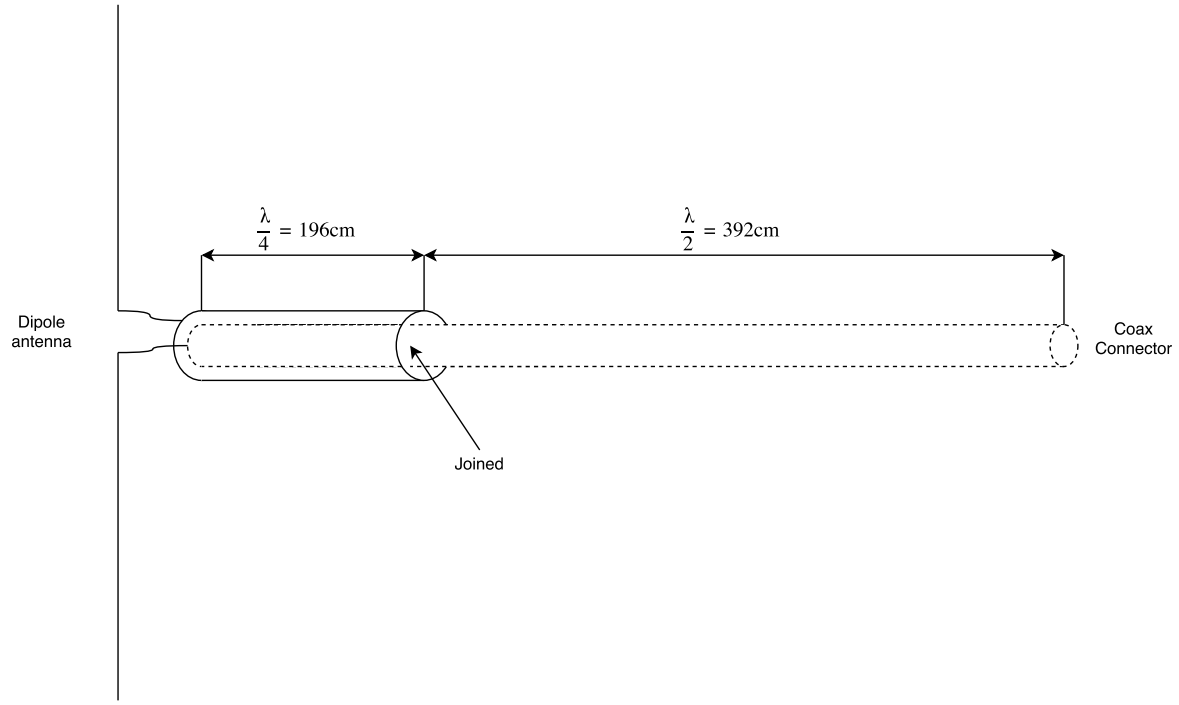


Figure 4.9: Design of the sleeve balun used by the antenna in trials three and four.

The combined S11 of the antenna-balun system as measured at the coaxial end of the balun is shown in Figure 4.10. The antenna has a good response at around 38 MHz with an S11 of almost -15 dB. The 3 dB bandwidth of the antenna is about 12 MHz between 32.5 MHz and 44.5 MHz. An additional resonance with an S11 of -7 dB is present at 14 MHz.

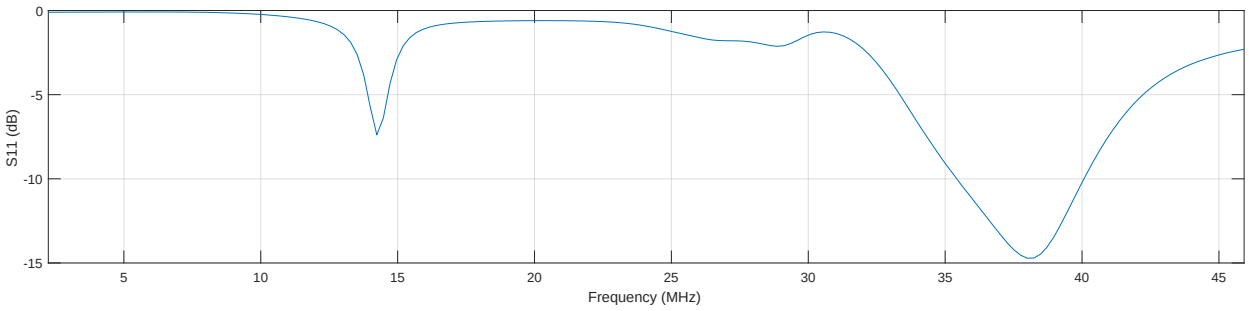


Figure 4.10: S11 parameter of the antenna used in trials three and four.

4.1.4 Representative Power Budget

The required gain was calculated by examining the typical noise power one would receive through the antenna. From Figure 3.3 the average noise power at 38 MHz is 2.9×10^4 K. The low noise amplifier attenuates everything above 42 MHz and the antenna only receives signals in a narrow band around its operating frequency at 38 MHz (assuming the antenna for trials three and four was used). As seen

in Figure 4.10 the antenna mainly picks up signals between 32 and 44 MHz. Therefore the combined antenna-filter response will have a bandwidth of 10 MHz between 32 and 42 MHz. Assuming a flat noise floor between 32 and 42 MHz, the received noise power will be -84 dBm. The ADC of the Red Pitaya can detect a maximum power of 10 dBm and a minimum power of -75 dBm, which means that 56 dB of gain was sufficient to push the ADC input power up to -28 dBm. Lying in the middle of the ADC's dynamic range allows clearance for interfering signals to be detected without saturating the amplifiers, which have a maximum output power 5 dBm, and also allows for strong absorption events to be detected.

4.2 The Red Pitaya

The Red Pitaya is a credit card sized System On a Chip (SOC), meaning it has both a CPU and FPGA. It also features two 14 bit ADC inputs and two 14 bit DAC outputs. These both sample at 125 $MSPS$ making them ideal for HF signal processing applications. Figure 4.11 details some of the features of the Red Pitaya. It has a dual core ARM Cortex A9 which is powerful enough to do some backend processing, while the majority of the processing is done on the Xilinx Zynq 7010 FPGA. A more detailed overview is available from its hardware specifications [2].

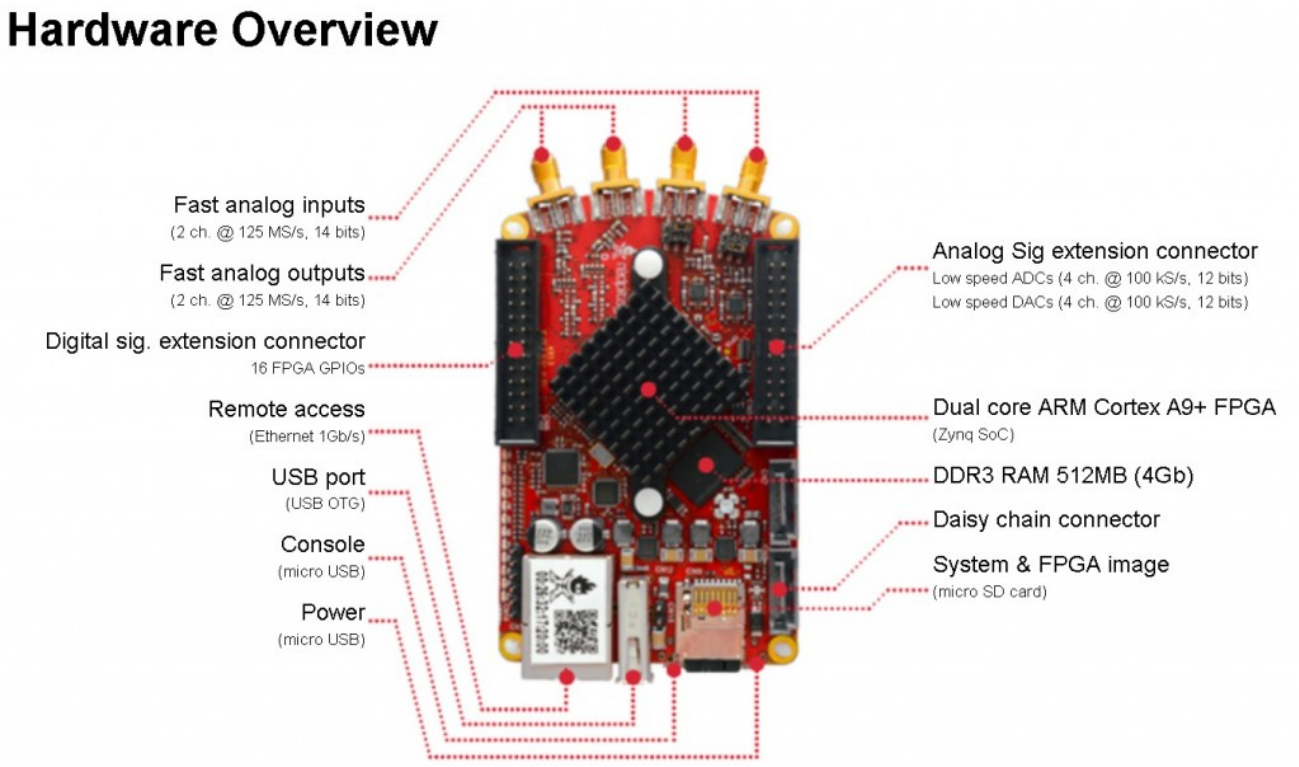


Figure 4.11: Features of the Red Pitaya SOC [45].

It can be programmed via JTAG, though the more common method is to Secure Shell (SSH) into the Red Pitaya and upload a bitstream by connecting to the provided ethernet port. Multiple Red Pitayas

can be daisy chained together allowing for multiple input systems to be implemented and synchronised.

The Red Pitaya was chosen because of its suitability for HF Radar, its relatively cheap cost and its accessibility as a prototyping system. Many programmers and engineers have already taken to using the Red Pitaya, opening up a wealth of useful information about it on the internet. Thus it makes for an easy prototyping device.

Communication between the operating system running on the Cortex CPU and the FPGA takes place via a shared memory block. The FPGA is allocated a certain portion of the memory to which it can write, preventing conflicts with the operating system. Both the operating system and the FPGA can read from this memory, allowing for data to be transferred from the FPGA to the operating system. Likewise, there is a portion of memory that only the operating system can write to, but the FPGA can read, allowing transfer in the opposite direction.

4.3 Programming the FPGA and the AXI Protocol

The ZYNQ 7010 was programmed using Verilog with Vivado as a development environment. Using a top-down approach, the designer adds blocks, called IP cores, to a block diagram and connects output ports to input ports to chain multiple cores together. Multiple blocks can be grouped together to form a hierarchy reinforcing the top-down approach. At the bottommost level one programmes each core using Verilog or VHDL. This is illustrated in Figure 4.12.

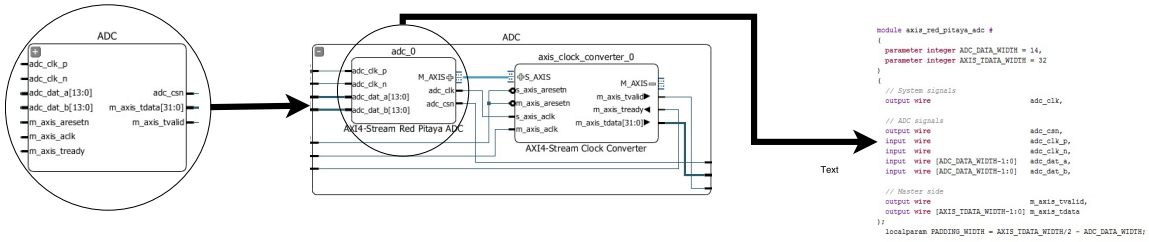


Figure 4.12: The hierachy used in Vivado, with the topmost level on the left and the bottommost on the right. The middle stage can be further nested depending on the design.

In this project the Advanced eXtensible Interface (AXI) protocol was mostly used to transfer data between blocks. The AXI protocol is defined by ARM and has a number of different levels [46]. It defines data transfer between a slave and a master, where the slave receives data from the master. The AXI4-Stream interface is the most basic interface and is widely used to transfer data between cores in this project. As shown in Figure 4.13, the interface defines a master and a slave core. The master transfers data to the slave via a simple handshake, only transferring data when the slave is ready to receive it, indicated by *t_ready* being held high. The master then asserts *t_valid* and puts valid data onto the *t_data* line.

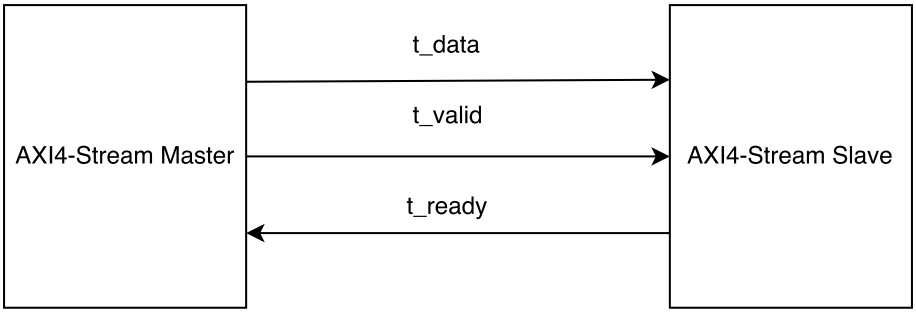


Figure 4.13: The AXI4-Stream interface. The slave asserts *t_ready* when it is ready to receive data and the master asserts *t_valid* when it has valid data to send. When both are asserted the data is streamed from the master to the slave.

Figure 4.14 shows how this handshake takes place. Data continues to be transferred until either no more valid data is available, in which case *t_valid* is pulled low, or the slave is no longer able to accept more data, in which case *t_ready* is pulled low. The AXI4-Stream interface is best suited to transferring data in signal processing applications where data flows through the system seamlessly.

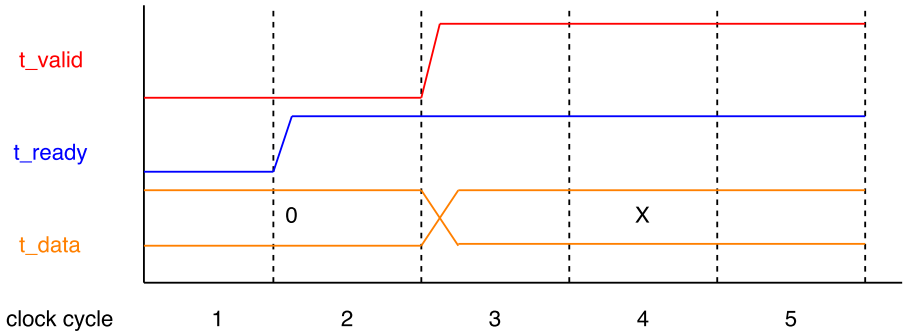


Figure 4.14: The AXI4-Stream interface. The master waits for *t_ready* to be held high before asserting *t_valid* and *t_data*.

A number of signal processing algorithms have been designed by Xilinx and made available, such as FIR filters and FFTs, and many of these cores were used in this project to speed up the design process. In addition, Demin Pavel designed a suite of cores specifically for the Red Pitaya to perform tasks such as reading from the ADC and writing to the Red Pitaya's RAM. These cores, which he made freely available online, have also been used in this project [47].

4.4 Riometer Designs

Two different riometer designs were implemented, the scanning riometer and the spectral riometer. The scanning riometer design and its conversion is explained first before detailing the spectral riometer design. The final section shows the integrated scanning and spectral riometer design.

4.4.1 Scanning Riometer

A problem with the basic riometer operating at a single frequency is that other HF signals can enter into the operating band of the riometer, creating a false impression of increased noise power. One of the solutions for this is the scanning riometer. It works on the same operating principle as the standard riometer, but scans across a small range of frequencies around the operating frequency selecting the minimum power in this frequency range as the power it records. Thus, should a free band be available, it will always sample the noise floor. One of the requirements for the scanning riometer to be accurate is that the scanning frequency range is narrow enough that the received noise power varies only a small amount over that frequency range. The tradeoff of the scanning riometer is reduced integration time due to the scanning.

Reduced integration time results directly in less instrument sensitivity or, alternatively, if the integration time is constant, then the riometer resolution will degrade. Thus, the riometer process was split into two sections: scanning and recording. In the scanning section one scans across a range of frequencies with a reduced integration time and selects the one with the minimum power. In the recording section the riometer is run at this frequency with an increased integration time. Thus the scanning part of the scanning riometer uses less overall integration time.

Figure 4.15 demonstrates the operation of the scanning riometer. If the riometer is set to scan between 38.5 MHz and 39.5 MHz in 19 increments then it will identify the minimum power in this range, at 38.64 MHz in this case, thereby avoiding the man-made noise at 38.94 MHz. Only if interference is present across the entire scanning range will the noise floor not be measured.

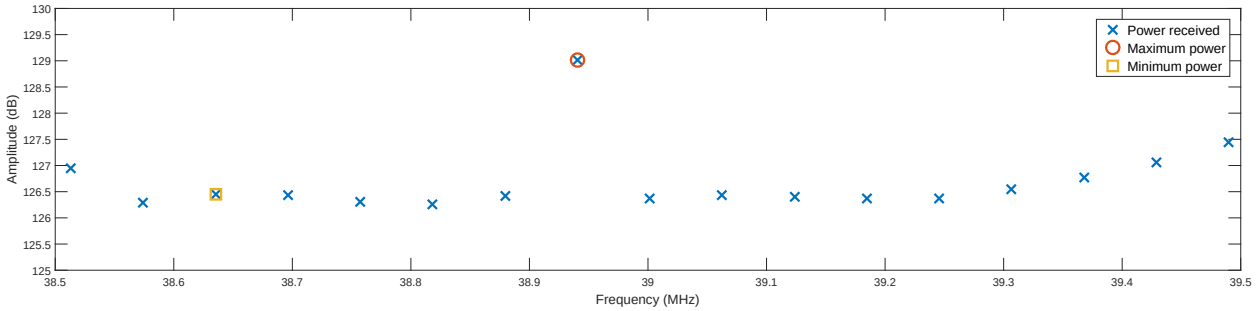


Figure 4.15: Example of the operation of the scanning riometer. It selects the minimum power in the scanning range, in this case, in this example at 38.64 MHz. The man-made noise spike at 38.94 MHz is avoided.

This section presents an overview of the digital implementation, as done on the Red Pitaya, of the scanning riometer. Figure 4.16 shows the layout of the scanning riometer. Starting at the top left hand side, the RF input from the front-end is converted to the digital domain by the ADC. This is fed to the first multiplexer which selects between four inputs to the scanning riometer: the ADC itself; a test signal; a noise generator and a sinusoid generator. The test signal produces three sinusoids of different

amplitudes summed together and the noise generator produces a white gaussian noise signal which can be programmed to have a constant power or increase its standard deviation linearly over time. The sinusoid generator is able to produce a variable frequency sinusoid at a fixed amplitude. The latter three signals are used for testing purposes.

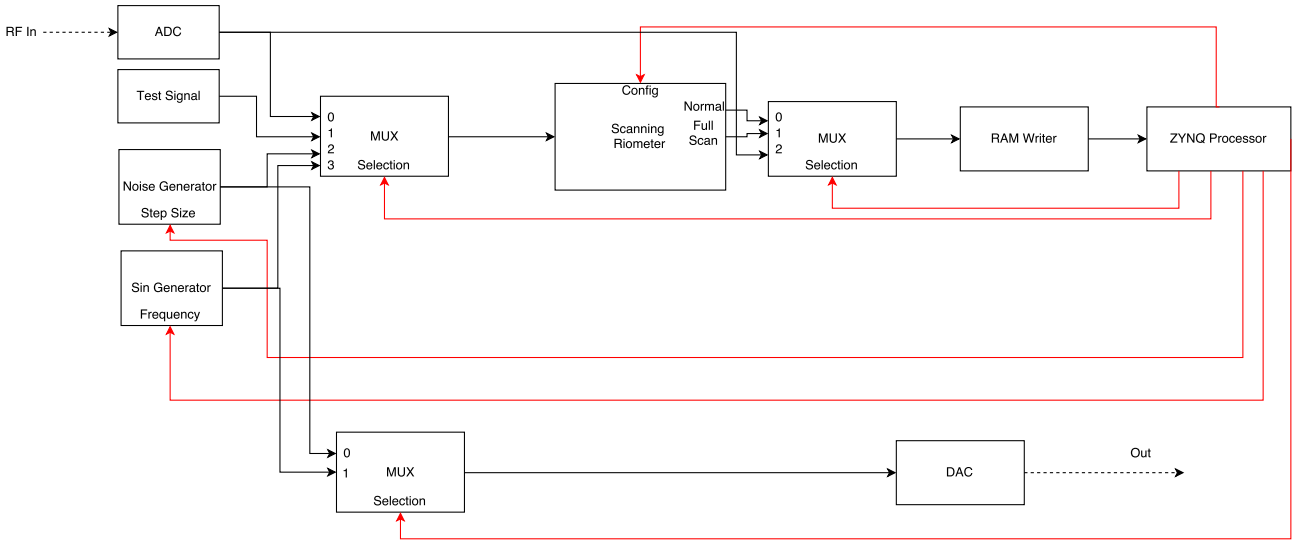


Figure 4.16: High level block diagram of the scanning riometer. **Black** lines represent data flow in the system, while **red** shows control lines.

