Lab #2: Thermal Radiation & the Statistics of Noise

Lab report is due Wednesday, Nov 14th, 2017, before 11:59 pm EDT

1. Overview

This handout provides a description of the second lab, which covers the basics of radio signal detection and the properties of thermal radiation and noise. This lab builds on many of the skills learned during the first lab, but will require you to handle larger data sets, develop more complex data processing algorithms, and fit models to your data.

1.1 Schedule

This is a three-week lab, with lectures and lab sessions the weeks of October 15th, 22nd, and 29th. Because of fall reading week, there is a compressed schedule for getting all the data you need: you should aim to complete everything through section 4.3 by during the 1st week, section 4.4 during the 2nd, and be sure your data looks reasonable for 4.5 before the end of the 3rd lab week. Section 4.5 and 4.6 draw on the same data taken in 4.4, but **do not underestimate the time the analysis and write-up will take!** The lab report is due following the break, to be submitted electronically on Nov 14th on or before 11:59pm EDT.

1.2 Goals

Investigate how radiometers work, and how to manipulate coherent measurements of light. As in lab #1, explore limitations in our ability to detect light, and quantify uncertainties. Fit models to data, and calibrate a radio receiver in physical units.

1.3 Reading assignments

- Normal and χ^2 ("chi-squared") distributions, first and second moments (mean, variance)
  Wikipedia has excellent reference pages on normal and χ^2 distributions.
- Radiometers and the Radiometer Equation
  Essential Radio Astronomy (Condon & Ransom, 2016) Ch 3.6.
  Available online, http://www.cv.nrao.edu/~sransom/web/Ch3.html
  Or older form, http://www.cv.nrao.edu/course/astr534/Radiometers.html
- Python: arrays, functions, and advanced plotting

1.4 Key Steps

The lab is broken down into several stages:

1. Familiarize yourself with the receiver, how to set gains, how heterodyning works, how to record and read back data.
2. Inspect a timestream of data, plot the distribution of electric fields and power measured.
3. Generate a spectrum. Plot the distribution of fields and power across the band of reception.

4. Vary the physical temperature of the load and record additional data. Generate plots and statistics describing the electric fields and power measured.
5. Calculate the contribution the receiver is adding to the measured signals.
6. Produce spectra from the data in step (4), generate statistics describing the fields and power and calculate the contribution of the receiver across the spectrum.
7. Write your lab report.

2. Radio Astronomy Background

When measuring signals at the long-wavelength end of the spectrum, the photoelectric effect can no longer be used: individual photons no longer have the energy to photo-ionize materials. In this lab, we will be looking at radio signals around 1GHz: how does the energy of one such radio photon compare to the energy you measured in a soft X-ray photon during lab #1?

> *A word of caution*: Radio astronomy grew out of a different background from other bands, and uses a wide array of specialized terms which may seem unfamiliar at first glance. Don't panic, the underlying principles are the same as ever, and you can usually find equivalents in more familiar bands. For example, radio astronomers don't talk of ***Point Spread Functions*** (PSFs), they refer to their **Beams**. Neither do they have ***cameras***, instead building **receivers** to measure and record incoming light. Keep an eye out for jargon, you'll get the hang of these terms as you go.

2.1 Radio receivers

While CCDs, CMOS, and other photon-counting technologies are out of the question, radio astronomers are able to measure sufficiently strong signals *coherently*: they can directly measure and record the oscillating electric fields which make up incident light. Radio receivers typically include:

- A **feed**, which efficiently couples light from free space into a waveguide.
- **Waveguide**, which constrains the propagation of light to a particular direction, and may be made of metal tubes, transmission lines, microstrip, coaxial cable, or at higher frequencies, optical fibre.
- **Mixers**, which are used to shift the frequency of light into a band which is easier to measure, through a process known as **heterodyning**.
- **Amplifiers**, which directly amplify the oscillating electric fields, making them bright enough to measure, at the cost of additional noise.
- **Filters**, which reflect or **attenuate** (absorb) undesired frequencies of light, limiting what colors are allowed to continue downstream. Typically, these are arranged as a **bandpass**, allowing some continuous-but-limited range of frequencies to continue through the receiver chain.
- **Analog-to-digital Converters (ADCs)**, which finally measure the amplified electric fields, returning a digital representation of their amplitude. By the Nyquist sampling theorem (see below), these must sample at least twice as fast as the bandwidth we want to digitize, e.g. if you want to record a 100MHz wide band, the ADC must sample at 200MHz, measuring a voltage every 5ns.

For most of this lab, we will be observing the radio signals generated by a thermal source, so the feed will be replaced by a resistor known as a **terminator**, which will radiate thermal noise directly into our waveguide cables.

2.2 Power vs Electric Fields

An important distinction between photon counters and coherent receivers is that photon counters are a type of **total power** receiver, they measure the incident power, often in units of electrons. Recall from your electromagnetics classes that the power in a traveling EM wave is

$$P = \frac{\epsilon_0 c\, E_0^2}{2} \propto \langle |E|^2 \rangle \tag{1}$$

That is, the power in a coherent detector is simply the time-average of the squared magnitude of the electric field. This is an important relation to remember: whenever measuring electric fields, you have to square things before they can be compared to the familiar world of photons, power, and energy.

