

# Automated Noise Temperature Meter

MADELINE NARDIN<sup>1</sup>

<sup>1</sup>*Dunlap Institute for Astronomy & Astrophysics, University of Toronto*

## 1. BACKGROUND

The front end of a radio telescope at the simplest form consists of a feed connected to a low-noise amplifier (LNA) connected to a receiver (Gary 2019). In radio astronomy noise sources are classified in terms of thermal noise in units of temperature. Noise temperature is proportional to the power of received by the system by the following relation

$$P = G \times T_{system} \quad (1)$$

Such that  $G$  and  $T_{system}$  denote the gain and temperature of the system respectively.

The total noise of a system is the sum of all sources, which can be simplified in the application of this project to encompass the load temperature and the LNA temperature shown in equation 2.

$$T_{sys} = T_{LNA} + T_{LOAD} \quad (2)$$

Characterizing the noise temperature of the LNA in a front-end system is typically done with the Y-factor method. The Y-factor method applies a linear regression to a set of measurements with different noise levels to determine the experimental parameters of gain and noise temperature of a Device Under Testing (DUT) at a given frequency (Agilent Technologies 2014). In this project the Y-factor method is applied to a power spectrum from each operational state of a noise source (on/off).

This paper explores the computation of the effective noise temperature of a ZX60-112LN+ LNA over its documented frequency band (400-1100 MHz) with an automated Python script. The following sections of this paper will first describe the experimental setup and data acquisition process (Section 2), then explain the methods used to develop the automated Python script (Section 3), and lastly results will be presented and implications will be discussed (Section 4).

## 2. EXPERIMENTAL SETUP AND DATA ACQUISITION

An AirSpy<sup>1</sup> Software Defined Radio (SDR) was used to record power spectra to determine the noise temperature of the LNA with the Y-factor test. The AirSpy SDR records a signal which is then transformed into a radio band and converted into raw radiometric voltage samples by an analog to digital converter (ADC). The **Kotekan**<sup>2</sup> software package is then utilized to transform the voltage samples into a power spectra.

The experimental setup consists of a switch controlled by a Trinket MO microcontroller, a 346A noise source, a ZX60-112LN+ LNA, and an AirSpy SDR. The experimental setup is shown in Figure 1. The AirSpy is connected via USB to a linux host which can be queried, and similarly the microcontroller is connected to the host and is controlled remotely. We first use **Kotekan** to remotely set the 20MHz frequency band at which the AirSpy samples. We then remotely set the three internal gain parameters: **gain\_lna**, **gain\_mix**, and **gain\_if**, to prevent the power source from overloading or under powering the device. When the AirSpy is configured, **Kotekan** returns the **railfrac** field (fraction of samples overloaded) and **rms** (root mean square of samples). These performance parameters are set to maintain the following conditions: **railfrac** = 0 and **rms** > 300. **Kotekan** can then be queried remotely to measure the power spectra with the desired gain and frequency settings.

## 3. METHODS

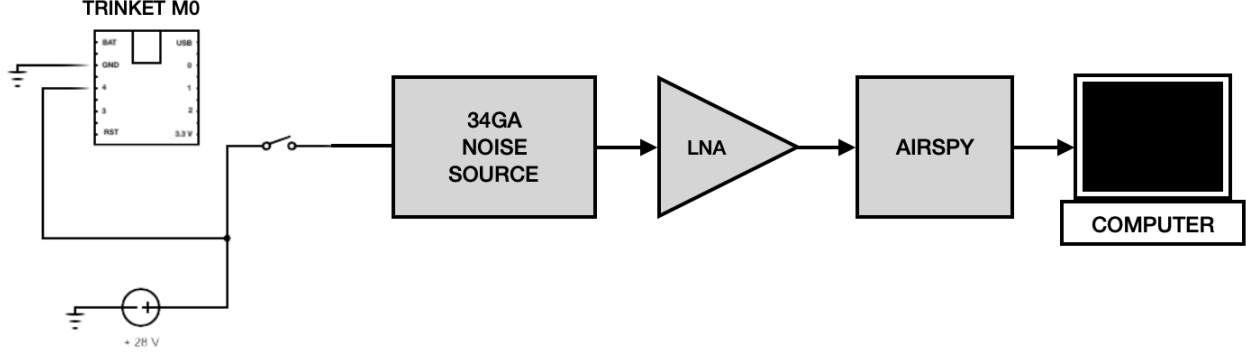
The following subsections describe the main steps of the noise temperature meter script utilized to determine the noise temperature of a LNA over its designated frequency band. The automated noise temperature meter script is publicly available<sup>3</sup> and can be used to test any LNA or more generally a Device Under Testing (DUT).