The scanning riometer produces two streams of outputs: a normal operation output and a full scan output. The normal output is the standard riometer output, which is the power after integration of the minimum power detected in the specified range of operation. The scanning mode, on the other hand, outputs both the normal output, and the results of the scanning process. Both these outputs, and the ADC values, are fed to a second multiplexer which selects which stream to write to the ZYNQ processor's RAM. In addition, the noise generator is fed to the DAC so that its output can be used as a feedback testing and calibration source for actual operation.

The purpose of the scanning riometer is to detect the power of the noise floor in the frequency range of interest. Figure 4.17 shows a more detailed block diagram of the scanning riometer. The data from the input multiplexer is multiplied by a sinusoid to convert it to baseband. The frequency of the sinusoid is varied by the controller. During the scanning phase it is stepped in increments between the lowest and highest frequency of interest. Once the lowest power level has been detected, the controller fixes the frequency to that of the lowest power and records for an extended period of time before outputting the integrated result as a single sample.

After converting to baseband, the signal is downsampled using a fourth order Cascaded Integrator-Comb (CIC) filter, which is immediately followed by a CIC compensation filter to flatten the spectral response of the CIC filter. The downsample factor is 500, leaving an effective downsampled rate of $250\,kHz$. In order to allow for a variable bandwidth, this is fed to a 64 bin FFT. Therefore each bin represents $1.95\,kHz$ of bandwidth. By summing a set number of consecutive bins a variable bandwidth

system can be implemented. This is what the bandwidth selector does, after converting each FFT bin to its equivalent power value: $Power = Im^2 + Re^2$, where Im and Re is the imaginary and real component of the power respectively. This can be shown to be proportional to the power in the time signal using Parseval's theorem.

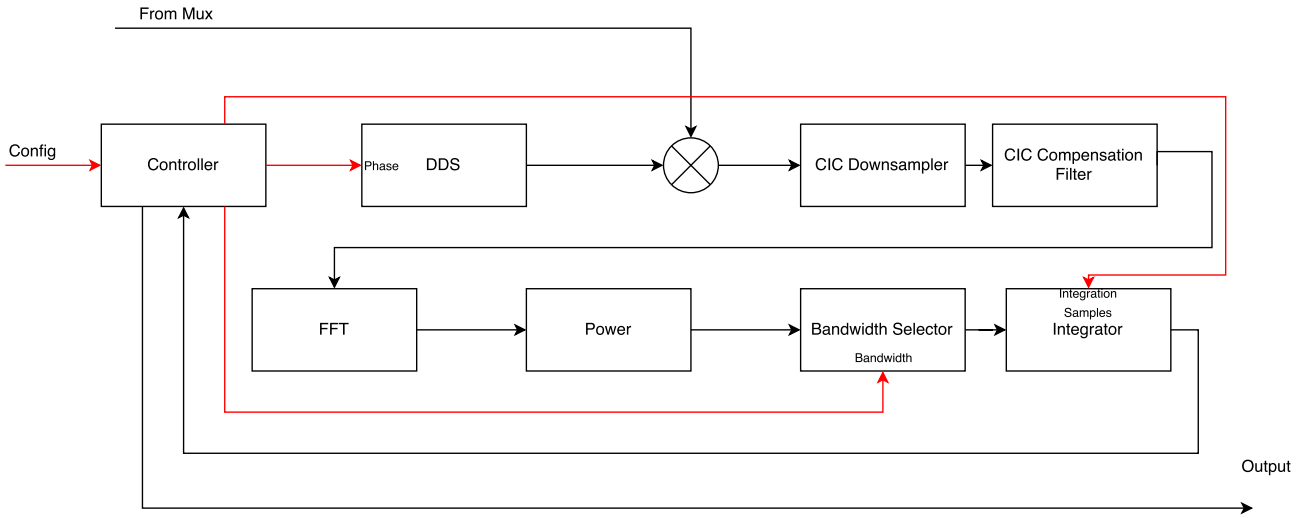


Figure 4.17: Block diagram of the inner working of the scanning riometer. **Black** lines represent data flow in the system, while **red** shows control lines.

Thereafter the power is integrated over a set number of samples. The integration time is usually set lower for the scanning operation and increased for the recording operation to increase the instrument sensitivity. The main purpose of the controller is to change the frequency of the Direct Digital Synthesis (DDS) device and the integration time of the integrator for the scanning and recording stages.

While most of the signal processing blocks presented in Figure 4.17 are fairly standard, the Cascaded Integrator-Comb filter requires some explanation. The Cascaded Integrator-Comb filter is a combination of a finite impulse response filter and a decimator (although interpolation can also be done). It is optimised for digital signal processing so that it only uses addition and subtraction for its operations and no multiplication. The filter consists of one or more cascaded integrators, followed by a downsampler and then the same number of comb sections as the number of integrators. Essentially this forms the equivalent of a moving average filter with the filter response:

$$H(z) = [\sum_{k=0}^{R-1} z^{-k}]^N, \tag{4.1}$$

where R is the decimation factor and N is the number of stages – an integrator-comb pair – in the filter. Figure 4.18 shows a block diagram of the structure of the CIC filter. Both the integrator, downsample and comb sections are present. The shape of the filter is influenced by the number of sections in the filter.

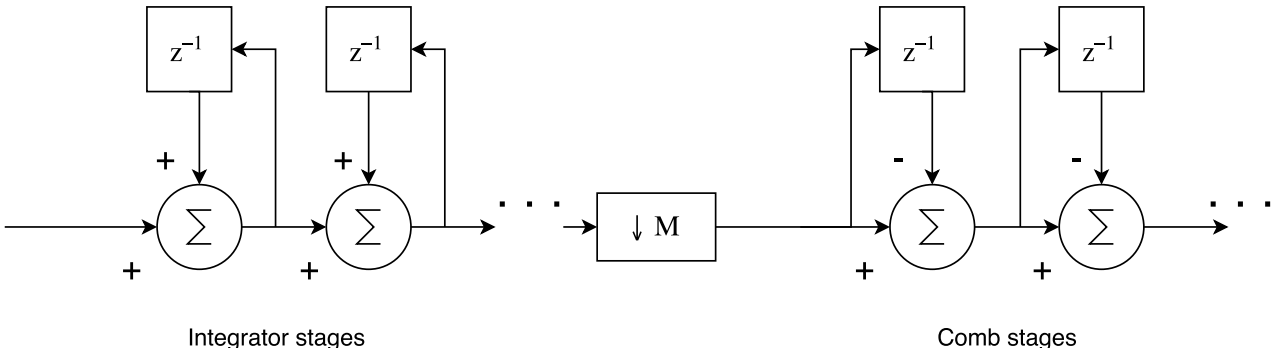


Figure 4.18: Block diagram of CIC downsample filter downsampling by a factor M.

After passing through the CIC filter, the spectrum is no longer flat and requires a CIC correction filter to obtain a flat spectrum. For the scanning riometer this is important because an FFT is applied after the compensation filter in order to allow for a variable bandwidth. If no correction is done then most of the power is shifted towards the lower end of the spectrum, skewing the results of bandwidth selection.

Figure 4.19 shows the magnitude of the CIC filter response used in this project plotted against normalised frequency. The CIC filter had a downsample factor of 500 and featured five sections for a sharper cut-off. The response is plotted before downsampling. After downsampling the frequency response will be limited to a normalised frequency of 0.002, and some aliasing will occur.

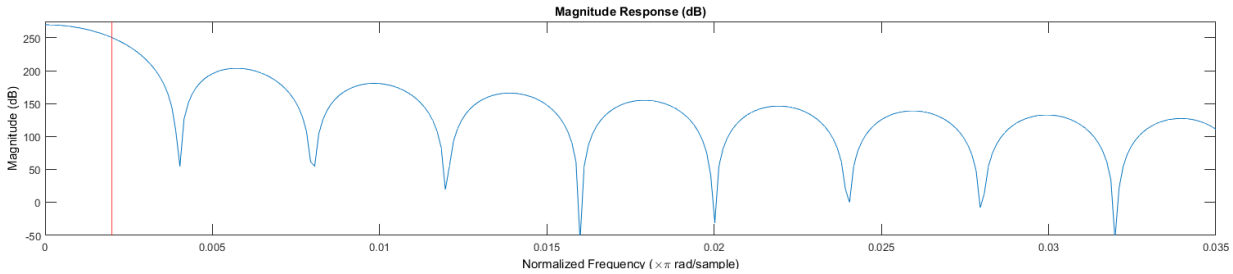


Figure 4.19: Normalised frequency response of the CIC filter. The red line indicates the maximum frequency after downsampling. Note the reduced magnitude of the response toward the cut-off frequency and the sidelobes that get aliased into the main spectrum after downsampling.

The CIC correction filter's response is shown in Figure 4.20. It should be able to flatten the spectrum over the passband, while also providing some further filtering. Since the incoming sample rate to the CIC filter will be 125 MHz its output sample rate will be 250 kHz. Taking into account the Nyquist sampling criterion and the cut-off of the CIC compensation filter, the maximum frequency after the CIC filter and its compensation filter will be 62.5 kHz.

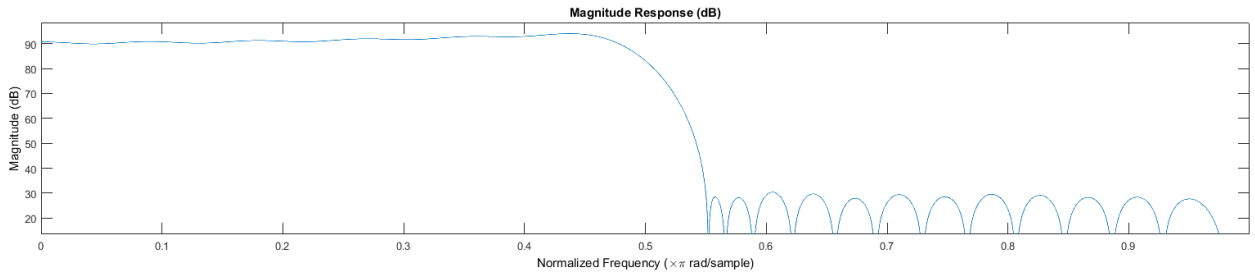


Figure 4.20: Normalised frequency response of the CIC compensation filter. The magnitude increases towards the cut-off to compensate for the decreasing magnitude of the CIC filter towards this frequency.

The CIC compensation filter has a frequency response that is the inverse of the CIC filter over the passband of the compensation filter. Thus, over the passband, the magnitude of the frequency response of the compensation filter is:

$$|F_{CIC-compensation}| = \frac{1}{|F_{CIC}|}. \tag{4.2}$$

In the frequency domain, the CIC filter response and CIC compensation filter response are multiplied out and will cancel with each other, leading to the overall response of the combined CIC and CIC compensation filter system being flat over the passband.

4.4.2 Dicke Riometer Converter

A Dicke riometer corrects for changes in the gain and bandwidth of the front-end using a known noise source to calibrate the receiver. Most of the processing in this riometer is done digitally, which means the analogue front-end is relatively small compared to a traditional riometer, containing only anti-aliasing filters, noise limiting filters and amplifiers. This means that gain variations are likely to be smaller and gain drift could take place over a longer period of time. The receivers at SANAE IV share this design and so the same principle applies for them.

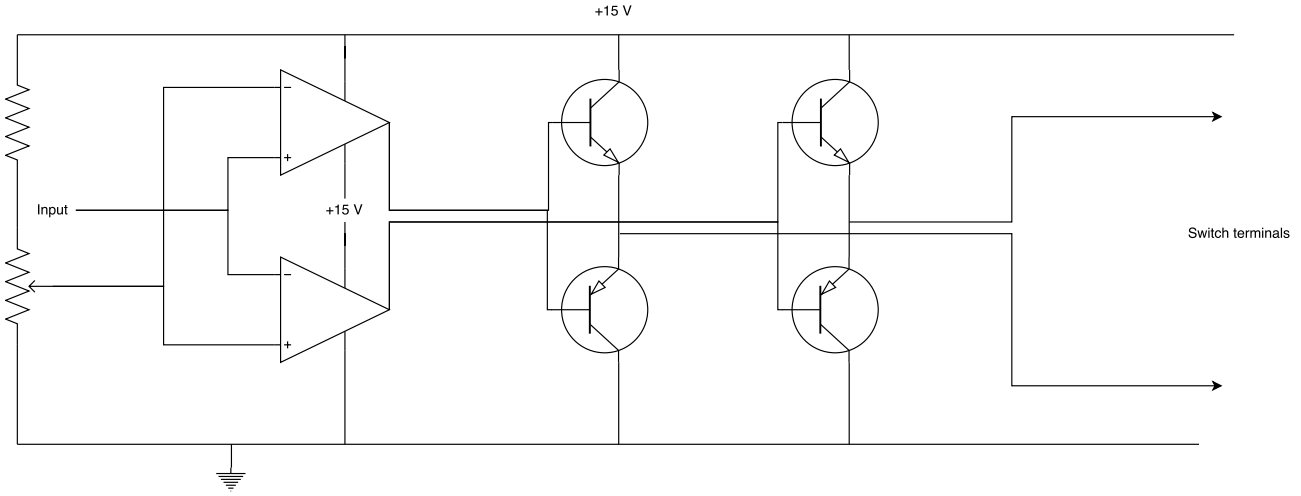


Figure 4.21: Switch driver circuitry for the Dicke riometer.

However, in the interest of increased accuracy, a Dicke riometer implementation has been completed. At the front-end this is just a switching unit, controlled by the FPGA, to switch between the antenna and a stable and known noise source. Every few integration cycles the front-end is switched onto the noise source for one cycle, yielding a sample of the noise power through the analogue front-end. Any changes in this power level can be used to calibrate out gain changes retroactively in post-processing. In addition, since the duty cycle of this is programmable, the time spent staring at the calibration source can be decreased for finer resolution or increased for finer accuracy.

The switch used is the HP8761A [48]. It is capable of operating at any frequency from DC to 18 GHz with an insertion loss of less than $0.2\,dB$ at HF and isolation greater than $60\,dB$. There are two limitations of the device: its switching speed and the fact that it needs a driver circuit. In testing it was found that the maximum switching frequency is about 20 Hz, well below the hundreds of Hz range a typical riometer would operate at [17]. However, considering the inherent stability of the digital riometer's front-end, a switching rate in the Hz range was found to be acceptable for tracking long term changes. In order to control the switch from the 3.3 V and limited current provided by the Red Pitaya GPIO pins, a driver circuit was constructed and is shown in Figure 4.21. The opamp stage converts the 3.3 V provided to 15 V. The HP8761A switches when the voltage across its terminals switches polarity. The two opamps in the first stage allow this to happen. There are then two driver circuits that provide the current required by the switch.

4.4.3 Spectral Riometer

A spectral riometer takes absorption readings over a range of frequencies. Figure 4.22 shows the implementation that was used in this project. Much like the scanning riometer implementation, an input multiplexer selects between the ADC, a test signal, a white gaussian noise generator and a variable frequency sinusoid generator. The test signal, noise generator and sinusoid generator are all

used for testing purposes. The input is then fed to both a 2048 point hamming window function and the second input of a separate multiplexer. This allows the operator to select between windowing before the FFT or not. The output of the second multiplexer is fed to a 2048 point FFT. Given the sampling frequency is 125 MHz, each frequency bin has a bandwidth of 61 kHz. Thereafter the power in each frequency bin is detected and integrated over each frequency bin for a set number of iterations. The output is essentially an integrated power spectrum which is written directly to RAM.

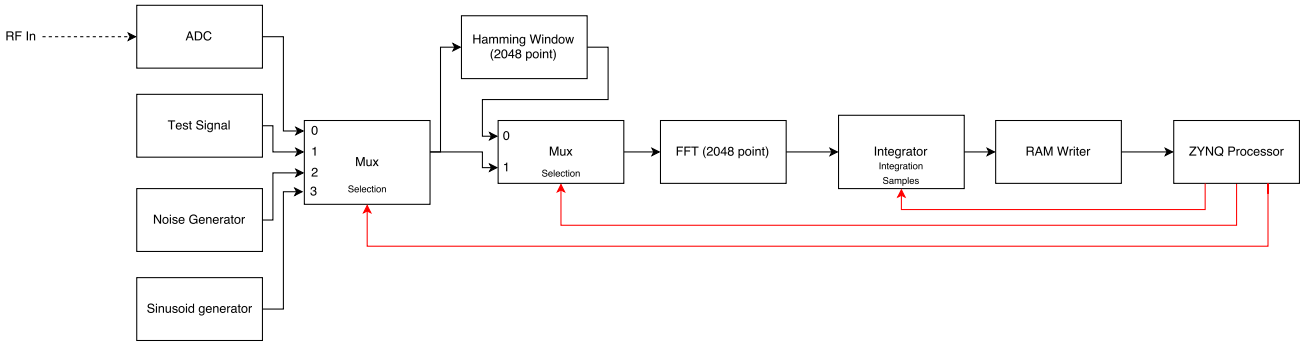


Figure 4.22: Spectral riometer block diagram. **Black** lines represent data flow in the system, while **red** shows control lines.

If, for instance, the test signal is injected into the spectral riometer, then the output over one integration cycle can be seen in Figure 4.23.

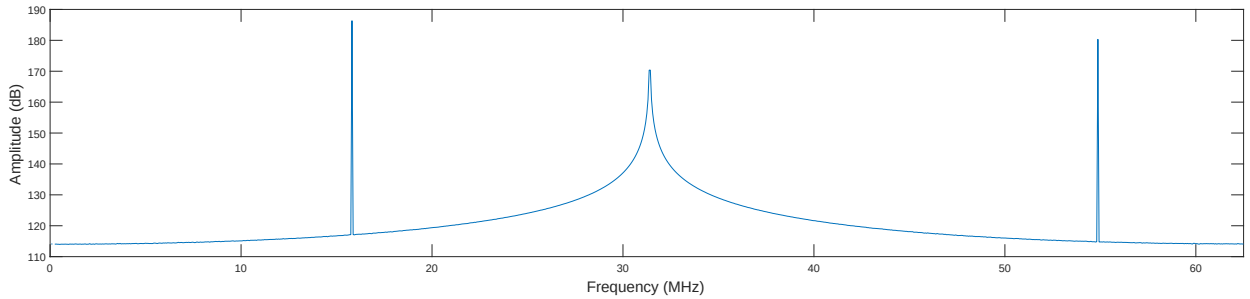


Figure 4.23: Output of the spectral riometer for the test signal. The peaks are those of the test signal, sinusoids at 15.68, 31.23 and 54.76 MHz respectively.

When adding the functionality of the Dicke riometer to the spectral riometer, the calibration signal is run through the FFT and integrator for a single integration cycle. The calibration takes place for each individual frequency bin, since the gain of the front-end could change over frequency as well, especially across the full 62.5 MHz that the spectral riometer operates over.

While most of the blocks presented in Figure 4.22 represent well known DSP functions with optimised implementations, the integrator was created from scratch and is significantly more complicated than a standard 1D integrator. Unlike the scanning riometer where a simple accumulator is appropriate for integration, the spectral riometer requires summation to take place across consecutive 2048 point

FFTs. The solution to this is to incorporate a First-In-First-Out (FIFO) buffer into the summation process. Figure 4.24 represents this process. After the first FFT is complete it outputs a 2048 point spectrum with one point, consisting of a real and imaginary part, outputted each clock cycle. This is converted to the absolute value squared ($Im^2 + Re^2$) and fed to the integrator. The first 2048 points are written directly to the FIFO. Thereafter the integrator simultaneously reads from the FIFO and the summer, adds the two values it reads and then writes the result back to the FIFO. The FIFO remains full throughout this process, having a value written to it and read from it at each clock cycle. After a set number of 2048 point segments have been integrated, the integrator continues reading from the FIFO, but diverts the values to its output while also continuing to read from its input, writing these values directly to the FIFO without summing. Thereafter, the integrator oscillates between the state of summing the values from the FIFO and its input, and outputting the integrated spectrum.

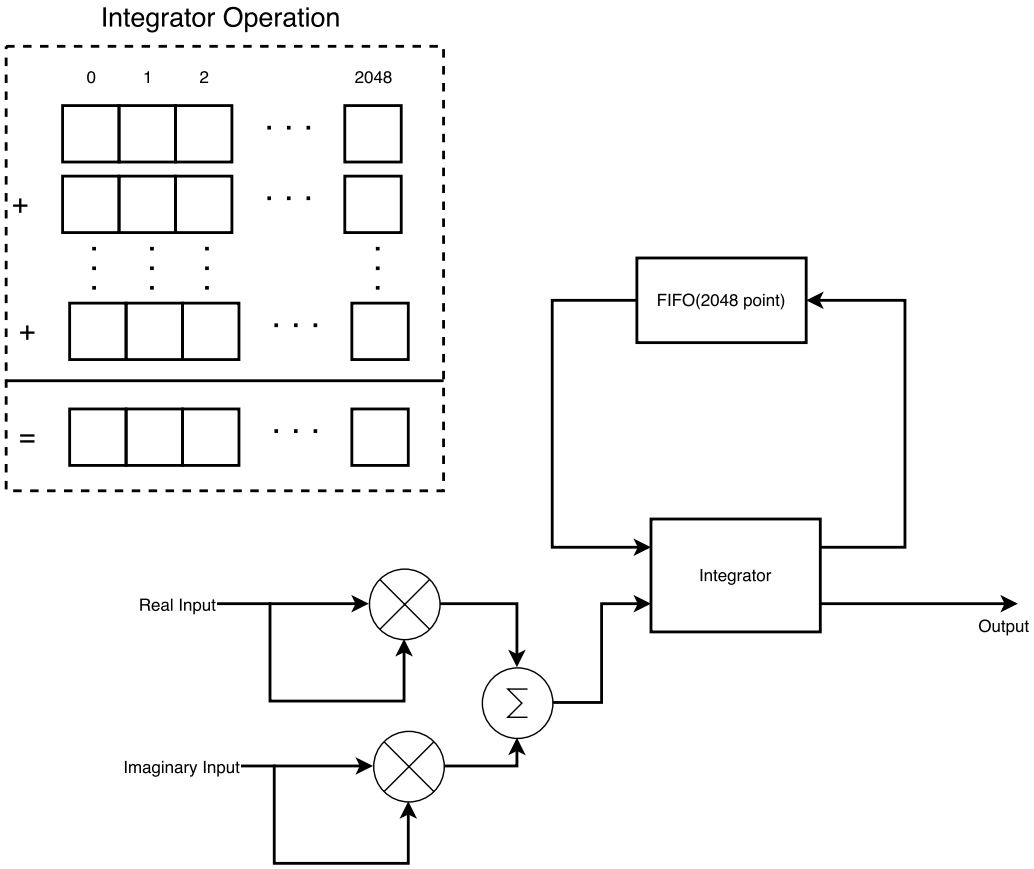


Figure 4.24: Integrator operation and block diagram.