In this lab, we will be making coherent measurements: sampling the raw electric fields by recording the voltage present in our waveguide moment by moment. Throughout this manual, we will refer to the electric fields, but in reality it is these voltages that are being sampled, as they trace the incoming electric fields. They are typically recorded as **timestreams**, time-ordered data showing the electric field at each instant in time.

2.3 Spectra and Fourier Transforms

One of the wonderful features of coherent measurements – of the electric field, not the incident power – is that measuring the **spectrum** of measured light becomes trivial. A spectrum shows how much power is arriving as a function of frequency (or wavelength), so the task is to take our timestream of electric field, and divide it into sinusoids of different frequencies.

Fourier transforms convert real-space functions into sums of sinusoids, so are exactly the tool needed to divide our timestream into components at different frequencies.

$$E(t) \rightarrow \mathcal{F} \rightarrow \tilde{E}(\nu) \tag{2}$$

Typically, these $\tilde{E}$ will describe the **amplitude** and **phase** of the sinusoid at that frequency, often as a complex number (containing a real component $\Re\{\tilde{E}\}$ and an imaginary component $\Im\{\tilde{E}\}$) which describes the amplitudes of the sine and cosine which make up the final sinusoid. The two conventions are easily converted between,

$$\left|\tilde{E}\right| = \sqrt{\Re\{\tilde{E}\}^2 + \Im\{\tilde{E}\}^2} \tag{3}$$
$$\tilde{E}_\phi = \mathrm{atan}\,(\Im\{\tilde{E}\}/\Re\{\tilde{E}\}) \tag{4}$$

In general, the phase of each sinusoid contains no useful information: it says something about the quantum mechanical uncertainty involved in the emission of photons by the source. *(The exception to this is in the case of interferometers, where the relative phase measured between two different detectors is of the utmost importance.)* For our purposes in this lab, the amplitude contains all the information we are interested in. Moreover, we want to measure the spectrum, which describes the power. As above, the power scales with the square of the time-average of the electric field, so to measure the spectral power, we square the Fourier amplitudes.

$$P(\upsilon) = \Re\{\tilde{E}(\upsilon)\}^2 + \Im\{\tilde{E}(\upsilon)\}^2 \qquad (5)$$

Strictly, these powers describe all the incident power across a range of nearby frequencies, in all the sinusoids so close in frequency that the timestream we measured can't distinguish between them, and would appear the same regardless of where in the narrow band that power exists. The width of these spectral bands, which we can think of as our spectral resolution, is given by the length of our timestream, sampled at cadence τ and containing N samples

$$\Delta\upsilon = 1/N\tau \qquad (6)$$

The highest frequncies we can measure the spectrum of without confusion is defined by the **Nyquist Sampling Theorem**, which states we must sample at twice the rate of the fastest varying sinusoid. Signals which vary faster than that, i.e. light at higher frequencies, will still appear in our timestream, but will masquerade as a lower frequency signal, so that we see an **alias** of these signals in our spectra.

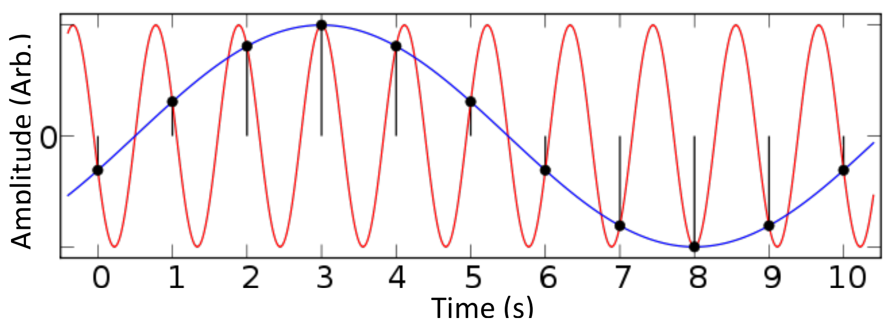

***Figure 1** – A timestream of measurements, marked by the **black** points, could be produced by the low frequency **blue** signal, or by the higher frequency **red** signal aliasing down. Typically, we add filters before the ADC to prevent this sort of confusion. Figure courtesy Wikipedia.*

Strictly, Nyquist states that we can measure without confusion a **bandwidth** defined by half the sampling rate; if we filter out low frequencies, we can measure aliased signals instead!

2.4 Thermal Noise

Objects at finite temperature will radiate (with varying efficiency) thermally, producing a blackbody spectrum of light. For anything warmer than a few millikelvin, radio frequencies will be well into the Rayleigh-Jeans portion of the thermal spectrum, meaning that the flux within any given radio band is a linear function of Temperature.

$$B_\upsilon(T) = \left(\frac{2\upsilon^2 k_B}{c^2}\right) T \qquad (7)$$

Radio astronomers typically describe signals in their receivers in terms of the temperature of the thermal source which would produce an equivalent amount of energy. This can be done in narrow spectral bands, or averaged across a larger band.

