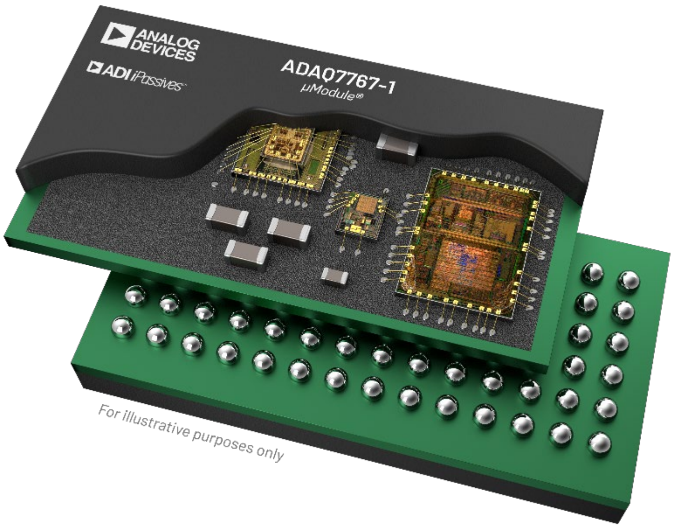
ADAQ7767-1 LTspice Simulation Guide

Example Simulation for ADAQ7767-1 Flexible Resistive Input, Anti-Alias, 24-Bit, 1 MSPS, μModule® DAQ Solution

This guide offers a walkthrough on running simulations with the ADAQ7767-1, a 24-bit, 1MSPS precision μModule® DAQ Solution, using the accompanying LTspice schematic (.asc file). The AC, transient, and noise simulation examples are also shown in this guide. A basic level of familiarity with LTspice is assumed; those new to the tool might want to go over a primer such as [1]. Make sure to check the attached readme.txt file before using this guide. For support, please log in (or create an account) and post the questions in the LTspice forum in ADI Engineer Zone [2].



***ADAQ7767-1** Flexible Resistive Input, Anti-Alias, 24-Bit, 1 MSPS, μModule® DAQ Solution*

ADAQ7767-1 Quick Overview

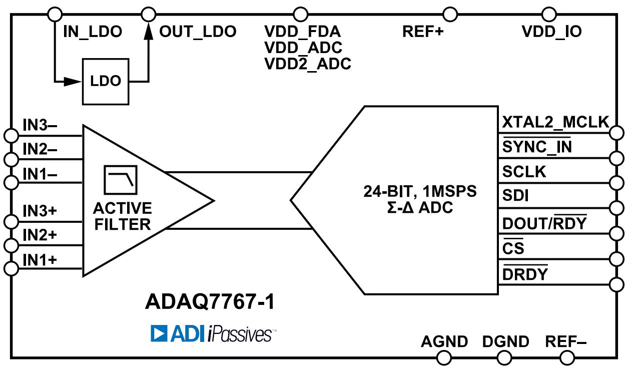


Figure 1. Simplified ADAQ7767-1 block diagram

A simplified block diagram of the ADAQ7767-1 can be seen in Figure 1.

The ADAQ7767-1 supports a wide variety of input types, including unipolar and bipolar single-ended, pseudo-differential, and differential signals, with a maximum differential range of ±28.672 V and a common-mode voltage range of −16 V to +12 V using IN3. The three pairs of pin-selectable gain settings are configured as wires that can be modified depending on the required input range.