Spectral leakage can be an issue for FFT based spectral riometers which require that the energy in a particular bin be an accurate representation of the energy at that frequency and is not overly influenced by surrounding frequencies. However, spectral leakage can be reduced by windowing the input: multiplying it by a function that tapers towards the edge. As an example consider the 2048 point FFT of a sinusoid at $37.5\,MHz$ which has been sampled at $62.5\,MHz$. Before windowing the bin immediately to the right of the main peak receives the full energy in the peak minus $1.76\,dB$, while the bin on the left receives the full energy minus $5.43\,dB$. Even the bins furthest from the peak receive the full energy minus $29\,dB$. If a hamming window is applied before taking the FFT then the energy in

the main peak is reduced slightly, but the spectral leakage is also greatly decreased, as shown in Figure 4.25.

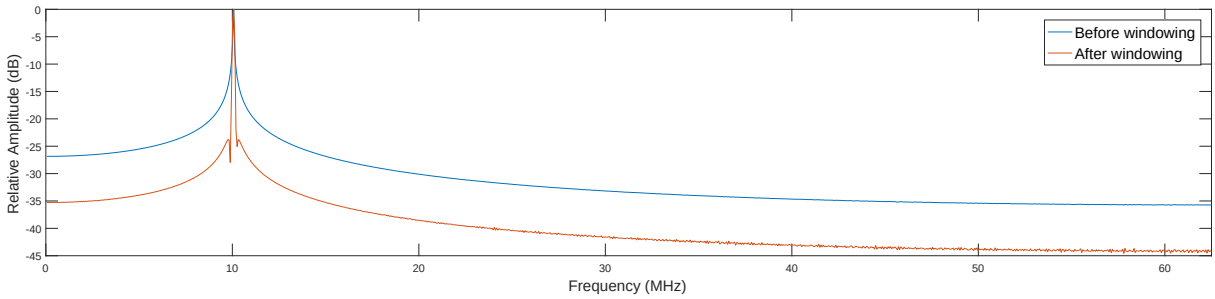


Figure 4.25: Windowing can be used to decrease spectral leakage in the spectral riometer.

Windowing functionality for the spectral riometer was only implemented for the fourth trial. As a result spectral leakage was a factor and did influence the results for the first three trials. Only hamming window functionality was added by the end of the project, but other windows can easily replace the hamming window by updating the ROM on which it is stored in the FPGA.

4.4.4 Integrated Scanning and Spectral Riometer

Figure 4.26 demonstrates how both the scanning and spectral riometer can be put onto the same FPGA. Since both accept the same data inputs, they are both connected directly to the input multiplexer. An output multiplexer selects between the outputs produced by the scanning riometer, spectral riometer and the ADC. The selected stream is written to RAM for more processing down the line. The first two outputs are the normal and full scan modes of the scanning riometer respectively. The third output is the ADC and the fourth output is the spectral riometer. It should be noted that data samples are 128 bits wide. As a consequence of this, and the fact that only 32 bits at a time can be written, some samples are discarded when writing directly from the ADC. Thus the ADC output channel is not very reliable.

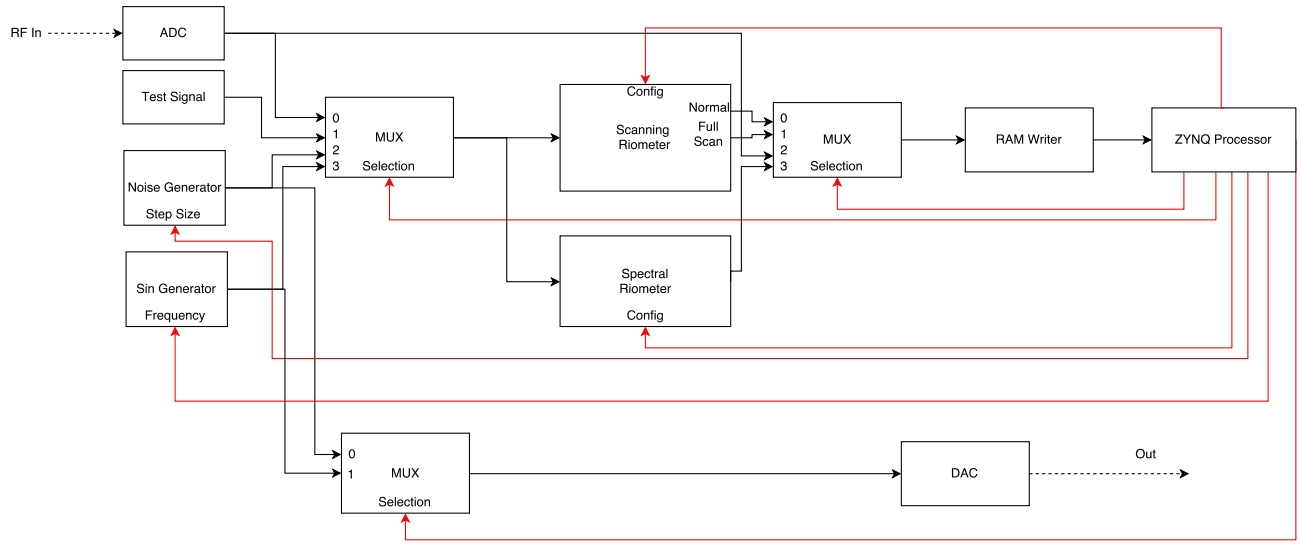


Figure 4.26: Block diagram of the scanning and spectral riometers integrated into a single unit. **Black** lines show data flow and **red** shows control lines.

Of interest is that the scanning riometer is capable of working as a spectral riometer. The drawback of operating it in this mode is that the integration time is cut by a factor of the number of frequency bins as compared to the spectral riometer. Thus, despite its increased flexibility in having variable bandwidths and operation frequency, it would be a less sensitive instrument for spectral riometry. Likewise the spectral riometer can be used as a scanning riometer, with the scanning taking place in post-processing. However, the drawback of this method is that more data must be temporarily stored before post-processing and the bandwidth is fixed at 61 kHz. The full layout of the design as it appears in Vivado can be found in Appendix A.

4.5 Additional Modules

Figure 4.26 shows a number of extra modules that are not strictly related to the function of the scanning or spectral riometer. The function of each of these modules is explained in this section. The additional modules are a noise generator, test signal generator and variable frequency sinusoid generator which can all be used for testing and calibration.

4.5.1 Noise Generator

Rather than develop a white gaussian noise generator, a readymade core from OpenCores was used [49]. It is quoted to produce a random distribution up to 9.1 σ. In testing it was found to produce a gaussian distribution with a flat power spectral density, tapering off towards the Nyquist frequency as expected. This module produced 16 bit noise at a fixed standard deviation. In order to make it a variable noise source some additional logic was included to multiply by a scaling factor and then bitshift the result

back down so that the maximum standard deviation is 16 bits. The noise source is fed back internally and through the DAC to allow for some basic testing of the riometer. For example, one test scaled the standard deviation of the noise linearly, which means the noise power should scale quadratically. This was fed to the input of the scanning riometer and the results observed are seen in Figure 4.27. Since noise power scales with σ^2, the power measured increases quadratically, as expected.

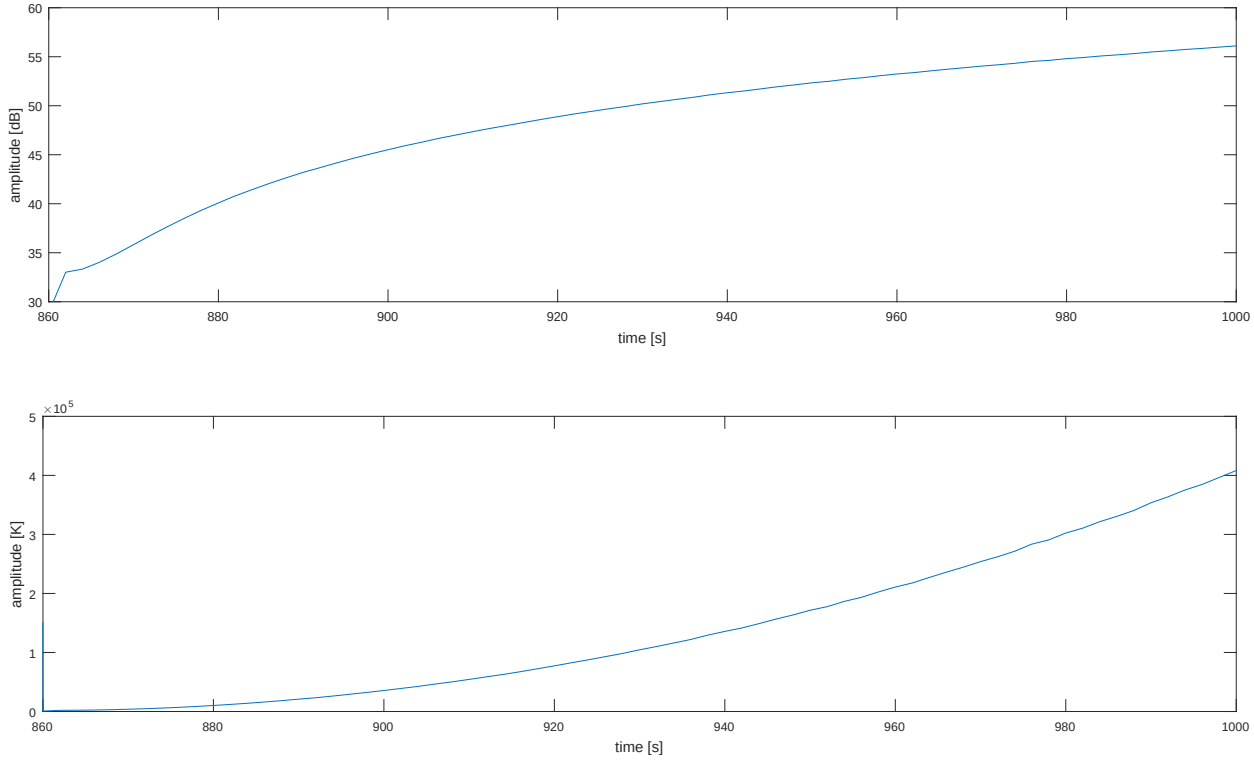


Figure 4.27: Result of quadratically increasing noise power test.

These results help to establish that the riometer is working properly as a power detector at the very least. Other tests shown later will be used to demonstrate the ability of the system to skip around man-made noise by using the scanning operation and changing the frequency slightly.

4.5.2 Test Signal

The test signal modules produces the sum of three sinusoids at different frequencies and with different amplitudes. Figure 4.28 shows the frequency spectrum of the test signal as computed using the scanning output of the scanning riometer across the frequency range of 0 to 62 MHz. The three sinusoids have frequencies of 15.68, 31.23 and 54.76 MHz, with amplitudes spanning 14, 12 and 13 bits respectively. The reason for the slightly spiky response near the actual peaks, as compared to the equivalent scanning riometer output in Figure 4.23, is the sidelobes in the CIC filter. Note, however, that they are upwards of 40 dB below the peaks and thus unlikely to affect normal operation with noise included.

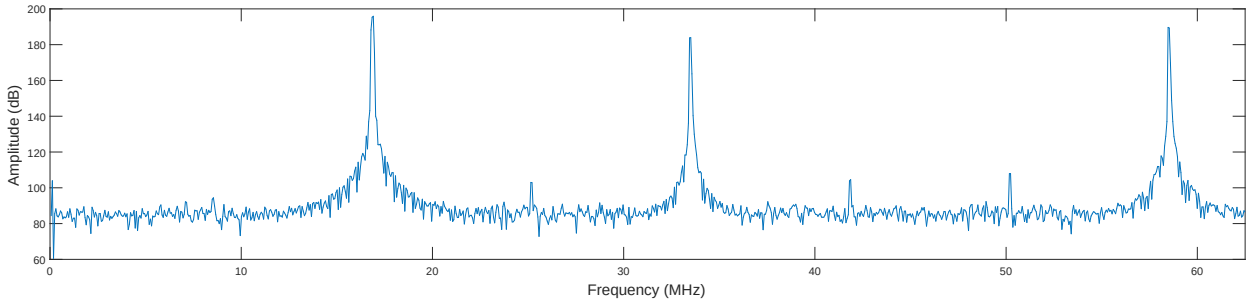


Figure 4.28: Output of the scanning riometer in full scan mode. The peaks are those of the test signal, sinusoids at 15.68, 31.23 and 54.76 MHz respectively.

Figure 4.28 also demonstrates the flexibility of the scanning riometer implementation, with the ability to work as both a single frequency power detector, able to avoid man-made noise, and the ability to work as a multiple frequency power detector or even a spectrum analyser.

4.5.3 Variable Frequency Sinusoid Generator

A useful test feature is the variable frequency sinusoid generator. It generates a sinusoid with an amplitude of the full 14 bit width of the DAC with a variable frequency specified by a 16 bit phase value. Each clock cycle the phase of the sinusoid is increased by the 16 bit phase value. Thus a sinusoid of any frequency between DC and 62.5 MHz can be selected with a resolution of 953 Hz. The variable frequency sinusoid generator, in conjunction with the spectral riometer, can be used to test the long term stability of amplifiers and other components. It is also a useful tool for testing the operation of the spectral riometer at a particular frequency.

4.6 Instruction Set

Interfacing between the FPGA and CPU on the Red Pitaya takes place via a shared memory block that both the FPGA and CPU are able to write to. To this end, a large number of parameters inside the FPGA design were identified that should be configurable. Each of these was mapped to a specific address that both the FPGA and CPU had access to. Thus a large number of parameters can be set by the user when recording riometer data.

Table 4.1 shows a set of parameters that are used to set up the scanning riometer. These are changed in a C file on the Red Pitaya which also records the data received from the riometer. The name of each parameter and the description for each has been provided in this table. Table 4.2 provides the parameters to configure when setting up the spectral riometer. Finally Table 4.3 shows the general

parameters that are used to, for instance, set up whether the output of the scanning or spectral riometer is being recorded. Perhaps the most important parameters to configure are the INPUT and OUTPUT, selecting the input source and output mode respectively. The other important parameters are setting the integration time for the scanning and spectral riometer individually and selecting whether Dicke switching mode is enabled.

Table 4.1: Scanning Riometer Parameters.

Parameter	**Description**
INT_TIME_SCAN	Number of samples to integrate over in the recording stage for the scanning riometer. Since the ADC samples at 125 $MSPS$, the integration time is $T_{scan} = \frac{INT\_TIME\_SCAN}{125 \times 10^6}$.
IND_INT_TIME	Number of samples to integrate over in the scanning stage of the scanning riometer. Much like for the recording stage, the integration time can be found using $T_{record} = \frac{IND\_INT\_TIME}{125 \times 10^6}$.
FFT_BLOCKS	Specifies the bandwidth used by the scanning riometer. One block corresponds to $3.9\,kHz$. More blocks increases the bandwidth accordingly.
START_FREQUENCY	Frequency, in MHz, to begin scanning from.
END_FREQUENCY	Frequency, in MHz, to end scanning on.
NO_INCREMENTS	Number of equally spaced increments between the start and end frequency when scanning. If, for instance, an increment value of 10 is chosen, the starting frequency is $10\,MHz$ and the ending frequency is $20\,MHz$, then the frequency increments will be $10\,MHz$, $11\,MHz$, $12\,MHz$ and so on up to $19\,MHz$.

Table 4.2: Spectral Riometer Parameters.

Parameter	**Description**
INT_TIME_SPECTRAL	Number of samples to integrate over for the scanning riometer. Since the ADC samples at 125 $MSPS$, the integration time is $T_{spectral} = \frac{INT\_TIME\_SCAN}{125 \times 10^6}$.
START_BIN	Determines which bin number starts recording at. This, along with END_BIN, limits the bins that are written to the Red Pitaya memory.
END_BIN	Determines which bin number recording stops at. This, along with START_BIN, limits the bins that are written to the Red Pitaya memory.
WINDOWING	Whether applying a hamming windowing before the FFT is enabled or not.

Table 4.3: General Riometer Parameters.

Parameter	Description
NOISE_STEP	Amount to increase noise amplitude by on each scan/record cycle when doing testing. Set to zero during normal operation.
FIX_NOISE	Set FIX_NOISE to 64 to keep the noise level constant at the NOISE_STEP amplitude. If FIX_NOISE is not set then amplitude increases by the step size after every scan/record cycle.
SWITCH	Determines whether Dicke riometer operation is enabled. If set to 16, the signal to switch between the input and internally generated noise will be enabled. The switch takes place every DUTY_CYCLE scan/record cycles.
DUTY_CYCLE	Sets the switch duty cycle. If DUTY_CYCLE is set to 1 the duty cycle will be 50 % with one scan/record cycle spent switched to the antenna and one switched to internally generated noise. Increasing the duty cycle decreases the relative time spent looking at the calibration source. Instead the riometer will stare at the antenna for DUTY_CYCLE scan/record cycles and then at the calibration source for one cycle..
INPUT	Configures which input is coming into the Red Pitaya, either the ADC (INPUT_ADC), internally generated noise (INPUT_INTERNAL_NOISE) or the test signal (INPUT_TEST_SIGNAL).
OUTPUT	Selects the output. Output modes are either normal (OUTPUT_NORMAL), scanning mode (OUTPUT_SCANNING), directly from the ADC (OUTPUT_ADC) or the spectral riometer mode (OUTPUT_SPECTRAL).
OUTPUT_SIN	Configures whether the DAC will output the internally generated noise source or an internally generated sinusoid. It should be set to 128 to output a sinusoid.
SIN_FREQUENCY	Configures the frequency of the internally generated sinusoid in MHz.

4.7 Comparison Between Scanning and Spectral Riometer

This section provides a brief comparison between the scanning and spectral riometer in terms of each one's effectiveness and practicality. As implemented in this project both the scanning and spectral

riometers have the ability to perform the task of the other one. However, each has their pros (+) and cons (–) for different tasks and as an overall package. The pros and cons of the scanning riometer are:

+ Bandwidth is configurable between $3.91\,kHz$ and $125\,kHz$.
+ Operating frequency and operating frequency range are easily configurable.
+ Requires less storage space than the spectral riometer due to fewer output samples per second.
+ Can resolve finer features of the spectrum when operating in spectrum analysis mode due to sharper filter roll off and smaller bin bandwidth.

– Integration time is effectively reduced when using the scanning riometer due to the scanning process.
– Takes far longer to produce a spectrum in spectrum analysis mode than the spectral riometer for an equivalent integration time and bin width.

Equivalently for the spectral riometer the following pros and cons apply:

+ Can produce real time spectrums when doing spectrum analysis.
+ Can be configured to only look at a smaller region of the spectrum to save on storage requirements.
+ In post-processing the recorded spectrum can be used to perform the same task as the scanning riometer, without the drawback of reduced integration time.

– Bandwidth is largely fixed due to the fixed size FFT used, but can be increased in post-processing by summing adjacent bins in $61\,kHz$ increments.
– Uses more storage space than the scanning riometer due to needing to store a large part of the spectrum after each integration step.

Despite the increased data rate and reduced bandwidth flexibility of the spectral riometer, it is overall the preferred mode of operation due to the options it provides in post-processing without reducing integration time. Modern storage technologies provide the capability of storing lots of data and increased flexibility down the line should be valued over storage space considerations. As an example, if the spectral riometer operates with an integration time of $1\,s$ and stores the full spectrum after each integration cycle it will require $1.3\,GB$ of data over a full 24 hours. While this is a lot, it is not unreasonably high and given a $1\,TB$ hard drive the riometer could be expected to operate for upwards of 2 years before running out of space.

Chapter 5

Testing Methodology and Results

This chapter covers the riometer testing that took place and incorporates the results that were achieved as the tests are covered. The tests are split into three sections: testing of the FPGA code in Section 5.1, some baseline front-end performance metrics in Section 5.2 and the readings of the scanning and spectral riometer as compared to that of a fully developed professional system in Section 5.3.

All the tests were performed using the Red Pitaya, antenna and front-end as described in Chapter 4 with some changes made between each field trial. Four trials were performed, each over a period of a week or more. Unfortunately the data from trial two was unusable due to damage to the balun of the antenna that was only detected after the trial. Therefore only the results of the first, third and fourth trials will be presented. Overall it is found that the demonstrator riometer system works satisfactorily.

