ADC Verification

Rapid Adoption Kit (RAK)

Product Version: IC23.1/IC6.1.8, SPECTRE 23.1/21.1
December, 2023

Copyright Statement

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide. Cadence and the Cadence logo are registered trademarks of Cadence Design Systems, Inc. All others are the property of their respective holders.

Contents

Purpose 4
Audience 4
Terms 4
Module 1: Overview 6
Module 2: Evaluate ADC ENOB 10
Module 3: Dynamic Comparator Characterization 15
Lab 3-1: Dynamic Comparator Testbench 15
Lab 3-2: Dynamic Comparator Transient Noise Analysis [Optional] 22
Lab 3-3: Dynamic Comparator Noise Characterization 39
Lab 3-4: Dynamic Comparator Offset Voltage Characterization 52
Lab 3-5: Dynamic Comparator Characterization Across Corners 71
Module 4: Capacitor D/A Converter Characterization 83
Lab 4-1: Characterizing Mismatch 85
Lab 4-2: Analyze Capacitor Mismatch with Monte Carlo Analysis 113
Module 5: Clock Generator Characterization 126
Lab 5-1: Analyzing the Clock Generator Noise 127
Lab 5-1.1 [Optional] Analyzing Clock Power Supply Noise Rejection with Transient Analysis 141
Lab 5-2: Analyzing Clock Power Supply Noise Rejection with PSS/Pnoise Analysis 160
Module 6: Top-Level ADC Calculations 175
Lab 6-1: Top-Level ADC Testbench 175
Lab 6-2: Top-Level ADC Transient Analysis 176
Lab 6-3: Top-Level ADC Transient Noise Analysis 181
Lab 6-4: Top-Level ADC Run Plan 188
Summary 201
References 205
Support 205
Feedback 205
Appendix A 206

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Purpose

This RAK will help users characterize an ADC using the Spectre® Transistor Level Simulator and the Virtuoso Analog Design Environment (ADE). The contents are:

- Evaluate ADC Effective Number of Bits (ENOB)
- Dynamic Comparator Characterization
- Capacitor D/A Converter Characterization
- Clock Generator Characterization

Audience

This document is intended for Analog/RF engineers designing data converters. It covers topics including transient noise analysis, periodic noise analysis (pnoise), Fourier analysis to characterize data converters, and data converter components. Furthermore, a Run Plan is used for characterization of the data converter's dynamic range.

Terms

ADC	Analog to Digital Converter
ADE	Analog Design Environment
APS	Accelerated Parallel Simulator
CAPDAC	Capacitor Digital to Analog Converter
DAC	Digital to Analog Converter
ENOB	Effective Number of Bits
FFT	Fast Fourier Transform
IRN	Input Referred Noise
LAN	Local Area Network
lsb	Least Significant Bit
PDK	Process Design Kit
Pnoise	Periodic Noise (Analysis)
SINAD	Signal to Noise and Distortion Ratio

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

SFDR	Spurious-Free Dynamic Range
SNR	Signal to Noise Ratio
THD	Total Harmonic Distortion
ViVA	Virtuoso Visualization and Analysis

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Module 1: Overview

1.1 Design Example

The ADC design example which has been selected for this workshop is intended for wireless LAN applications (IEEE 802.11b). In a wireless LAN system, the ADC converts the analog output of the receiver into a digital signal; so, the baseband processor can perform demodulation. In this example, a 10-bit Successive Approximation register ADC will be characterized.

The code name for this ADC design is **`saradc`**. It has been implemented using the Cadence generic PDK 45nm CMOS process.

Table 1: ADC Parameters

Parameter	Condition Notes	Typ.	Min	Max	Unit
Power Supply		1.2	1.1	1.3	Volts
Input Range	For each input differential signal: -480mV to +480mV Referred to the Input Common Mode Level	0.7-1.2	VDD-0.5V	VDD	Volts
Input Common Mode		0.96	0.6	VDD	Volts
Aperture Jitter		3			ps_{rms}
Power Dissipation					
Conversion Rate		50	1	50	MS/s
ENOB	Typical: Nominal Conditions, fin=50MHz, fsample=100MHz Min: Worst Case Corner, fin =24.9MHz, fsample=100MS/s*	9.6	9.0	9.8	Bits

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Bit Error Rate	Typical: Nominal Conditions, fin=50MHz, fsample=100MHz Min: Worst Case Corner, fin =24.9MHz, fsample=100MS/s*	0.3	1		Errors per second
Full Power Bandwidth	Typical: Nominal Conditions, fsample=100MHz	25			MS/s

**Mega Samples per second*

The key analog components of the **`saradc`** are the Comparator, the CAPDAC, and the Clock Generator. There are two main factors that dominate the conversion time of the ADC: Comparator decision time and Capacitor DAC settling time.

A Capacitor DAC has been used to eliminate the static power that would have been consumed if a Current Switch DAC was implemented. Furthermore, using a Capacitor DAC allows the Sample and Hold function to be merged into the DAC, eliminating the power and area needed for a Sample and Hold circuit.

Additionally, a dynamic comparator is used to further reduce the static power. The top-level schematic for the **saradc** ADC is shown in Figure 1.

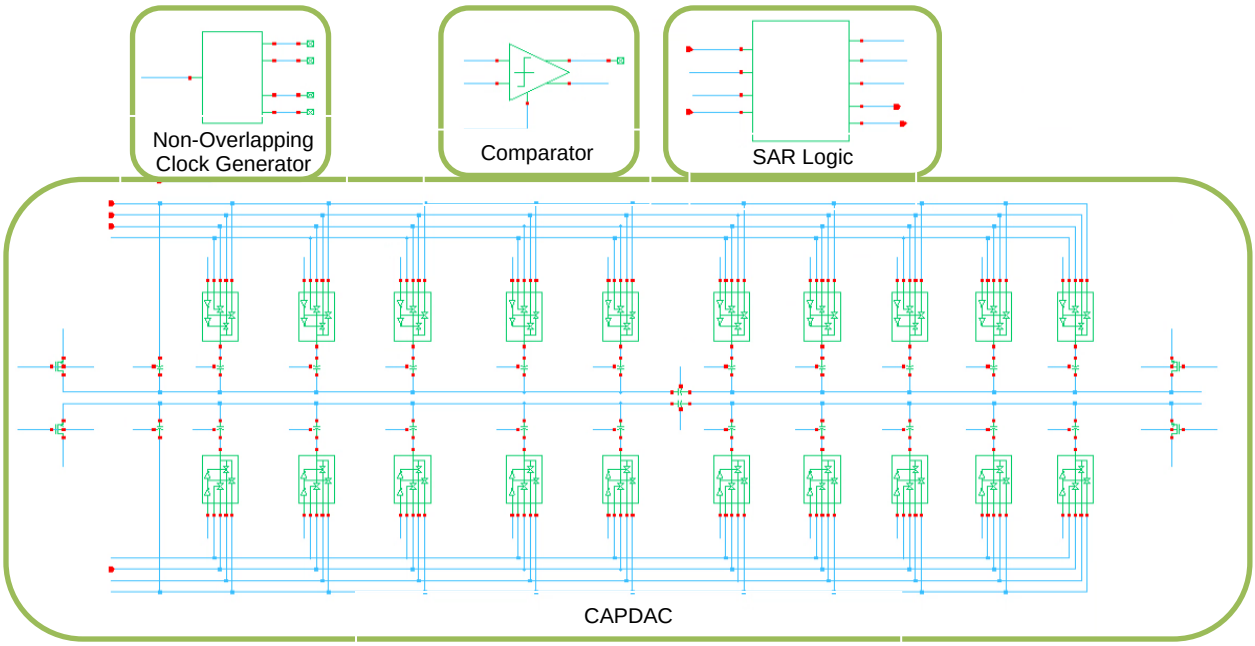


Figure 1: Top-level Schematic for the **saradc** ADC

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

1.2 Design Flow Section

In this RAK, you will use the methodology for characterizing an ADC's ENOB presented in the ADC Design and Verification Seminar, which is also explained in detail in Module 2 of this kit. You will characterize each of the key components of the design to extract the actual parameters required to calculate the ENOB. After this is completed, you will characterize the ADC at the top level by simulating the SINAD. Finally, you will calculate the ADC's ENOB and identify potential issues that will need to be resolved.

1.3 Tools Used

This RAK uses the following Cadence® products for all its labs:

Product Name	Release	Release Name	Products Used
Virtuoso	6.1.8/23.1	IC6.1.8-64b.500.33/ IC23.1-64b.ISR3.27	• Virtuoso IC Design Framework • Virtuoso ADE
Spectre	23.1/21.1	23.1.0.275.isr2/ 21.1.0.879.isr23	• Spectre RF Circuit Simulator • Spectre APS

In this RAK, you will be using ADE Explorer and Assembler, which provide all capabilities required to fully explore and analyze a design against your desired goals. This allows you to maintain the design intent throughout the design cycle.

ADE Explorer provides a quick entry into the analysis process with easy execution of simulations, including support for running Monte Carlo statistics, corner sweeps, pass/fail analysis, and Real-Time Tuning with Spectre® APS.

Virtuoso ADE Assembler extends Explorer's capabilities, providing all those capabilities across multiple testbenches simultaneously. This allows designers to monitor all aspects of the larger analog block they are creating by enabling easy reviewing of all results directly; and generate spec-comparison sheets and datasheets as needed. ADE Assembler also contains targeted tools that aid with key design challenges with early parasitic analysis and design centering.

1.4 Before You Begin

The following instructions assume the SARADC and SARADCII example design databases and proper Cadence® tools have been properly installed on your computer

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

and they are accessible for your environment. Follow the installation and configuration instructions given in the database root installation directory `README.txt` file.

1.5 Definitions

- **Top level** – Refers to the **10Bit_ADC_TB_new** block. This is the top-level cell view in the design.
- **CIW** – Virtuoso Command Interpreter Window – Main window that first appears when the Virtuoso design framework is started.
- **$Project** – Root directory of the seminar workshop workspace.

1.6 Acronyms Used in the Document

Abbreviation	Description
LMB	Left Mouse Button. This indicated that you are to press the left mouse button
MMB	Middle Mouse Button. This indicated that you are to press the middle mouse button
RMB	Right Mouse Button. This indicated that you are to press the right mouse button

1.7 Before You Run the Workshop

This RAK is a processing- and memory-intensive RAK. It requires at least 100GB of memory for the completion of the RAK, and you will also need multithreading (4-8 threads) enabled for a couple of labs. Make sure that you have enough memory and processing power before you continue.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Module 2: Evaluate ADC ENOB

In this section, you will review the methodology to characterize an ADC's dynamic range, its SINAD, and its ENOB. The challenge is that there is no easy method to directly simulate an ADC's ENOB with a single simulation. Instead, you will use a different approach for calculating the ENOB. The approach will be to "divide and conquer". Each source of non-ideality, noise, or distortion will be identified, and a simulation will be performed to characterize its noise contribution. The results will be combined to provide the ENOB for the ADC. Figure 2 (in the "ADC ENOB Calculation" section below) shows an example of the spreadsheet used for the ADC ENOB calculation.

Before discussing the individual terms, let's review the overall methodology. Each term in the spreadsheet represents a source of non-ideality that degrades the ENOB. Each term will be simulated or calculated individually, and then; it will be converted into a noise contribution. The contributions are then converted to voltage and referred to the input of the ADC. Since each term is uncorrelated with the other terms, the contributions are combined by root-sum-squaring them. Therefore, at the ADC input, you will know the maximum signal level and the total noise and distortion due to non-idealities. From this, the SINAD and ENOB can be calculated.

Below is a review of the discussion in the ADC Design and Verification Seminar, outlining what each of the parameters is and how it can be measured.

IDEAL SINAD is the SINAD for an ideal 10-bit ADC and is given by the standard equation:

$$\text{SINAD} = (6.02\text{dB/bit}) * (\text{Number of bits}) + 1.76\text{dB}$$

For a 10-bit ADC:

$$\text{SINAD} = 60.2\text{dB} + 1.76\text{dB} = \mathbf{61.96dB}$$

ADC Transient SINAD is the SINAD of the ADC for a nearly full-scale input signal near the Nyquist sampling rate. This simulation is used to extract the contribution of the ADC dynamic non-idealities (for example, incomplete settling of the DAC, etc.) on the overall SINAD.

Matching SINAD is the SINAD of the CAPDAC and comes from the Monte Carlo simulation of the DAC. This parameter captures the effect of statistical variation and capacitor mismatch on the SINAD of the ADC.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Ideal DAC Full Scale Output is the full-scale output voltage of the Successive Approximation DAC. This value is a designer-specified input parameter; so, simulation is not required. It is used to determine the full-scale signal level of the ADC.

Quantization Noise is derived from the ideal SINAD. For this ADC with a full scale of 0.96 V_{pk-pk}, or 0.339V_{rms}, and 1024 is the lsb for 10 bits, then each lsb is about 0.339V_{rms} /1024= 331.4µV_{rms}. The quantization noise is 331.4µV_{rms} /sqrt(12), or 95.57µV_{rms} .

Aperture Jitter Noise is an ADC designer-specified parameter used to specify the quality of the clock source.

Total CAPDAC Capacitance is the sum of all individual capacitors in the CAPDAC. The CAPDAC is a differential DAC, therefore containing dual DACs. Each CAPDAC consists of two 5-bit, fully segmented capacitors + one series capacitor to scale the lower CAPDAC values. Total CAPDAC Capacitance = (2 * (2 * (16 + 8 + 4 + 2 + 1 + 1))) * C_{unit}, where C_{unit} is the series capacitor value.

Total CAPDAC Gain (or Attenuation) accounts for the gain (loss) of the DAC due to the parasitic backplate capacitance of the DAC capacitors. It is used to transform the Capacitor DAC noise for the output of the DAC to the input of the ADC.

Total CAPDAC Gain Input Referred Noise is the kTC noise of the DAC. This contribution can be directly calculated from the total capacitance of the DAC.

CAPDAC Gain (attenuation from parasitic cap) corresponds to the parasitic backplate capacitance of the device capacitors in the CAPDAC, which results in attenuation of the signal from the ADC, the CAPDAC, the input to the comparator, and the CAPDAC output. This phenomenon is equivalent to the noise gain from the CAPDAC output to the CAPDAC input. Since the noise is referenced to the ADC input, it needs to be scaled up by the CAPDAC attenuation.

Input Referred PSS Measured Comparator Noise is the noise of the dynamic comparator (the comparator noise simulation will be discussed in Lab 3.2). The comparator noise needs to be “input referred”. The DAC gain is used to refer the comparator noise to the input of the ADC.

CAPDAC Mismatch is calculated from the matching SINAD of the CAPDAC. The SINAD is converted into an equivalent noise.

Clock Phase Generator Noise (Input Referred) is the jitter caused by the clock phase generator. It is measured from a PSS analysis in Lab 5.1.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Supply - Clock Phase Generator Noise (Input Referred) is the jitter caused by the clock phase generator with a noisy voltage supply. It is measured from a PSS analysis in Lab 5.1.1.

