

Johnson Noise

Michael Mulhearn

May 16, 2018

1 Introduction

Johnson–Nyquist noise is the electronic noise that results from the thermal excitation of electrons in a conductor independent of any applied voltage. As the applied voltage is zero, there is no net potential difference due to Johnson noise ($\langle V \rangle = 0$), but generally the average power $\langle V^2 \rangle$ is non-zero. As we will show, the one-sided power spectral distribution turns out to be simply:

$$\mathcal{S}_+(f) = 4Rk_bT, \quad (1)$$

By measuring all of the other quantities, we will use this relation to experimentally determine Boltzmann's constant, which has the value

$$k_b = 1.38064852(79) \times 10^{-23} \text{ J/K}$$

with the uncertainty in parenthesis.

2 Autocorrelation and Power Spectrum Distribution

The signal $V(t)$ associated with random noise does not have a Fourier Transform, and so the obvious mechanism for finding the frequency spectrum associated with $V(t)$ is ruled out. Instead, we define the autocorrelation by:

$$\mathcal{R}(\tau) \equiv \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} dt V(t)V(t - \tau) = \langle V(t)V(t - \tau) \rangle \quad (2)$$

which is part of a Fourier transform pair:

$$\mathcal{S}(f) \equiv \int_{-\infty}^{\infty} d\tau \mathcal{R}(\tau) \exp(-i2\pi f\tau) \quad (3)$$

$$\mathcal{R}(\tau) = \int_{-\infty}^{\infty} df \mathcal{S}(f) \exp(i2\pi f\tau) \quad (4)$$

To interpret $\mathcal{S}(f)$ we note that Definition 2 implies that:

$$R(0) = \langle V^2(t) \rangle \equiv P_{\text{avg}}$$

but from Equation 4 we also have:

$$\begin{aligned} R(0) &= \int_{-\infty}^{\infty} df \mathcal{S}(f) \exp(i2\pi f0) \\ &= \int_{-\infty}^{\infty} df \mathcal{S}(f) \end{aligned}$$

or in other words:

$$P_{\text{avg}} \equiv \langle V^2(t) \rangle = \int_{-\infty}^{\infty} df \mathcal{S}(f)$$

That is to say $\mathcal{S}(f)$ is the average power contained at frequency f . To calculate $\mathcal{S}(f)$ we simply calculate the Fourier transform of the autocorrelation function:

$$\mathcal{R}(\tau) \equiv \langle V(t)V(t - \tau) \rangle. \quad (5)$$

There are a few practical simplifications we can make resulting from the fact that $\mathcal{R}(\tau)$ is a real even function, and therefore need only calculate:

$$\mathcal{S}(f) = 2 \int_0^{\infty} d\tau \mathcal{R}(\tau) \cos(2\pi f\tau).$$

But now clearly $\mathcal{S}(f)$ is also an even function, and we can therefore consider the one-sided-power spectral distribution:

$$\mathcal{S}_+(f) \equiv 2\mathcal{S}(f) \quad (6)$$

$$= 4 \int_0^{\infty} d\tau \mathcal{R}(\tau) \cos(2\pi f\tau) \quad (7)$$

which has the same interpretation as the power spectral distribution but simply restricted to positive frequencies:

$$P_{\text{avg}} \equiv \langle V^2(t) \rangle = \int_0^{\infty} df \mathcal{S}_+(f) \quad (8)$$

3 Theory of Johnson Noise

Johnson–Nyquist noise is the electronic noise that results from the thermal excitation of electrons in a conductor independent of any applied voltage. As the applied voltage is zero, there is no net potential difference due to Johnson noise ($\langle V \rangle = 0$), but generally the average power $\langle V^2 \rangle$ is non-zero. As we will show, the one-sided power spectral distribution turns out to be simply:

$$\mathcal{S}_+(f) = 4RkT, \quad (9)$$

where the one-sided power spectral distribution $\mathcal{S}_+(f)$ is the average power per bandwidth, i.e.