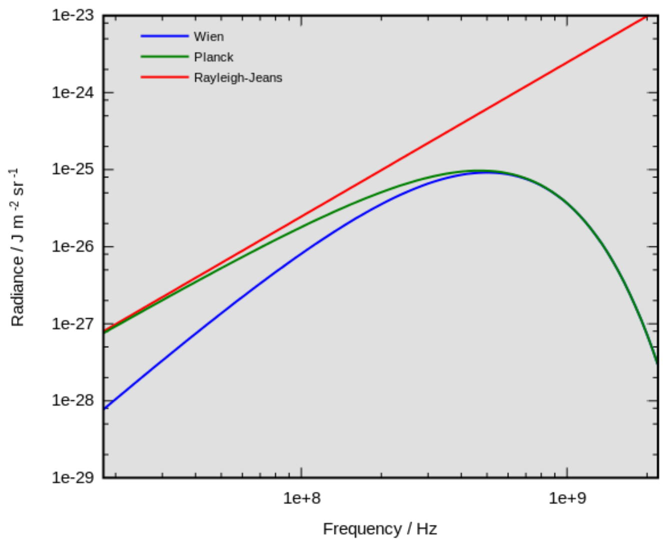

***Figure 2** – The Blackbody spectrum of an 8mK radiator, showing the Rayleigh-Jeans and Wien approximations at low and high frequencies, respectively. Figure courtesy Wikipedia.*

This shorthand, describing signal levels in terms of an equivalent thermal load, makes the calibration and understanding of an instrument much easier, as can be seen with a quick thought experiment. Imagine placing a radio feed in a blackened box, with a terminating resistor connected to its input in an adjacent box, with both boxes at the same temperature.

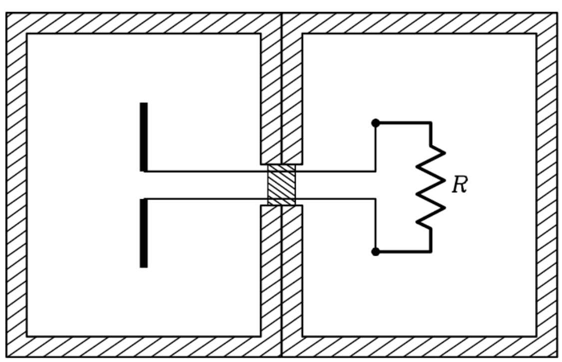

***Figure 3** – A thought experiment showing the relation between power and temperature. An antenna is placed in a blackened box (left), with its input connected to a terminating resistor in an adjacent blackened box (right). If the two boxes begin in thermal equilibrium, the power absorbed and emitted on both sides must be exactly equal to maintain thermal equilibrium. Figure from NRAO course on Antenna Fundamentals.*

The antenna will collect thermal light from the box, which will be dissipated in the resistor. At the same time, the resistor will generate noise due to the random thermal motion of its electrons known as Johnson Noise, with power per unit frequency

$$P_\nu = k_B T \tag{8}$$

or equivalently, the total power across some bandwidth

$$P = k_B T \Delta \upsilon \tag{9}$$

The second law of thermodynamics says this **must** be equal to the power absorbed by the antenna. Otherwise, the device would pump heat from one side to the other, moving things away from thermal equilibrium, allowing a perpetual motion machine. This convenient relation allows us to describe all signals in terms of an equivalent thermal load, and notice that that thermal load could be either a full-sky blackbody, or a simple terminating resistor.

Note also that since powers from different sources will add, the equivalent temperature simply adds up. If we have multiple sources of signals and / or noise in our system, we can sum their contributions to determine the total **system temperature**,

$$T_{sys} = T_{CMB} + \Delta T_{source} + T_{atm} + T_{spill} + T_{RFI} + T_{receiver} + \cdots \tag{10}$$

which is generally composed of multiple sources in our field-of-view, additional noise from ground pickup and nearby Radio Frequency Interference (RFI), the contribution of our receiver itself, and a host of additional terms.

2.5 Normal and χ^2 Distributions

The electric fields which make up a thermal distribution are drawn from a vast number of quantum mechanically randomized events. The **central limit theorem** tells us that the addition of a large number of independent random variables – regardless of the underlying distribution they are drawn from – will tend toward a **normal distribution**, also known as a Gaussian. The probability of measuring a given electric field is then

$$Pr(E) = \ A\, e^{-(E-\langle E\rangle)^2/\sigma^2} \tag{11}$$

The mean field $\langle E\rangle$ will typically be zero: static electric fields almost never persist for long in free space, and any such field would appear as zero-frequency power, which will be filtered early in the signal chain of any radio receiver.

As described above, the signal we're interested in often isn't actually the electric field, but rather the incident power, or in radio astronomy language, the temperature of the system.

$$T \propto P \propto \langle E^2 \rangle \qquad (12)$$

Conveniently, this can be expressed in terms of the variance of the E-field distribution,

$$\begin{aligned} Var(E) &\equiv \langle (E - \langle E \rangle)^2 \rangle \\ &= \langle E^2 \rangle - \langle E \rangle^2 \\ &= \langle E^2 \rangle \\ &\propto T \end{aligned} \qquad (13)$$

By the time we measure the electric fields, they may have been filtered and amplified in several stages, such that the constant of proportionality is not known apriori, and may vary across the band and even over time. Measuring that constant is a primary goal when calibrating a receiver.