Settling + DC Accuracy is the noise corresponding to the Transient SINAD subtracted by the quantization noise.

Matching (1 Sigma) is the noise corresponding to the matching SINAD of the CAPDAC subtracted by the quantization noise.

Equivalent Input Referred Thermal Noise is calculated by taking the magnitude of all the individual non-ideal noise contributions.

Total Noise represents the total degradation to the ideal SINAD from each of the individual effects. Since the degradation has been expressed as noise, the individual terms are root-sum-squared; that is, the square root of the sum of the squares of the individual terms. The terms to be summed are in the yellow boxes.

This methodology can be applied from the start of the design until the final verification of the design is complete. Early in the design, the estimates can be implemented, and the actual block values can be used as they become available. This methodology allows designers to understand where the design bottlenecks and margins are so that the designers can re-allocate the error budget.

After characterizing each of the block parameters, you will look at the ADC SINAD after the block design is complete.

ADC ENOB Calculation

# of Bits	=	10.00	
Full Scale Input	=	0.96	V(pk-pk) Diff
Full Scale Input		339.41	mV (rms)
Quanta Size (q)	=	0.94	mV
Aperture Jitter	=	3.00	ps (rms)
Sampling Frequency	=	100.00	MHz
Test Frequency	=	49.12	MHz
Temperature		27.00	C

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

IDEAL SINAD		61.96	dB
ADC Transient SINAD	=		dB
Matching SINAD	=		dB
Ideal DAC Full Scale Output	=		V (p-p)
Measured Ideal DAC Output	=		V (p-p)
Quantization Noise			
Quantization Noise	=		µV (rms)
Jitter Noise			
Aperture Jitter Noise	=		µV (rms)
CAPDAC Noise			
Total CAPDAC Capacitance			pF
Total CAPDAC Gain (or Attenuation)			
Total CAPDAC Gain Input Referred Noise	x		µV (rms)
CAPDAC Gain (attenuation from parasitic cap)			
StrongARM Latch (Dynamic Capacitor) Noise			
Input Referred PSS Measured **Comparator** Noise			µV (rms)
Accounting for CAPDAC attenuation	x		µV (rms)
Jitter from Phase Clock Generator Noise			
Clock Phase Generator Jitter			fs (rms)
Clock Phase Generator Noise (Input Referred)	=		µV (rms)
Jitter from Clock Phase Generator Supply			
Clock Phase Generator Jitter			fs (rms)
Clock Phase Generator Noise (Input Referred)	=		µV (rms)

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

SINAD Transient "Noise"			
Settling + DC Accuracy	=		µV (rms)
CAPDAC Mismatch (1 Sigma)			
Matching (1 Sigma)	=		µV (rms)
Equivalent Input referred Thermal Noise	=		µV (rms)
Total Noise	=		µV (rms)
SINAD (with Noise)	=		dB
ENOB	=		Bits

Figure 2: ADC ENOB Calculation

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Module 3: Dynamic Comparator Characterization

In this section, you will characterize a dynamic comparator. The key parameters for the dynamic comparator are noise and offset voltage. The key technologies for these simulations are transient analysis and analog periodic steady-state analysis. Spectre RF Shooting Newton analysis will be used to calculate the periodic steady-state of the comparator. The periodic small-signal noise analysis will be used to calculate the noise of the comparator, and the offset voltage of the comparator will be calculated using transient analysis.

Lab 3-1: Dynamic Comparator Testbench

ACTION 1: Start Virtuoso. In the command line, type the following commands:

```
% cd $PROJECT/WORK/saradc
% source project.cshrc
% virtuoso -log CDS.log &
```

The **project.cshrc** file should specify the locations of all required tool executables (location of the IC product installation) and other environment setup such as the location of the Cadence license file. If Virtuoso fails to start, make the required corrections to this file.

After Virtuoso starts, you should see the main Virtuoso window (referred to as the CIW) as shown below:

Figure 3: Command Interpreter Window

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The Cadence Library Manager window should also appear as follows:

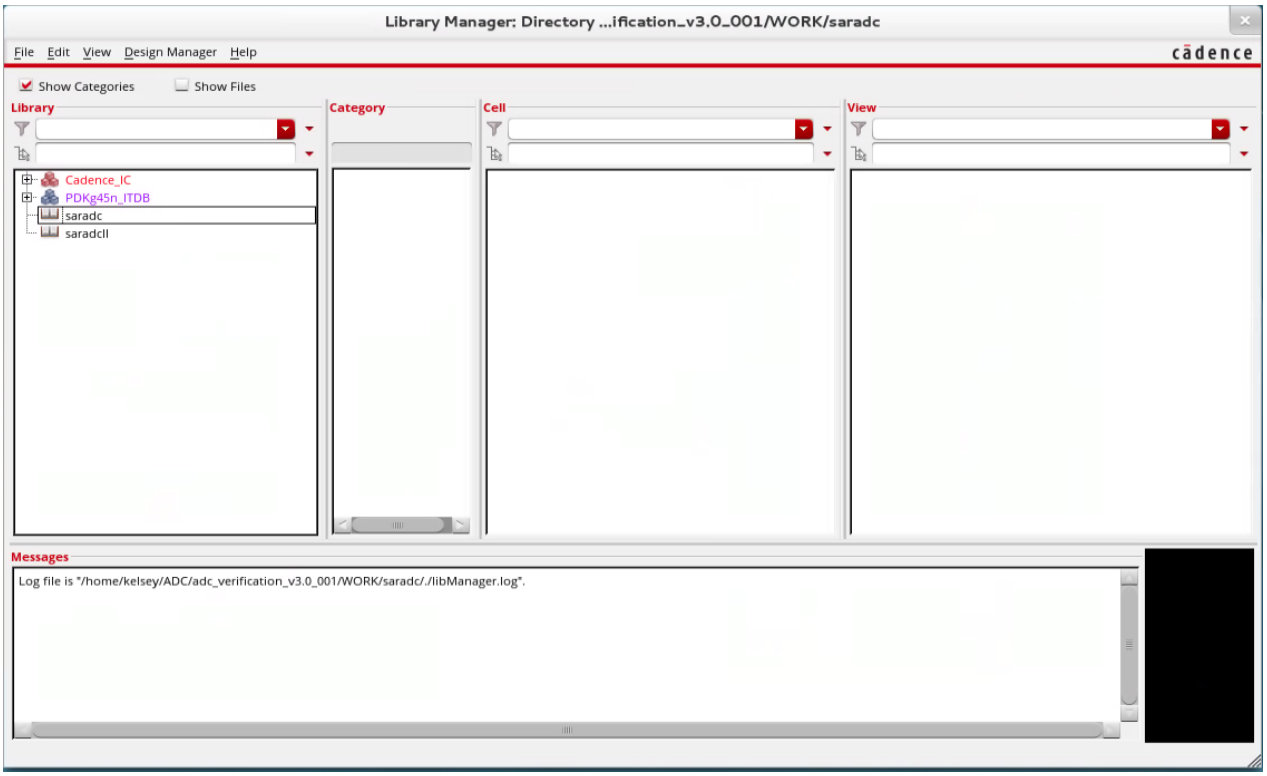


Figure 4: Virtuoso Library Manager

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 2: Use the Library Manager to open the following testbench schematic:

Library: saradc
Cell: 10Bit_ADC_TB_new
View: schematic

The schematic shown below will appear. This is the testbench that you will use to characterize the ADC Transient SINAD. First, you are going to characterize the dynamic comparator. You will be descending several levels to explore its design hierarchy.

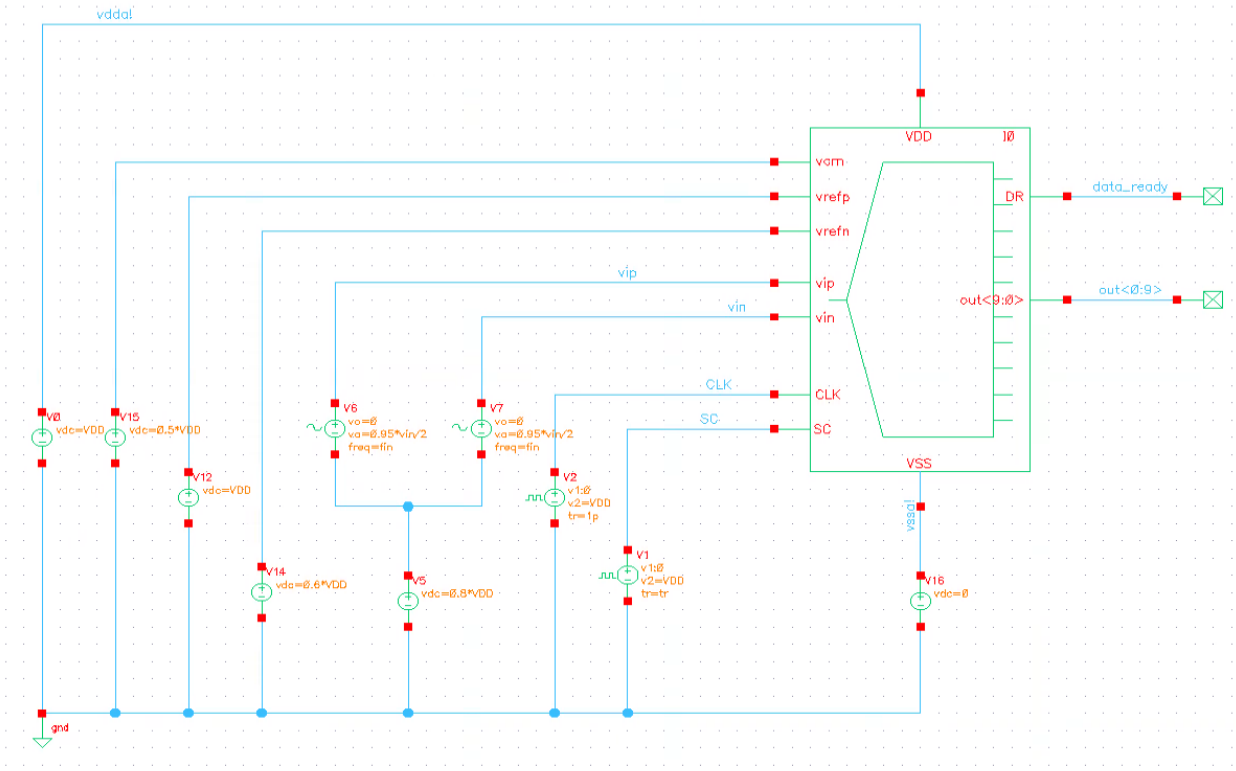


Figure 5: ADC Testbench Schematic

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 3: LMB-select instance **I0** in the **10Bit_ADC_TB_new** cell. Then do **RMB > Descend Read**. The following form will appear. Make sure that the schematic view is selected and then click **OK**.

Figure 6: Descend Read Window

The core of the 10-Bit ADC should now be visible.

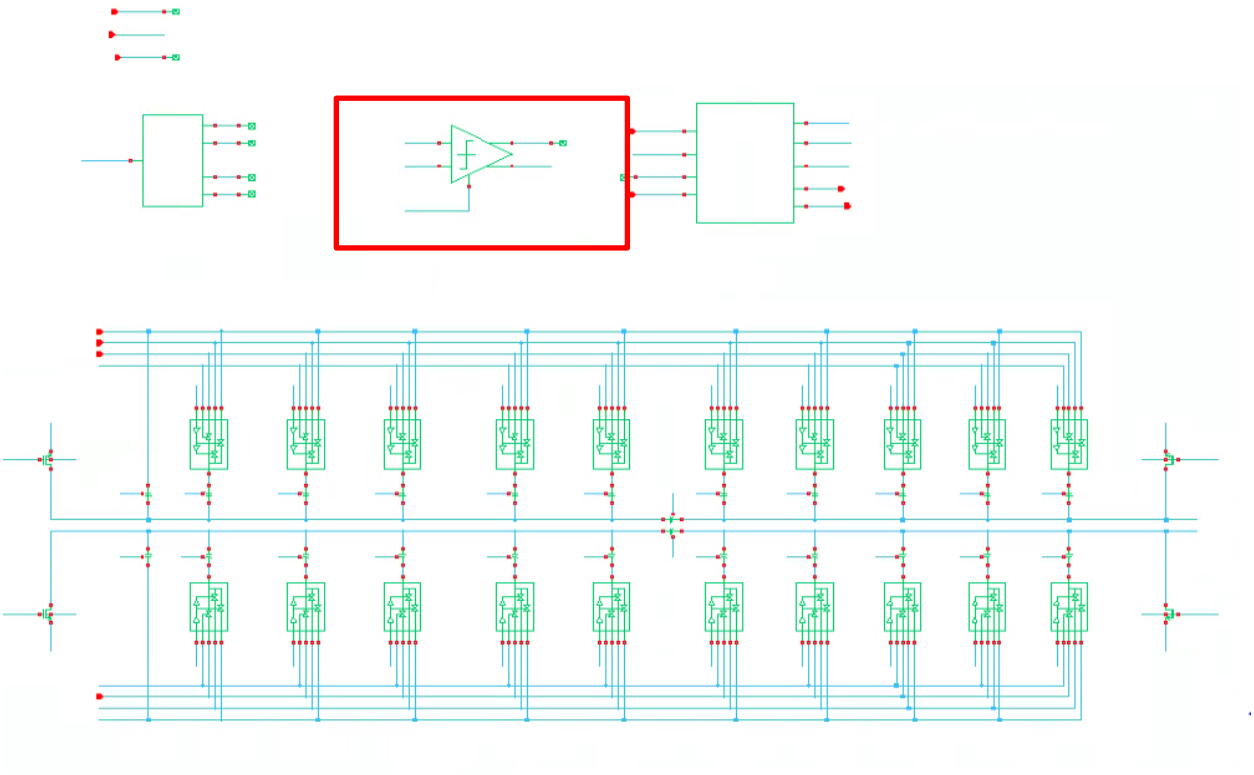


Figure 7: ADC core schematic, with the dynamic comparator shown in red

You will now descend into the dynamic comparator to study its design in more detail.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 4: LMB-select the **I55** comparator instance shown in red in Figure 7, and then do **RMB > Descend Edit**. In the form that appears, make sure that the schematic view is selected and then click **OK**.