$$P_{\text{avg}} \equiv \langle V^2(t) \rangle = \int_0^{\infty} df \mathcal{S}_+(f)$$

Consider a resistor of resistance R with no potential imposed across it. Using the superposition principle, we'll consider that the total potential V across the resistor then results from sum of the voltage caused by each individual electron due to its current I_i :

$$V_i(t) = RI_i = \frac{Re}{L}u_i(t)$$

where L is length of the resistor and u_i is the velocity along the axis of the resistor. With no applied voltage and therefore no preferred direction we must have $\langle u_i \rangle = 0$. However, because each electron is in thermal equilibrium we have $m\langle u_i^2 \rangle = kT$ and so:

$$\langle V_i^2(t) \rangle = R^2 \frac{e^2}{mL^2} kT.$$

Assuming each electron is uncorrelated with the other electrons,

$$\langle V_i(t)V_j(t) \rangle = 0,$$

for $i \neq j$ and so:

$$\langle V^2(t) \rangle = \sum_{ij} \langle V_i(t)V_j(t) \rangle = \sum_i \langle V_i^2(t) \rangle = R^2 \frac{Ne^2}{mL^2} kT,$$

where N is the number of electrons.

The Fourier transform of $V(t)$ is not defined for continuous random noise. However, we can still determine the power spectral distribution from the autocorrelation function, as discussed above. In this case, the electrons are subject to collisions which alter their trajectory on a timescale τ_c , and we therefore expect the signal to become uncorrelated on that timescale:

$$\mathcal{R}(\tau) \equiv \langle V(t)V(t-\tau) \rangle = A \exp(-\tau/\tau_c)$$

for $\tau \geq 0$, which is all we will need. We note that by definition

$$\mathcal{R}(0) = \langle V^2(t) \rangle$$

and so we must have:

$$\mathcal{R}(\tau) = R^2 \frac{Ne^2}{mL^2} kT \exp(-\tau/\tau_c).$$

With the autocorrelation function in hand, the one-sided power spectrum distribution is only an integral away:

$$\begin{aligned} \mathcal{S}_+(f) &= 4R^2 \frac{Ne^2}{mL^2} kT \int_0^{-\infty} \cos(2\pi f\tau) \exp(-\tau/\tau_c) d\tau \\ &= 4R^2 \frac{Ne^2}{mL^2} kT \frac{\tau_c}{1 + (2\pi f\tau_c)^2} \\ &= 4RkT \frac{1}{1 + (2\pi f\tau_c)^2} \end{aligned}$$

Where in the last step we have used the fact the resistance is given by:

$$R = \frac{mL^2}{Ne^2\tau_c}.$$

Noting that $\tau_c \sim 10^{-14}$ s for typical conductors, then at any frequency we are likely to encounter in an electric circuit $f\tau_c \ll 1$ so that the one-sided power spectral distribution is simply

$$\mathcal{S}_+(f) = 4RkT. \quad (10)$$

4 Experimental Expectations

We saw in the previous sections that the one-sided power spectral distribution is defined by

$$P_{\text{avg}} \equiv \langle V^2(t) \rangle = \int_0^{\infty} df \mathcal{S}_+(f) \quad (11)$$

The average power is something we can easily measure. When we measure the RMS of a signal we are measuring

$$V_{\text{rms}} = \sqrt{\langle V^2(t) \rangle} = \sqrt{P_{\text{avg}}}$$

To make a quantitative prediction for the result of this measurement, we integrate the power spectral distribution across the entire bandwidth for the circuit (all electronic circuits cut off at some high-frequency).