Understanding errors and uncertainties is key across all of science, and radio astronomy is no exception. Often, science can be more a question of how well you know something than what exactly that thing is! Thankfully, uncertainties in radiometric data can be characterized from first principles.

The χ^2 **distribution** describes random variables formed from summing the squares of values drawn from a normal distribution with unit variance, and is described by the number k of values summed (k here describes the **degrees of freedom**, or DoF for short). When we measure a temperature, we are measuring the variance of the electric field using k samples, and this estimate will follow a χ_k^2 distribution, scaled by T/k. The uncertainty on such a measurement is

$$\begin{aligned} \sigma_T &= \sqrt{Var\left(\frac{T}{k} \cdot \chi_k^2\right)} \\ &= \sqrt{\left(\frac{T}{k}\right)^2 \cdot 2k} \\ &= T\sqrt{2/k} \end{aligned} \qquad (14)$$

With a timestream containing N samples, we can estimate the fractional uncertainty in a Temperature measurement,

$$\frac{\sigma_T}{T} = \sqrt{2/N} \qquad (15)$$

If these samples arrive with cadence τ, Nyquist tells us they will cover a bandwidth $\Delta\nu = 1/2\tau$, while taking a total time $t = N\tau$. Substituting these into eq 15, we finally arrive at the canonical **radiometer equation**:

$$\frac{\sigma_T}{T} = 1/\sqrt{t\Delta\upsilon} \qquad (16)$$

This equation holds for any normally distributed data, independent of any linear amplification, filtering, or other instrumental effects. As such, it makes an excellent check for any non-Gaussian events, or non-linear processing in the signal chain.

2.6 Logarithmic data: Decibels

Radio instrumentation grew out of telecommunications and electrical engineering, and as noted above, uses some particularly oddball terminology and units. One feature endemic throughout radio is the use of decibels, marked as dB, in place of SI prefixes. Decibels are unitless fractional ratios, which describe the relative strength of two components. For example, the gain of an amplifier will be described in terms of the ratio of input and output powers.

$$Gain\ [dB] = 10 \cdot log_{10}\left(\frac{P_{out}}{P_{in}}\right) \qquad (17)$$

An amplifier with 20dB gain will produce an output power 100x greater than the input power. Each 10dB corresponds to a 10-fold increase, so 20dB means a 100x gain.

> *A word of caution*: In the case of amplifiers, they are always quoted with regard to power gain. In the above example, although P is amplified 100x, $|E|$ is only amplified 10x, since $P \propto \langle |E|^2 \rangle$.

Absolute units are sometimes quoted in dB, such as dBm (dB power relative to 1mW), or dBs (dB duration relative to 1 sec). In all cases, the important thing to remember is that they are a logarithmic ratio, such that 30dB is a factor of 1,000x, 60dB is 1,000,000x, etc.

2.7 Heterodyning

In radio signal processing, we often make use of a technique (invented by Canadian Reginald Fessenden in 1901) called **heterodyning,** which mixes a band-limited signal with a **carrier** to shift the original signal to a band where it is easier to manipulate. The word 'Heterodyne' has Greek roots and refers to different powers rather than different frequencies (*hetero*: other, *dunamis*→*dyne*: power), but this may have just been referring to the fact that the carrier and signal were of differing powers.

A heterodyning circuit (sometimes called a **superheterodyne receiver**) consists of an amplifier, an input signal, a tunable Local Oscillator (LO), a mixer, and a filter. The LO is a narrow-band oscillator, which can be taken as a pure tone, while the mixer is a device which takes two inputs, and produces an output which is proportional to the product of the inputs. Feeding the mixer with the LO and our original signal, we have an output

$$\begin{aligned} E_{out} &= E_{in} \cdot A \sin(\omega_{LO} t + \phi_{LO}) \\ &= \textstyle\sum_{\Delta v} \widetilde{\mathrm{E}}_{in}(\omega) \sin(\omega t + \phi_{in}) \cdot A \sin(\omega_{LO} t + \phi_{LO}) \end{aligned} \qquad (18)$$

where each spectral component of the original signal is multiplied by the LO signal. Invoking the trigonometric identity

$$\sin(\theta_1) \cdot \sin(\theta_2) = \tfrac{1}{2}\cos(\theta_1 - \theta_2) - \tfrac{1}{2}\cos(\theta_1 + \theta_2) \qquad (19)$$

we can see that each spectral component in the input signal becomes two components in the output, one copy shifted spectrally up by ω_{LO}, and one copy shifted down by ω_{LO}. If we then add some amplification and a low-pass filter which removes all signals above ω_{LO}, we are left with a single copy of the input signal, which has been shifted to a lower spectral window. This is referred to as the **Intermediate Frequency (IF)** band.

This technique is frequently applied in radio astronomy, to allow us to focus on a narrow spectral window. For example, if we wanted to coherently sample 100MHz of bandwidth starting at 1.4GHz (near the 21cm line of HI), we could naively use a 3GHz digitizer, then Fourier transform and discard the unwanted 0-1.4GHz portion; alternately, we could mix the signal with a 1.4GHz LO, and sample at 200MHz, recording just the 100MHz band from 1.4-1.5GHz. This latter option is much more efficient, and the standard across coherent measurements.