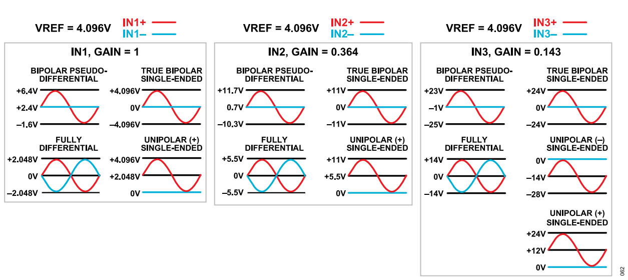


Figure 2. Example Signal Inputs

A fourth-order low-pass analog filter, combined with the user-programmable digital filter, ensures that the signal chain is protected against the high frequency noise and out-of-band tones presented at the input node from aliasing back into the band of interest. The analog low-pass filter is carefully designed to achieve high phase linearity and maximum in-band magnitude response flatness. Additionally, a high-performance analog-to-digital (ADC) driver amplifier ensures the full settling of the ADC input at the maximum sampling rate. The driver circuit is designed to have minimum additive noise, error, and distortion while maintaining stability. The fully differential architecture helps maximize the signal chain dynamic range.

At this stage in the signal chain, the output is displayed as ANALOG_OUT+ and ANALOG_OUT-. These are also the signals routed into the ADC pins. Transient and AC simulation coverage applies only up to this block.

The ADC integrated within the ADAQ7767-1 is a high-performance, 24-bit precision, single-channel, sigma-delta (Σ-Δ) converter with excellent AC performance and DC precision. This ADC also supports throughput rates up to 1 MSPS from a 16.384 MHz MCLK.

The noise model of this example circuit includes signal chain noise contributions from the inputs up to the ADC noise. Noise from the reference block is assumed negligible—given that the voltage reference to be used with this device should have excellent noise performance.

Please note that the digital outputs of the ADC is not included in this model; nor is the device's frequency response derived from digital filtering and decimation. Finally, a +5 V voltage source is used to supply the circuit.

ADAQ7767-1 Model Presentation

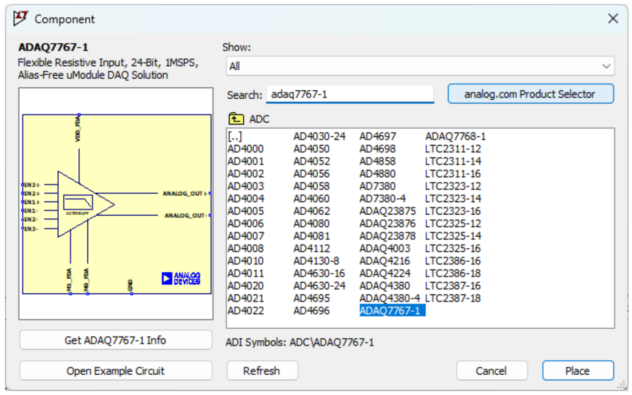


Figure 3. ADAQ7767-1 in Component Selection

Figure 3 illustrates the ADAQ7767-1 within the Component Selection tab. To view the device's default configuration, click **Open Example Circuit**. The following sections provide step-by-step guidance on navigating and operating the LTspice simulation circuit.

ADAQ7767-1 LTspice Pin Descriptions

Table 2 summarizes the pin names, types, and their specific functions in LTspice simulation for the ADAQ7767-1 model. It provides guidance on how each pin should be configured or connected during simulation, along with important limitations to ensure accurate results.

Table 1. Pin Function Descriptions

Pin Names	Type	Pin Usage in LTspice Simulation
IN1+ IN1– IN2+ IN2– IN3+ IN3–	Analog Pins	**Signal Inputs** • Apply the appropriate input signals here based on the required configuration
VDD_FDA	Power Pin	**FDA Supply** • Provide a 5 V source to power the Fully Differential Amplifier (FDA) inside the ADAQ7767-1.