Figure 8: Descend Edit Window

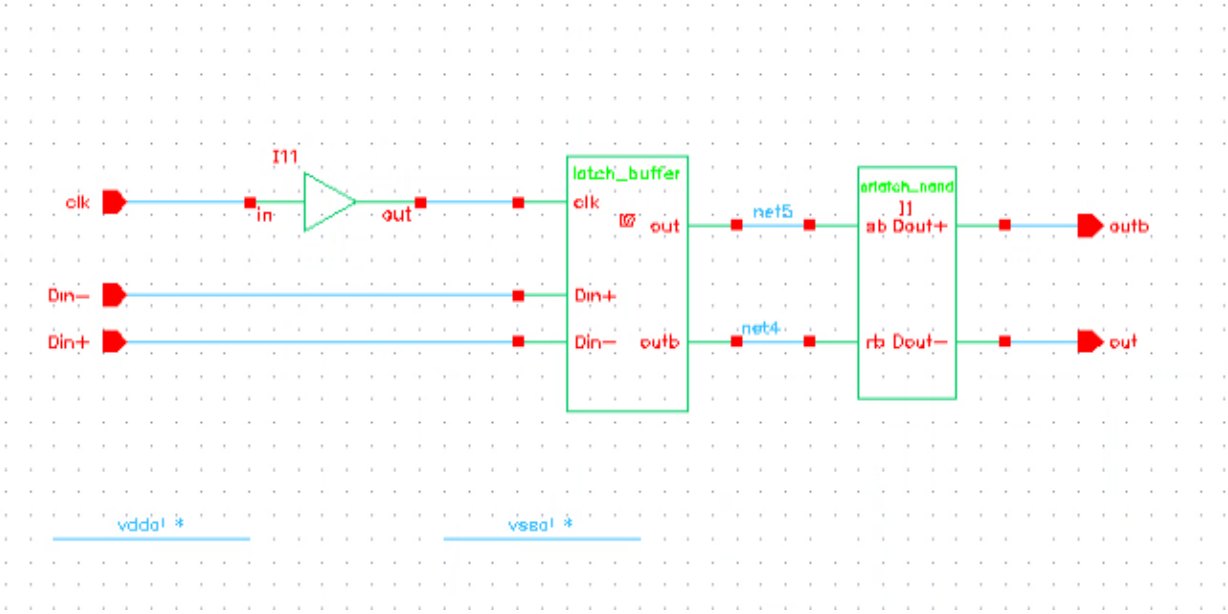


Figure 9: ADC Comparator Schematic

The comparator design consists of a StrongArm latch followed by an RS flip-flop. The StrongArm latch is triggered by a clock signal and converts the differential signal applied at the input into a 1-bit digital code. The output levels are either VDD or GND.

The StrongArm latch uses regenerative gain to quickly convert the differential voltage into digital levels. The RS flip-flop provides additional gain and turns the output of the latch into digital levels.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 5: LMB-select the **I0 latch_buffer** instance and then do **RMB > Descend Edit**.

Click **OK** and the next open tab should display Figure 10, which is the schematic of the ADC's Comparator Latch Buffer.

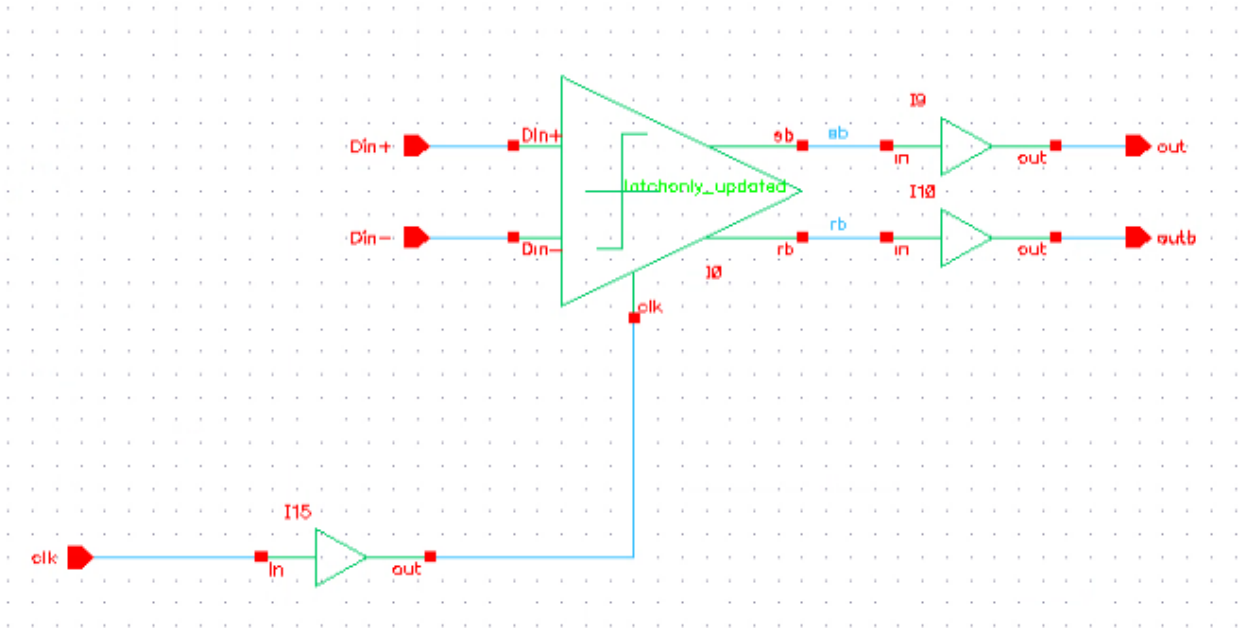


Figure 10: ADC Comparator Latch Buffer

ACTION 6: LMB-select the **latchonly_updated** instance and do **RMB > Descend Edit**. The following form will appear. In this form, make sure that the **schematic** view is selected and then click **OK**.

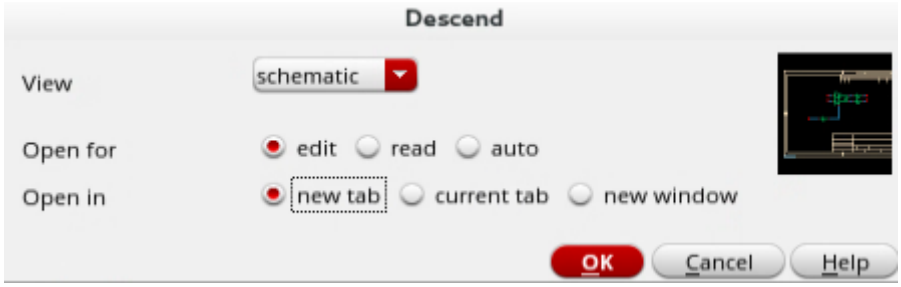


Figure 11: Descend Edit Window

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The schematic that will open is the core block of the comparator: A StrongArm latch. StrongArm latches are widely used in custom digital designs such as logic, sense amps, and decision circuits. A StrongArm latch is fast, operates at low voltage, and dissipates no static power, making it a good choice for use as a comparator for advanced node data converters.

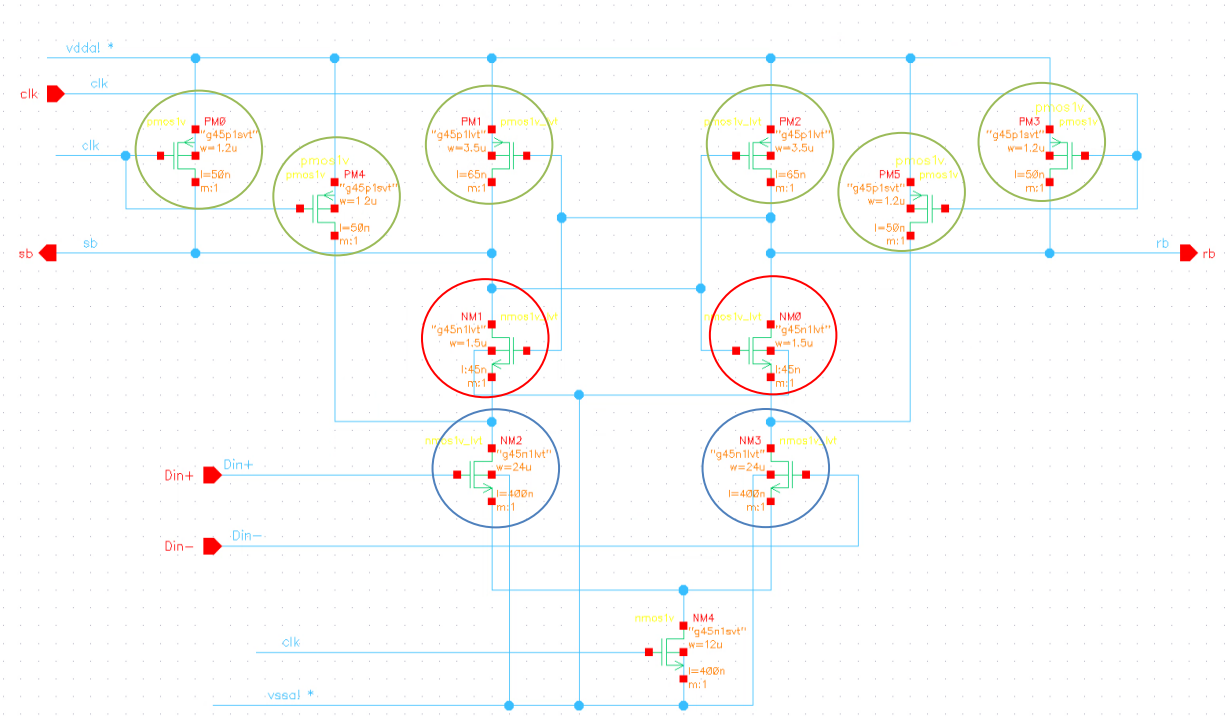


Figure 12: StrongArm Latch Schematic

When the latch signal goes high, the p-channel transistors (green) that shunt the output nodes are disabled and the n-channel transistors (red) turn on, enabling the differential pair (blue). The voltage difference between the input nodes is converted into current and is used to drive the load. The load is a latch made from two back-to-back inverters. The load uses the regenerative gain to amplify the input differential signal and convert it into full-scale output level. In addition, further improvements have been made to increase performance:

1) The input clock is buffered by an inverter. The inverter is in the **latchonly_updated** schematic.

2) The output drives an RS flip-flop to retain the output level when the latch is disabled. The RS flip-flop is in the latch schematic.

The latch is reset at the start of every conversion. The series n-channel transistor turns off and the p-channel transistors clamp the output nodes to VDD.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 7: When you are done reviewing the design, close the schematic window by selecting **File > Close All**.

Lab 3-2: Dynamic Comparator Transient Noise Analysis [Optional]

There are two methods to analyze the noise of a comparator. In this section, you will use a transient noise analysis to characterize the noise of the dynamic comparator. In Lab 3-3, you will use a periodic steady-state analysis to characterize the noise of the comparator and compare it to the results from this lab.

The transient noise analysis of a comparator is easy to understand and has an easy testbench to simulate. The main downside of performing a transient noise analysis is that the simulation requires a long time to achieve the desired accuracy. Another fault in the transient noise analysis is its inability to give in-depth insight into the sources of noise inside the comparator circuit.

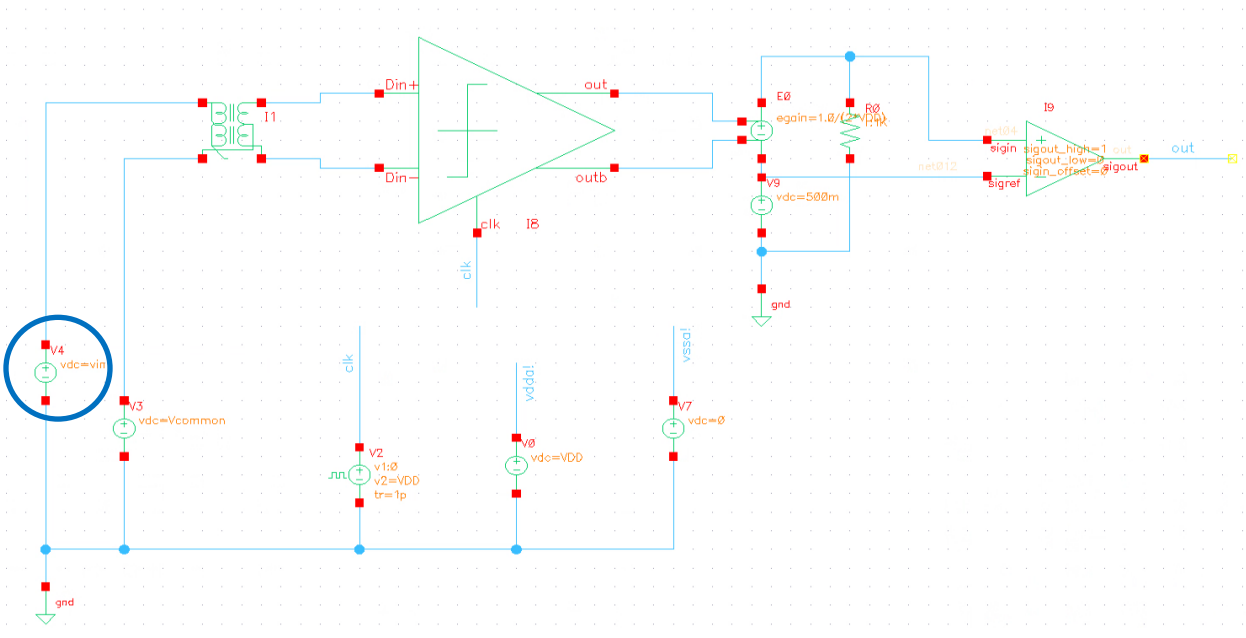


Figure 13: Comparator Transient Noise Testbench

To simulate a transient noise analysis, first a DC input must be applied to the input of the comparator. The DC input must have an amplitude within the expected noise excursions. For example, if the expected noise is ~100µV_{rms}, one could put in a 50µV DC input. In Figure 13, the voltage source **vin** (blue circle) is responsible for this offset voltage.

After the DC input is applied, the comparator is clocked a large number of times and the number of 1s and the number of 0s out of the comparator are counted. From the ratio of

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

1s to the total number of clock cycles, you can calculate the required amount of equivalent Gaussian noise that would give that ratio. To do this calculation, you are assuming that this noise is Gaussian in distribution. For larger positive offset voltages, the probability of an error due to noise is lower, shown in the top waveform of Figure 14. Furthermore, for larger negative offset voltages, the probability of an error due to noise is higher, as seen in the bottom waveform of Figure 14.

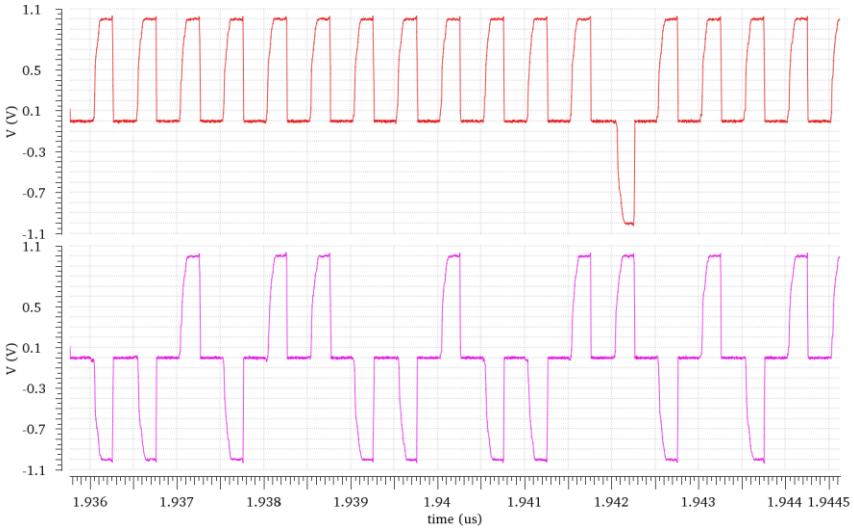


Figure 14: Plots of the output of the comparator with different offset (vin) values

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

To estimate the IRN of the comparator by using the transient noise analysis, a sweep of the input offset voltages from -X mV to X mV must be done. The first method to estimate the IRN requires the results to be fit with an error function. Figure 15 illustrates the shape of an error function, and this is what you expect to get after running the sweep of offset voltages.