5.1 Verification of FPGA Code

It is important to verify that the digital design of the riometer is functioning correctly. To do this a number of experiments were designed and carried out. Since the functioning of the scanning and spectral riometer are largely separate, each has its own set of verification tests and results. The scanning riometer experiments are presented first and then the spectral riometer experiments. Each experiment is presented in the form of a description, expected result, obtained by simulation using MATLAB code, and actual result, before comparing the two. The digital processing for the riometer is required to adhere closely to the simulated results and discrepancies are explained.

5.1.1 Scanning Riometer Tests

Two different tests were designed to test different aspects of the scanning riometer. In the first test a sinusoid of known frequency was injected and the output before minimum detection was monitored. The scanning riometer was set to scan across the full frequency range, in the full scan mode, to test that the correct power level was being obtained. A spectrum was obtained and this should closely correspond to the predicted spectrum. In the second test a sinusoid was once again injected, but this time the minimum detection functionality was brought into the circuit to determine if the circuitry correctly avoids the sinusoid peak and takes the minimum power in the frequency range(ie. the normal mode of operation).

Sinusoid Injection test

In the sinusoid injection test a sinusoid was fed internally to the input of the scanning riometer with a peak-to-peak amplitude spanning the full 14 bit width. The sinusoid's frequency started at 10 MHz and was increased by 10 MHz up to 50 MHz on each run through of the test. For each run the integration time was changed in three increments: 0.01 s, 0.1 s and 1 s. The bandwidth was changed between 15.6, 31.2 and 62.4 kHz for each time increment. The spectrum is divided into 1024 equal increments.

MATLAB code was written to simulate the expected operation of the scanning riometer. The simulation attempted to adhere as closely as possible to the functionality of each block, as shown in Figure 4.17, within the limitations of MATLAB. Thus, while bit width changes were largely followed, some MATLAB functions do not accept fixed point numbers, and so double precision numbers were used here before converting back to fixed point numbers. These resulted in some discrepancies in the results which are discussed later in this section. Each simulation produced a spectrum corresponding to that produced by the scanning riometer in this test.

The results of the simulation and the scanning riometer could then be directly compared by finding the Root Mean Square (RMS) difference the two spectrums. The two are compared in decibels since absorption measurements are taken in decibels. Given a 1024 element simulation spectrum F_{sim} and a 1024 element riometer spectrum F_{riom}, both in dB, the mean error would be:

$$Error_{RMS} = \frac{\sqrt{\sum_{i=0}^{1023}(F_{sim}[i] - F_{riom}[i])^2}}{1024}[dB]. \tag{5.1}$$

The mean error for each of the configurations is shown in Table 5.1. Overall the error is quite high, well above zero where it should be for all of the frequency, integration time and bandwidth permutations. It is above 5.6 dB over all of the measurements. There are two reasons for this. The first is the difference in implementations of the simulation and the Vivado FFT. Since both are implemented

slightly differently, disparities between the two could exist. The differences in FFT implementations are discussed in Section 5.1.2. The second is that the bit width changes, including truncations, in the Cascaded Integrator-Comb filter are not accurately modelled by the MATLAB code, leading to some further differences in the results. It should be noted, that for a particular frequency and bandwidth, the error is identical no matter the integration time, indicating that integration in the scanning riometer is working as expected.

Table 5.1: RMS error in dB for the sinusoid test on the scanning riometer.

Int. time (s)	0.01			0.1			1		
Bandwidth (kHz)	15.6	31.2	60.4	15.6	31.2	60.4	15.6	31.2	60.4
Frequency									
$10\,MHz$	6.29	6.29	6.29	6.20	6.20	6.20	5.96	5.96	5.96
$20\,MHz$	6.29	6.29	6.29	5.73	5.73	5.73	5.63	5.63	5.63
$30\,MHz$	7.32	7.32	7.32	6.90	6.90	6.90	6.88	6.88	6.88
$40\,MHz$	6.79	6.79	6.79	6.46	6.46	6.46	6.38	6.38	6.38
$50\,MHz$	7.01	7.01	7.01	6.74	6.74	6.74	6.42	6.42	6.42

Figure 5.1 shows the simulated and scanning riometer spectrums plotted on the same figure for the $10\,MHz$ sinusoid. While the curves largely correspond, there are nonetheless significant disparities between the two across the frequency range. If the two curves were smoothed out using an averaging filter they would be very similar, but taking a closer look at any narrow frequency range shows that the local peaks and troughs of each curve do not correspond. This is where the majority of the error is present.

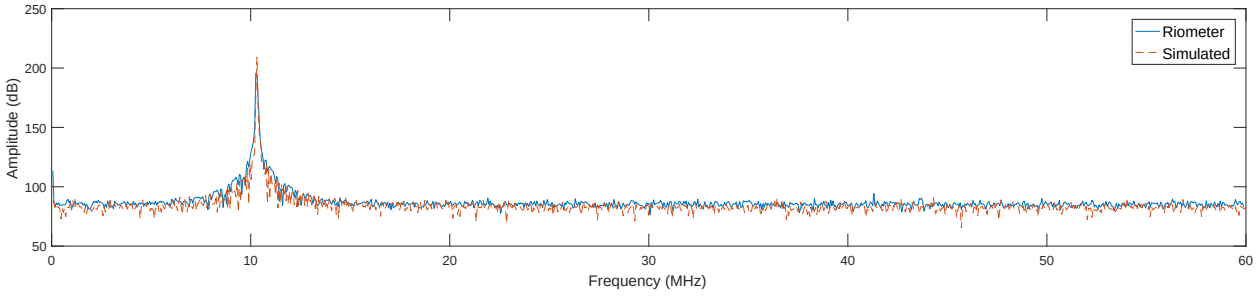


Figure 5.1: Spectrum of the $10\,MHz$ sinusoid produced in MATLAB by simulation and by the scanning riometer for an integration time of $1\,s$ and a bandwidth of $15.6\,kHz$.

From these results it is difficult to say whether the scanning riometer is functioning as expected. The differences between the simulated and actual results can be explained, but there is nonetheless a possibility that the scanning riometer is not working quite as predicted and further research into this is warranted.

Minimum Detection Circuit Test

The minimum detector circuit is tested by injecting a sinusoid at $15\,MHz$ with a peak-to-peak amplitude spanning the full 14 bit range of the ADC. The integration time is set to $1\,s$ and the scanning integration time is set to $0.1\,s$. The frequency range is 13 to $17\,MHz$ with 10 increments over that range. Thus one power reading will be recorded at the output of the scanning riometer every two seconds.

In addition, the individual spectrums over each integration time were recorded so that the minimum could be found in post-processing and compared against the minimum that the scanning riometer found. After the test was run for 5 minutes, it was found that the minimum detection circuit correctly identified the minimum power value in the spectrum every time. For this test no simulations were required to verify the results.

5.1.2 Spectral Riometer Tests

Two different tests were designed to check the functionality of the spectral riometer. These were designed around the recognition that, taking the FPGA implementation on its own, the spectral riometer is basically a spectrum analyser and can be tested as such. In the first test a sinusoid of known frequency and amplitude is injected into the ADC and compared against a simulation. A more complex signal consisting of multiple sinusoids is injected in the second test. In each test the integration time and windowing are varied to check that they work as expected.

Sinusoid Injection Test

In the sinusoid injection test a sinusoid is fed internally to the input of the spectral riometer with a peak-to-peak amplitude spanning the ADC's full 14 bit width. The sinusoid's frequency starts at $10\,MHz$ and is increased by $10\,MHz$ up to $60\,MHz$ on each run through of the test. For each run the integration time was changed in three increments: $0.01\,s$, $0.1\,s$ and $1\,s$. For each integration time tested the hamming window before the FFT was either enabled or disabled.

A simulation of the expected functionality of the spectral riometer was implemented in MATLAB and each permutation of the test run through it. The error metric used was the same error metric that was used for the sinusoidal scanning riometer tests (see Equation 5.1).

The mean error metrics are shown in Table 5.2. The mean error is very close to $0\,dB$ for the 10, 20, 30, 40 and $50\,MHz$ sinusoids when not windowed with the notable exception being $60\,MHz$. For $60\,MHz$ the error is $0.05\,dB$ when not windowed but $1.10\,dB$ when windowed. Both these errors are higher than for the equivalent integration time of any other frequency. In general, when the sinusoid was windowed,

the error measured is largest for frequencies near to $0\,MHz$ and near to $60\,MHz$. Integration time has almost no effect on the error measured and it is therefore safe to assume that the integration time feature is working as expected.

Table 5.2: RMS error in dB for the sinusoid test on the spectral riometer.

	Not Windowed			Windowed		
Integration time (s)	0.01	0.1	1	0.01	0.1	1
Frequency						
$10\,MHz$	0.01	0.01	0.01	0.35	0.35	0.35
$20\,MHz$	0.02	0.02	0.00	0.19	0.13	0.13
$30\,MHz$	0.01	0.01	0.01	0.05	0.05	0.05
$40\,MHz$	0.01	0.01	0.00	0.13	0.12	0.12
$50\,MHz$	0.01	0.01	0.01	0.33	0.33	0.33
$60\,MHz$	0.05	0.05	0.05	1.10	1.10	1.10

Figure 5.2 shows the unwindowed spectrums of the $10\,MHz$ sinusoid with an integration time of $1\,s$ produced by the riometer and simulation respectively overlayed. Note how they overlap almost exactly. The error versus frequency is plotted in Figure 5.3 to give some more detail on its variation over the spectrum. The error at a particular frequency bin, i, is calculated using:

$$Error[i] = (F_{sim}[i] - F_{riom}[i])^2[db^2]. \tag{5.2}$$

Clearly at $10\,MHz$ they agree almost perfectly, however, further away from the peak the variation in error starts to increase. The reason for this is the discrepancy between the FFT used for the simulation and the FFT used in the digital design. MATLAB's FFT only has provision for double precision floating point numbers, whereas the digital design works on fixed point numbers and uses rounding at each butterfly stage to keep the bit width of the output from becoming too large. In this case the input is 16 bits wide, while two 28 bit numbers are used to represent each real and imaginary part at the output. This rounding, compared to the precision doubles used by MATLAB, is thought to play the major part in the error produced here. The rest of the chain can be accurately modelled using integer numbers in MATLAB.

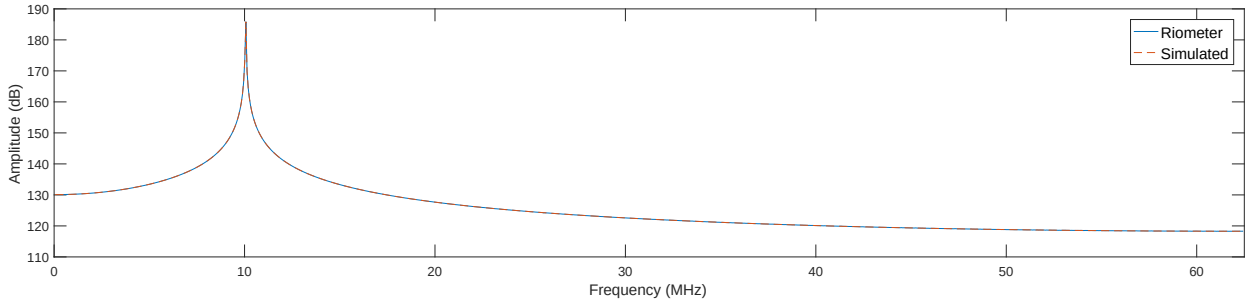


Figure 5.2: Spectrum of the 10 MHz sinusoid produced in MATLAB simulation and by the riometer without windowing for an integration time of 1 s. They overlap almost exactly.

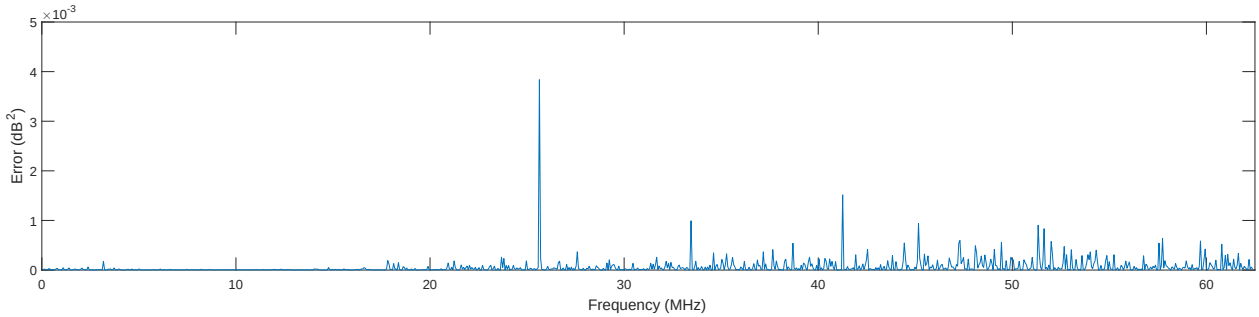


Figure 5.3: Error versus frequency for the 10 MHz sinusoid without windowing for an integration time of 1 s. The error is almost exactly zero at the peak at 10 MHz and has greater variation further away from the peak.

The increased error for the windowed 60 MHz spectrum can be also be explained by the discrepancy in the FFT implementations. 60 MHz is very near to the maximum frequency that the spectral riometer operates at, 62.5 MHz. Thus the sidelobes of the peak at 60 MHz have to be presented all the way down to zero decreasing all the time, further than any of the other frequencies tested. Due to the rounding process taking place in the spectral riometer FPGA only so many bits are able to represent these very low amplitudes, resulting in increased error. The windowing process also contributes to the error as it decreases the sidelobe level even further. This also explains why the error increases as the windowed sinusoid is nearer to the edge of the measured spectrum, near 0 MHz and 62.5 MHz. The error for the 60 MHz sinusoid after windowing is shown in Figure 5.4, with the error calculated as per Equation 5.2.

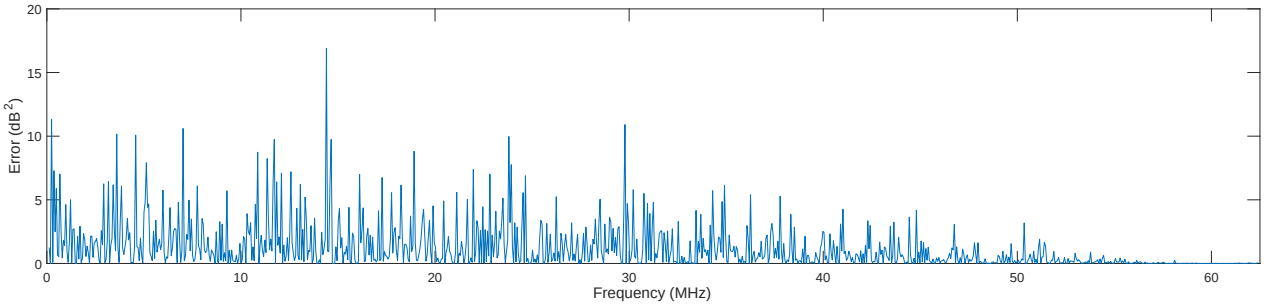


Figure 5.4: Error versus frequency for the 60 MHz sinusoid with windowing for an integration time of 1 s. The error is almost exactly zero at the peak at 60 MHz with overall error increasing at lower frequencies.

Complex Signal Test

The complex signal test sums three sinusoids at different frequencies and amplitudes together and feeds it internally to the input of the spectral riometer. The three frequencies are at 15.68, 31.23 and 54.76 MHz. The peak-to-peak amplitudes of these sinusoids are scaled according to the full 14 bit width of the spectral riometer input corresponding, respectively, to 14, 12 and 13 bits. A MATLAB simulation of the expected functionality of the riometer had already been implemented for the sinusoid test and the same implementation was used for this test. The integration time was changed between 0.01 s, 0.1 s and 1 s with windowing being enabled and then disabled for each test. The same error metric as used for the sinusoid test is used here (see Equation 5.1).

Table 5.3 shows the normalised error metrics obtained for the complex signal test. As expected the error remains constant with changing integration times and is acceptably low for the non-windowed signal: 0.04 dB. It goes up to 0.23 dB for the windowed complex signal. This error can again be explained by the discrepancy in the FFT implementations in the simulation and on the FPGA.

Table 5.3: RMS error in dB for the complex signal test on the spectral riometer.

	Not Windowed			Windowed		
Integration time (s)	0.01	0.1	1	0.01	0.1	1
	0.04	0.04	0.04	1.32	1.32	1.32

Figure 5.5 shows the spectrum of the non-windowed signal as simulated and produced by the FPGA. The two match very well over the entire spectrum with a few minor discrepancies seen at the lower amplitudes as rounding differences between the two implementations start to appear.

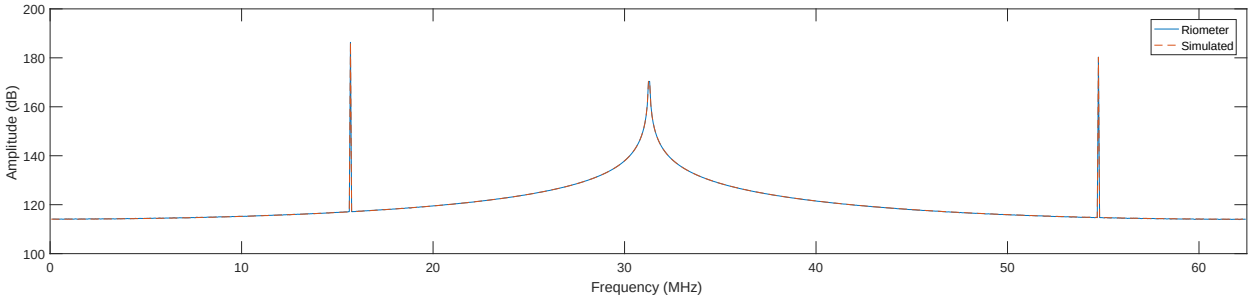


Figure 5.5: Spectrum of the complex, multi-sinusoid signal from the simulated and real data with an integration time of 1 s with no windowing.

After windowing the difference between the two spectrums becomes much more apparent (Figure 5.6). The spectrums follow each other very well at higher amplitudes, however, towards the lower amplitudes significant differences appear, and as Figure 5.7 shows, the error only increases the further away from the main peak the spectrum lies.

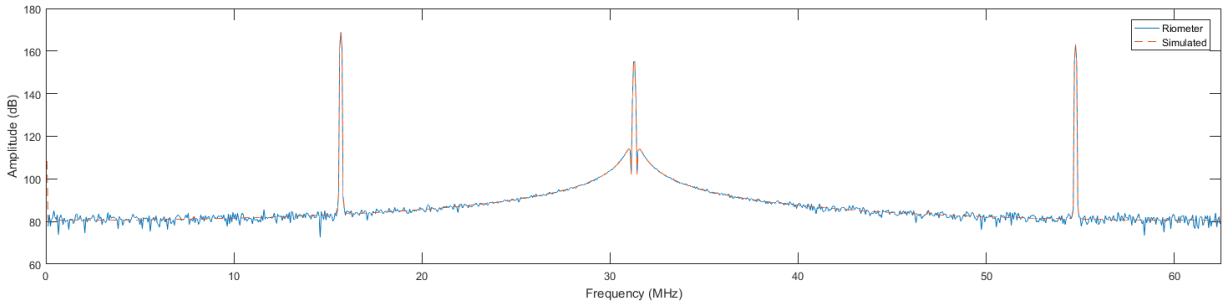


Figure 5.6: Spectrum of the complex, multi-sinusoid signal from the simulated and real data with an integration time of 1 s and windowing enabled.

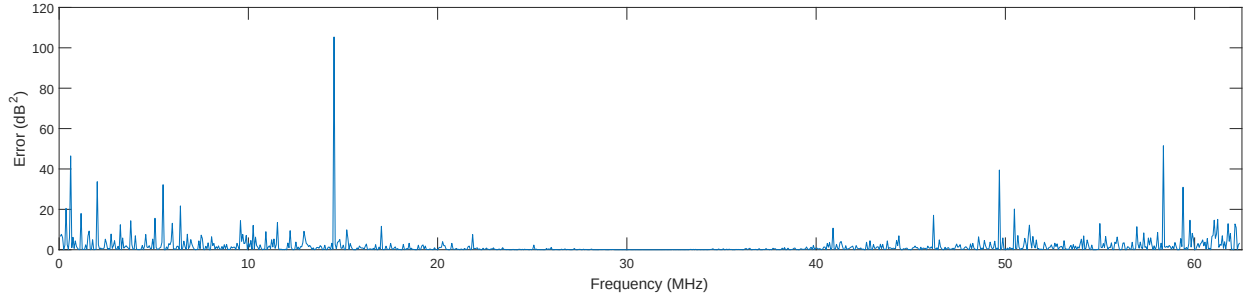


Figure 5.7: Error versus frequency of the complex signal for an integration time of 1 s.

The performance of the spectral riometer is satisfactory overall with some minor concerns about discrepancies between the simulation and riometer power levels at low amplitudes. However, these can be explained and in field trials the noise floor was significantly above these levels.

5.2 Front-end Riometer Performance Metrics

Performance metrics help to characterise the baseline performance of an instrument. In this case the key characteristics are stability, linearity and dynamic range. For each characteristic a theoretical result is presented, and practical tests are performed for the stability and linearity metrics.

5.2.1 Stability

Stability refers to how much the gain and bandwidth of the RF front-end changes over time. In other words, if the input noise power is held constant, the output of the riometer will be constant if it is a stable system. Dicke and Ryle-Vonberg switching are both methods to mitigate the effect of these changes. Due to the short and simple front-end of the demonstrator riometer, as compared to analogue riometers, it is predicted that the system will be very stable even without gain correction methods. In particular the demonstrator system has no analogue multipliers or any analogue minimum detection circuitry which all help to provide a more stable instrument.