<table>
<tr><th>Pin Names</th><th>Type</th><th>Pin Usage in LTspice Simulation</th></tr>
<tr><td>M0_FDA
M1_FDA</td><td>Digital Pins</td><td><u>FDA Mode Control Outputs</u>
• These pins control the FDA power mode in hardware. In this LTspice model:
✓ The device defaults to Full-Power Mode.
✓ Pins only simulate leakage current; active mode switching is not modeled.
✓ In real hardware configuration, these pins interface with the ADC for mode control.
• Limitations in this LTspice model:
⚠ Both pins are tied to 5 V by default, preventing mode changes.
⚠ Only Full-Power Mode is supported; Low-Power and Standby modes are not functional.
⚠ Connecting pins to GND does not enable Low-Power Mode in simulation.</td></tr>
<tr><td>ANALOG_OUT+
ANALOG_OUT–</td><td>Output Pins</td><td><u>Simulated Analog Outputs</u>
• Connect a net name or a voltage-controlled voltage source to observe the AFE circuit output of the ADAQ7767-1.
Note: These pins do not exist on the actual device; they are only provided for simulation purposes.</td></tr>
</table>

LTspice Simulation Guide

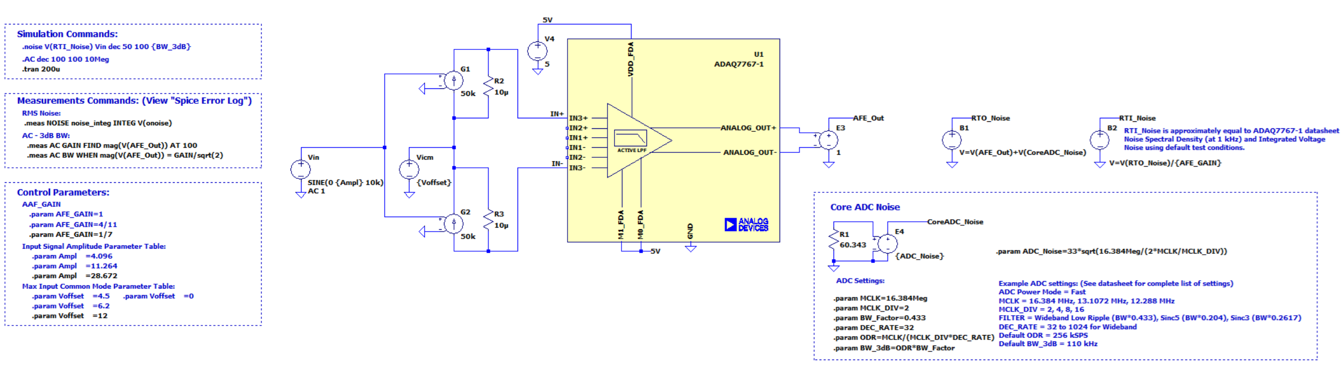


Figure 4. LTspice Schematic Screenshot

Simulation File Usage

Import the ADAQ7767-1 example circuit found on LTspice component search. The accompanying LTspice simulation file is divided into boxed sections for readability. Some of these sections are:

Simulation Commands

LTspice simulation commands for three kinds of analyses can be found in this block, namely: transient (.TRAN), AC (.AC), and noise (.NOISE).

LTspice requires only one active simulation command present in the schematic at a time. The most convenient use of this block is to right-click on the simulation command that the user wants to run and then click the OK button in the ***Configure Analysis*** dialog. This action enables the command that the user just clicked on while automatically commenting out all other simulation commands in the schematic.

Alternatively, the ***shift + left click*** shortcut that turns commands into comments and vice versa is also a good alternative to control what command is active.

Measurement Commands

The measure (.MEAS) command in LTspice is used to calculate and extract specific quantities from simulation results based on user-defined expressions. These measurements are saved to a .log file, which can be accessed by pressing the ***Ctrl + L*** hotkey or by right-clicking on a window, selecting ***View***, and then choosing ***Spice Output Log***.

Control Parameters

Most variable elements in this simulation can be controlled by modifying the parameters defined in this section. LTspice parameters are created through the .PARAM command. This allows quick modification of the simulation, keeping track of its current status and is convenient for running stepped simulations, which are displayed in the examples further ahead (if unfamiliar with the .STEP command, see [3]).