In the case of Johnson noise we found that:

$$\mathcal{S}_+(f) = 4Rk_bT. \quad (12)$$

which is flat in frequency space. Note that at room temperature:

$$4k_bT = 1.6 \times 10^{-5} \frac{\text{mV}^2}{\text{kHzM}\Omega}$$

so even with a $1 \text{ M}\Omega$ and a 10 kHz bandwidth, the RMS voltage due to Johnson Noise will be much smaller than 1 mV . We will need to amplify this signal with very high gain in order to see the effect of Johnson Noise. Our gain will have a frequency dependence, even though our power spectral distribution does not, so we have:

$$\begin{aligned} V_{\text{rms}}^2 &= \int_0^\infty df g^2(f) \mathcal{S}_+(f) \\ &= 4k_bT R \int_0^\infty df g^2(f) \end{aligned}$$

If we approximate our circuit as having a constant gain G across a bandwidth Δf , we have:

$$V_{\text{rms}}^2 = 4k_bT R G^2 \Delta f$$

The high gain devices we will be using have a bandwidth of about 10 kHz and a fairly constant gain in the range $3000 - 5000$. The devices allow you to select between resistors with values $R = 1 \text{ M}\Omega, 500 \text{ k}\Omega, 100 \text{ k}\Omega, 60 \text{ k}\Omega, 30 \text{ k}\Omega$.

For a gain in the range $G = 3500$ to 4000 and $R = 1 \text{ M}\Omega$ this would yield a power spectral density in the range of:

$$200 - 262 \text{ mV}^2/\text{kHz}$$

For a 10 kHz bandwidth, the corresponding expected RMS voltage would be $44 - 51 \text{ mV}$.

In the first week of this experiment, you will measure the gain of the Johnson Noise circuit as a function of frequency using the attenuated output from your function generator. You will then measure the RMS voltage resulting from amplified Johnson noise and use this to determine k_b .

In the second week of this experiment, you will build a power spectrum analyzer, which will allow you to qualitatively investigate the (flat) Johnson noise spectrum, as well as extract k_b more directly from the average PSD.

5 Johnson Noise Device Gain Measurement

In this section, we will measure the gain of the Johnson Noise device as function of frequency.

Obtain a Johnson Noise (JN) measurement device (Model JN-A or JN-B) and note the serial number (e.g. #4). These are high gain devices that are very susceptible to damage. Do not drive them directly from a function generator, but use a 1000:1 attenuated, taking care that the small

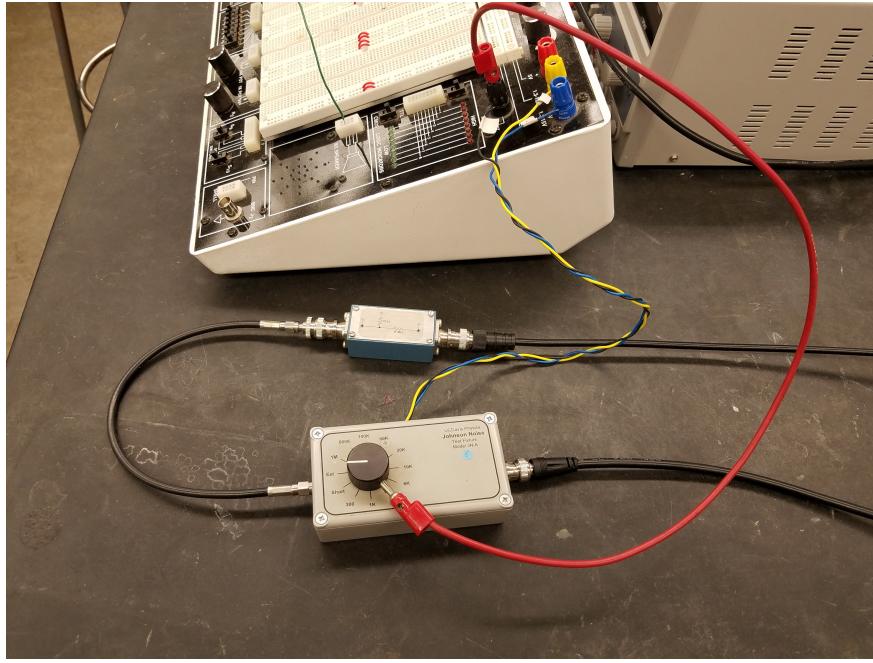


Figure 1: Connections to the Johnson Noise device. Note the banana plug used to reduce noise from the knob acting as an antenna!

resistor is toward the JN device (else you will reduce voltage by a factor 999/1000, and likely damage the equipment!)