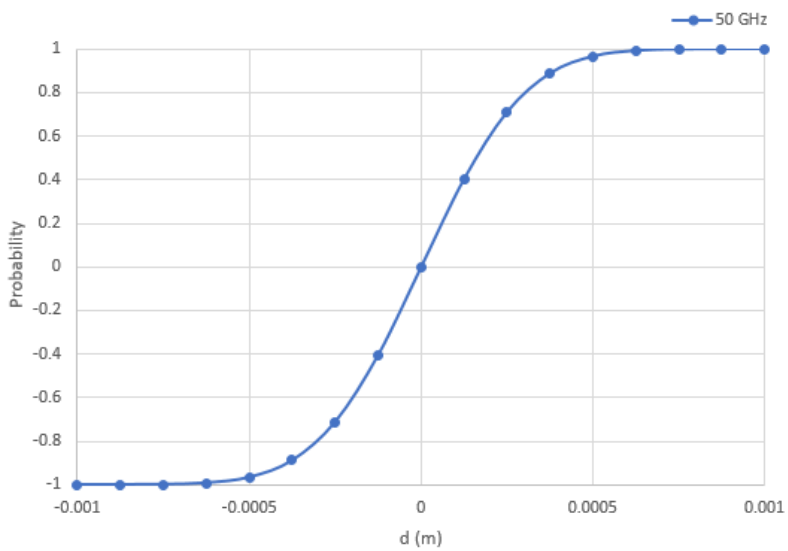


Figure 15: IRN Calculation by Fitting Error Function

Traditionally, Excel has been used to fit the results to the error function. This is an iterative method for estimating the transient IRN of the comparator. One downfall is that it is difficult to control the accuracy of the simulation using this method. In addition, the fitting method does not provide insight into the measurement error.

The alternative method to calculate the IRN of the comparator is by calculating the noise voltage directly from the probability data. At each offset voltage in the sweep, the average of the noise voltage is calculated. Using this method gives a better noise estimate over the former method since the results are generated from averaging a large sample. It also gives more insight into the accuracy of the measurement.

Both methods for measuring the dynamic comparator noise with the transient noise analysis are highly dependent on statistics. Regions with large offset are less sensitive to numerical error in simulation but are more sensitive to transient noise, whereas regions of low offset are more sensitive to numerical error in simulation but are less sensitive to transient noise. It is up to the designer to either simulate fewer events at high accuracy or more events at low accuracy.

Before you begin, you first need to discuss the value of Noise F_{max}, which is the main parameter needed to run a transient noise analysis. Noise F_{max} is used to control the highest noise bandwidth (BW); therefore, it controls the accuracy and efficiency of the

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

simulation. If Noise F_{max} is set too small, the noise sources have a very narrow bandwidth (BW) and you can lose accuracy. If Noise F_{max} is set too large, the transient simulation will run with very small timesteps, leading to slow simulation.

For this lab, you will be using Noise F_{max} = 50GHz. The reason for this value is illustrated in Figure 16. The graph shows the relationship between the value of Noise F_{max} and the noise of the comparator. As the value of Noise F_{max} is increased from 0 to 100 GHz, the noise of the comparator levels out to around 575.6µV_{rms}. Because of this leveling out, there is no need to run the simulation for values of Noise F_{max} greater than 50GHz.

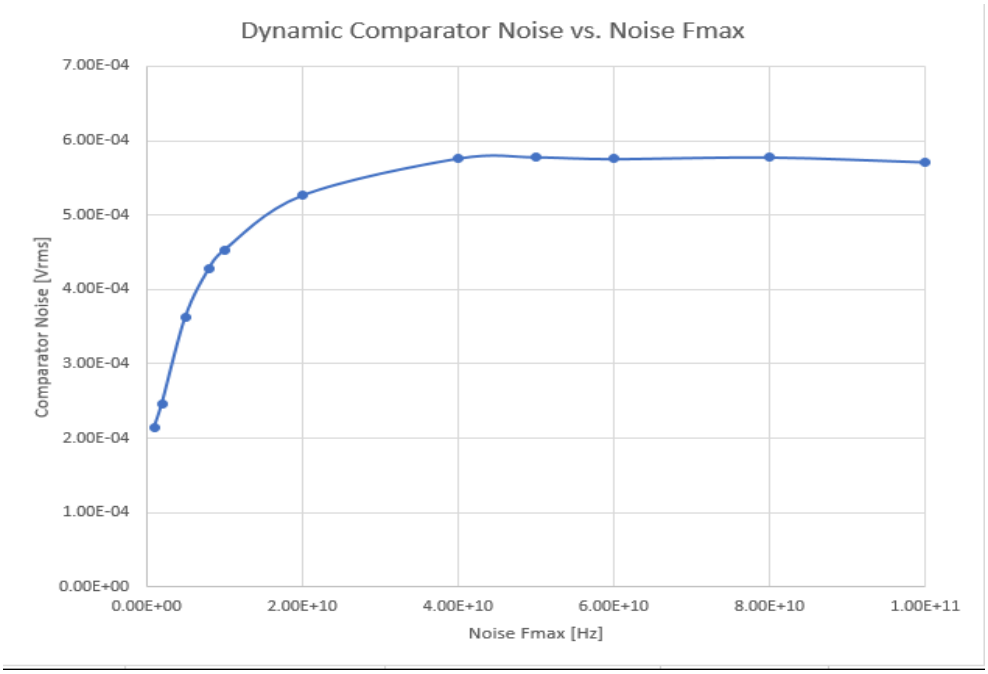


Figure 16: Dynamic Comparator Noise vs. *Noise F_{max}* Value

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 8: In the Library Manager, select **saradc > Everything > comparator_tran_noise_TB**. Double-click on **schematic** to open the schematic.

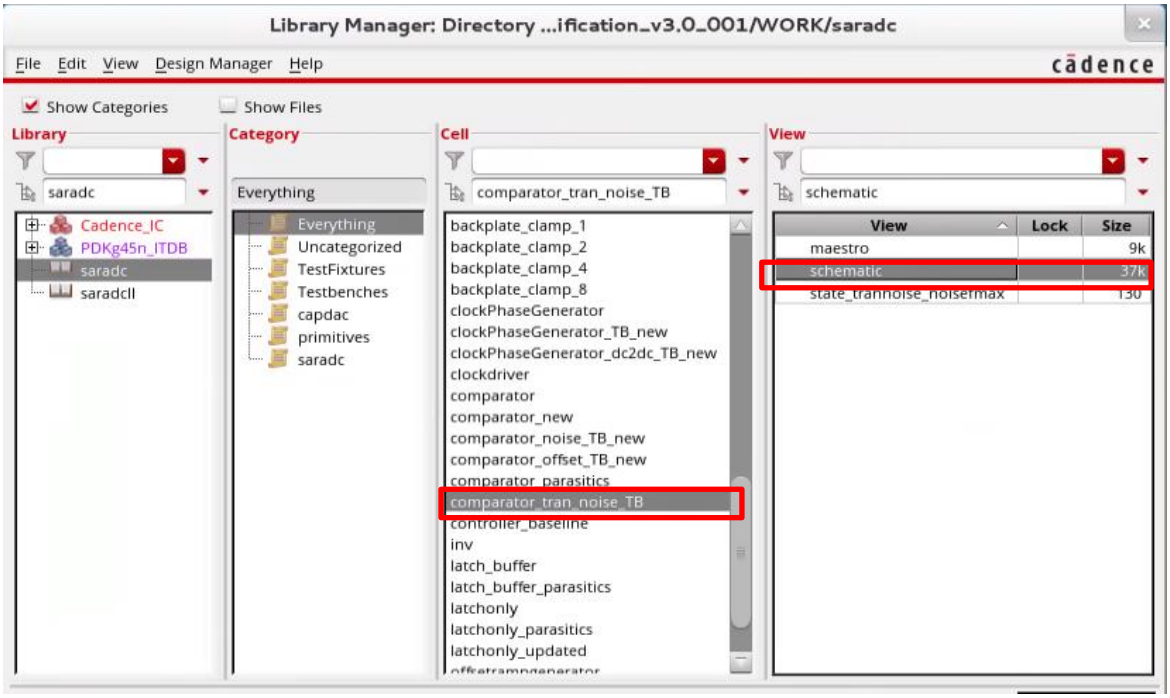


Figure 17: Library Manager

ACTION 9: In the schematic window, select **Launch > ADE Assembler**.

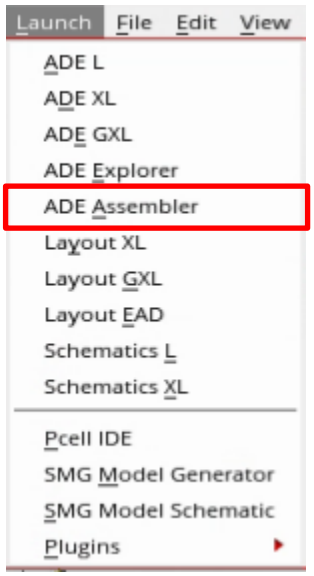


Figure 18: Launch ADE Assembler

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 10: In the **Launch ADE Assembler** window, select **Open Existing View**. Then, click **OK**.

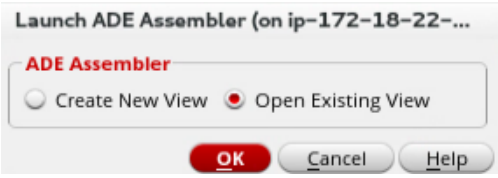


Figure 19: Open Existing Assembler View

ACTION 11: In the **Open ADE Assembler View** window, verify that the view is set to **maestro**. Then, click **OK**.

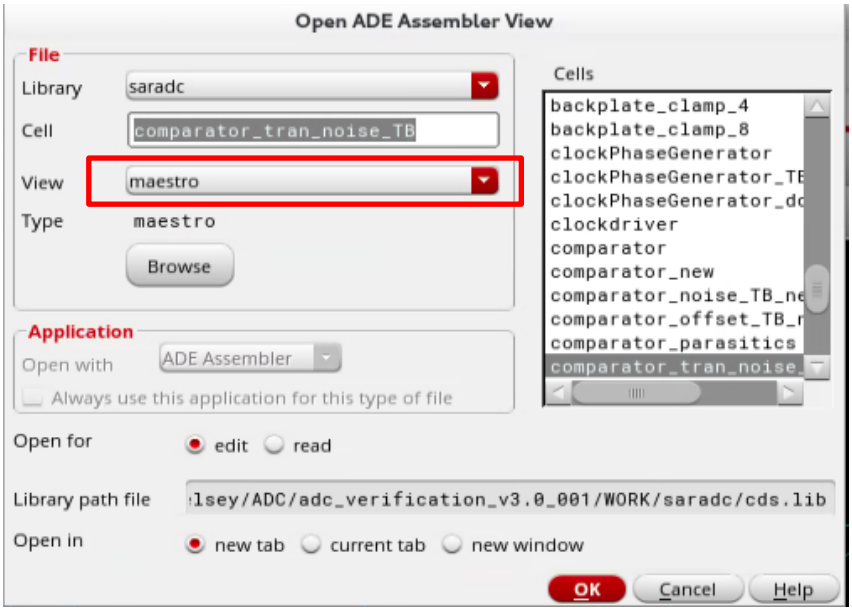


Figure 20: Open ADE Assembler View

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 12: Once the **maestro** tab is loaded, it should look like Figure 21.

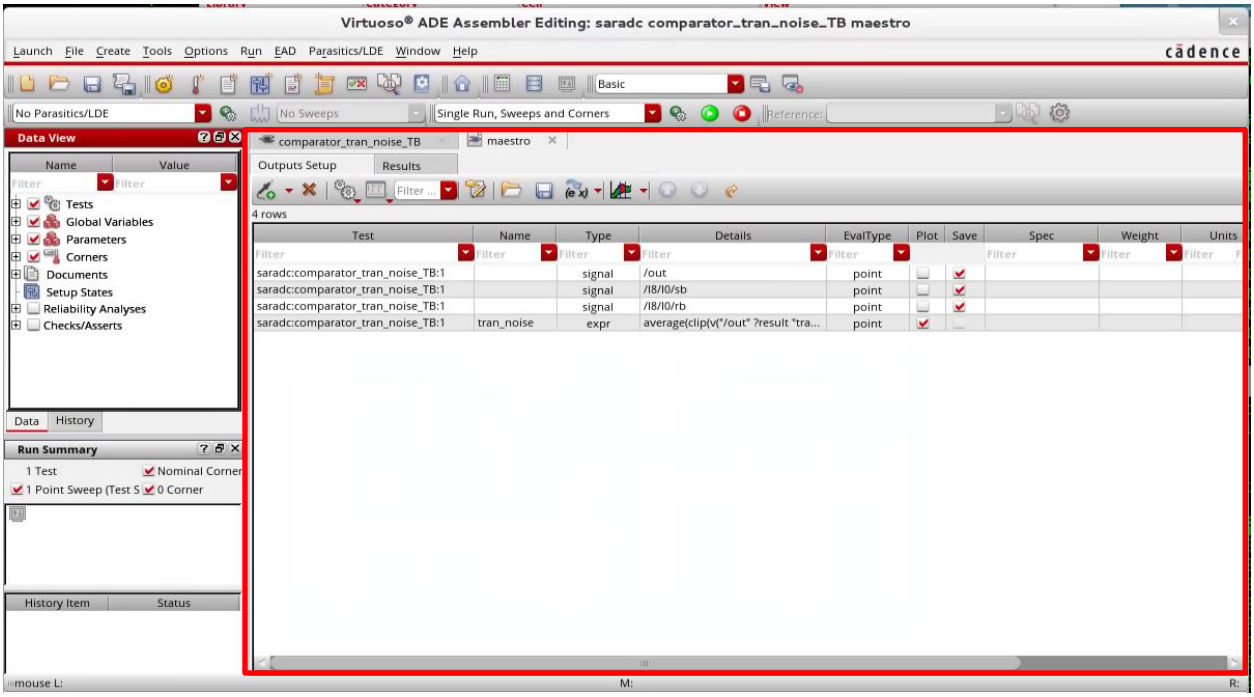


Figure 21: ADE Assembler Window

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 13: Click on the "**+**" symbol next to **Tests**, and then click on the "**+**" next to **saradc:comparator_tran_noise_TB:1**.

Furthermore, click on the "**+**" next to **Analyses** to view the type of analyses set up. Double-click on **tran** to open the **Choosing Analyses** window.