Input Configuration

The ADAQ7767-1 has 3 pin-selectable gain options, namely: 1 (IN1), 0.364 (IN2), or 0.143 (IN3). Select only one gain option and adjust the control parameters by commenting out the other alternatives. Additionally, ensure proper wiring connections to match the selected gain setting and enable correct simulation operation. Please refer to the "Analog Inputs" section of the datasheet for additional information.

Table 2. Input Range Selection Table

INx	AFE_GAIN	Input Signal	Max Input Common Mode	IN+ & IN− Wiring
IN1	1	4.096	4.5	IN1+ IN1−
IN2	4/11	11.264	6.2	IN2+ IN2−
IN3	1/7	28.672	12	IN3+ IN3−

Noise Simulation: Core ADC Noise and -3dB Signal Bandwidth

The ADAQ7767-1's integrated core ADC noise spectral density (NSD) is specified at 33 nV/√Hz. This value, together with the system bandwidth, is used to calculate the noise performance of the signal chain.

The input signal bandwidth of the ADAQ7767-1 is primarily determined by its digital filter. The variables in the example circuit must be configured to match the ADC settings required for the intended application. The device offers three digital filter options, and the appropriate filter must be selected based on the specific use case. These settings are essential for obtaining accurate noise results in the simulation.

To calculate for the -3dB bandwidth, below is an example:

1. Set variables according to your requirements.
    a. MCLK = 16.384 MHz
    b. MCLK_DIV = 2
    c. BW_Factor = 0.433 (Wideband)
2. Calculate for the f_{MOD} and ODR using the formulas.
    a. $f_{MOD} = \frac{MCLK}{MCLK_{DIV}}$
    $f_{MOD} = \frac{16.384\ MHz}{2}$
    $\boxed{f_{MOD} = 8.192\ MHz}$
    b. $ODR = \frac{f_{MOD}}{DEC_{RATE}}$
    $ODR = \frac{8.192\ MHz}{32}$
    $\boxed{ODR = 256\ ksps}$
3. These values are used to compute for the -3dB bandwidth.
    a. $f_{-3dB} = BW\_Factor \times ODR$
    $f_{-3dB} = 0.433 \times 256\ ksps$
    $\boxed{f_{-3dB} = 110.848kHz}$

In LTspice, the simulated NSD at the AFE_Out pin is combined with the core ADC NSD specified in the datasheet. To obtain the RMS noise, multiply the resulting total NSD by the square root of the calculated −3 dB bandwidth. Modify the command as needed to display the RMS noise in either input-referred (RTI) or output-referred (RTO).

Table 3. Clock, Digital Filter, and Decimation Control Options

Parameter	Options	Text to Input
MCLK	16.384 MHz 13.1072 MHz 12.288 MHz	16.384 13.1072 12.288
MCLK_DIV	MCLK/2 MCLK/4 MCLK/8 MCLK/16	2 4 8 16
Digital Filter	Wideband Low Ripple (BW*0.433), Sinc5 (BW*0.204), Sinc3 (BW*0.2617)	0.433 0.204 0.2617
DEC_RATE	<u>Wideband Low Ripple</u> ×32, ×64, ×128, ×256, ×512, ×1024	32 to 1024
	<u>Sinc5</u> ×8, ×16, ×32, ×64, ×128, ×256, ×512, ×1024	8 to 1024
	<u>Sinc3</u> Programmable decimation rate	Programmable decimation rate

Transient Simulation

Transient simulation solves the circuit in time-domain using the .TRAN command.

Transient simulation example: time domain exploration of the circuit

In this example, a transient simulation is executed to explore the signals across the signal chain in time-domain.