3. Familiarizing yourself with the AirSpy receiver

In this lab we will be using the commercially available AirSpy Device, a tunable 24-1800 MHz heterodyne receiver, which can record up to 10^7 samples per second (10MSps). A photo of the inside of the device is shown in **Figure 4**, while the basic circuit diagram is shown in **Figure 5**. As well as having a tunable Local Oscillator (LO), a mixer, and an Intermediate Frequency filter, the Raphael tuner chip has three gain adjustments. There is the Low Noise Amplifer (LNA) which increases the strength of the input signal, the mixer amplifier to boost the signal after the two signals are mixed, and the Intermediate Frequency (IF) amplifier, which boosts the resulting signal after it has been shifted and filtered for a desired frequency band.

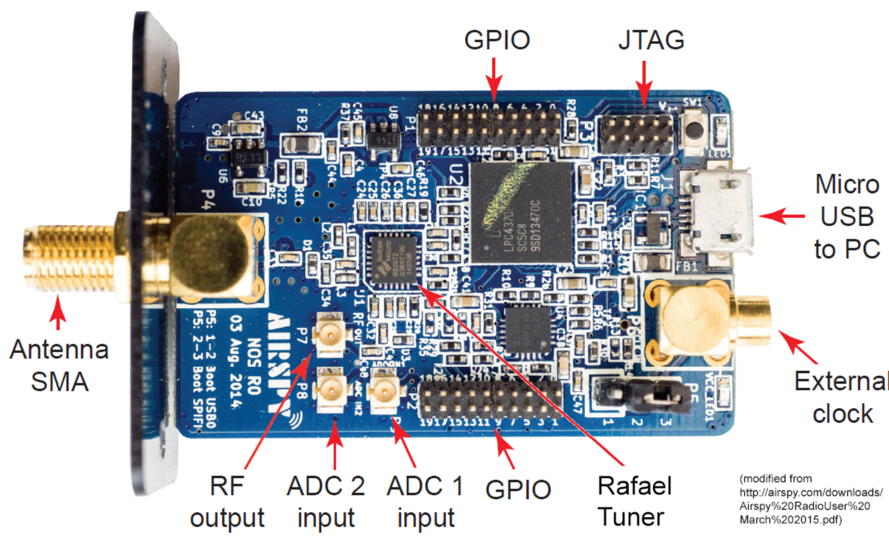

Figure 4 *– A photo showing the inside of the* ***AirSpy****, the small radio receiver which will be used throughout this lab (as well as labs #4 and #6 for those of you taking AST326). Photo from airspy.com.*

In this lab, we will be connecting to the AirSpy units via USB, and feeding an input signal into their SMA connector. You will be configuring the three gain levels, the LO, and the ADC sampling rate digitally, so it's worth understanding which is where in the signal chain.

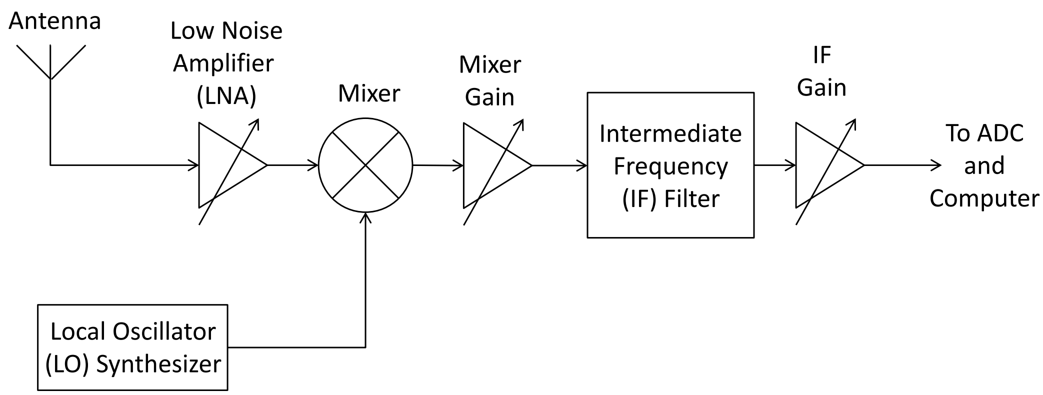

***Figure 5** – A schematic diagram of the **AirSpy** receiver, which contains a Low Noise Amplifier (LNA), Local Oscillator (LO), heterodyning mixer, mixer amplifier, Intermediate Frequency (IF) filter, and IF amplifier. Diagram courtesy Dunlap Institute Summer Instrumentation School.*

3.1 Control Software

Throughout this lab, we will need access to the raw ADC samples, so the data recording will **not** be using any sort of Graphical User Interface. Instead, we'll be relying on a simple command-line program called `airspy_rx` which allows us to control various parameters of operation and record data to disk.