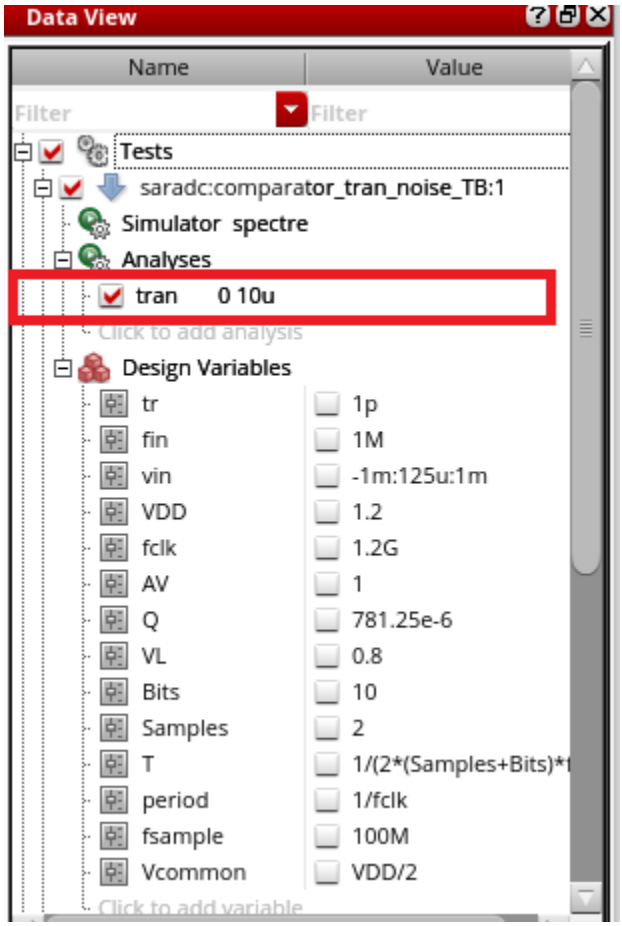


Figure 22: Transient Analysis

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 14: In the **Choosing Analyses – ADE Assembler** window, verify that the transient analysis is as shown in the picture below.

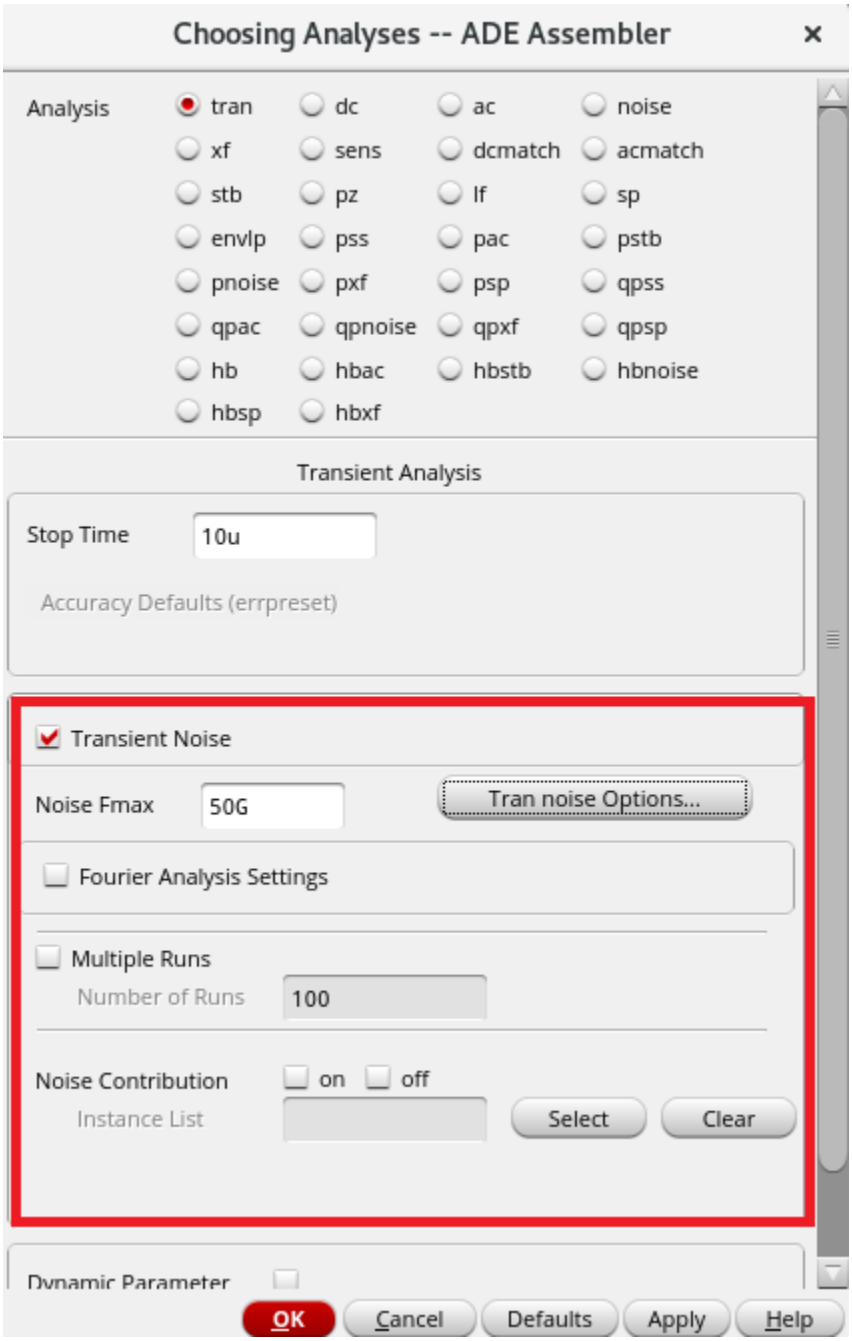


Figure 23: Transient Analysis Setup

The **Stop Time** value is set to **10µs** because this will allow enough time for the simulation to get an accurate measure of the transient noise.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Make sure that the **Transient Noise** option is selected. The **Noise Fmax** value should be set to **50G**. Noise F_{max} sets the bandwidth of the random noise sources that are injected at each time point in the transient analysis. Since your ADC's clock frequency is set to about 1.2GHz, you want your maximum noise frequency to be about 40 times the clocked frequency of the ADC.

ACTION 15: After you are done reviewing the **Choosing Analyses** window, click **OK** to close the window. Next, you will review the design variables of this simulation, which are shown in Figure 24.

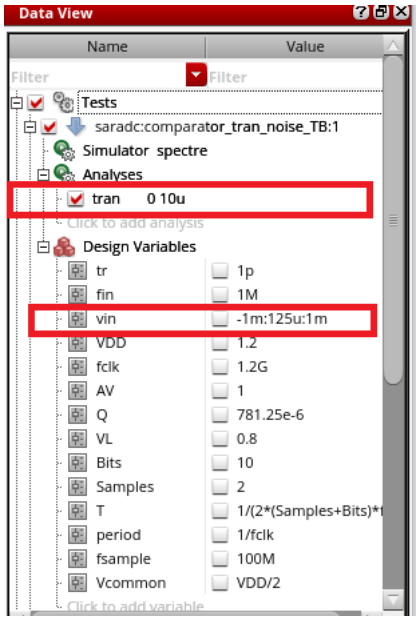


Figure 24: Design Variables of Transient Noise Analysis

The variable **vin** controls the offset applied to the comparator. In this simulation, you will be varying the value from -1mV to 1mV in steps of 125µV for a total of 17 simulations.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Note: You can either run the simulation as ACTION 16 below describes, or you can load the data from a previous run in the **History** tab of the Assembler view. Figure 25 shows what the Assembler view will look like if you RMB-click on **Tran_Data** and then select "**Load Setup to Active**". If you do load the data from the **History** tab, skip to ACTION 17.

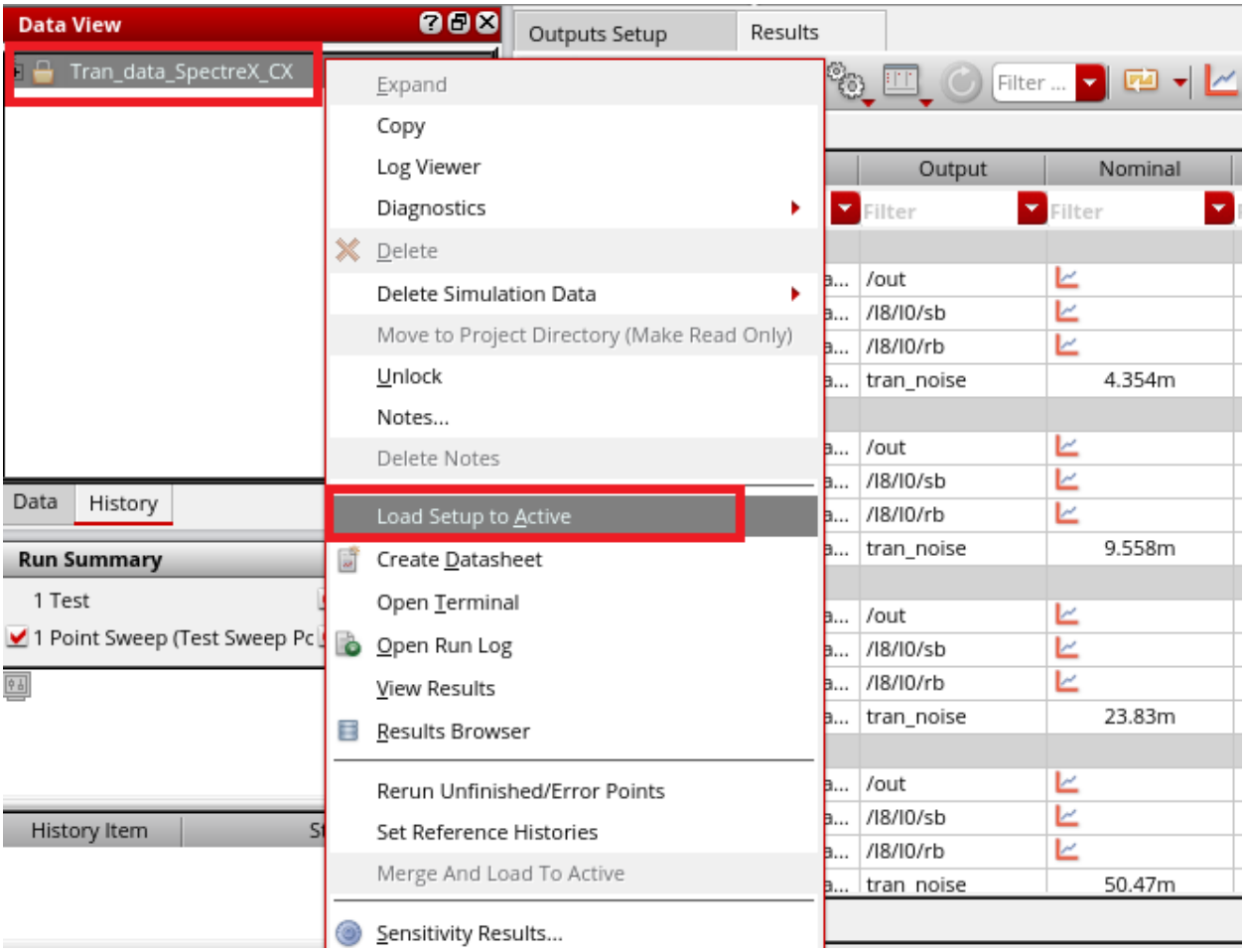


Figure 25: Assembler View of Comparator Transient Noise

ACTION 16: To start the simulation, select the ("**Run Simulation**") button. When the simulation completes, a ViVA window will appear, showing a plot of all the transient noise values calculated at each offset voltage value. This plot should appear similar to.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

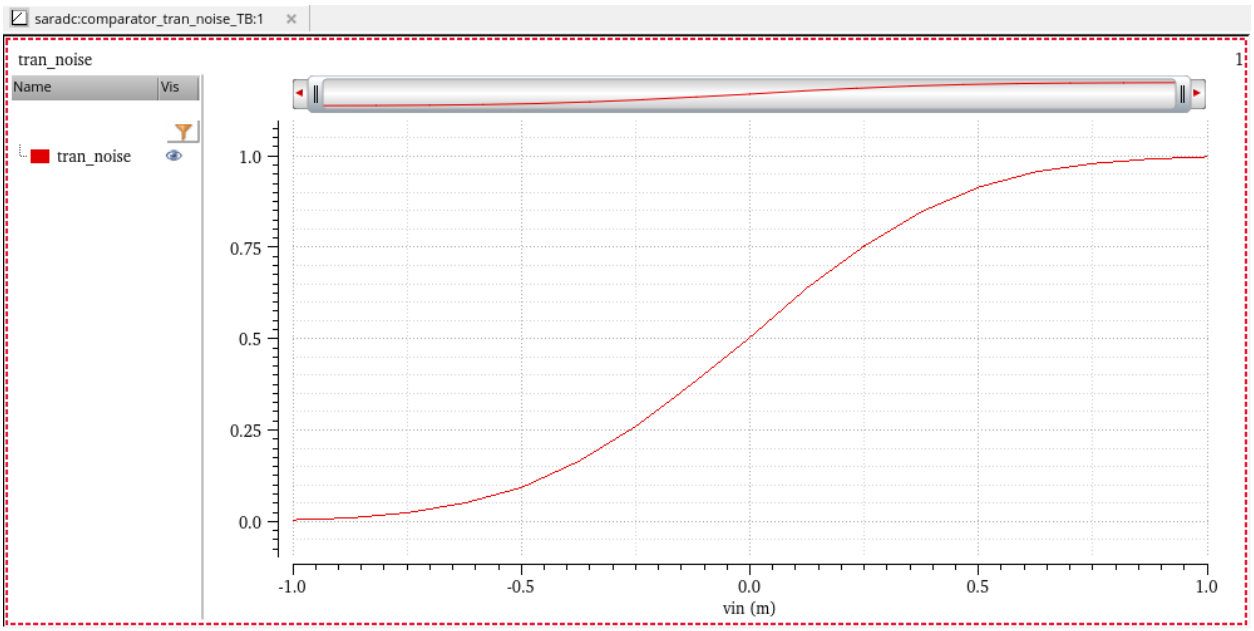
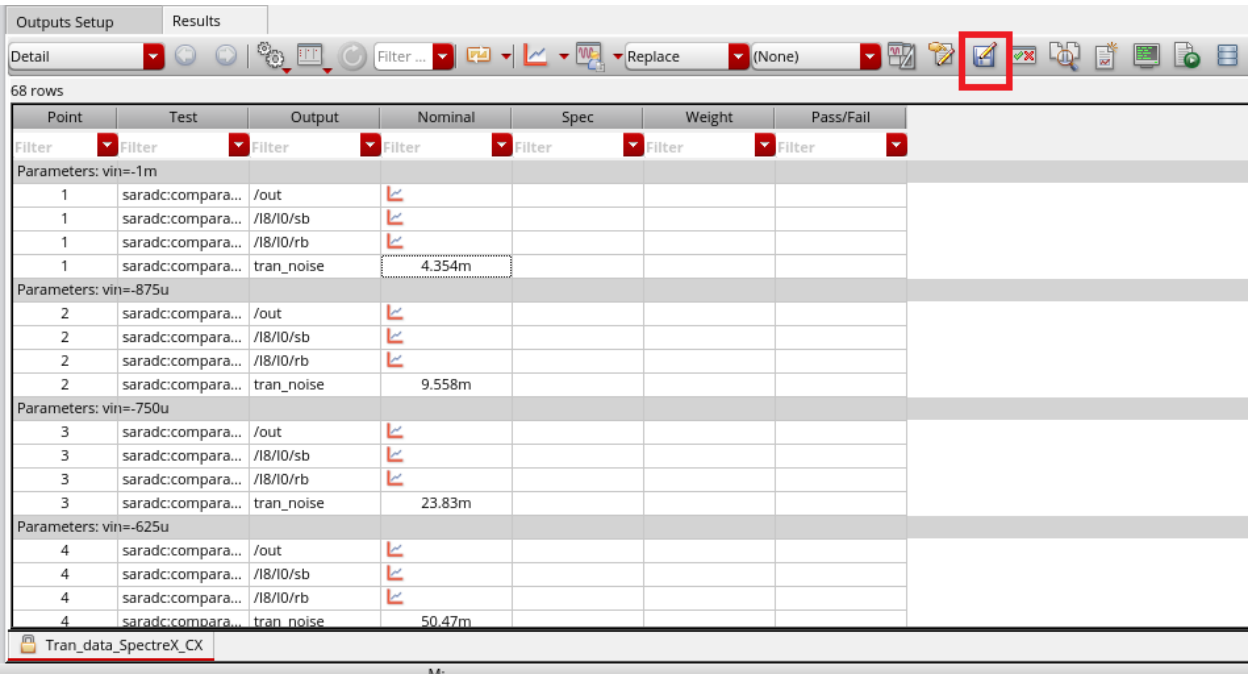