### 3.1. Averaging Power Spectra

<sup>1</sup> <https://airspy.com>

<sup>2</sup> <https://kotecan.io>

<sup>3</sup> <https://github.com/maddynardin>



**Figure 1.** Experimental set up consisting of a switch controlled by a Trinket MO microcontroller, a 346A noise source, a ZX60-112LN+ LNA, and an AirSpy SDR.

The AirSpy SDR samples a voltage signal into 1024 frequency bins over a 20 MHz frequency range centered at a designated frequency. To determine the average value of this spectrum, disturbances caused by Radio Frequency Interference (RFI) and DC offsets must be removed from the spectrum. This is done by first trimming the spectra to remove the roll-off that occurs at the high and low ends of the spectrum, then creating a mask which enables frequencies within the band that have a power output within the range of two factors of the standard deviation of the median power. The average value of the masked spectrum is then taken to be the average power of a given power spectrum.

### 3.2. Y-Factor Method with Linear Regression

With the average value of the power spectra for the on and off state of the noise source known, we can apply the Y-factor method. We determine the Excess Noise Ratio (ENR) of the noise source at any frequency in decibel units by applying linear interpolation to the ENR settings intrinsic to the 346A noise source listed in Table 1. The `interp()` function from the `numpy` Python library is used to linear interpolate. The ENR parameter is then converted from decibels to a unitless decimal value by equation 3.

$$ENR = 10^{ENR_{dB}/10} \quad (3)$$

The load temperature is determined by the equation below, based on the operational state of the noise source.

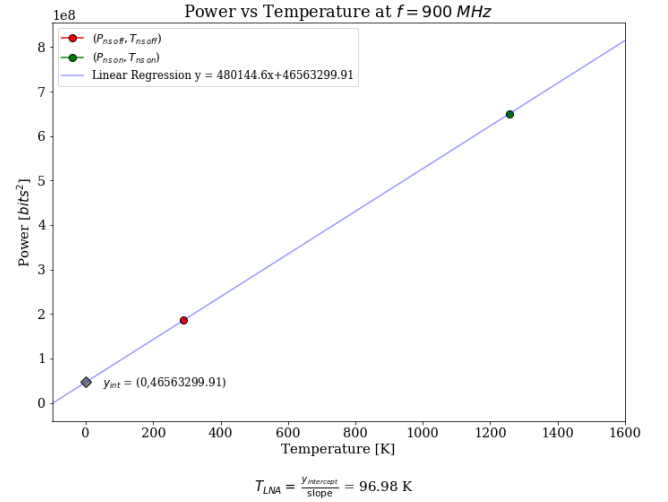
$$T_{LOAD} = \begin{cases} 290 \text{ K} & \text{noise source off} \\ 290K(ENR + 1) & \text{noise source on} \end{cases}$$

From equation 1, we define the front end radio receiver

system with the following system of equations.

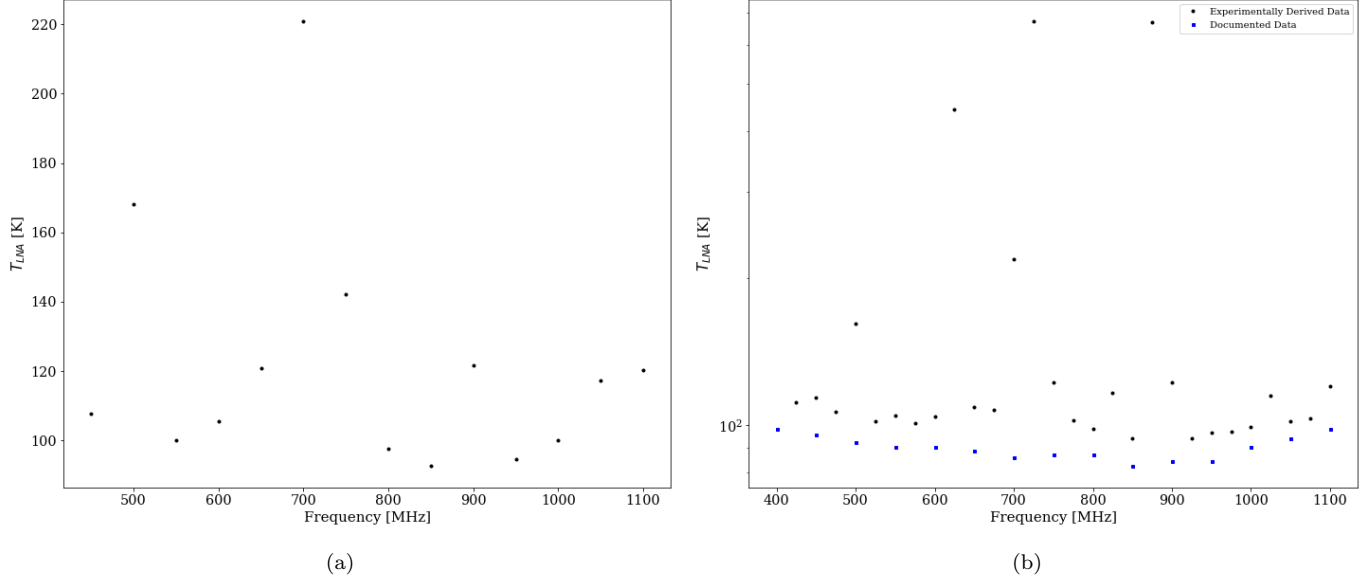
$$P_{(ns \text{ off})} = G \times (T_{LNA} + T_{LOAD \text{ ns off}}) \quad (4)$$