1. **Simulation Setup**

   All simulation commands except for .TRAN command are commented out. The simulation command box should look like this:

```
Simulation Commands:
 .noise V(RTI_Noise) Vin dec 50 100 {BW_3dB}
 .AC dec 100 100 10Meg
 .tran 200u
```

Figure 5. TRAN example: simulation commands

2. **Gain Configuration and Input Signal**

   Since the ADAQ7767-1 has three pin-selectable gain settings, only IN3 (AFE_GAIN = 1/7) is used in this example. Please refer to Table 1 for the other available options.For the input, a full-scale differential sinusoidal signal of 28.672 V_{peak} (i.e., -0 dBFS) at 10 kHz with 0 V common mode is applied using the following control parameters:

```
Control Parameters:
 AAF_GAIN
   .param AFE_GAIN=1
   .param AFE_GAIN=4/11
   .param AFE_GAIN=1/7
 Input Signal Amplitude Parameter Table:
   .param Ampl    =4.096
   .param Ampl    =11.264
   .param Ampl    =28.672
 Max Input Common Mode Parameter Table:
   .param Voffset   =4.5     .param Voffset   =0
   .param Voffset   =6.2
   .param Voffset   =12
```

Figure 6. TRAN example: input signal configuration

3. **Running the Simulation**

   Press the **Run** button to start the simulation. Once the simulation is finished, check if the input signal is properly set, as displayed in Figure 5.

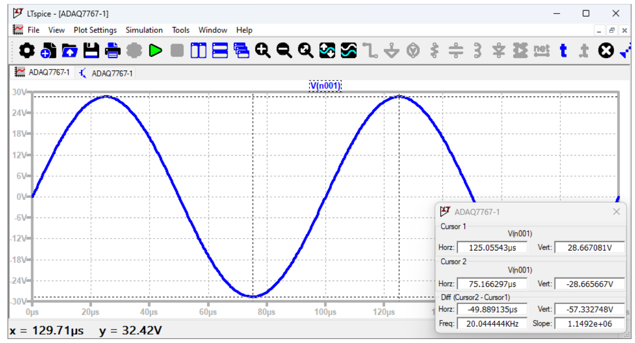


Figure 7. TRAN example: measuring input signal

Next, check the time domain signal at the output by clicking on the AFE_Out node. The following measurement can be observed, where placing cursors confirms the desired amplitude and frequency was obtained:

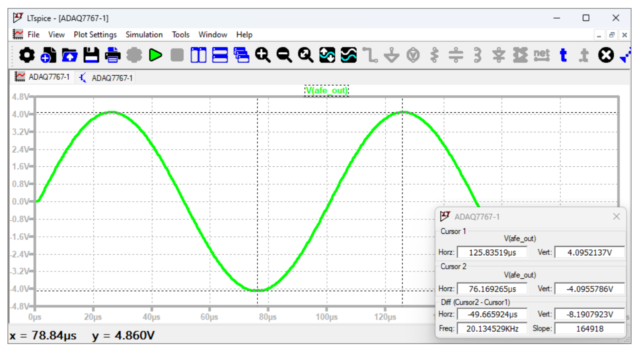


Figure 8. TRAN example: measuring analog front-end (AFE_Out) output signal

Figure 7 confirms the intended AFE_GAIN = 1/7 based on the signal amplitude. This also represents the analog front-end output signal that is fed into the ADC.

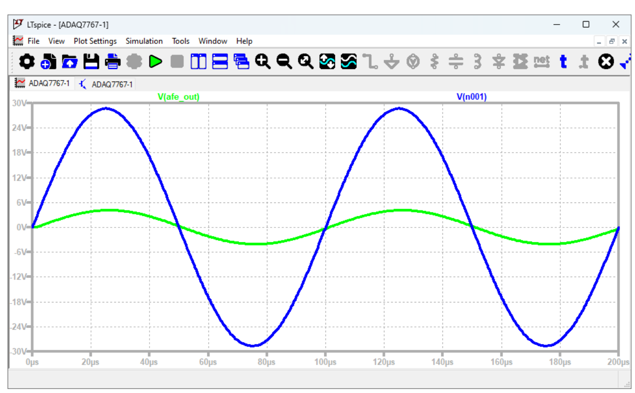


Figure 9. TRAN example: input and analog front-end (AFE_Out) output signal

AC Simulation

In an .AC simulation, LTspice performs a small-signal analysis over a frequency domain sweep, after calculating the bias operating point of the circuit. This type of simulation is useful for evaluating gain, bandwidth, and frequency response characteristics.