This recording program supports a variety of command-line options (sometimes called "flags") to allow control of the AirSpy, for example to set gains, the LO frequency, or the sample rate:

```
-r <filename>: Receive data into file
-w Receive data into file with WAV header and automatic name
 This is for SDR# compatibility and may not work with other software
[-s serial_number_64bits]: Open device with specified 64bits serial
number
[-p packing]: Set packing for samples,
 1=enabled(12bits packed), 0=disabled(default 16bits not packed)
[-f frequency_MHz]: Set frequency in MHz between [24, 1900] (default
900MHz)
[-a sample_rate]: Set sample rate
[-t sample_type]: Set sample type,
 0=FLOAT32_IQ, 1=FLOAT32_REAL, 2=INT16_IQ(default), 3=INT16_REAL,
4=U16_REAL
[-b biast]: Set Bias Tee, 1=enabled, 0=disabled(default)
[-v vga_gain]: Set VGA/IF gain, 0-15 (default 5)
[-m mixer_gain]: Set Mixer gain, 0-15 (default 5)
[-l lna_gain]: Set LNA gain, 0-14 (default 1)
[-g linearity_gain]: Set linearity simplified gain, 0-21
[-h sensivity_gain]: Set sensitivity simplified gain, 0-21
[-n num_samples]: Number of samples to transfer (default is unlimited)
[-d]: Verbose mode
```

As an example, if you wanted to record 1,000,000 samples of RAW data at 5MSps, with the LO set to 1GHz, and all gains maxed out, you should run:

```
airspy_rx –f 1000 –a 1 –t 4 –v 15 –m 15 -l 14 –n 1000000 –d –r my_data.dat
```

This should take a moment, printing out various messages telling you how it's going. (You may not be able to copy/paste the line above, some systems do odd things with the hyphens. Please type it in on the acquisition computer for yourself!)

> *Important Note*: Radio data arrives & adds up fast! You won't be able to record it directly to your home directory, which would require sending it all across the network. Instead, please record all your data to **/tmp**, then copy or move it to your home directory after you've got a good set. Also make sure to record to **different filenames** each time, so that you can compare the different recordings later!

4. Key Lab Activities

This section outlines the key measurements you should be making throughout the lab, and which should be discussed in your lab report. Remember that the goal is to study the statistical properties of thermal noise, and to understand the function and properties of a radio receiver.

4.1 Record and inspect a test data set

Time to get your hands dirty! Well, not very dirty, but time to start running things.

Record a first test run of data, as described in the previous section. This will produce a binary file consisting of 16-bit unsigned integers representing the ADC samples. The AirSpy's ADC is actually only a 12-bit unit, with values representing its full range of allowable voltages. The range of recorded values therefore runs from $0 – 2^{12}$-1, i.e. 0 – 4095, but is meant to represent a signed voltage, centred at zero. The voltage values are stored using "offset encoding", that is, as positive numbers, but offset by half their range, in this case, 2^{11}. To convert into useful digital data in python, use `numpy.fromfile` and remove this offset:

```
In [1]: import numpy
In [2]: data = numpy.fromfile('my_data.dat',dtype='int16')-2.**11
```

Take a look at the data, and inspect some of its properties. Calculate its mean, median, standard deviation, and variance. ***Notice that the ADC might not be perfectly symmetric around zero: you'll have to make sure that you remove that mean before calculating any powers!***

Try plotting a small section of the data you just took. Is it organized, distributed peak-to-peak across allowable values, or constrained? It should look something like shown in **Figure 6**, below.

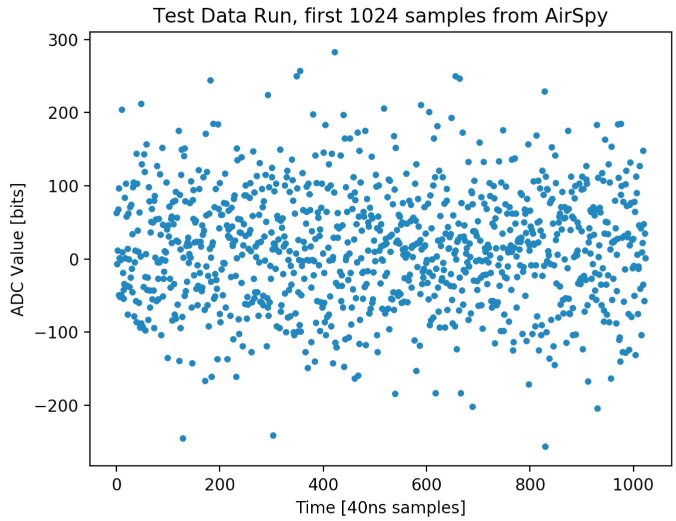

***Figure 6** – A sample timestream from the AirSpy, plotted so we can see the major features.*

4.2 Examine Normal and χ^2 Distributions

It's hard to see what exactly is going on by looking at the raw timestream. Go dig out your histogram function from Lab #1 and try histogramming the measured E-field values. Does the distribution look familiar? Is it single-peaked, double, or multiple? Can you conclude anything about the signals being measured? If you have a hard time seeing all the features of your distribution, try plotting it on logarithmic Y axis. ***If your distribution looks wildly non-Gaussian, you might have an RFI problem! Try re-tuning 10MHz higher and give it another go.***

Calculate the power measured from individual time samples. Generate a histogram from these power estimates, and plot on logarithmic axes. Since you are inspecting individual samples drawn from a normal distribution and squared, this should be comparable to a χ^2 with DoF=1, i.e. χ_1^2. You can generate a χ_1^2 probability distribution (or χ_1^2 distributed numbers) using python's `scipy.stats.chi2` function.