Figure 26: Comparator Transient Noise Values Across -1mV to 1mV Offset Values

Note: If you loaded the data from the **History** tab, you can RMB-click on **tran_noise** and then select "**Plot All**" to recreate Figure 26.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 17: Once you are done looking at the transient noise plot, return to the **Results** tab, which should be populated with each simulation value, as shown in Figure 27. Click on the icon to export the data in CSV or HTML format.



Point	Test	Output	Nominal	Spec	Weight	Pass/Fail
Parameters: vin=-1m						
1	saradc:compara...	/out				
1	saradc:compara...	/I8/I0/sb				
1	saradc:compara...	/I8/I0/rb				
1	saradc:compara...	tran_noise	4.354m			
Parameters: vin=-875u						
2	saradc:compara...	/out				
2	saradc:compara...	/I8/I0/sb				
2	saradc:compara...	/I8/I0/rb				
2	saradc:compara...	tran_noise	9.558m			
Parameters: vin=-750u						
3	saradc:compara...	/out				
3	saradc:compara...	/I8/I0/sb				
3	saradc:compara...	/I8/I0/rb				
3	saradc:compara...	tran_noise	23.83m			
Parameters: vin=-625u						
4	saradc:compara...	/out				
4	saradc:compara...	/I8/I0/sb				
4	saradc:compara...	/I8/I0/rb				
4	saradc:compara...	tran_noise	50.47m			

Figure 27: Comparator Transient Noise Tabular Results

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 18: The **Export Results** window will open, as shown in Figure 28. Save the data under any name. Here, the name "**Comparator_Transient_Noise.csv**" is chosen for the data file.

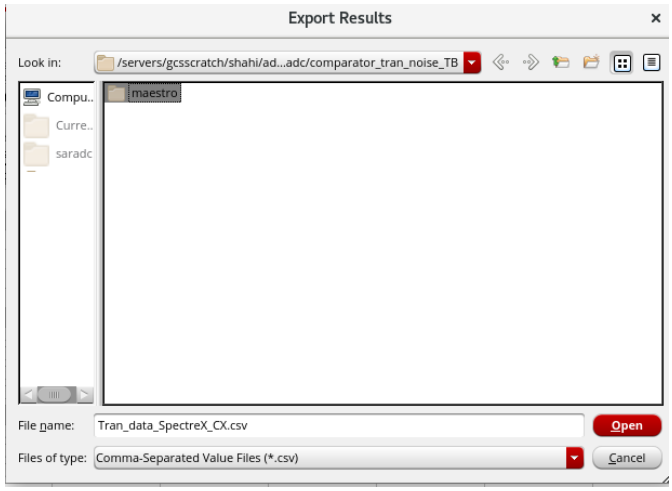
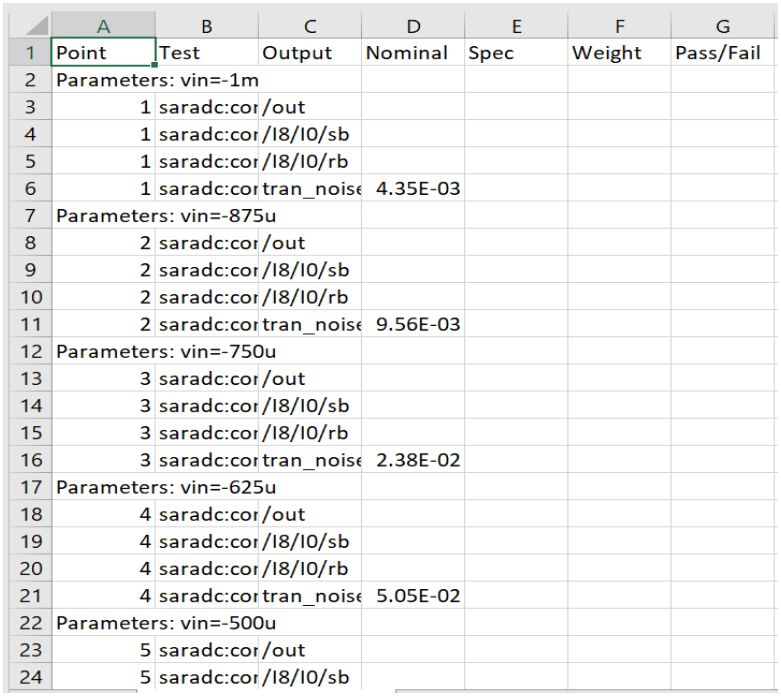


Figure 28: Export Data to File

ACTION 19: Once the data is saved to a .csv file, open it in Excel. The file will appear as shown in Figure 29.



	A	B	C	D	E	F	G
1	Point	Test	Output	Nominal	Spec	Weight	Pass/Fail
2	Parameters: vin=-1m						
3	1	saradc:cor	/out				
4	1	saradc:cor	/I8/I0/sb				
5	1	saradc:cor	/I8/I0/rb				
6	1	saradc:cor	tran_noise	4.35E-03			
7	Parameters: vin=-875u						
8	2	saradc:cor	/out				
9	2	saradc:cor	/I8/I0/sb				
10	2	saradc:cor	/I8/I0/rb				
11	2	saradc:cor	tran_noise	9.56E-03			
12	Parameters: vin=-750u						
13	3	saradc:cor	/out				
14	3	saradc:cor	/I8/I0/sb				
15	3	saradc:cor	/I8/I0/rb				
16	3	saradc:cor	tran_noise	2.38E-02			
17	Parameters: vin=-625u						
18	4	saradc:cor	/out				
19	4	saradc:cor	/I8/I0/sb				
20	4	saradc:cor	/I8/I0/rb				
21	4	saradc:cor	tran_noise	5.05E-02			
22	Parameters: vin=-500u						
23	5	saradc:cor	/out				
24	5	saradc:cor	/I8/I0/sb				

Figure 29: Exported Data Opened in Excel

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 20: From the imported data, create a plot for the parameter **vin** versus the values calculated at each offset voltage, **tran_noise**. The resulting plot should resemble Figure 26.

ACTION 21: After you finish making the measured density plot, the next step is to fit an error function to the probabilities. The error function equation is shown below:

$$erf(x) = \frac{1}{\sqrt{\pi}} \int_{-x}^{x} e^{-t^2} dt$$

First, you need to estimate a transient IRN value. For this lab, you will use 0.536μV_{rms}. This value was chosen after multiple iterations to find the best estimate to fit the data. After selecting your estimate, you can use Excel to create an error function with your estimated transient IRN value. The equation used in Excel is as follows:

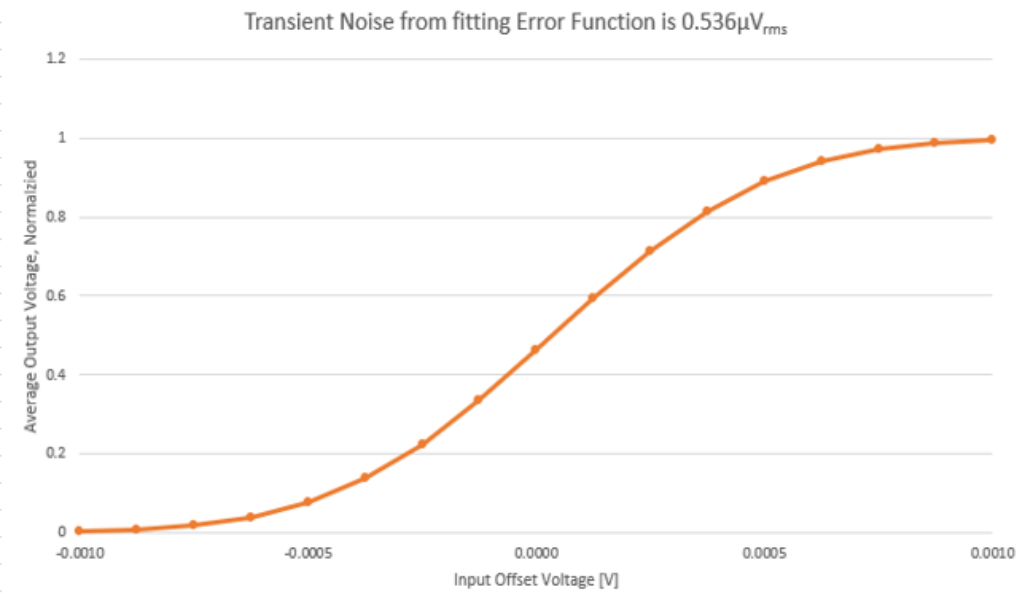
Fitted Density = (1 + ERF(vin / estimated_transient_IRN)) / 2

estimated_transient_IRN in the above equation corresponds to your estimate of the transient IRN value (0.536μV_{rms}). By evaluating this function and then comparing it to the measured density (**tran_noise**) that you received from your simulation, you can calculate the Least Mean Square error of the fit.

The RMS Error value was calculated to be about 9% error between your fitted estimate and the measured data. The following formula is used to calculate the RMS Error, and the results for this calculation are shown in Figure 30.

RMS Error = SQRT(SUM(Difference^2))

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.



Estimated Transient Noise	0.000536				
	Vin	Measured Density	Fitted Density	Difference^2	RMS Error
	-1.00E-03	4.35E-03	4.16E-03	3.60E-08	0.023640328
	-8.75E-04	9.56E-03	1.05E-02	8.53E-07	
	-7.50E-04	2.38E-02	2.39E-02	7.54E-09	
	-6.25E-04	5.05E-02	4.96E-02	8.10E-07	
	-5.00E-04	9.31E-02	9.35E-02	1.90E-07	
	-3.75E-04	1.65E-01	1.61E-01	1.42E-05	
	-2.50E-04	2.61E-01	2.55E-01	3.66E-05	
	-1.25E-04	3.79E-01	3.71E-01	6.61E-05	
	0	5.05E-01	5.00E-01	2.21E-05	
	1.25E-04	6.40E-01	6.29E-01	1.25E-04	
	2.50E-04	7.55E-01	7.45E-01	8.74E-05	
	3.75E-04	8.48E-01	8.39E-01	8.33E-05	
	5.00E-04	9.15E-01	9.06E-01	6.64E-05	
	6.25E-04	9.57E-01	9.50E-01	4.06E-05	
	7.50E-04	9.79E-01	9.76E-01	9.10E-06	
	8.75E-04	9.92E-01	9.90E-01	3.93E-06	
	1.00E-03	9.97E-01	9.96E-01	2.45E-06	

Figure 30: Transient Noise from Fitting Error Function

ACTION 22: From the imported data, you can also plot the IRN of the comparator by using the following Excel formula:

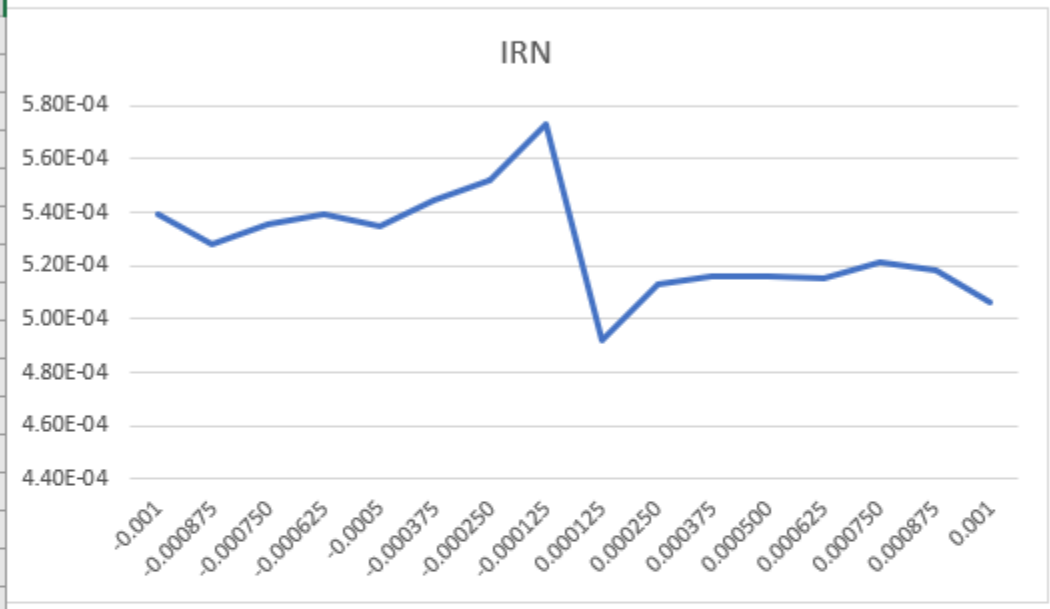
IRN = SQRT(2) * vin / NORM.S.INV(Measured Density)

NORM.S.INV is the Excel formula to return the inverse of the normal distribution with mu=0 and sigma=1 of a specified selection of data. **Measured Density** is the parameter **tran_noise** that you calculated from your simulation.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Noise Vs Input Offset

Input Refered Noise=528uVrms



Vin	Measured Density	Norm.s.inv(M.D)	IRN	Avergae_IRN
-0.001	4.35E-03	-2.62E+00	5.39E-04	5.28E-04
-0.000875	9.56E-03	-2.34E+00	5.28E-04	
-0.000750	2.38E-02	-1.98E+00	5.36E-04	
-0.000625	5.05E-02	-1.64E+00	5.39E-04	
-0.0005	9.31E-02	-1.32E+00	5.35E-04	
-0.000375	1.65E-01	-9.74E-01	5.44E-04	
-0.000250	2.61E-01	-6.41E-01	5.52E-04	
-0.000125	3.79E-01	-3.08E-01	5.73E-04	
0.000125	6.40E-01	3.60E-01	4.92E-04	
0.000250	7.55E-01	6.89E-01	5.13E-04	
0.000375	8.48E-01	1.03E+00	5.16E-04	
0.000500	9.15E-01	1.37E+00	5.16E-04	
0.000625	9.57E-01	1.71E+00	5.15E-04	
0.000750	9.79E-01	2.04E+00	5.21E-04	
0.000875	9.92E-01	2.39E+00	5.18E-04	
0.001	9.97E-01	2.79E+00	5.06E-04	

Figure 31: IRN of Comparator

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The final value reveals an average transient IRN value of 528 μV_{rms}.