To test the stability of the RF front-end, with only the low pass filter and two LNAs connected, the input was tied to a $50\,\Omega$ termination, with Dicke switching disabled. The riometer was set to record at $30\,MHz$ over a bandwidth of $62.5\,kHz$ and an integration time of $1\,s$. According to the riometer sensitivity equation the measured output noise temperature should be accurate to within $1.8\,K$. In practice, the ambient temperature also varied by a few degrees over the measurement period. Any changes in the gain-bandwidth product should result in a variation in the output temperature measured. Figure 5.8 shows the variation of output temperature measured with time.

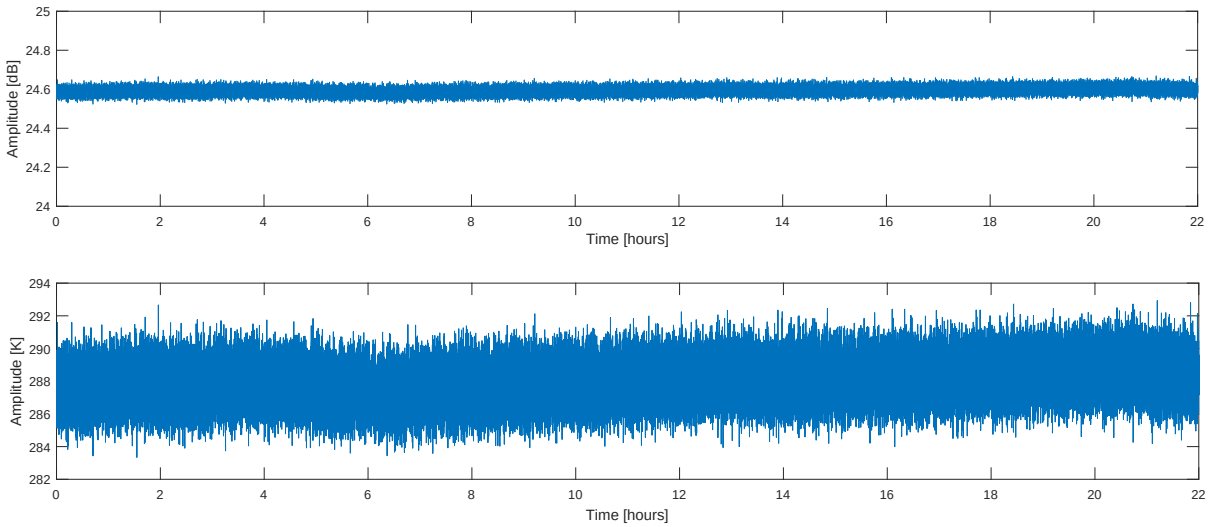


Figure 5.8: Stability of the RF front-end over time.

The difference between the maximum and minimum temperature measured was $9.6\,K$, corresponding to a $0.14\,dB$ overall change. Most of the large scale changes can probably be attributed to changes in ambient temperature, as the test was run over a 22 hour period. For instance around hour 6 the recorded noise temperature is seen to drop slightly, probably due to changes in ambient temperature. While less stable than predicted, accuracy to within $9.6\,K$ is still satisfactory.

5.2.2 Linearity

The linearity of the system is influenced by the analogue front-end and the ADC. All the digital signal processing will be linear by its nature, with the exception of the power detection algorithm. Since the front-end for a digital riometer is much shorter than that of a typical analogue riometer, the receiver will generally have more linear characteristics. In this analysis it is assumed that the major contributing factor towards non-linearities would be operating the amplifiers outside of their linear range, near their maximum rated input power. The amplifier used is the ZFL-500LN+ from Mini-Circuits [43]. It has maximum output power of $5\,dBm$, compared to the maximum input power of $10\,dBm$ for the ADC. This means that the amplifier will never saturate the ADC, however, the amplifier can itself saturate, leading to non-linearities.

In order to demonstrate the linearity of the system a white gaussian noise signal with smoothly increasing standard deviation is injected into the input of the riometer. Since the noise power spectral density level is equivalent to $\frac{N_0}{2} W/Hz$, where N_0 is the standard deviation of the noise squared, the noise power measured at the output of the riometer should increase quadratically over time.

The test was run with an integration time of $1\,s$, scanning over a frequency range of 38 to $40\,MHz$ in 10 intervals. The bandwidth used was $250\,kHz$. Figure 5.9 shows the results of this test. The noise power measured does increase quadratically, with a slight unexpected deviation at the beginning of the test. This can be attributed to the fact that when the input noise power is not present, the system is measuring the noise floor of the front-end. As soon as the input noise power increases for the first time a jump in the measured noise power is observed.

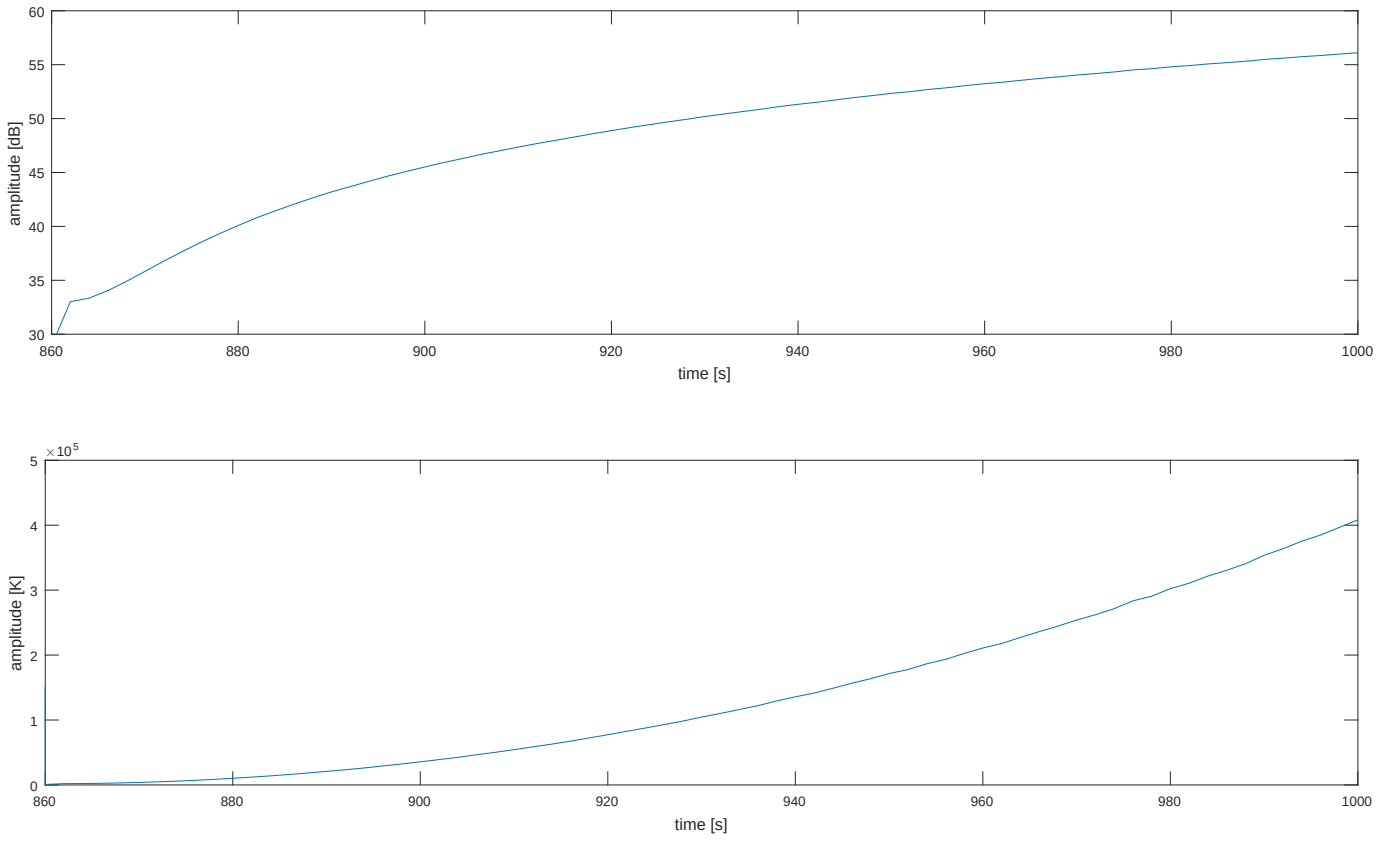


Figure 5.9: The results of the linearity test - quadratically increasing noise power over time.

Overall the system behaves linearly in accordance with the initial theory.

5.2.3 Dynamic Range

Dynamic range is the ability of the ADC to pick up both very small and very large signal powers. There are multiple different types of dynamic ranges that one can measure, with two of the most widely used ones being the Single Signal Received Dynamic Range (SDR) and the Two Signal Instantaneous Dynamic Range (IDR). SDR is the ability of a received to measure a single frequency from the minimum input level to the maximum input level [50]. IDR is the ability of a receiver to simultaneously process a very strong and a very weak signal [50]. However, these traditional metrics require a known Signal-to-Noise Ratio (SNR) and a riometer is simply trying to measure the noise floor, making these difficult to quantify. Therefore, instead, a simple dynamic range analysis of the front-end is presented.

The LTC2208-14 ADC on the Red Pitaya has a minimum input power of about -75 dBm and a maximum power of 10 dBm, with a dynamic range of 85 dB [51]. A spectral riometer must be able to operate over a wide range of frequencies, leading to an input low pass filter that cuts off at 42 MHz. This means that at a typical galactic noise level of 2.9×10^4 K, the power received at the input of the RF

chain will be -77 dBm. This is then amplified by 52 dB to -25 dBm, including losses from the filters. Strong absorption events can reduce the power received by up to 20 dB. This would lead to -45 dBm appearing at the input to the ADC, still well above the minimum power it can receive at -75 dBm, which allows for deep absorption events to be captured.

Since the maximum power that the amplifiers can output is 5 dBm, the receiver has a clearance of 30 dB above the nominal noise floor for other, man-made signals. This could be a problem, particularly in urban areas, where there are a large number of interfering signals at HF that add to the input power and could cause saturation. In Antarctica, however, this is less likely to occur as most HF signals are likely to have propagated from far away and will therefore be significantly weaker than a signal received from nearby.

5.3 Field Trials

Testing of the system took place at SANSA in Hermanus (34°42'48.63"S , 19°22'37.96"E), and at Fish Hoek (34°13'18.46"S , 18°39'89.85"E) in Cape Town, whose locations are shown on the map in Figure 5.10. SANSA already had two riometers running there against which the demonstrator riometer could be compared. The riometers at Hermanus operated at 38.2 MHz and 30 MHz. Each employed two dipole antennas of the required dimensions for the frequency of interest. The two antennas are spaced and the beams added to allow for a narrower upwards pointing beam to be formed. The riometers were designed and built by La Jolla sciences [44].

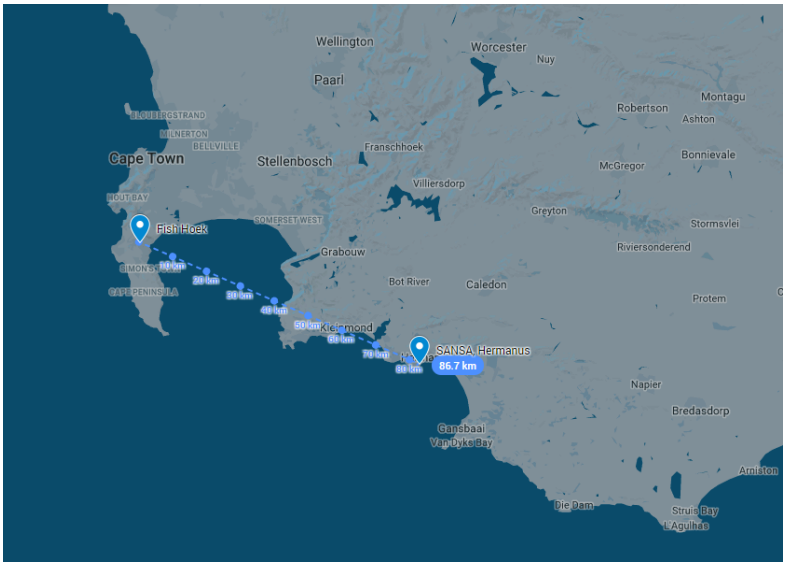


Figure 5.10: Trial locations at Fish Hoek and SANSA, Hermanus. The distance between the two locations is 86 km.

Four trials were performed with the antenna, front-end and FPGA code changing between trials, and the power recordings received compared against the professional riometer, as the ground truth. Additionally on examining the data from the 30 MHz riometer it was found that its ADC had saturated for most of

the day making the readings unusable and so only comparisons with the 38.2 MHz riometer's results were made. Each trial ran for a week or more to compare the power received across multiple days in an attempt to identify a quiet day curve in the recordings. The design of the demonstrator riometer did not allow for the measurements of the spectral and scanning riometer to be recorded simultaneously. Thus only the spectral riometer was used in the field trials since it allows for multiple frequency readings and storage space was not an issue.

Table 5.4 shows the setup used for each of the four trials, with changes between each trial being noted. The integration time was kept constant at 2 s for each trial. 2 s was chosen since it is short enough that ionospheric changes are easily tracked and not too short that the Red Pitaya runs out of storage space in a week's worth of recording. As described in Chapter 4, the antenna and balun were changed between trials one and two, and trials three and four. Windowing before the FFT was only implemented before the fourth trial where it is applied. Also note that trial three was run in Fish Hoek instead of Hermanus.

Table 5.4: Setup for each of the four trials.

	Trial 1	Trial 2	Trial 3	Trial 4
Integration time (s)	2	2	2	2
Antenna (MHz)	42	42	38.2	38.2
Balun	1:1 balun	1:1 balun	Sleeve	Sleeve
Windowing	No	No	No	Yes
Low pass filter	Yes	Yes	Yes	Yes
High pass filter	No	No	No	Yes
Location	Hermanus	Hermanus	Fish Hoek	Hermanus

For each trial the results were displayed as a series of curves directly comparing the power received over several days and the power received by the ground truth riometer. It was expected that the curves would correspond to quiet day curves over this period, which would not be removed since each trial was run over a period of about a week which was too short to determine the quiet day curve.

The setup and results of each of the four trials will now be discussed.

5.3.1 Trial One: 2 - 9 June

The antenna was setup as shown in Figure 5.11a. The balun is housed in the box between the two metallic struts, with the stand used to keep the dipole at the correct height. The antenna was connected to the front-end via a 15 m long coaxial cable, seen in Figures 5.11a and 5.11b. Figure 5.11b also shows the front-end box on the right and the power supply in the box on the left. In terms of the front-end,

only the low pass filter was used for this trial. This was followed by both of the Mini-Circuits amplifiers at 28 dB gain each.

Spectral riometer recordings were taken between 30.5 and 42.7 MHz therefore covering a wide frequency range and including the 38.2 MHz operation frequency of the existing riometer at SANSA. An integration time of 2 s was chosen. The existing riometers both operate with an integration time of 1 s but the integration time can easily be doubled in post-processing by adding together every two consecutive samples. The bandwidth of the spectral riometer was also increased to 244 kHz in post processing by adding adjacent frequency bins to match as closely as possible the 250 kHz bandwidth of the La Jolla receiver.

(a) The 42.2 MHz antenna.

(b) Coaxial cable leading from the antenna to the box housing the front-end and Red Pitaya.

Figure 5.11: The antenna and front-end as setup in Hermanus.

Figure 5.12 shows the setup of the 38.2 MHz riometer at SANSA. One can see that the dipoles are made up of wires that have been held taught. The top set of wires receives the incoming signal while the bottom set acts as a small reflector, allowing for a more focussed beam. A similarly designed antenna was used for trials three and four in an attempt to closer match the readings of the professional riometer systems and uses the same dimensions, as shown in Figure 4.8. The main difference between the two is that the professional system had two antennas whose beams were added to form a single, narrower upwards pointing beam, as compared to the single antenna used in trials three and four.

(a) Single antenna of the 38.2 MHz riometer.

(b) Both antennas of the 38.2 MHz riometer.

Figure 5.12: The 38.2 MHz riometer.

The first trial took place from 2 June 2017 at 10:45:30 am until 9 June 10:28:00 am, a period of almost exactly one week. The reading of the instrument for 42.2 MHz over the full recording period are shown in Figure 5.13. Unfortunately the recording of these results corresponded with one of the worst storms in Cape Town's recent history with gale force winds over the 7th and 8th of June. The result of this was that the antenna blew over around midnight on the 7th June which meant the instrument readings after this date were unusable.

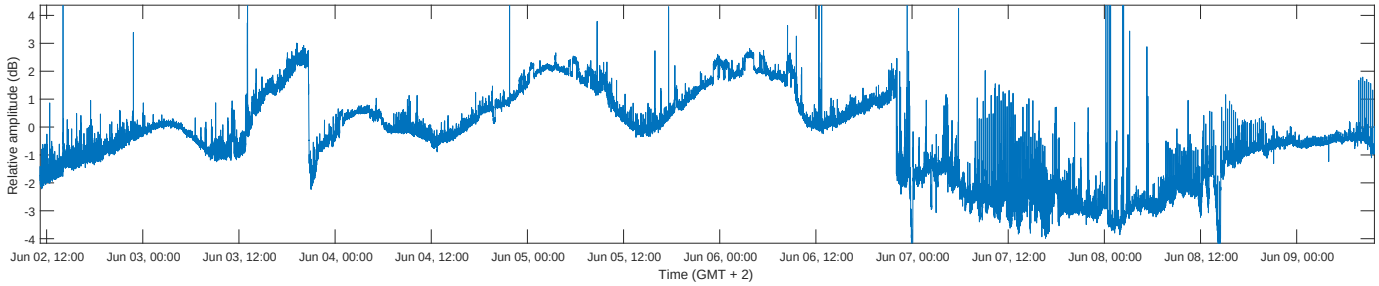


Figure 5.13: Results from the first trial at 42.2 MHz with a bandwidth of 244 kHz.

One can see a quiet day curve over the fourth, fifth and sixth of June. In order to compare the power readings to that of the professional 38.2 MHz system, the plots of both the data from the 38.2 MHz riometer and the spectral riometer are shown overlayed in Figure 5.14. Since the scale of the professional riometer readings was not known, the two were compared using relative amplitude by subtracting the mean value, in decibels, over the entire period from the both readings. Note how the curves seem to have the same general shape. The absorption experienced does change at higher frequencies and thus the data should not match exactly. The power readings from the SANSA riometer contains periodic spikes which are not related to the actual power received. These spikes are as a result of SANSA's ionosonde operating nearby. The ionosonde transmits across a small range of frequencies in the 10-20 MHz range every 15 minutes and, while the riometer systems here are operating at 38.2 MHz, the ionosonde signal seems to be powerful enough that it overcomes the filters and leaks into the passband, causing increased power to be received when the ionosonde transmits.

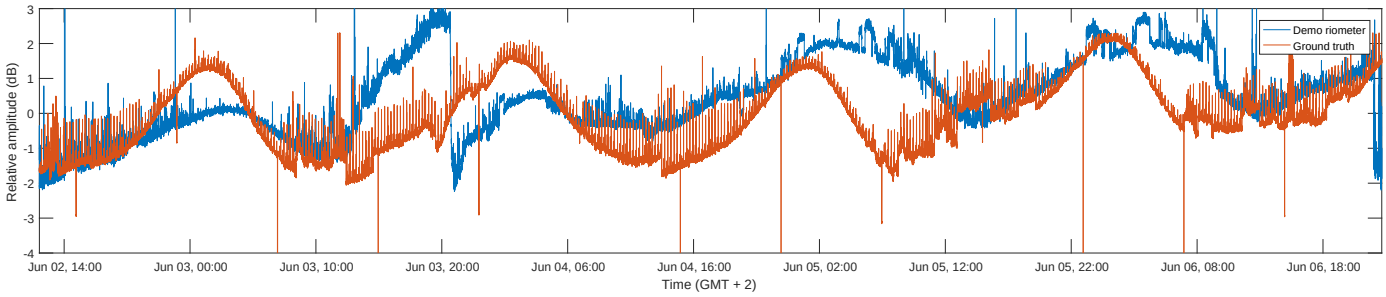


Figure 5.14: Comparison of the 38.2 MHz riometer at SANSA and the results of the demonstrator riometer at 42 MHz.

Even though the antenna was designed to work at 42 MHz, a comparison between the data recorded at 38.2 MHz by the demonstrator riometer and the ground truth data is shown in Figure 5.15. The power received was, however, greatly reduced which means the sensitivity was also compromised leading to more noise in the power readings. The plots show that the two seem to concur for the most part from about midnight on the third of June.