The JN devices are very sensitive, easily pickup noise, and ring at a characteristic frequency of about 27 kHz. The tuning knob in particular picks up noise. An old version banana plug cable plugged into the knob at the set screw grounds the knob, eliminated this source of noise. The effect can be quite dramatic.

Adjust the supply levels on your bench-top supply to +12 V and -12 V. Then, with the supply turned off, connect the JN device to power and ground, as labeled on the wires. Once connected, power the device on and adjust the current thresholds to just slightly above where the supplies become current limited. Under normal operation, the JN device should draw about 20 mA from each supply... keep an eye on it. If one of the supplies is inadvertently disconnected, the circuit latches in a high current state, which should be avoided if possible.

Set your function generator to produce a sine wave with $V_{\text{rms}} = 100 \text{ mV}$ and $f = 1 \text{ kHz}$. Connect the function generator to channel 1 of your scope, then to the 1000 : 1 attenuator, and then to your JN device. The output of the JN device should then be plugged directly into the scope channel 2.

Make a series of gain measurements at different frequencies that will allow to determine the integral

$$\int_{-\infty}^{\infty} df g(f)^2.$$

While making the gain measurement, you will see that the output of the Johnson Noise seems to bounce around. The effect can be seen in Fig. 2. There is a significant amount of low frequency noise in the output. I found the best approach was to trigger on the AC line, which allows you to find a region of the plot with relatively little 60 Hz noise. Use the scopes Measure capability to find the RMS voltage of the input and output of the Johnson Noise. This is likely the major systematic uncertainty in your measurement, so spend some time thinking about how to quantify it.

You might be tempted to try and use your DMM to measure the RMS voltage of your signal,

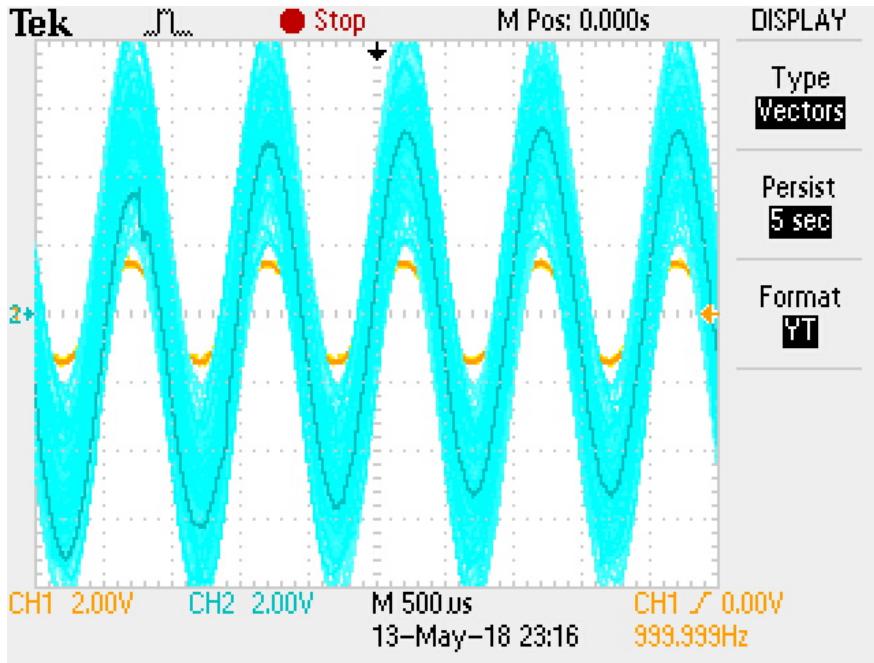


Figure 2: The output of the Johnson Noise device bounces around a lot due to low frequency noise. Here the output is shown with display persistence.

but this is a mistake. Your DMM is designed to measure 60 Hz AC.