As discussed before, estimating the transient IRN of the comparator by fitting it to an error function is the least accurate method of the two. In the next lab, you will use PSS to calculate the noise of the comparator and see that it is very similar in value to the noise calculations done in this lab.

ACTION 23: Close the Excel, ViVA, and ADE Assembler windows once you are done analyzing the data. Move on to the next lab.

Lab 3-3: Dynamic Comparator Noise Characterization

Pnoise analysis can be used to measure the noise of a dynamic comparator. Periodic noise is a small-signal analysis performed after the periodic steady-state has been calculated using Spectre RF Shooting Newton periodic steady-state analysis (PSS). The difference between pnoise and traditional noise analysis is that pnoise includes the effect of noise folding. Noise folding is caused from noise aliasing from out of band into the band of interest. The methodology for calculating the comparator noise is as follows:

1. Apply a small DC input voltage to the comparator (1mV differential) and clock the comparator.
2. Set up Shooting Newton analysis to calculate the periodic operating point (OP).
3. Use sample pnoise analysis with the sample point determined by the differential output levels (internal latching nodes) of the comparator.
4. Set the sampling point to where the comparator differential output levels are separated by about 50mV.
5. Integrate the squared noise of sampler output and take the square root to find the RMS noise on ideal sampler from DC to Nyquist frequency.
6. Divide noise by gain from input to sampler output (50mV/1mV).

The dynamic comparator is a decision circuit, where it makes a decision in an instant. In the case of a comparator, any noise at the instant when the comparator is making the high/low logic-level decision will impact the result. However, noise after the decision is made, when the comparator output is high or low, has no impact and can effectively be ignored. Therefore, for this measurement, you will use sample noise, which is an extension to pnoise analysis.

Pnoise reports the average noise value across the period, which is equal to 1/ (PSS fundamental frequency). In the past, Sample and Hold [S/H] circuits have been used to characterize decision circuits using pnoise analysis. However, an S/H is no longer required since pnoise has been enhanced to support sample noise analysis for the analysis of instantaneous noise.

It should be noted that sample noise is not the same as transient noise analysis. Sample noise is an extension of pnoise that allows analysis of the noise at an instant of

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

time. This replaces using pnoise with an S/H to characterize the noise of a decision circuit. Since it is based on pnoise analysis, it is still a small-signal noise analysis. Transient noise analysis is an extension of transient analysis and is a large-signal noise analysis.

ACTION 24: In the Library Manager, expand **saradc > Everything > comparator_noise_TB_new > schematic**. Double-click on **schematic** to open the schematic.

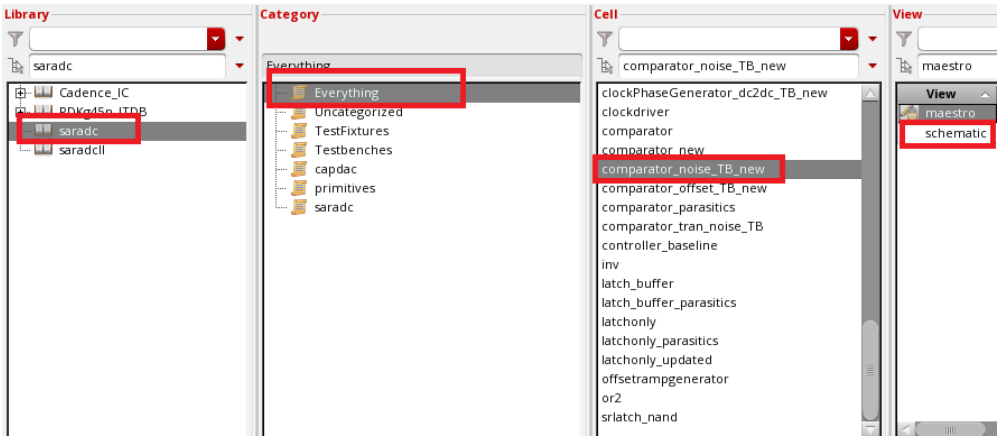
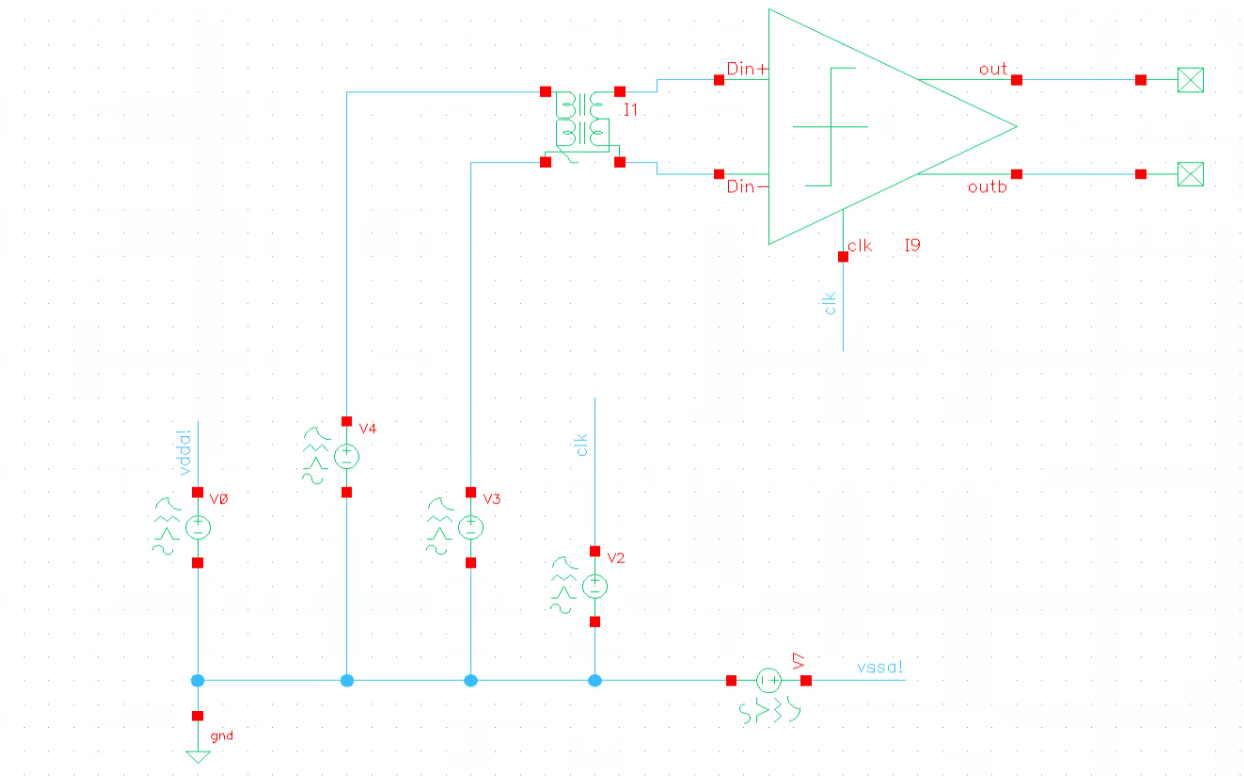


Figure 32: Library Manager



© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Figure 33: Comparator Noise Testbench Schematic

ACTION 25: In the schematic window, select **Launch > ADE Explorer**.

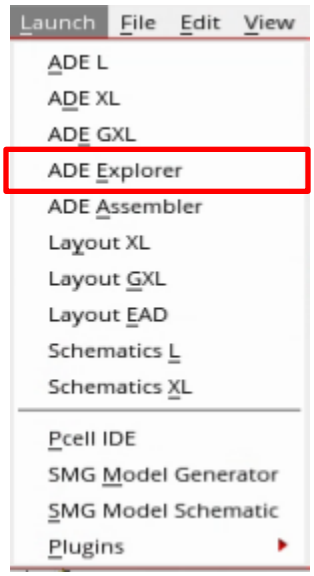


Figure 34: Launch ADE Explorer

ACTION 26: In the **Launch ADE Explorer** window, select **Open Existing View**. Then, click **OK**.

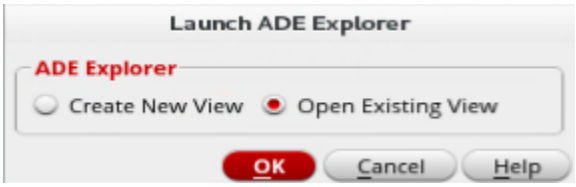


Figure 35: Open Existing View

ACTION 27: In the **Open ADE Explorer View** window, verify that the view is set to **maestro**. Then, click **OK**.

To provide better performance, the saved state file has been centralized for all the new ADE tools into a view called "**maestro**". This view replaces the ADE L state files and the current ADE XL view from the classic tools. By centralizing all the information, moving between the tools and sharing information at different stages of the design cycle becomes easier.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

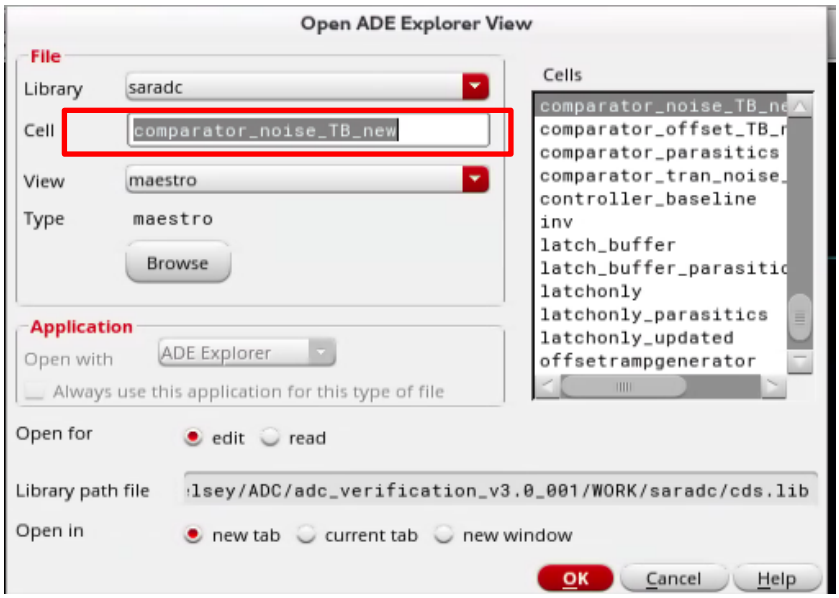


Figure 36: ADE Explorer View

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 28: Once the **maestro** view is loaded, it should look like Figure 37.

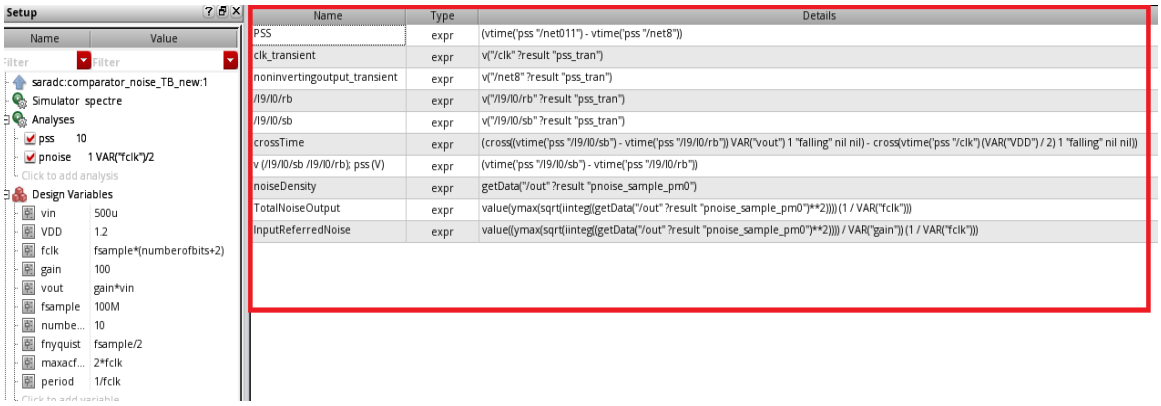


Figure 37: ADE Explorer Window

The ADE window will be populated with the analysis and output setup information.

This ADE setup starts with a Shooting Newton analysis to calculate the periodic steady-state of the circuit, and then, the time domain steady-state response is calculated. In addition, a pnoise analysis is performed to calculate the comparator noise. The measurement setup located on the **Outputs** panel records the total integrated output noise for a fixed gain. This result is used to calculate the IRN.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Pnoise analysis is an extension of traditional noise analysis. Pnoise has been enhanced to include the effect of noise folding. Noise folding occurs due to aliasing, where the noise is folded down into the band of interest from out-of-band frequencies, as shown in Figure 38.