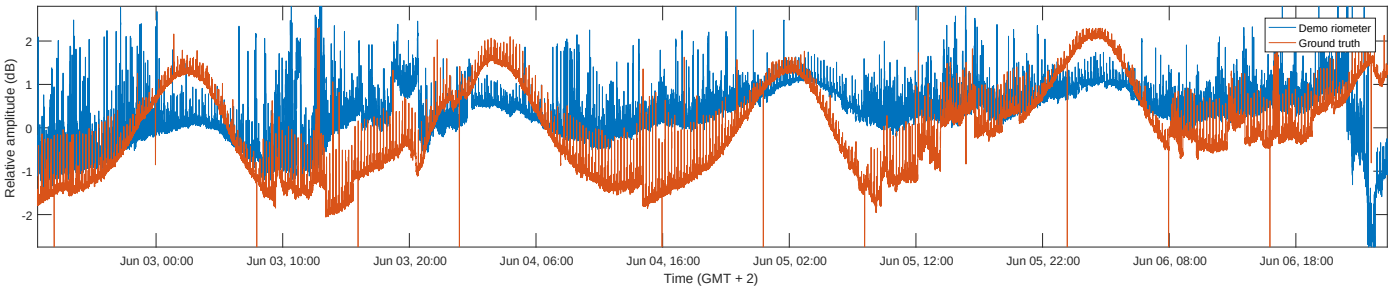


Figure 5.15: Comparison of the 38.2 MHz riometer at SANSA and the results of the riometer developed here at 38.2 MHz.

The reason for what appears to be a lack of sensitivity on the part of the demonstrator riometer could be as the result of a lack of adequate electromagnetic interference shielding on the receiver. The SANSA riometers were also buried underground, insulating them from temperature changes to a large extent, which could improve their sensitivity.

5.3.2 Trial Two: 9 - 16 June

Trial two was unsuccessful. When the antenna blew over in trial one it damaged the balun. Unfortunately the fault was only detected upon recovering the antenna, and the data recorded, at the end of the trial. The data was unusable.

5.3.3 Trial Three: 28 June - 7 July

The third trial took place in Fish Hoek, Cape Town with the second iteration of the antenna as described in Section 4.1.3. This antenna was tuned to work best at 38.2 MHz to match the frequency of the riometer at SANSA. The front-end was kept the same as for trials one and two.

Initially it was hoped to be able to run the trials with the new antenna at UCT, but it was thought that UCT had too much man-made noise to be a suitable site, thus Fish Hoek, a more remote location, was chosen. The reason for not running the trial in Hermanus (and thus being able to compare it directly against data from the professional systems there) was simply a lack of time over those weeks and the long distance from Cape Town.

Figure 5.16: Antenna setup and location in Fish Hoek for trial three.

Since the new antenna was matched to $50\,\Omega$ at $38.2\,MHz$ through its balun, at this frequency more power should be received than for the 1:1 balun used in the previous antenna. The main disadvantage of the sleeve balun is that it is a narrowband device around the frequency of interest, unlike the 1:1 balun which can work across a wider band of frequencies but with no matching between the $75\,\Omega$ dipole and the $50\,\Omega$ coaxial cable. From transmission line theory the reflection coefficient at the discontinuity will be:

$$\rho = \frac{Z_L - Z_0}{Z_L + Z_0}, \tag{5.3}$$

where Z_L is the load impedance and Z_0 is the source impedance. Therefore the transmitted power will be multiplied by $1 - \rho^2$. For a mismatch of 50 and $75\,\Omega$ this equates to 0.96 W/W of the power that would have been transmitted if they were matched. So the increase in power due to matching is only a small amount in reality, less than $0.2\,dB$, but it will nonetheless result in greater sensitivity.

The results from this trial were compared against the $38.2\,MHz$ riometer data captured by the professional system at Hermanus, assuming that overall shape of the curve would not change a great deal due to the change of location. The relative amplitude of the received power is shown in Figure 5.17. At a first glance the curves appear roughly sinusoidal, however, there seems to be periods of intense interference from about midday until midnight on most days, after which the curve is well defined until about 8 am. Once again the data exhibits periodic spikes in received noise power, despite being not being collocated with the ionosonde at SANSA in Hermanus. This could nonetheless be the ionosonde transmitting at Hermanus, as HF signals can easily travel the 90 km between Fish Hoek and Hermanus via the ionosphere.

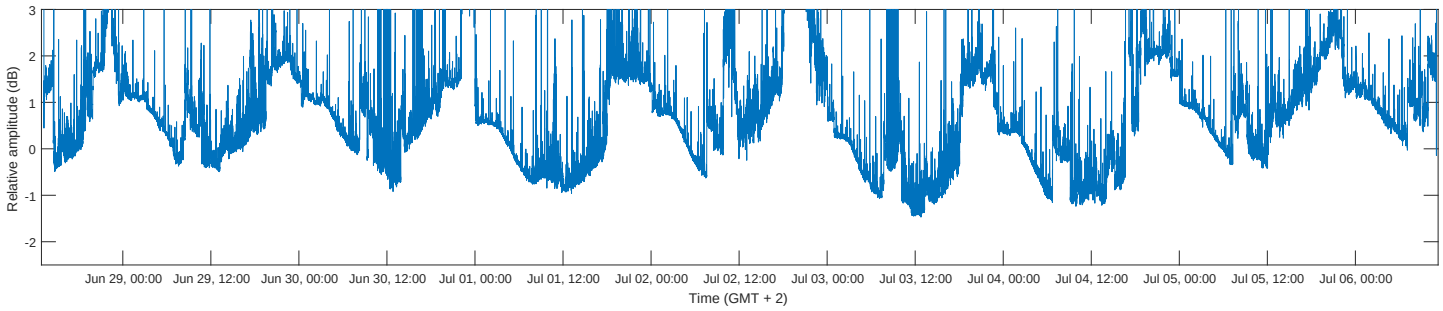


Figure 5.17: Demonstrator riometer measurements for trial three at 38.2 MHz trial three.

Figure 5.18 shows the demonstrator riometer data superimposed upon the ground truth data readings from the riometer in Hermanus. The curves are once again roughly the same, with the main difference being the reduced interference experienced by the ground truth riometer. Between about midnight and 8 am on most days both curves follow each other quite well, with the demonstrator riometer displaying a reduced amplitude change. The reduced amplitude change can be attributed to the single dipole antenna's smaller beamwidth, as well as spectral leakage due to the FFT technique used. The professional system uses an analogue filter that has a sharper cut-off than each FFT bin, resulting in much reduced spectral leakage.

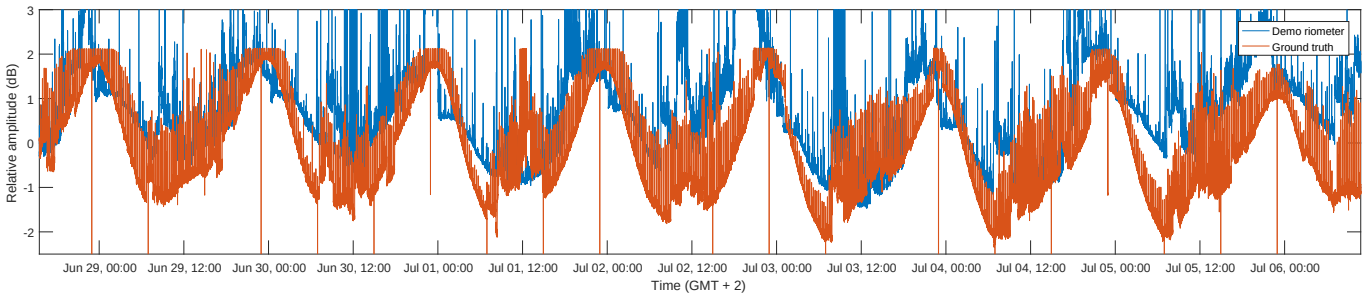


Figure 5.18: Demonstrator riometer data and ground truth data at 38.2 MHz.

A closer look at the two sets of data between the first and third of July, as seen in Figure 5.19, reveals the similarities in the curves present in the early hours of the morning, and the increased interference experienced by the demonstrator system at other times. In the hours after midnight the curves appear to be highly correlated, but scaled differently.

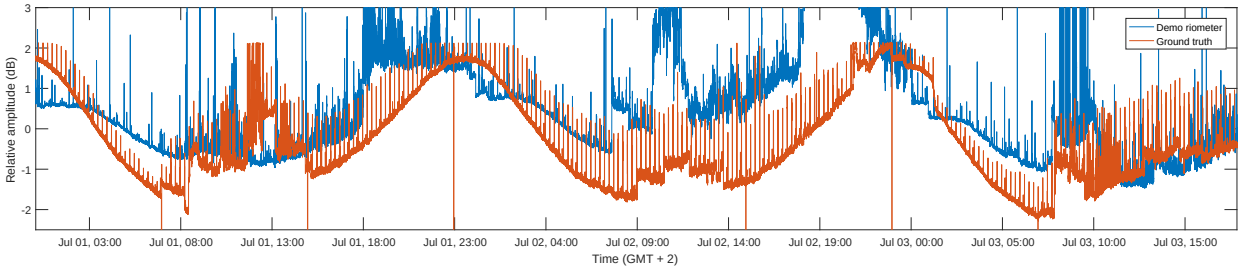


Figure 5.19: Demonstrator riometer data for trial three at 38.2 MHz and comparison with ground truth between 1 and 3 July.

5.3.4 Trial Four: 2 August - 11 August

Trial four took place in Hermanus at SANSA again. The setup was similar to that of trial three, with the same antenna being used. Two key changes were made: the windowing functionality for the spectral riometer was enabled, placing a hamming window before the FFT, and a high-pass filter with a cut-off of 11.5 MHz was included in the design. It was hoped that windowing before taking the FFT would help to reduce spectral leakage and thus interference from nearby frequencies. The antenna setup used in trial four is shown in Figure 5.20. The dipole antenna, reflector and sleeve balun running from the dipole are all clearly visible in the image.

Figure 5.20: Antenna setup and location for trial four.

The results from trial four at 38.2 MHz with a bandwidth of 244 kHz are shown in Figure 5.21. This time the power received varies in a clear sinusoidal pattern, mirroring the quiet day curve. Once again periodic spikes in the power received can be seen, occurring approximately every 15 minutes due to the ionosonde transmission. In this trial the spike in power due to the ionosonde were upwards of 30 dB

higher than the noise floor being measured.

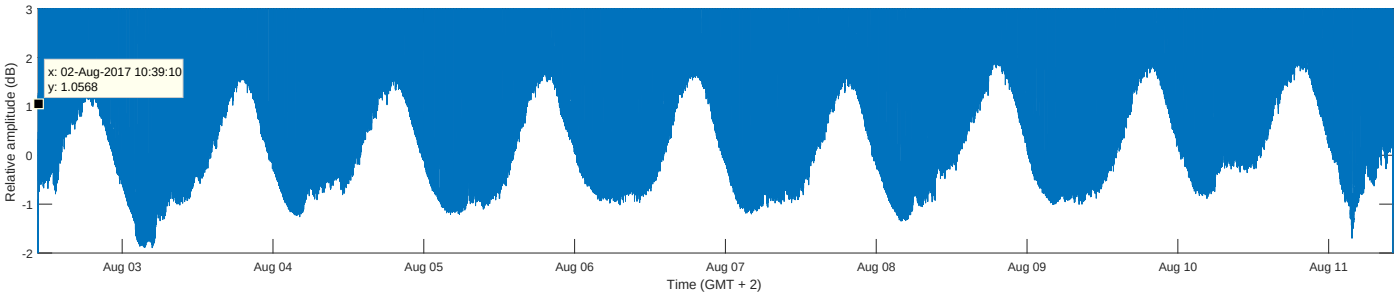


Figure 5.21: Demonstrator riometer data at 38.2 MHz.

Figure 5.22 shows a comparison between the recorded power at 38.2 MHz and the ground truth data recorded by the riometer system at SANSA. The curves follow each other more closely than what was observed in previous trials. One immediately notable difference is the greater amplitude measured by the demonstrator riometer while the ionosonde is transmitting. It could be that while the ionosonde is transmitting it saturates the amplifiers of the demonstrator riometer causing images to appear at higher frequencies which are then measured as much larger amplitudes at higher frequencies.

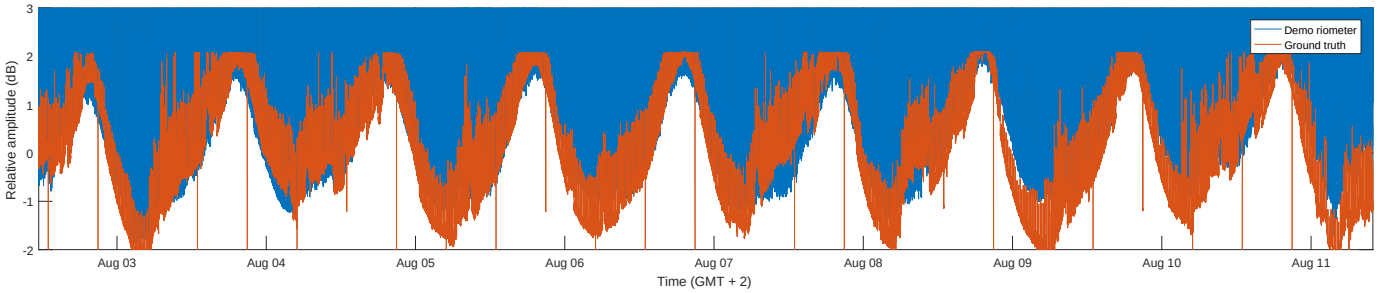


Figure 5.22: Demonstrator riometer data and ground truth data at 38.2 MHz.

A closer look at the recorded results over the fourth to the seventh of August is shown in Figure 5.23. Here we see how closely the curves match each other over these three days, with about 0.5 dB separation between the two curves and less for the most part. The curves show particularly good correlation between about 10 pm and 8 am when the ionosphere is less disturbed and fewer interfering signals are present.

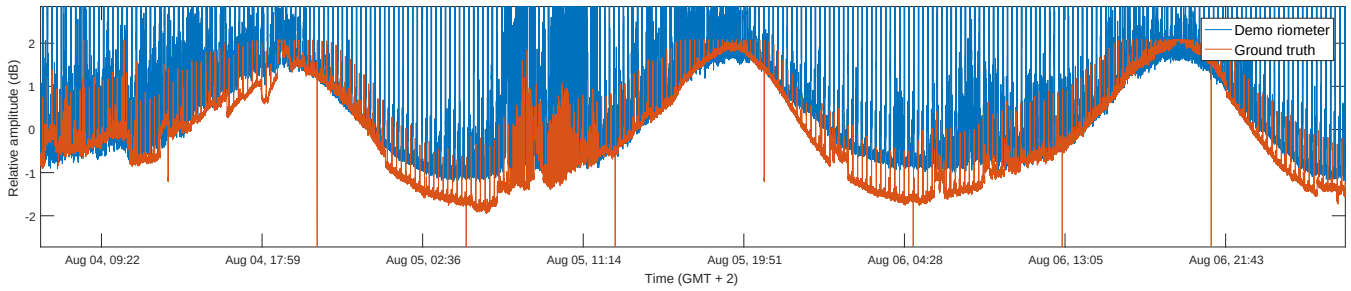


Figure 5.23: Demonstrator riometer data and ground truth data at 38.2 MHz from 4 August to 8 August.

Overall adding windowing to the spectral riometer seems to have greatly reduced spectral leakage and thus improved the agreement between the ground truth data and that recorded by the spectral riometer. It could now be confirmed, with a reasonable degree of certainty, that the demonstrator spectral riometer recorded a day night curve over those days.

5.3.5 Spectrum at Hermanus

The final test that was performed was to connect the antenna directly to a spectrum analyser at the test site in Hermanus in an attempt to gauge both the general expected signal levels and to identify free bands in the spectrum. The resolution bandwidth (RBW) of the spectrum analyser was set to 30 kHz and the video bandwidth (VBW) to 1 MHz. The RBW is mainly of interest since it limits the input power to the IF stage of the spectrum analyser right before power detection and thus effectively determines the bandwidth across which the noise floor is measured. Here it is set to a value in the same order of magnitude as the spectral riometer bin size, which is 64 kHz.

The spectrum was recorded once in real time and once after applying a maximum hold for 15 minutes, where the maximum hold records the maximum magnitude that occurs in a particular frequency bin over the full 15 minutes. The real time measurement was used to determine the general noise power into the input of the front-end of the riometer, while the maximum hold is used to determined free bands in the spectrum. The spectrums produced are shown in Figure 5.24.

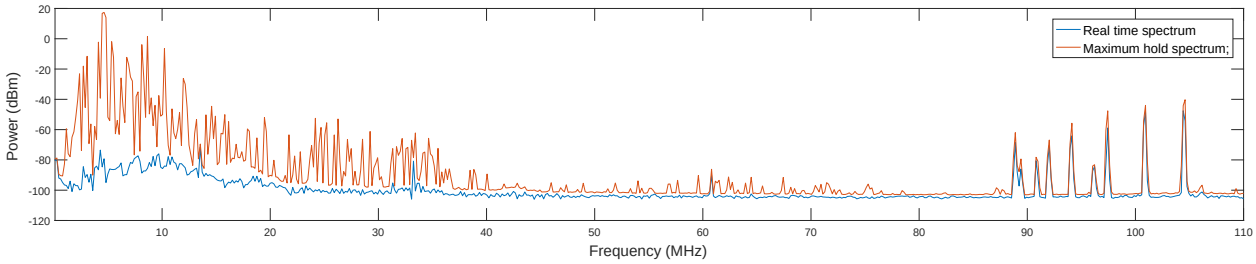


Figure 5.24: Spectrum levels as measured in Hermanus at SANSA.

Examining the real time spectrum one notices significant interference between about 90 and 110 MHz. Each of the spikes here corresponds to an FM radio station in South Africa. It is also noticeable that the noise seems to increase at low frequencies, as expected as cosmic noise increases towards lower frequencies. In the middle of the spectrum, between 30 and 80 MHz the spectrum seems to be mostly flat, however, this is likely the noise floor of the instrument and not the actual noise floor that the riometer would be measuring. The maximum hold readings confirm the interference due to FM radio stations and also reveal some small interfering signals in the mid range of frequencies. However, the low frequency interference is greatly increased as compared to the real time spectrum. Part of this is no doubt due to the ionosonde which scans its transmitting frequency across a wide range below 20 MHz.

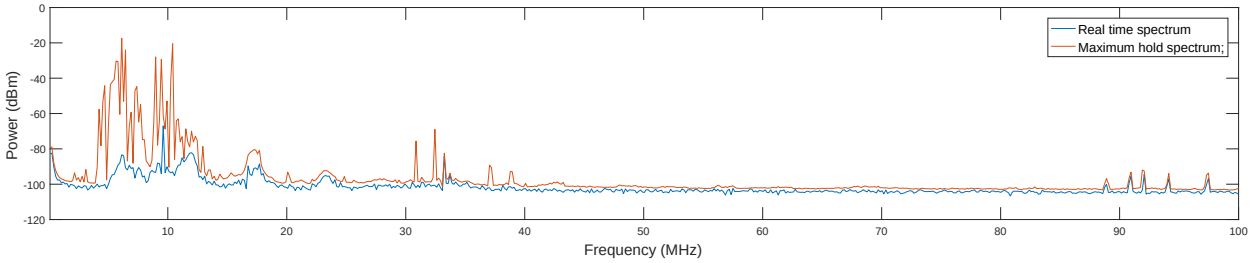


Figure 5.25: Spectrum levels as measured in Hermanus at SANSA with the addition of both LNAs and the low pass filter.

Two more spectrums were obtained with some of the riometer front-end added in – the two LNAs in series followed by the low pass filter. It was hoped that with the addition of the amplifiers that the noise floor received via the antenna would be pushed above the noise floor of the spectrum analyser. It was also an attempt at testing the characteristics of the front-end. The spectrum levels for the real time and maximum hold are shown in Figure 5.25. The spectrum looks very different as compared to the spectrum obtained in Figure 5.24. One can clearly see the influence of the low pass filter on the spectrum. It is also noticeable that the amplifiers pushed the noise floor above that of the spectrum analyser, revealing some additional information about the noise floor. The FM band is suppressed by the low pass filter and some additional power spikes can be seen between about 45 and 55 MHz.

Chapter 6

Discussion of Results

The stated hypothesis of this project was to create a digital HF riometer that samples at the RF. In Chapter 4 the design of both the single frequency (scanning) and spectral riometers were detailed. From there a number of tests were run to assess the hypothesis. In this chapter the results and their relevance to the initial hypothesis will be discussed. In Section 6.1 the FPGA code verification results are discussed, with the front-end test results being discussed in Section 6.2. In Sections 6.3 and 6.4 the trial results are discussed and compared, both with each other and the professional riometer. In addition Section 6.5 will evaluate how the demonstrator riometer could be ported to the radar at SANAE IV, as one of the secondary aims of this project is to evaluate the changes that might need to be made to the SuperDARN radar to allow it to operate as a riometer.

6.1 Comments on FPGA Code Verification Results

Four different tests were carried out to verify the correctness of the FPGA code, two each for the scanning riometer and the spectral riometer respectively.

For the scanning riometer, a sinusoid injection test and minimum detection test were run. The minimum detection test showed that the minimum detection circuitry was working as expected, with the scanning riometer consistently choosing the frequency bin with the lowest power in the spectrum it received. For the sinusoid injection test some inconsistencies between the simulated results and actual results were recorded. The average error achieved across all the sinusoid frequencies and different bandwidths and integration time was $6.46\,dB$. Most of this error occurred at the lower end of the amplitude scale, as seen in Figure 5.1 and can be attributed to differences in the MATLAB and Vivado implementations of both the FFT and Cascaded Integrator-Comb filter.

The differences in FFT implementations are explained further down in this section when talking about

the spectral riometer FPGA code verification tests and results, however, the CIC filter is discussed here. In the FPGA code the CIC filter is implemented with 5 stages. At each stage the bit width changes, with the width decreasing at each stage. Thus truncations take place to accommodate the decreasing bit width. MATLAB's CIC filter implementation is also capable of modelling changing bit widths, but whether the result is truncated or rounded at each stage is not detailed, and this difference in implementation could be part of the reason that the simulated and actual results are quite different.