Try summing N adjacent samples together and generating new histograms, for each of N=2, 4, 10, 100. ***This will be MUCH easier if you write a python function to do it for you!*** Compare to χ_N^2 distributions. What happens to the distribution as N becomes large?

Let's generate realistic measurements of the temperature, averaging across 1000 samples. Plot these temperature estimates as a timestream. What do they look like? ***Keep an eye out for sudden jumps in the power, those are RFI and not thermal noise! For this lab, you'll want to cut those from your data set.*** What is the mean, and the standard deviation? How much time is included in each sample? How much bandwidth do they cover? According to the radiometer equation, what should the ratio between the mean temperature and the uncertainty be? How does this compare to the measured value?

4.3 Generate a spectrum

Next, you should explore the spectral properties of the data you just collected. As above, you'll want to write a few functions to help you out with this, it's probably something you'll be doing often going forward!

To begin, divide your data into smaller chunks, with lengths that are a power of two. I suggest starting with *n* = 1024 as a good length, and you can do this with loops, or using the `numpy.reshape` function. For each of these shorter chunks, you'll want to run a Fourier Transform to convert from a timestream into a sum of sinusoids. Fourier Transforms are generally quite expensive to run, but for data whose length contains no large primes, a **Fast Fourier Transform (FFT)** makes the task trivial for a modern computer. I suggest using `numpy.fft.fft`, but you're welcome to use whatever library you like.

The FFT will return an array of *n* complex values, representing the sine and cosine components. These will correspond to frequencies from $0\ Hz \rightarrow 2/\tau$, with a spacing of $1/N\tau$. You may notice that only produces *n/2* spectral bands! Because of Nyquist's theorem, the output only contains a bandwidth equal to half the sampling rate; the second half of the returned values are redundant information, complex conjugate to the first half, and can be discarded. As one last complication, you may remember that this is a heterodyne receiver, and that you mixed these signals down from some higher frequency band. You'll have to add the LO frequency back in to properly calibrate the frequency axis.

You've almost got a spectrum now. You can use Eq. (5) to calculate the power for each frequency band, and try plotting up the result. You've made your first spectrum! It probably doesn't look like much, so try plotting it in dB, `10*numpy.log10(spectrum)` — awfully noisy, isn't it? Estimate the uncertainty on each point using the radiometer equation, and add error bars to your plot. You can use `matplotlib.pyplot.errorbar` to do this.

To get a nice measurement of the spectrum, you'll want to average many of these individual spectra together. Using the radiometer equation, estimate how many you need to average over to achieve a 1% uncertainty on the temperature in each spectral bin. Good thing you started with millions of E-field samples, and wrote a function to generate your spectrum! Make a spectrum which achieves 1% fractional error, and plot.

Once you've got everything working, your newly-measured spectrum should look something like **Figure 7**. Make sure your axes make sense. Identify the major features of the spectrum: the high- and low-pass filters which define the band, and significant RFI, and any other odd features. How strong is the bandpass in the AirSpy? (How much fainter is the edge of the band than the middle?) Are there other features you notice?

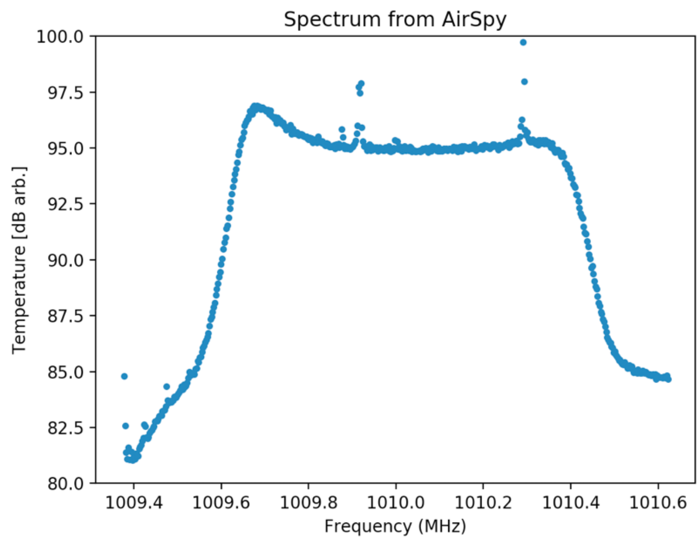

Figure 7 *– A typical spectrum generated from an AirSpy dump. We can see the shape of the bandpass filter, and a few regions of RFI which may corrupt estimates of noise properties and Gaussianity.*

Re-tune the AirSpy to a few different portions of the radio spectrum, and look for spectral features. Try FM radio (87.5-108 MHz), or the LTE cell phone band (720 MHz). Plot spectra and identify features.

> *Since your AirSpy isn't actually connected to an antenna, and in fact has a terminator across its input, we would hope that no outside signals were being picked up. Toronto is unfortunately a very radio-loud environment, and even a tiny leakage of -90dB – one part per billion – is enough to be clearly visible in the measured data.*

4.4 Vary the physical temperature of the load

Now that you've got a basic understanding of the AirSpy and of radio signals, let's vary the load a bit and see what happens to the data.

> *A word of caution*: You'll be using hot- or cold-water immersion to change the temperature of the terminator, but these connectors are not liquid tight: please make sure you keep the load dry by covering it in a plastic glove or balloon. If you notice water leaking through, please swap to a new protection layer!