6 Johnson Noise Measurement

You are now ready to make your first-pass Johnson Noise measurement:

- Turn down the function generator and disconnect it from the JN device.
- Turn the knob to select a resistor, and note the RMS voltage from the scope. Repeat this measurement for every resistance.

These measurements, combined with the gain measurement, should allow you to calculate k_b . Before leaving, do make certain that the data you have collected is reasonable. At this stage, within an order of magnitude of k_b is acceptable. **This is the end point for week one.**

7 Digital Dynamic Range Adjustment

In the second week of the lab, you will build a digital spectrum analyzer based on the Arduino Uno. The Arduino's analog inputs have an input voltage range from 0 to 5 V, but our signals are ground referenced (symmetric about 0 V). A DC level shifter is therefore needed to map the ground referenced output of the Johnson Noise device to the 0 to 5 V range expected by the Arduino analog inputs.

Of course we'll be running the ADC as fast as possible which comes at the price of reduced resolution to 8 bits. That would result in a voltage precision of $5 \text{ V}/2^8 \sim 20 \text{ mV}$. Considering we will be looking at $\sim 100 \text{ mV}$ signals, this is just not enough precision. Fortunately, we have one last trick up our sleeve! The ADC has an (optional) external voltage reference, which sets the scale used

by the ADC. Experimentation showed that the ADC works reliably down to a voltage reference of ~ 780 mV, which, at 8-bit precision, would give us a 3 mv voltage precision, plenty good enough!

The low-pass filter in Fig. 3 is used to convert PWM output (on pin 5 of the Arduino) into a DC reference voltage V_{ref} . This reference voltage is used as (1) the reference for the ADC, at pin AREF, setting the dynamic range of the ADC, and (2) the reference to the DC level shifter circuit in Fig. 4 so that input to the ADC is referenced to $V_{\text{ref}}/2$. In this way, we can adjust V_{ref} to match the dynamic range of our signal, and the DC offset will be set to the middle of this range. This will ensure that even for small signals, the full 8-bit precision of the ADC is being put to use.

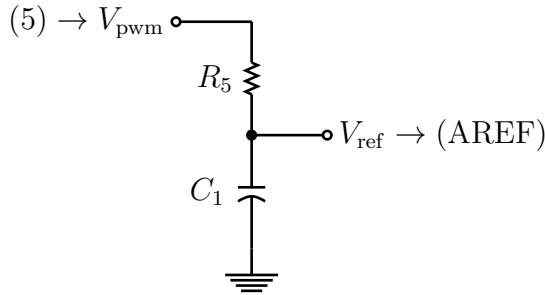


Figure 3: A low pass filter used to convert the Arduino PWM output (on pin 5) to an analog DC reference voltage (sent to pin AREF and also used by the DC level shifter.)

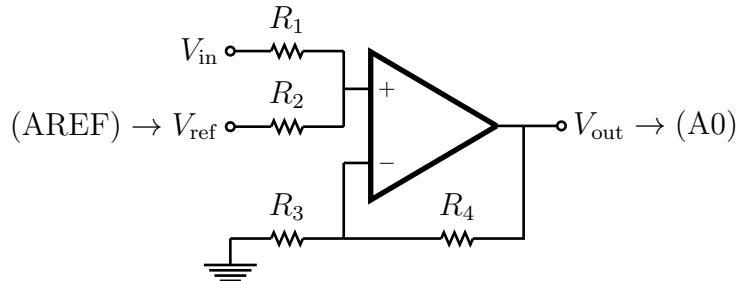


Figure 4: A DC level-shifter with unit gain. The points (AREF) and (A0) refer to the Arduino pins, which should be connected as shown.

For our op-amp we will use a LM358 low-power dual op-amp IC with pinout in Fig. 5. The LM358 is designed to run off a single supply down to 5 V, so we can power this circuit entirely from the Arduino. This is not just convenient, it also ensures that its output cannot exceed the 0 to 5 V range expected by the Arduino.