For the spectral riometer the sinusoid injection test and complex signal test both showed similar results. The mean error for the sinusoid injection test, unwindowed, was $0.02\,dB$, which is extremely low. Most of this error was from the $60\,MHz$ test which made up half of the total error measured. The mean error for the windowed tests was $0.35\,dB$. Both of these results are much closer to zero than those of the scanning riometer. Discrepancies can be explained by the differences between the simulation's MATLAB FFT and that of the FPGA implementation. The MATLAB implementation only allows for double precision inputs and outputs, as compared to the Vivado implementation that inputs and outputs fixed point numbers of a given bit width. In the case of the spectral riometer, a 16 bit input is used and two 28 bit outputs are used to represent the real and imaginary parts of the output. At low amplitudes the limited precision of the Vivado FFT differs from that of the MATLAB implementation, leading to differences in the mean error observed.

6.2 Comments on Front-end Test Results

The front-end was tested using three criteria: stability, linearity and dynamic range. The details of the tests and results are given in Section 5.2. Practical tests were run for the stability and linearity tests. The results of these tests were promising, with the front-end shown to be only slightly less stable than the predicted results and the linearity condition being met. Dynamic range was only examined theoretically.

In the stability test the front-end was tied to $50\,\Omega$ termination and the noise power at $30\,MHz$ measured over a 22 hour period. In theory the measured noise temperature should have been accurate to within $1.8\,K$. In reality it was only accurate to within $9.6\,K$. The difference between the theoretical and actual noise temperature measured can be attributed to changes in front-end gain, since the theoretical result does not take front-end gain change into account. If the modified sensitivity equation is considered:

$$T_{min} = T_{sys}(\frac{1}{\sqrt{BT}} + \frac{\Delta G}{G}), \tag{6.1}$$

and rearrange it so that $\frac{\Delta G}{G}$ is the subject of the equation such that

$$\frac{\Delta G}{G} = \frac{T_{min}}{T_{sys}} \frac{1}{\sqrt{BT}}, \tag{6.2}$$

then the ratio of gain change to gain can be calculated. Given that T_{min} is 9.6 K, T_{sys} is 440 K, bandwidth, B, is 32.5 kHz and integration time, T, is 1 s, the gain change over gain ratio is 0.018. This means that the gain is varying by almost 2 % over the period that the stability calculations were made.

The linearity of the front-end is largely influenced by the only active components in the receiver chain, the amplifiers. As long as these are operated in their linear operation region, the receiver chain should be linear. Operating them near to their maximum input power at 5 dBm would lead to non-linear amplification of the incoming signal. To test the linearity over a wide range of input power, that were nonetheless well below the maximum input power, a noise signal with linearly increasing standard deviation was injected into the demonstrator riometer front-end. The results, as shown in Figure 5.9, are as expected with the power received increasing quadratically, due to the received power being proportional to the standard deviation of the noise squared. In the field trials the system seemed to be operating linearly, except possibly when very large man-made signals were detected. However, this is unconfirmed and these observations are only made with reference to the spectrum obtained in Hermanus and the measurements made for each trial.

Since the dynamic range front-end performance was a purely theoretical analysis, there is not much to add here. The dynamic range of the 14 bit ADC is more than sufficient to record even small changes in ionospheric absorption. One issue that could be addressed is the fact that the ZFL-500LN+ amplifier can only provide a maximum output power of 5 dBm. In order to utilise the full 10 dBm that the ADC can accommodate an amplifier with a maximum output power of 10 dBm could be considered. Overall the benefit of this is likely to be minimal, however, and operating the ADC too near to its maximum input power is also unwise as gain change could then cause the ADC to saturate.

6.3 Comparison and Progression of Results Across Trials

Four field trials were conducted of which three trials produced usable data. The second trial failed to produce any usable data due to an undetected fault in the antenna's balun. The results tended to improve as changes were made between each trial in an attempt to achieve a result closer to the ground truth data.

In trial one the dipole antenna was made to operate at 42.2 MHz. The SANSA riometer was working at 38.2 MHz which meant there was a mismatch between the ground truth data and that of the demonstrator system. This, in addition to the lack of windowing, meant that the curves recorded, shown in Section 5.3.1, did not match well to the ground truth data.

One major change that was made between trial one and trial three was that the antenna design was changed. A new dipole antenna was designed to operate at $38.2\,MHz$, complete with a sleeve balun for matching purposes, as opposed to the 1:1 design used for trial one. Trial three was also held in Fish Hoek as opposed to Hermanus. Overall the power received has a better fit to the ground truth data, when less interference was present, in the early hours of the morning. However, outside of this region much more interference was present. This was most likely due to Fish Hoek being a noisier EM region during these times, and increased spectral leakage into the bins of interest from nearby interfering frequencies.

Therefore in trial four a hamming window before the FFT was applied, reducing the sidelobe level of the power received in each bin at the cost of increased mainlobe width and reduced peak height. A high pass filter was also added to the front-end to suppress some of the lower frequency interference. The results of trial four, as detailed in Section 5.3.4, show a marked improvement from previous trials in following the trend of the ground truth riometer power readings. A QDC can clearly be observed in the riometer data at $38.2\,MHz$ and the spikes in power due to the nearby ionosonde correspond in both the ground truth and actual power recordings.

Trial four helped to confirm the hypothesis that it is possible to create a digital HF riometer that samples at the RF.

6.4 Reasons for Differences between Ground Truth and Recorded Data

As the results across the trials have demonstrated, the demonstrator riometer produces significantly noisier readings than the professionally built system. Even in trial four, while the quiet day curves closely matched, the demonstrator system produced noisier results. There are some factors that may account for the difference in performance:

- The front-end of the professional system was buried underground in an insulated box. This has the benefit of providing a form of temperature stability which is important in reducing analogue component variations. The front-end of the riometer produced here was located outside exposed to the elements and associated temperature changes.
- The professional system's front-end had shielding to reduce electromagnetic interference, unlike the system produced here. It was noticed, in the lab, that operating a signal generator producing a sinusoid near to the riometer's front-end produced interference at the sinusoid's frequency even when the antenna was completely disconnected. Clearly some form of shielding would have helped reduce this interference.
- The professional riometer system had two dipole antennas from which a single narrower beam

was formed by adding the signals. The narrower beam could have helped to reduce interference from HF signals bouncing off the ionosphere.

– Differences in the front-end filters used could have contributed to differences in the received power. As Figures 4.2, 4.3 and 4.4 show, the performance of the demonstrator riometer filters is not satisfactory across the passband. Better filters, with lower S11 parameters across the passband, would no doubt improve the results achieved.

In addition, trial one had some extra issues, such as the antenna operating at a higher frequency than the professional system, which made direct comparison difficult, and the 1:1 balun not being able to match the $75\,\Omega$ antenna to the $50\,\Omega$ cable at the frequencies of interest. Direct access to the La Jolla system was not possible so some of the factors mentioned here are largely speculative. In particular since access to the La Jolla front-end was not possible a direct comparison between the filters used in their riometer could not be carried out.

6.5 Adapting the Demonstrator System to the SuperDARN Radar

While the demonstrator system was developed largely independently of the SuperDARN radar at SANAE IV, it should be noted that this design will have to be adapted when implemented on the SuperDARN radar. This section will start with an overview of the TIGER-3 radar, including both hardware and FPGA software, before describing how the SuperDARN radar could be adapted.

6.5.1 TIGER-3 Overview

The antenna array of the SuperDARN radar at SANAE IV consists of 16 horizontally polarised twin-terminated folded dipoles (TTFD) in the main array and 4 TTFD dipoles in the auxiliary array $100\,m$ away from the main array. Each dipole has its own transceiver, shown in Figure 6.1. The receiver part of the transceiver consists of a high-pass filter at $8\,MHz$, followed by a low noise amplifier with $28\,dB$ gain. This is followed by a variable attenuator and another amplifier at $24\,dB$. Finally there is a low-pass filter at $30\,MHz$. This feeds into the FPGA via the 16 bit ADC which samples at $125\,MHz$, capturing the entire filtered frequency spectrum (8 - $30\,MHz$). The transmit portion of the transceiver is also shown in Figure 6.1, but is not significant in the operation of the riometer.

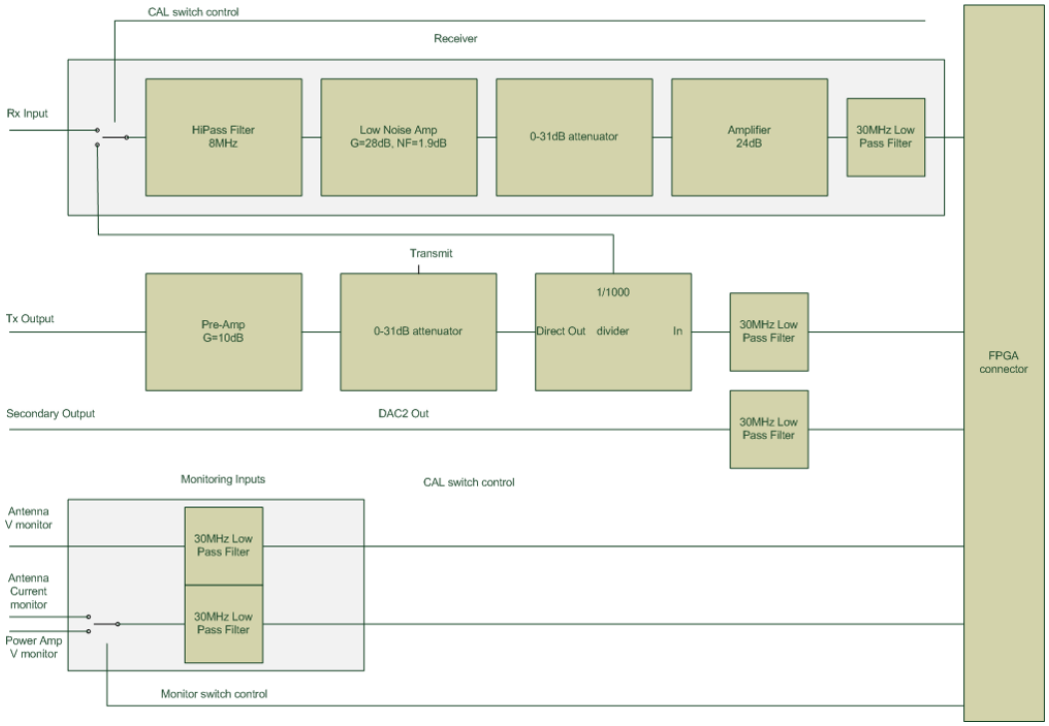


Figure 6.1: An overview of the TIGER-3 receiver [7].

Figure 6.2 shows the FPGA layout for the TIGER-3 radar processing. Of particular importance is the receive chain inside the FPGA, as well as the timing sequence control. The received samples are passed to a digital receiver which performs digital down-conversion to the frequency and bandwidth of interest which is determined by the current pulse in the pulse sequence. The pulse sequence is designed with frequency diversity, to provide unambiguous doppler, and uses a varying PRI to provide far unambiguous range. After down-conversion the data stream from each FPGA is sent to a central radar server which combines the data streams into a single output product. A riometer could be implemented at this stage in the processing pipeline, however, it would only be able to operate at the frequencies that the radar transmits and receives and would only receive data when the transceivers are receiving data. Also, due to the down-conversion process, the received stream would include the echoes from the transmitted waveform, possibly obscuring the noise floor. An Automatic Gain Control (AGC) function is also implemented on the FPGA and controls the variable attenuator in the front-end. Other blocks in the FPGA are dedicated to transmission, general monitoring of the system and communications.

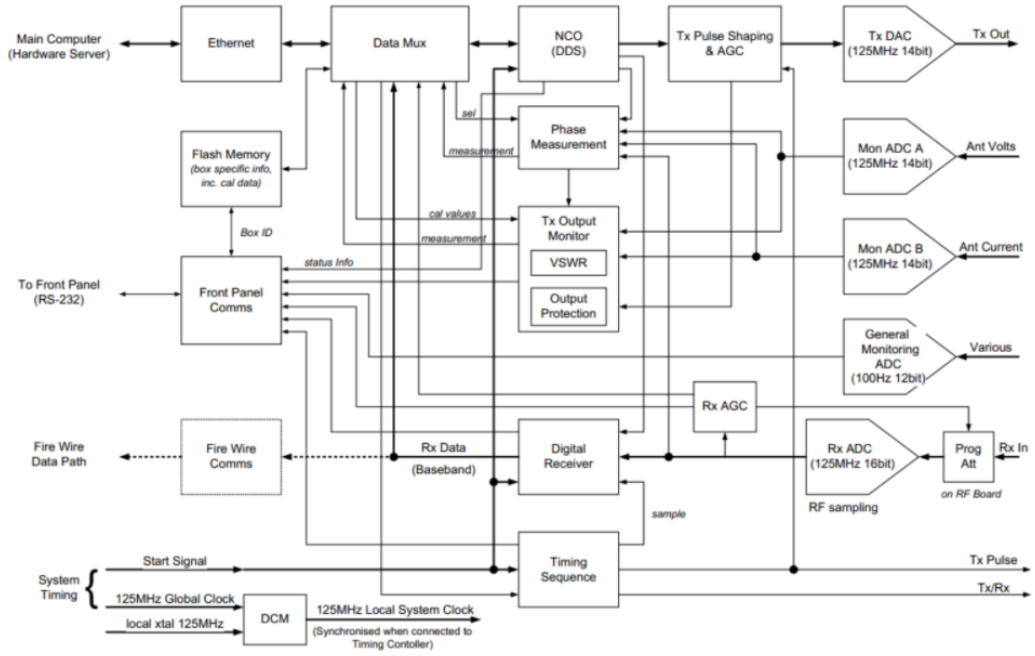


Figure 6.2: An overview of the current TIGER-3 FPGA layout for radar processing [7].

6.5.2 Proposed Changes to the SuperDARN radar

In practice a riometer conversion would have to modify the FPGA code itself. While the necessary Verilog code to create a working riometer has already been implemented in this project, it will only work on a single antenna system. The change to a multi-antenna, multi-FPGA system has some very important ramifications on the design. A more sparse design is proposed. Each transceiver forwards the sampled data directly to a central FPGA. This FPGA performs the task of beam-forming and, once the 16 streams have been combined, performs the spectral or scanning riometry process on it. By changing the phase factors that control the beam-forming the beam can be steered to create an imaging system. Figure 6.3 shows the proposed design layout of the upgraded system.

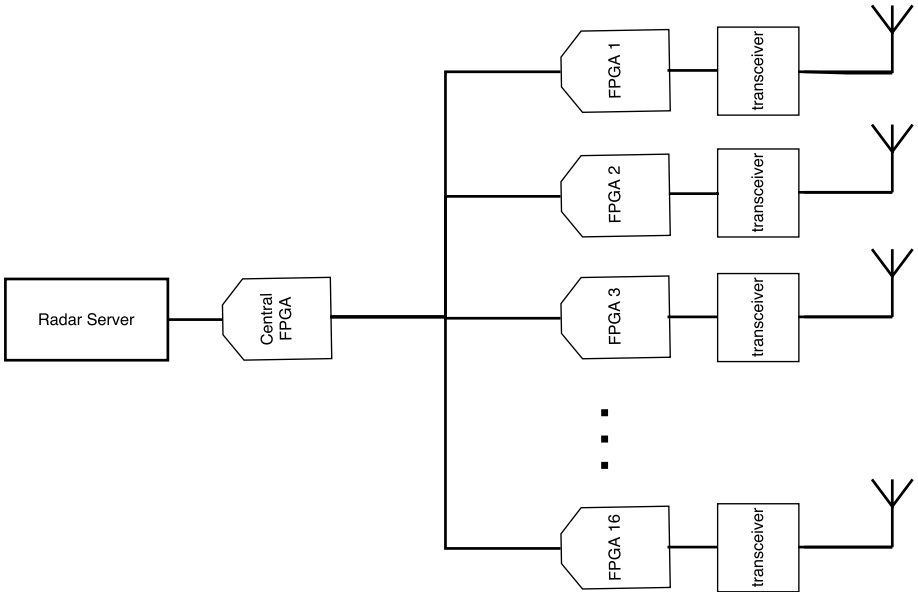


Figure 6.3: Proposed layout of the SuperDARN riometer system. An extra FPGA is added for riometer processing.

The conversion of the SuperDARN radar at SANAE IV to a riometer will be addressed by starting at the antennas and moving steadily through the front-end and into the FPGA and back-end thereby covering all the key areas.

The antennas are designed to operate between 8 and 20 MHz, well below the typical riometer frequencies of 25 to 50 MHz. There are two reasons this could be problematic. The first reason is that incoming cosmic radiation is more likely to be refracted by the ionosphere and between 8 and 20 MHz most of the signal will probably be refracted back into space. This, however, is slightly counteracted by the increased cosmic noise radiation intensity at lower frequencies and the reduced absorption at lower frequencies. Refraction could nonetheless lead to spurious absorption readings. The second reason is that atmospheric noise, caused by lightning storms, produces increased noise at these frequencies which can interfere with riometer measurements, even if they are extremely far away as the noise will be carried for very long distances via refraction off the ionosphere. Another issue is that the antenna beams do not point orthogonally to the earth and have an elevation angle that varies with frequency. For example, at 14 MHz the elevation angle is 20^{o}[52]. If the beam is not orthogonal to the ionosphere then cosmic noise radiation entering into that beam will pass through a greater volume of the ionosphere, resulting in increased absorption as compared to the orthogonal. This effect can be corrected for in post-processing [27].

Next the front-end is examined. Each TTFD antenna is attached to a transceiver and FPGA. The front-end is shown in Figure 6.1 and in the receiver the incoming signal is high-pass filtered at 8 MHz before passing through a 28 dB amplifier, a variable attenuator of up to 31 dB and another 24 dB amplifier. It is then passed through a low-pass anti-aliasing filter at 30 MHz. Purely in terms of amplification, the receiver can provide a maximum of 52 db of gain which is only 4 dB below what the demonstrator riometer provides. Thus, provided the variable attenuator can be controlled without interrupting the

radar operations, enough gain is present. The filters are sufficient for operating in the region of 8 to 20 MHz. A switch might need to be added to each receiver if Dicke switching is desired, however, it has been shown thus far that a short receiver chain like this one does not necessarily need any switching scheme to maintain good results.

Moving onto the ADC, it has 16 bits of resolution and samples at 125 $MSPS$, meaning it has better resolution than the 14 bit one used by the Red Pitaya and has exactly the same sampling rate. So there is no need to modify the current ADC on the SuperDARN radar.

After the ADC, the sampled data from each antenna is passed to the FPGA. Riometer processing needs to be done after beam-forming. Naturally, since the code was written in Verilog significant amounts of it can easily be ported over. However, the interface between the FPGA and CPU would have to be re-factored. The input to the riometer would also need to be pointed at the output of the beam-forming process instead of directly at the ADC. On top of this, the current Verilog code would have to be integrated with the logic already present on the radar's FPGA, adding a layer of complexity and requiring additional logic elements.

The riometer design as it currently stands uses 17491 slice LUTs out of 17600 available on the Xilinx 7010 chip. The Virtex-5 XC5VSX50T FPGA used by the SuperDARN radar has 32640 slice LUTs [53]. If the exact same design that was implemented for the demonstrator riometer is used in the exact same way, 15149 LUT slices would be left for the normal radar operation. In terms of DSP slices, FPGA slices optimised for common signal processing applications such as multiplying, the riometer design uses 45 out of the 80 available. The Virtex-5 has 288 available, more than enough to accommodate the 40 DSP slices required. Finally, the riometer design requires 1458 kb of RAM. The Virtex-5 can provide 4752 kb of RAM, leaving 3492 kb of RAM for the radar design. Unfortunately the FPGA code for each transceiver was not available at the time of writing and so the resource utilisation was not available.

In summary, adapting the radar at SANAE IV to include a riometer mode will require the addition of another FPGA, possibly a switch at the front-end of each transceiver and some significant changes to the FPGA code as implemented at the moment. No changes would need to be made to the antenna, front-end filters and amplifiers or the radar server hardware. The Virtex-5 is also theoretically capable of supporting both the current radar digital design and the additional logic required for riometer support.

6.5.3 Interoperability with Radar Processing

Initially the riometer operation could be implemented as a standalone mode on the SuperDARN radar. This seems relatively straightforward given the riometer will have direct access to the ADC streams and transceiver hardware. Thus it will be able to override the automatic gain control that the radar uses and set the gain to the highest possible value, as well as performing its own processing on each of the beams. The FPGA code would simply require a switch to determine whether the standard radar logic is to run, or the riometer logic is to run.

Operating the radar and the riometer at the same time is more complicated. Since the functioning of the radar as normal is the most important directive, the riometer must not interfere with the operation of the radar. To achieve this the radar logic would be allowed to operate as normal, with the riometer having enough information about what the radar is doing in order to operate itself. Some of the key aspects of these changes that need consideration are:

- During transmission the radar switches its front-end to a dummy load so as not to saturate or damage its amplifiers. The riometer will need to know about this so that it does not record at these times.
- The radar automatically adjusts the value of the front-end attenuator as it operates using Automatic Gain Control. Clearly the gain of the front-end changing during operation is not ideal for riometer operation. At the very least the gain that is being set will need to be recorded so that it can be taken into account in post-processing. However, it would be better if the attenuation could be set to zero while the riometer is recording.