AC simulation example: AFE DC Gain and -3 dB bandwidth measurement

In the next example, an AC simulation is used to measure the DC gain magnitude, the -3 dB bandwidth, and the frequency response of the analog front-end circuit.

1. Simulation Setup

Begin by enabling only the AC simulation command in the schematic. The signal chain is configured exactly as in the previous transient simulation example.

Simulation Commands:

```
.noise V(RTI_Noise) Vin dec 50 100 {BW_3dB}
.AC dec 100 100 10Meg
.tran 200u
```

Figure 10. AC simulation enabled

2. Running the Simulation

Run the simulation and probe the output at the AFE_Out node.

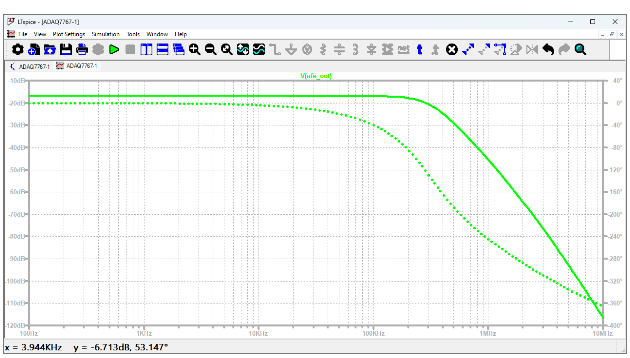


Figure 11. AC example: ADC input and output default plot

a. Manual Measurement using Cursors

Measure the DC Gain and -3 dB bandwidth of the filter by using the XY cursors in the plot window.

Click twice on the signal label (e.g., V(AFE_Out)) to activate the two cursors.

Drag Cursor 1 to the leftmost end of the plot (100 Hz). Carefully slide Cursor 2 to the frequency where the signal is attenuated by 3 dB.

The cursor window then displays an anti-aliasing filter bandwidth measurement of approximately 286 kHz (see Figure 10), closely matching the datasheet value specified at 283 kHz.

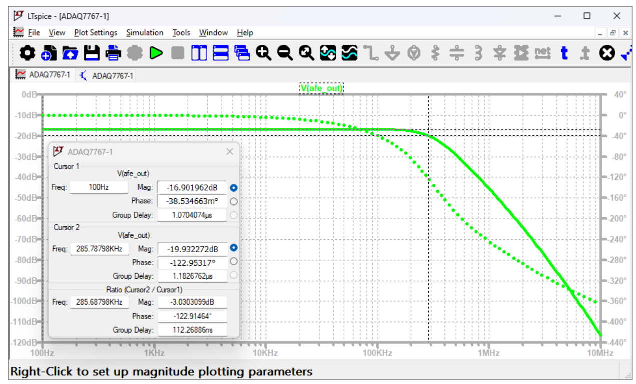


Figure 12. Measuring -3 dB bandwidth with cursors

On the other hand, Cursor 1 placed at the frequency of 100 Hz shows a DC Gain measurement of -16.901 dB. This is validated through the formula:

$$DC\ Gain = 20\log_{10}(AFE\_GAIN) = 20log_{10}(1/7)$$
$$\boldsymbol{DC\ Gain = -16.902\ dB}$$

b. Automated Measurement using .MEAS Command

Alternatively, the .MEAS command easily automates the bandwidth and DC gain measurement of the circuit.