$$P_{(ns \text{ on})} = G \times (T_{LNA} + T_{LOAD \text{ ns on}}) \quad (5)$$



**Figure 2.** Linear regression applied to average power vs. load temperature data at a 900 MHz frequency. Parameters associated with linear regression are later used to determine the noise temperature of the LNA.

The average power vs temperature is plotted for each operational state of the noise source and a linear regression of the form  $y = mx + b$  is applied to solve for the unknown gain and LNA temperature parameters. The `polyfit()` function from the `numpy` Python library is used to interpolate the documented ENR-frequency values. The slope  $m$ , of this linear regression is the gain of



**Figure 3.** Noise temperature of the ZX60-112LN+ LNA over the the frequency band 400-1100 MHz (a). Experimentally determined and documented noise temperature of the ZX60-112LN+ LNA over the the frequency band 400-1100 MHz (b).

the system and the y-intercept  $b$  is  $\frac{T_{LNA}}{G}$ , of A graphical representation of the linear regression is shown in Figure 2, at a frequency of 900 MHz.

The noise temperature of the LNA is then given by the following equation.

$$T_{LNA} = \frac{b}{m} \quad (6)$$

Such that  $b$  and  $m$  denote the y-intercept and slope parameters of the linear regression.

### 3.3. Automating over frequency band of the LNA

The steps outlined above were then implemented into a function to automate obtaining the noise temperature at a given frequency. The function *NTM*, (an abbreviation for Noise Temperature Meter) was written to loop over a defined frequency band, and at each frequency, acquire a power spectra at each operational state of the noise meter, determine the ENR and load temperatures, apply a linear regression and obtain the noise temperature of

the LNA. The function then generates a plot of a plot of  $T_{LNA}$  vs. frequency for the entire frequency band of the LNA. Figure 3.2 displays the generated  $T_{LNA}$  vs. frequency plot for the ZX60-112LN+ LNA.

## 4. RESULTS

In comparison to the documented<sup>4</sup> noise temperature spectrum of the LNA, the automated output is within the expected range. Discrepancies between the two data sets is explained by RFI since most of the outlier points found in the derived spectrum, occur at either the cell phone and smart meter frequency range ( $\sim 900$  MHz) and water meter frequency range ( $\sim 470$  MHz). Note these RFI frequencies are specific to Toronto, Ontario and are obtained from municipal documentation. The RFI disturbance observed in this project, only emphasizes the importance of conserving radio quiet frequencies for radio astronomy, allowing for further advancements in the field.

## REFERENCES

Agilent Technologies. 2014, Noise figure Measurement Accuracy - The Y-Factor Method.  
<https://www.testworld.com/wp-content/uploads/noise-figure-measurement-accuracy-the-y-factor-method.pdf>

Gary, D. E. 2019, Front End Receiving System.

<https://web.njit.edu/~gary/728/Lecture5.html>

<sup>4</sup> <https://www.minicircuits.com/pdfs/ZX60-112LN+.pdf>

**Table 1.** Keysight 346A Noise Source Calibration Label

Frequency (GHz)	ENR (dB)
0.01	5.20
0.1	5.38
1.0	5.19
2.0	5.01
3.0	4.85