You'll only be able to vary the load temperature from roughly 0C to 90C, which is a small fractional shift, and it is easy for other effects to overwhelm this change in the signal. If you run into trouble, try again, check that your spectrum is relatively RFI-clear, and re-tune if necessary. Check your timestreams and check that the temperature you're measuring is relatively stable throughout your measurement. The ADC may warm up as it records samples, and you may see an initial drift in the temperature: check these things and take more data until you get what looks like a clean recording!

Start by taking a room temperature sample, and recording the reading of the thermometer. As before, calculate the mean and variance of the data, inspect the timestream and power, plot a histogram of the raw samples.

Next, fill a container with boiling water, place the thermometer in it, and immerse the rubber-jacketed load. Give things a minute or so to reach equilibrium, give the liquid a stir, record the liquid bath temperature, and take another recording of radiometric data. Calculate the mean and variance, and plot the histogram of data over of the original histogram. Has anything changed? How do you explain that? Again estimate your uncertainty on the variance, from the radiometer equation.

As the liquid cools, record 2-3 additional points, ideally evenly spaced between the hottest temperature you were able to sample, and room temperature. Replace your hot liquid with ice water, and cool the load. Repeat the above measurements. Again, calculate variances, examine histograms and timestreams. Do you observe any trends with temperature of the load?

Because thermal power in the Rayleigh-Jeans portion of a blackbody spectrum scales with the temperature in Kelvin, the ≈ 0 – 90C range we can probe with liquid water doesn't give a huge lever arm on measuring the properties of the receiver. Ask the TAs for the container of dry ice, which will allow you to cool the load to below -100C, and record the temperature they give you along with radiometric data. Lastly, ask the TAs for the canister of liquid nitrogen, which will let you probe down to 77K, almost -200C.

4.5 Calculate the receiver temperature

Make a table of results, showing the load temperature and variance. As described above, $T \propto Var(E)$. This temperature includes all contributions from throughout the system, so we refer to it as the total system temperature, T_{sys}. We don't know yet how to convert this number to Kelvin, so for the time being, leave your temperature estimate in units of [bits^2]. Estimate the uncertainty in this temperature from the radiometer equation, and add to the table. Plot T_{sys}[bits^2] vs T_{load}[K] including error bars. The plot will be easier to read if you ensure the ranges on your axes both include zero.

Fitting models to data is key to science, and the simplest fit is known as **linear least-squares**. The idea here is straightforward, that given a model described by some parameters p_i, the most likely (best-fit) parameter set will be the one which minimizes the discrepancy between your data and the model. A least-squares fit will minimize the variance in the residuals,

$$S = \sum_i (y_i - M_i)^2$$

When our model is a line, $y(x) = p_0 x + p_1$, the least-squares best fit line will be the one which minimizes

$$S = \sum_i \left(y_i - (p_0 x + p_1)\right)^2$$

Minimization is a common problem in science and mathematics, the general trick is to find when the slope goes to zero, which must represent either a maximum or a minimum. Differentiate this equation with respect to p_0 and p_1 and set each equal to zero to find the minimum. Rearrange and show that the solutions are of the form

$$p_0 = \frac{n \sum_i x_i y_i - \sum_i x_i \sum_i y_i}{n \sum_i x_i^2 - (\sum_i x_i)^2}$$

$$p_1 = \frac{1}{n}\left(\sum_i y_i - p_0 \sum_i x_i\right)$$

Calculate the best-fit line for your data points, and add it to the plot you generated above. What does the slope of this line represent? What does the offset represent? Calculate the x-intercept. What does this indicate? What fraction of the signal you are measuring comes from the terminator thermal load, and how much comes from the receiver itself?

You may notice that the line doesn't appear to fit your data within uncertainties. There are a few reasons for this. First, you haven't accounted for measurement uncertainties in your fitting, to do so would require a **weighted linear least-squares**. We will return to this in a future lab, but the general idea is a straightforward extension, minimizing the weighted variance,

$$S = \sum_i \frac{\left(y_i - (p_0 x + p_1)\right)^2}{\sigma_i^2}$$

More importantly, though, the radiometer-equation-derived uncertainties are only **statistical uncertainties**. Variations in your instrumental setup, fluctuations in your receiver's gain, errors in recording data, and all other difficult-to-pin-down problems fall into the category of **systematic uncertainties**. Discuss a few systematic uncertainties which could be manifesting.

4.6 Measure spectral properties

Go back to your measurements from step (4), and calculate a spectrum for each. Plot these spectra on the same set of axes: can you see the thermal load trend?

Repeat step (5) for each spectral bin, calculating the receiver gain and temperature as a function of frequency. You'll want to automate this as much as possible: make a function to calculate the receiver temperature given an array of (load temperature, signal variance) pairs, and make sure it reproduces your results from the previous section. Now that you've got a function, it should only be a little more work to calculate thousands of fits, one for each narrow spectral band!

Plot up the resulting gains G and receiver temperatures $T_{receiver}$ as a function of frequency. These will very likely be extremely noisy measurements! Try plotting them on logarithmic axes, and see if you can measure their evolution across the band.

With these fits in hand, you're all set to convert measured variances into load temperatures! Congratulations, you've calibrated your first radio receiver!