Measurements Commands: (View "Spice Error Log")

RMS Noise:

```
.meas NOISE noise_integ INTEG V(onoise)
```

AC - 3dB BW:

```
.meas AC GAIN FIND mag(V(AFE_Out)) AT 100
.meas AC BW WHEN mag(V(AFE_Out)) = GAIN/sqrt(2)
```

Figure 13. AC example: commands for automating -3 dB bandwidth measurement

- First, the **.MEAS AC GAIN** command finds the value of the transfer function at 100 Hz.
- Afterwards, the **.MEAS AC BW** returns the frequency value where the transfer function is 3 dB below the previously calculated value.
  - ⚠ *Note: This method is not valid for transfer functions with peaking, as the maximum may not represent the flat-band gain.*

To access the measurement results, press ***Ctrl + L*** after the simulation is finished to open the SPICE Output Log and scroll down to the end of the file:

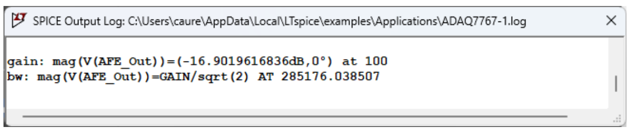


Figure 14. .MEAS statement results found in the SPICE Output Log

These are the measurements at AFE_Out node, just before the ADC inputs. Figure 12 displays the anti-aliasing filter bandwidth only. The overall signal chain bandwidth is still dominated by the ADAQ7767-1's digital filter. Please refer to the datasheet for more information.

A result of approximately 285.1 kHz is observed in the SPICE Output Log, confirming the manual measurement previously performed using cursors. This automated approach using measurement command is particularly useful for repeated and stepped simulation runs, as it improves efficiency and consistency.

For more information on the .MEAS command and its capabilities, refer to the built-in LTspice Help documentation.

Noise Simulation

The .NOISE command performs a noise simulation to extract the noise spectral density at a specified node within the circuit. This analysis focuses exclusively on random noise generated by circuit components. It does not account for other unwanted signals such as coupled interference, out-of-band signal components, crosstalk, power supply harmonics, etc.

The .NOISE analysis is a particular case of small-signal AC analysis, and it operates independently from of both transient (.TRAN) and AC (.AC) simulations. That is, even though .NOISE analysis exposes noise voltages present in the circuit, those voltages cannot be observed in the time domain through a .TRAN simulation or have any effect on an .AC simulation. When performing full signal chain noise analysis, it is a good practice to check beforehand that components in the schematic have their noise modeled in. In this case, the example circuit of the ADAQ7767-1 includes the front-end circuit noise model, and the core ADC formula-based noise model.

For a reference first-order theoretical approach to signal chain noise analysis, see [4].

Noise simulation example: total signal chain noise

In this example, the .NOISE simulation is used to demonstrate referred-to-input (RTI) noise performance of the complete signal chain.

1. **Simulation Setup**
   Start by enabling only the noise simulation command in the schematic. The signal chain is configured exactly as in the previous examples.

```
Simulation Commands:
.noise V(RTI_Noise) Vin dec 50 100 {BW_3dB}
.AC dec 100 100 10Meg
.tran 200u
```

Figure 15. .NOISE simulation enabled

2. **Running the Simulation**
   The .NOISE simulation command requires specifying the circuit node where noise will be measured. Run the simulation and probe the RTI_Noise node.

Figure 14 displays the result in the waveform viewer, which corresponds to the input-referred noise power spectral density of the signal chain. The RMS noise voltage is obtained by integrating the power spectral density over the bandwidth of interest. This integration can be performed in the waveform viewer by: (a) ***Ctrl + left click*** on the trace label, or; (b) ***Ctrl + L*** to display the result in SPICE Output Log .