You will build the circuits in Fig. 3 and 4 using $R_1 = R_4 = 33$ k Ω , $R_2 = R_3 = 15$ k Ω , $R_5 = 1$ k Ω . For C_1 use a large polarized capacitor ($C_1 = 330$ μ F) and **make certain that the negative terminal is connected (as shown in the circuit) to ground**. Connect the positive supply of the Op Amp to the 5 V supply of the Arduino and the 0 V supply to the Arduino ground.

A light-weight sketch CircuitTester is provided which is useful while developing your circuit. This sketch implements an RMS voltmeter (significantly better, if less durable, than the one provided by your DMM.) based on the circuit you are building. It provides control of the PWM on pin 5, reads the analog input on A0, and regularly reports the RMS voltage (and other useful quantities) on the serial output. The level of the PWM is specified in the parameter RUN_SCALE. A value of 0 results in 0% duty-cycle for the PWM, a value of 255 corresponds to 100% duty-cycle.

**D, P, and NAB Package
8-Pin SOIC, PDIP, and CDIP
Top View**

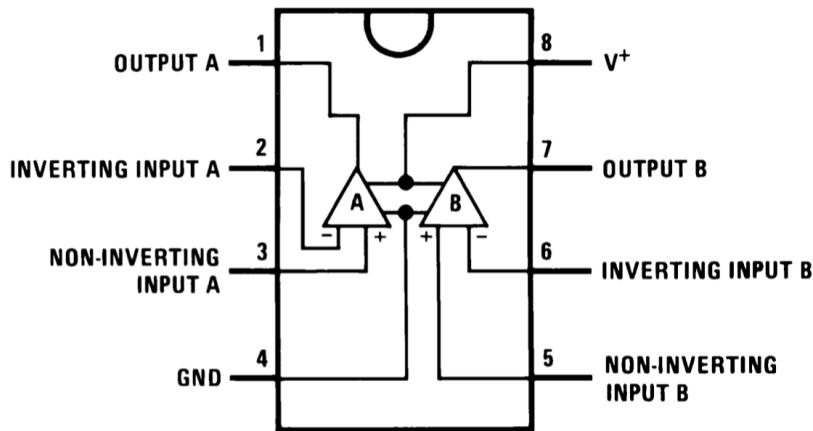


Figure 5: Pinout for the LM358 dual-op-amp with single-ended supply. We'll only need one of the op-amps.

While testing your circuit, you should use your function generator to drive the input V_{in} directly. A 5 kHz sine function with an RMS amplitude of 100 mV is a good starting point. Make sure the ground of the function generator is connected to the ground of your Arduino.

By now we've learned to divide and conquer technical challenges! I'd propose something like:

- Leave the Arduino unpowered.
- Build the level shifter as shown but at first (1) set the reference voltage to 5 V and (2) connect the output to channel one on your scope instead of pin A0.
- Power up the Arduino.
- On your scope you should see the function generator referenced to approximately 2.5 V.
- Power down the Arduino.
- Build the lowpass filter as shown, but connect V_{ref} to channel two on our scope instead of to pin AREF.
- Power up the Arduino and download the CircuitTester sketch, setting the parameter RUN_SCALE to 112.
- On your scope, you should observe V_{ref} , it should be at about 2.5 V with very little ripple (use AC line trigger and DC offset).
- Adjust the RUN_SCALE parameter and ensure you have the expected behavior.
- Power down the Arduino.
- Connect the lowpass filter output V_{ref} to both (AREF) and the DC level-shifter input V_{ref} .

- Power up the Arduino.
- Check that level-shifter output has the correct DC level when you adjust `RUN_SCALE`.
- Connect the output of the DC level-shifter to the Arduino analog input A0.
- Start the Serial Monitor.

Once the whole circuit is built, the Arduino is running the `CircuitTester` sketch, and the Serial Monitor is open, you should observe periodic RMS voltage readings that should correspond approximately with the output of the function generator.

8 Calibration of Arduino Circuit

9 Improved Johnson Noise Device Gain Measurement

10 Johnson Noise Measurement