The radar is operated via the Radar Operating System (ROS). The riometer mode will have to be added to the radar setup to allow operators to decide whether they want the riometer running or not, as well as whether it is running on its own or simultaneously with the radar. The ROS will also need to be able to configure the riometer. Currently the configuration is done on the Red Pitaya itself in a C++ file. This file would have to be directly modified by the ROS or a different method of configuring the riometer would have to be proposed.

Chapter 7

Recommendations and Conclusions

The stated objective of this project was to develop a demonstrator HF riometer in aid of adding a riometer mode to the HF radar at SANAE IV. A demonstrator riometer system was designed using the Red Pitaya as its backend, with two distinct riometer modes: scanning and spectral. The digital design of the scanning and spectral riometers were tested by applying a known input and seeing if it matched the expected output. The spectral riometer implementation was found to perform satisfactorily, however, there were some doubts about the scanning riometer implementation. The front-end of the system was tested against a few key criteria: stability, linearity and dynamic range and was found to operate sufficiently well under these criteria.

After confirming that the digital design and front-end of the demonstrator riometer was sound, four field trials were performed of which three produced usable results. The first two trials took place at Hermanus with only the first trial being successful and the second one suffering from technical issues. The third trial was run in Fish Hoek with a different antenna design tuned to work at $38.2\,MHz$. The fourth field trial took place in Hermanus again using the same antenna design as trial three. In addition, windowing was added before the FFT and produced the results most closely matched to the ground truth data, a La Jolla riometer system on SANSA's premises. Comparisons of the results from the professional riometer system and the demonstrator system showed that they both followed roughly the same day night curve with discrepancies attributed to different antenna designs, lack of shielding, handmade filters and the professional system's improved insulation.

7.1 Conclusions

The demonstrator riometer successfully showed that low cost, powerful and flexible HF riometry is possible through its sound digital processing and real world performance. It was also shown that it is feasible to upgrade the SANAE IV SuperDARN radar to act as a dual riometer/radar system.

7.1.1 The Digital Processing behind the Demonstrator Riometer

The signal processing steps presented in Chapter 4 were shown to be working correctly in Chapter 5. Discrepancies between the simulated data and that produced by the digital implementation of the riometer can be described by the differences in the implementation of the FFT in MATLAB and in the FPGA. The spectral riometer was therefore shown to be working correctly, while there was some doubt over the scanning riometer due to the large discrepancies between the simulated results and those obtained experimentally from the demonstrator riometer.

A combination of tests running directly through the digital implementation on the FPGA were carried out to test the correctness of the implementation. For the scanning riometer a sinusoid injection test was used to test that the correct power level at a particular frequency was being obtained, and a minimum detection test was used to test the minimum detection circuitry.

Likewise, a sinusoid injection test was used to test the spectral riometer's power level reading across the frequency range. Thereafter, a complex signal made up of multiple sinusoids was injected into the spectral riometer implementation to check the ability of the riometer to deal with more complex signals.

7.1.2 Real World Performance

In real world testing against a professional riometer system used as a ground truth, the demonstrator HF riometer showed promise in following the curves produced by the professional system, with the results proving to be particularly promising for the fourth trial. It was thought that the reasons for differences between the ground truth data and that obtained by the riometer over the course of the trials were lack of electromagnetic shielding on the demonstrator system, the exposed location of the front-end electronics, without any temperature stability mechanism, and the larger beamwidth of the antenna design used, as compared to the professional system. The front-end filters were also thought to be at a lower standard than those of the professional system's. Although the stability of the front-end was shown to be excellent in the lab under controlled conditions, in the field it is possible that more variation in gain and bandwidth of the front-end could be expected and this may have influenced the results obtained as well, especially considering that the Dicke riometer method of mitigating changes in gain and bandwidth was not used for the trials.

In the fourth trial a clear day night curve could be seen, corresponding closely to that produced by the ground truth riometer. Therefore, overall, the field trials showed that low cost, digital HF riometry is a possibility and can be explored further.

7.1.3 Feasibility of Upgrading the Radar at SANAE IV to a Dual Purpose Radar/Riometer

The radar at SANAE IV can be converted into a dual purpose radar/riometer system. However, some hardware changes are required: both an additional FPGA for processing and an additional switch (should a Dicke-switched riometer be desired). On the software side the new mode will have to be integrated into the Radar Operating System and the FPGA code rewritten for a multiple antenna system. Beam-forming will also have to be implemented if an imaging riometer is desired.

7.2 Recommendations and Future Work

The work presented here builds upon the work already done on using flexible and powerful FPGA technology along with ADCs and DACs capable of sampling at hundreds of MHz to build HF systems. It is hoped that this will enable a number of future projects to expand upon this work. Foremost amongst those is the implementation of a riometer mode on the SuperDARN radar at SANAE IV. Should this be possible, the riometer design could be ported to other SuperDARN radars. Thereafter, this work, and other recent work done at Universities, such as the University of Cape Town and La Trobe University, should allow for the design and implementation of a new class of HF instruments on a flexible, FPGA based backend.

7.2.1 Improvements to the Demonstrator Riometer

Designing or buying some improved front-end filters, with a lower S11 parameter over the passband, below -10 dB, would help to improve performance. Likewise, adding EM shielding to the front-end would reduce interference, especially as the hand wound coils in the filters can act like loop antennas and pick up unwanted interference. Enclosing the entire front-end in a metal box should provide sufficient shielding.

On the FPGA side, windowing functions other than the hamming window could be implemented. Different windowing functions have different properties, such as sidelobe level and mainlobe width, that can be desirable for spectral riometers. Another feature that could be added is a digital filter before the window to further narrow the operation band without requiring narrower analogue filters. For instance, in the field trials done, the only frequencies of interest were very near to 38.2 MHz and much of the lower frequencies could have been filtered out digitally without much consequence.

7.2.2 Upgrade the SANAE IV Radar to Include a Riometer

Upgrading the radar at SANAE IV to include a riometer mode will require some adaptation to the riometer implementation presented here. While the front-end and antenna are mostly suitable for riometer operation, an additional switch would be required for a Dicke switched mode. In addition an extra FPGA would need to be added to the existing FPGAs present in the SuperDARN receiver to act as the central processing unit for the riometer. This will likely require a significant time investment to achieve as the adaptation is not necessarily straight forward.

7.2.3 Determine the Feasibility of Implementation on Other SuperDARN Radars

Most SuperDARN radars in the network have a similar design, however, most of them have yet to implement the fully digital transceiver design used by the SANAE IV radar and the Buckland park radar in Australia. Until such a time as these upgrades take place, adding a riometer mode to operate simultaneously with the radar will simply not be possible. Once these upgrades take place then transitioning to dual radar/riometer should be achievable with only a relatively minor effort. However, as each radar is not exactly the same, changes will have to be made to the original riometer design on a case-by-case basis and the feasibility will have to be examined separately for each radar.

7.2.4 Test Other HF Instruments on a Flexible FPGA Based Back-end

It is hoped that more advanced FPGAs and SOCs will allow the digitisation of HF systems to be shifted ever closer to the antenna. Considering that even ADCs of moderate capabilities are able to completely sample the HF spectrum, thereby eliminating the need for analogue down-conversion , this seems like a strong possibility. Other HF instruments can therefore be implemented on existing digital HF platforms simply by modifying the FPGA code. Some instruments that could be explored are ionosondes, surface wave radar and over-the-horizon radars. HF radar in particular stands to benefit from the hyper-flexibility that FPGAs can provide, allowing, for instance, the ability to easily change transmitted waveforms to a more suitable one for the task at hand.

Bibliography

[1] G. Chisham, M. Lester, S. E. Milan, M. P. Freeman, W. A. Bristow, A. Grocott, K. A. McWilliams, J. M. Ruohoniemi, T. K. Yeoman, P. L. Dyson, R. A. Greenwald, T. Kikuchi, M. Pinnock, J. P. S. Rash, N. Sato, G. J. Sofko, J. P. Villain, and A. D. M. Walker, "A decade of the Super Dual Auroral Radar Network (SuperDARN): Scientific achievements, new techniques and future directions," *Surveys in Geophysics*, vol. 28, no. 1, pp. 33–109, 2007.

[2] B.B., B.R., and Z.L., "Electrical schematics for red pitaya," https://dl.dropboxusercontent.com/s/jkdy0p05a2vfcba/Red_Pitaya_Schematics_v1.0.1.pdf, Red Pitaya, accessed: 2017-08-19.

[3] "Radar overview: Virginia tech superdarn," http://vt.superdarn.org/tiki-index.php?page=Radar+Overview, accessed: 2017-08-22.

[4] R. a. Greenwald, K. B. Baker, J. R. Dudeney, M. Pinnock, T. B. Jones, E. C. Thomas, J. P. Villain, J. C. Cerisier, C. Senior, C. Hanuise, R. D. Hunsucker, G. Sofko, J. Koehler, E. Nielsen, R. Pellinen, a. D. M. Walker, N. Sato, and H. Yamagishi, "DARN/SuperDARN," *Space Science Reviews*, vol. 71, pp. 761–796, 1995.

[5] "Convection map overview: Virginiatech superdarn." accessed: 2017-06-30. [Online]. Available: http://vt.superdarn.org/tiki-index.php?page=Conv+map+overview.

[6] R. Van Zyl, D. Visser, P. Cilliers, and B. Opperman, "Zacube-1 space weather mission: Characterize the superdarn hf radar antenna array at sanae-iv," *Space Weather*, vol. 11, no. 2, pp. 52–54, 2013.

[7] J. Devlin, A. Console, B. Bienvenu, V. Vu, and H. Nguyen, "Buckland park radar overview," June 2013.

[8] M. L. Parkinson, J. S. Whittington, J. C. Devlin *et al.*, "Why do we need digidarn a global network of digital superdarn radars," in *Proceedings of WARS'06 (Workshop on Applications of Radio Science), Leura, New South Wales*, 2006, p. 15.

[9] P. L. Dyson, J. C. Devlin, M. L. Parkinson, and J. Whittington, "The tasman international geospace environment radar (tiger)-current development and future plans," in *Radar Conference, 2003. Proceedings of the International.* IEEE, 2003, pp. 282–287.

[10] G. Fabrizio, *High Frequency Over-the-Horizon Radar: Fundamental Principles, Signal Processing, and Practical Applications*. McGraw-Hill Education, June 2013. [Online]. Available: https://books.google.co.za/books?id=R2qAyffnAaIC

[11] K. Davies and I. of Electrical Engineers, *Ionospheric Radio*, ser. Electromagnetics and Radar Series. Peregrinus, 1990. [Online]. Available: https://books.google.co.za/books?id=qdWUKSj5PCcC

[12] J. Ratcliffe, "The ionosphere and the engineer," *Electronics and Power*, vol. 12, no. 11, pp. 381–382, 1966.

[13] S. Andersen and J. Headricks, "Hf over-the-horizon radar," in *Radar Handbook, Third Edition*, 3rd ed., M. Skolnik, Ed. McGraw-Hill Education, 2008, p. 20.

[14] S. Chapman, "The absorption and dissociative or ionizing effect of monochromatic radiation in an atmosphere on a rotating earth," *Proceedings of the Physical Society*, vol. 43, no. 1, pp. 26–45, 1931.

[15] M. K. Öztürk, "Trajectories of charged particles trapped in earths magnetic field," *American Journal of Physics*, vol. 80, no. 5, pp. 420–428, 2012.

[16] V. Jodalen and E. Thrane, "A study of the relation between ionospheric absorption and predicted hf propagation parameters at high latitudes," pp. 64–68, 1994.

[17] C. Little and H. Leinbach, "The Riometer-A Device for the Continuous Measurement of Ionospheric Absorption," *Proceedings of the IRE*, vol. 47, no. 2, pp. 315–320, 1959.

[18] A. Mitra and C. Shain, "The measurement of ionospheric absorption using observations of 18·3 mc/s cosmic radio noise," *Journal of Atmospheric and Terrestrial Physics*, vol. 4, no. 4, pp. 204–218, 1953.

[19] R. Shaw and D. Routledge, "A digital-analogue minimum detector for riometers," *Radio and Electronic Engineer*, vol. 44, no. 7, pp. 355–356, 1974.

[20] A. T. Weatherwax, C. D. Fry, T. J. Rosenberg, L. Lutz, D. L. Detrick, E. Knouse, H. Breden, and J. Giganti, "A programmable riometer for Earth and Mars," vol. 35, no. 1, pp. 263–274, 2000.

[21] K. Tapping and M. Leech, "Rtlsdr-based, software defined radio alternative to switched radiometers for continuum radio astronomy," 2015.

[22] L. Marcus, T. Trondsen, and T. Matthews, "An advanced riometer platform based on sdr techniques."

[23] I. T. U. Radiocommunication Assembly, *RECOMMENDATION ITU-R: Radio Noise*, Std., 2001.

[24] R. H. Dicke, "The measurement of thermal radiation at microwave frequencies," *Review of Scientific Instruments*, vol. 17, no. 7, pp. 268–275, 1946.

[25] M. Ryle and D. Vonberg, "Solar radiation on 175 mc./s." in *Classics in Radio Astronomy*. Springer, 1946, pp. 184–187.

[26] D. Detrick and T. Rosenberg, "A phased-array radiowave imager for studies of cosmic noise absorption," *Radio Science*, vol. 25, no. 4, pp. 325–338, 1990.

[27] S. Browne, J. K. Hargreaves, and B. Honary, "An imaging riometer for ionospheric studies," *Electronics & Communication Engineering Journal*, vol. 7, no. 5, pp. 209–217, 1995. [Online]. Available: http://eprints.lancs.ac.uk/9824/

[28] Y. Murayama, H. Mori, S. Kainuma, M. Ishii, I. Nishimuta, K. Igarashi, H. Yamagish, and M. Nishino, "Development of a high-resolution imaging riometer for the middle and upper atmosphere observation program at poker flat, alaska," *Journal of Atmospheric and Solar-Terrestrial Physics*, vol. 59, no. 8, pp. 925–937, 1997.

[29] F. Honary, S. R. Marple, K. Barratt, P. Chapman, M. Grill, and E. Nielsen, "Digital beam-forming imaging riometer systems," *REVIEW OF SCIENTIFIC INSTRUMENTS*, vol. 82, 2011.

[30] A. Kero, J. Vierinen, D. McKay-Bukowski, C.-F. Enell, M. Sinor, L. Roininen, and Y. Ogawa, "Ionospheric electron density profiles inverted from a spectral riometer measurement," *Geophysical Research Letters*, vol. 41, no. 15, pp. 5370–5375, 2014.

[31] D. McKay-Bukowski, J. Vierinen, I. I. Virtanen, R. Fallows, M. Postila, T. Ulich, O. Wucknitz, M. Brentjens, N. Ebbendorf, C.-F. Enell *et al.*, "Kaira: The kilpisjärvi atmospheric imaging receiver arraysystem overview and first results," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 3, pp. 1440–1451, 2015.

[32] R. Parthasarathy, G. Lerfald, and C. Little, "Derivation of electron-density profiles in the lower ionosphere using radio absorption measurements at multiple frequencies," *Journal of Geophysical Research*, vol. 68, no. 12, pp. 3581–3588, 1963.

[33] B. Lusignan, "Cosmic noise absorption measurements at stanford, california, and pullman, washington," *Journal of Geophysical Research*, vol. 65, no. 12, pp. 3895–3902, 1960.

[34] Y. Tanaka, K. Makita, M. Nishino, and T. Ookawa, "Development of data analysis program for imaging riometer by using matlab," *Bulletin of science and engineering, Takushoku University*, vol. 10, no. 1, pp. 61–66, 2007.

[35] J. Moro, C. Denardini, E. Correia, M. Abdu, N. Schuch, and K. Makita, "A comparison of two different techniques for deriving the quiet day curve from sarinet riometer data," in *Annales Geophysicae*, vol. 30, no. 8. European Geophysical Society, Max-Planck-Str. 13 Katlenburg-Lindau Germany, 2012, pp. 1159–1168.

[36] J. Wu and P. Wilkinson, "Time-weighted magnetic indices as predictors of ionospheric behaviour," *Journal of Atmospheric and Terrestrial Physics*, vol. 57, no. 14, pp. 1763–1770, 1995.

[37] J. L. Hook, "Some observations of ionospheric absorption at geomagnetic conjugate stations in the auroral zone," *Journal of Geophysical Research*, vol. 67, no. 1, pp. 115–122, 1962.

[38] C. Holt, B. Landmark, and F. Lied, "Analysis of riometer observations obtained during polar radio blackouts," *Journal of Atmospheric and Terrestrial Physics*, vol. 23, pp. 229IN1231IN3243–230IN2242, 1961.

[39] D. Bailey and M. Pomerantz, "Relativistic electron precipitation into the mesosphere at subauroral latitudes," *Journal of Geophysical Research*, vol. 70, no. 23, pp. 5823–5830, 1965.

[40] V. G. Bezrodny, O. V. Charkina, V. G. Galushko, K. Groves, A. S. Kashcheyev, B. J. Watkins, Y. M. Yampolski, and Y. Murayama, "Application of an imaging HF riometer for the observation of scintillations of discrete cosmic sources," vol. 43, pp. 1–14, 2008.

[41] P. Stauning, "High-latitude d-and e-region investigations using imaging riometer observations," *Journal of Atmospheric and Terrestrial Physics*, vol. 58, no. 6, pp. 765–783, 1996.

[42] N. C. Mehta and N. D'Angelo, "Cosmic noise absorptionby e region plasma waves," *Journal of Geophysical Research: Space Physics*, vol. 85, no. A4, pp. 1779–1782, 1980.

[43] *Coaxial Low Noise Amplifier ZFL-500LN+*, Mini-Circuits, July 2014, rev. D.

[44] S. Maagoe, *Solid State Riometer Manual*, La Jolla, 1972.

[45] "Red pitaya - getting started," accessed: 2017-06-01. [Online]. Available: http://www.sparxeng.com/blog/hardware/red-pitaya-getting-started

[46] *AXI Reference Guide*, Xilinx, March 2011, v 13.1.

[47] P. Demin, "Red pitaya notes," accessed: 2017-08-19. [Online]. Available: http://pavel-demin.github.io/red-pitaya-notes/

[48] *8761A/B Microwave Switches*, Keysight Technologies, July 2014.

[49] L. Guangxi, "Gaussian noise generator," Feb 2015. [Online]. Available: https://opencores.org/project,gng

[50] J. Tsui, *Special Design Topics in Digital Wideband Receivers*, ser. Artech House Radar. Artech House, 2010.

[51] *LTC2208 14 bit, 130 Msps ADC*, Linear Technology, 2006, rev. B.

[52] E. Custovic, A. J. McDonald, J. Whittington, D. Elton, T. A. Kane, and J. C. Devlin, "New antenna layout for a superdarn hf radar," *Radio Science*, vol. 48, no. 6, pp. 722–728, 2013.

[53] *Virtex-5 Family Overview*, Xilinx, August 2015, v 5.1.

Appendix A

Full FPGA Layout in Vivado

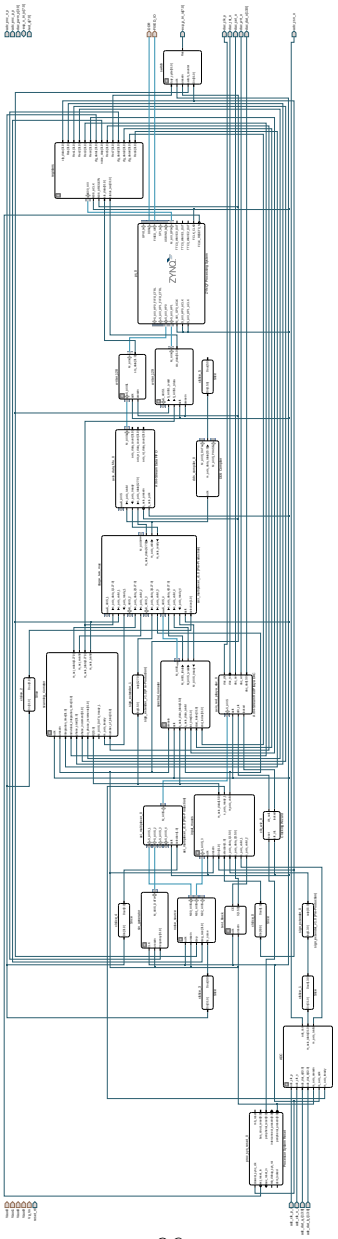


Figure A.1: Full layout of the digital design on the FPGA.

Appendix B

Radiometer Sensitivity Equation

The radiometer equation describes the sensitivity of a radiometer in terms of its system noise temperature, bandwidth and integration time. Consider a simple receiver consisting of an antenna, perfect bandpass filter, integrator and square law detector. At the output one would observe what is essentially bandlimited noise. Since the bandwidth is limited to that of the bandpass filter, B, the sampling theorem states that the signal can be represented by $2N$ independent samples. N is given by the product of the integration time and bandwidth, BT. This is due to the fact that the Nyquist sampling theorem says it takes two samples per period to quantify a given sinusoid. Therefore for BHz, $2B$ samples are required. This is multiplied by the number of seconds, T - the integration time.

The central limit theorem states that given enough samples from a given population, the samples will follow an approximate normal distribution. Given that T_{sys} is the variance in the noise, the minimum detectable noise temperature can then be given by:

$$T_{min} = \frac{\sqrt{2}T_{sys}}{\sqrt{N}}, \tag{B.1}$$

which simplifies to

$$T_{min} = \frac{\sqrt{2}T_{sys}}{\sqrt{2}\sqrt{BT}} \tag{B.2}$$

and finally:

$$T_{min} = \frac{T_{sys}}{\sqrt{BT}} \tag{B.3}$$