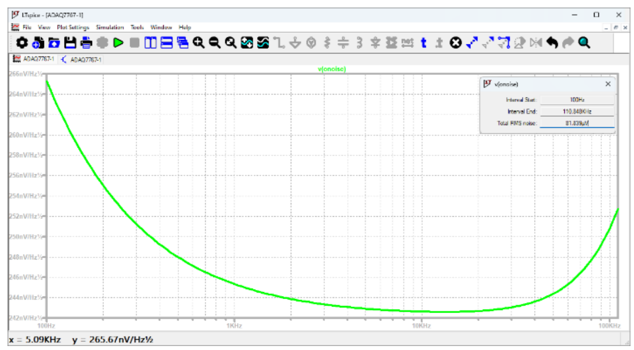


Figure 16. .NOISE example: input-referred noise spectral density

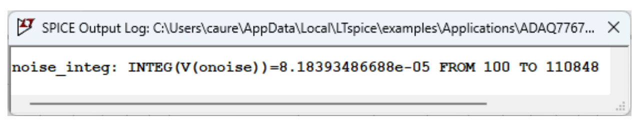


Figure 17. NOISE example: input-referred RMS noise

Results show 81.839 µVrms of input-referred noise. Note that the result is the integrated RMS noise of the **displayed** waveform. This means the result will vary if the frequency interval over which the analysis is performed is modified (in the simulation command parameters). It will also vary if x-axis is zoomed-in.

Additionally, noise can be measured at the RTO_Noise node. These values represent combined noise contributions from the AFE_Out node and the formula-based ADC noise model. Adjust the parameters (e.g., RTO_Noise) on the simulation commands based on specific analysis requirements.

Other Considerations

Component models

LTspice models for all electronic components in this signal chain are included in the LTspice built-in library, so no external files are needed. Make sure both the LTspice version and component library are up to date. For the LTspice version, go to the *Help* menu and click on "*Check for LTspice updates…*". For the component library, open the *Tools* menu, then click on "*Update components*".

Simulation models for components don't necessarily cover the full behavior of the real device. In general, it is safe to assume that the component will display the correct behavior under standard operating conditions (room temperature, nominal supply voltage…) while second order effects such as distortion, crosstalk, etc. might not be included in the models.

For ADCs, the models focus on analog behavior so most of the component behavior, pins, etc. related to digital I/O are not present in the models. The LTspice model will not be giving out the coded output over a digital interface (SPI, LVDS) as the actual device does. Instead, the ADC output is provided as a regular (analog) signal that is quantized and restricted to the maximum input span of the ADC. This makes it easy to seamlessly perform signal chain analysis, especially noise and AC, up to and including the ADC output (which is a digital domain signal) in analog units of volts and hertz. For TRAN analysis, note that transient variations in the voltage reference of an ADC may not have a clearly identifiable effect in this analog version of the output, even though they could indeed cause code changes in the digitally coded output of the real device.

Simulation Speed

Full signal chains might experience long simulation times. First because the signal chain will be composed by multiple components, but also because ADCs and DACs can have complex models, so it is not uncommon for them to be the main cause for a slow simulation. To alleviate this, consider removing the ADC temporarily from the schematic when exploring aspects of the signal chain that are not impacted by it.

For information on the topic of speeding up simulations see this article [6].

References

[1] G. Alonso, "Get Up and Running with LTspice," Analog Devices, [Online]. Available: https://www.analog.com/en/analog-dialogue/articles/get-up-and-running-with-ltspice.html.

[2] Analog Devices, "ADI EngineerZone LTspice forum," [Online]. Available: https://ez.analog.com/design-tools-and-calculators/ltspice/.

[3] G. Alonso, "LTspice: Using the .STEP Command to Perform Repeated Analysis," [Online]. Available: https://www.analog.com/en/technical-articles/ltspice-using-the-step-command-to-perform-repeatedanalysis.html.

[4] R. Delaney and P. Delizia, "Step-by-Step Noise Analysis Guide for Your Signal Chain," [Online]. Available: https://www.analog.com/en/analog-dialogue/articles/step-by-step-noise-analysis-guide-for-your-signal-chain.html

[5] G. Alonso, "LTspice: Speed Up Your Simulations," [Online]. Available: https://www.analog.com/en/technical-articles/ltspice-speed-up-your-simulations.html.