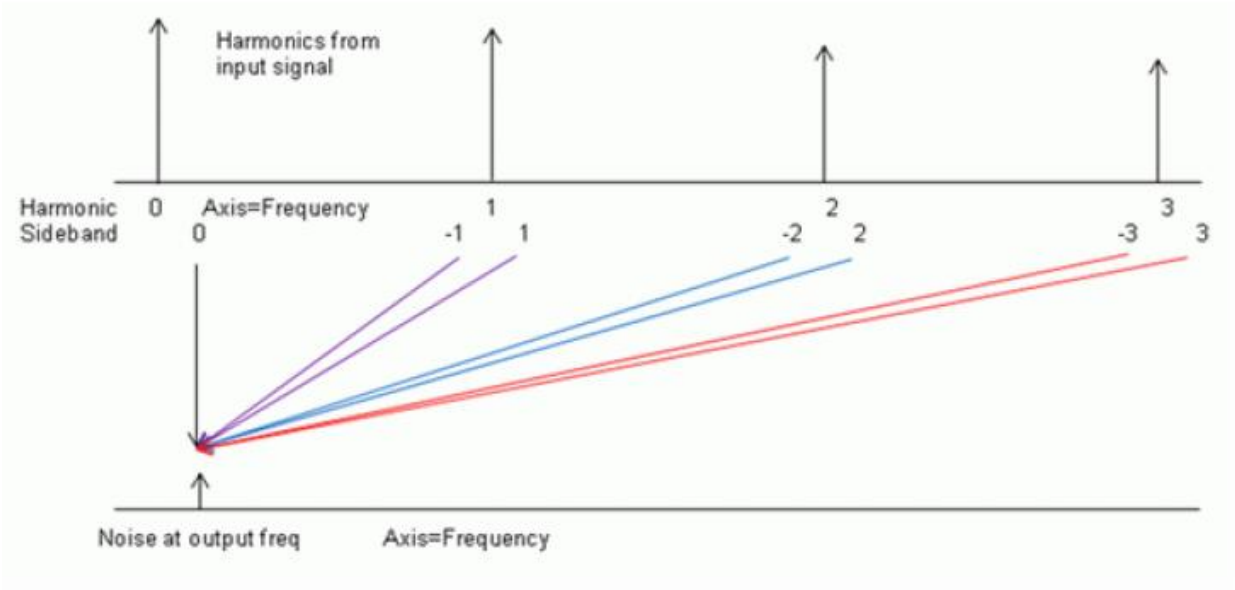


Figure 38: Noise Folding

Periodic noise calculates the average noise for the entire period. However, for a dynamic comparator, noise affects the circuit instantaneously at the decision point. So, a special type of pnoise analysis is used called sampled(jitter). You will be using this sampled(jitter) analysis to calculate the noise at an instant.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 29: In the ADE **Analysis** panel, double-click on **pss** analysis to view the setup.

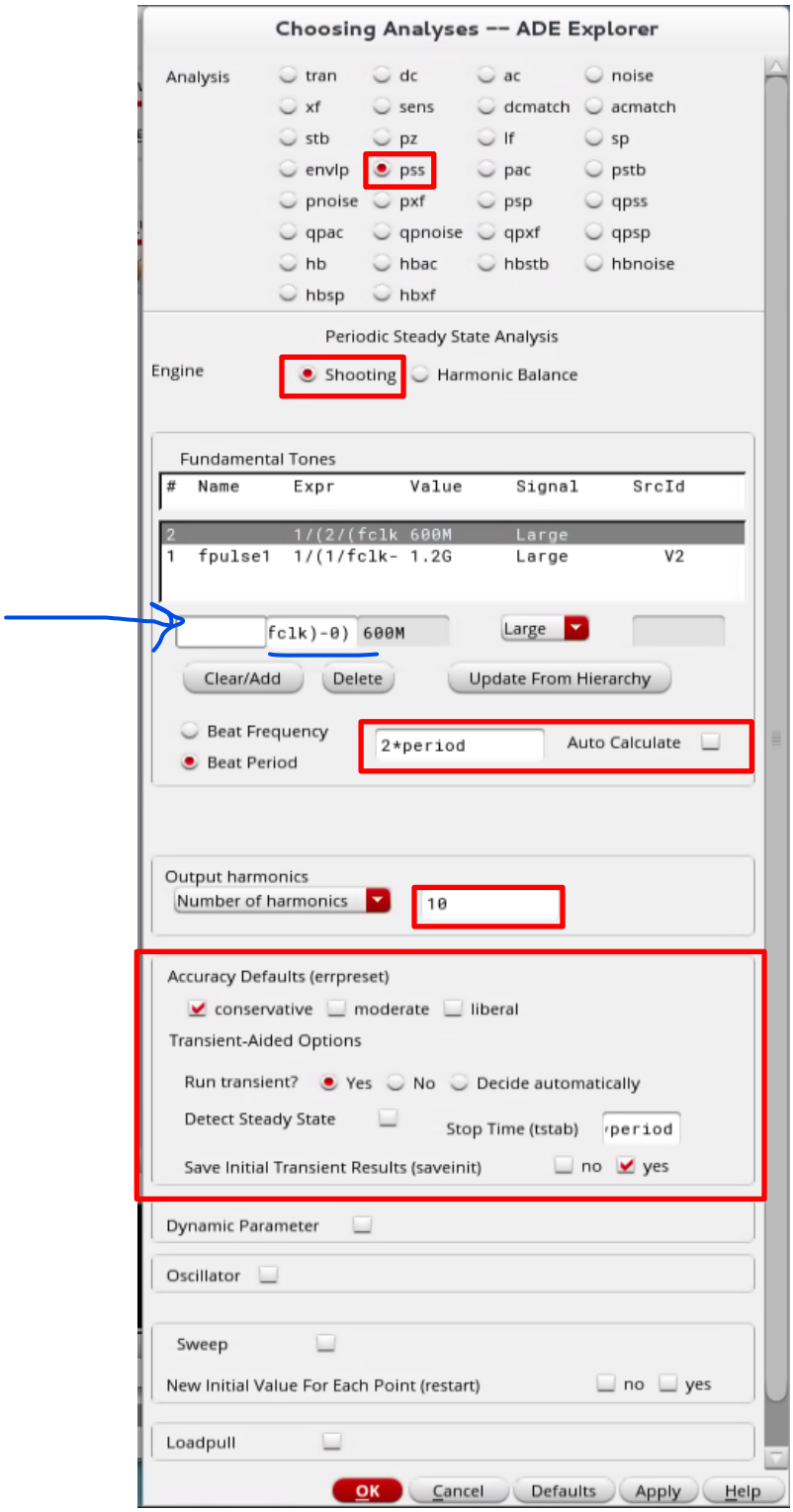


Figure 39: Choosing PSS Analysis – ADE Explorer

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Engine → Shooting

Fundamental Tones → There should be two fundamental tones to analyze. The first one **fpulse1** is the frequency of the clock, which is **1.2GHz**. The second frequency tone is inputted by you by changing the Beat Period to 2*period. 2*period is the same as fclk/2, or **600MHz**. This will be the value of your second fundamental tone.

Output harmonics → 10. The number of harmonics is used to determine the number of tones to use for the THD calculation. The remaining tones will be counted in the SNR calculation.

Accuracy Defaults (errpreset) → You use **conservative** because you want high-accuracy results. "**moderate**" is the general-purpose setting, and "**liberal**" is not recommended. We are using SpectreX with CX mode which is for high accuracy.

Run transient? → When you choose "**Yes**", the **Detect Steady State** option is enabled automatically and the simulation can process to pss without finishing the tstab simulation.

tstab → This is the user-defined stabilization time interval. You will be using the design variable "period" to define the time interval. A longer tstab allows the circuit to get closer to steady-state and thus, converge better. You use the value of the Beat Period as your tstab.

You also want to save your initial transient results; so, you select "**Yes**" for this part.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 30: Click on **Options** in the **Choosing Analysis** window to load the **Periodic Steady State Options** window. Click **Accuracy** to view the accuracy setup and verify that it looks like Figure 40. When you are done, click **Cancel** to exit the window.

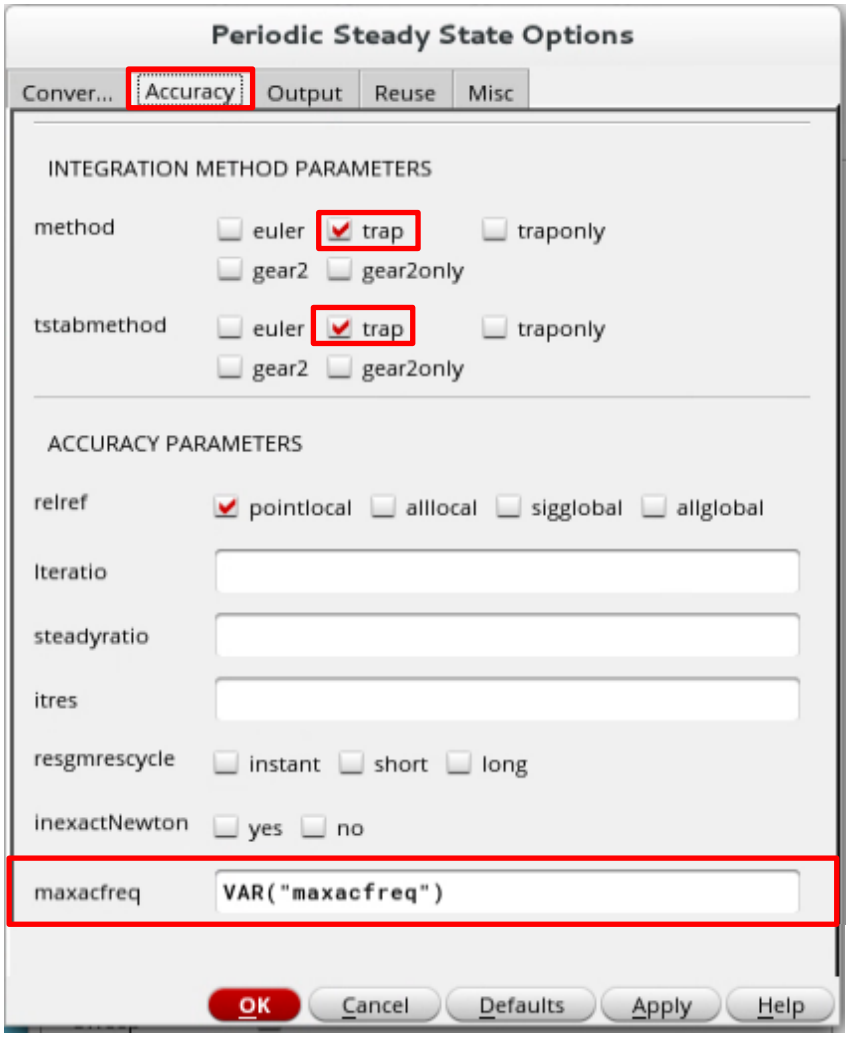
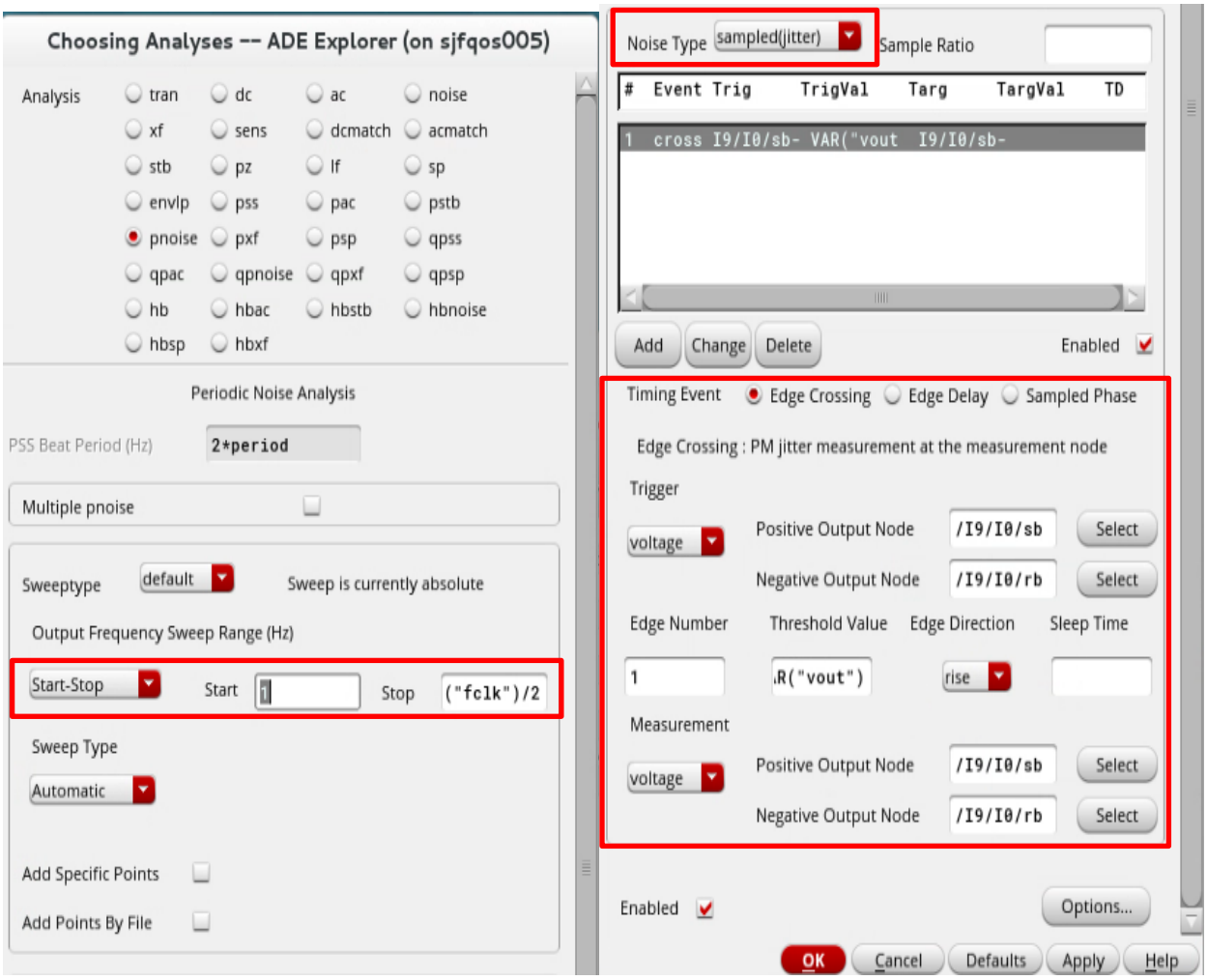


Figure 40: PSS Options → Accuracy Window

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 31: In the ADE **Analysis** panel, double-click on **pnoise** analysis to view the setup.



Top *Bottom*

Figure 41: Choosing pnoise Analysis – ADE Explorer Window

The pnoise analysis setup includes:

The **Output Frequency Sweep Range (Hz)** field, set **Start →1** and **Stop →VAR("fclk/2")**.

The **Sidebands →Method →full spectrum** is used because this method is especially useful for circuits where aliasing occurs through very high harmonics of the clock. The run-time advantages of using **full spectrum** are large with no loss in accuracy of the result. Using the full spectrum method setting allows pnoise to calculate the noise translations through the full spectrum of the pss analysis.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Trigger → voltage

Positive Output Node → /I9/I0/sb

Negative Output Node → /I9/I0/rb

Edge Number → 1

Threshold → VAR ("vout")

Measurement → voltage

Positive Output Node → /I9/I0/sb

Negative Output Node → /I9/I0/rb

In this case, the noise is measured directly at the output nodes of the StrongArm latch since the comparator includes an additional RS latch, which is connected to the output of the StrongArm latch. The RS latch holds the output steady, while the StrongArm latch is being reset and making decisions. However, the RS latch also affects the noise simulation; therefore, the StrongArm latch outputs are directly probed for noise analysis. The noise is calculated based on the Nyquist rate of the ADC (fsample/2).

In this design, the sample noise at 100ps will be calculated. The value 100ps is derived from the Shooting Newton analysis. In the testbench, the designer sets a threshold level and it is used to determine the time at which to perform noise analysis. The threshold level is called vout.

ACTION 32: Click **OK** in the **Choosing Analyses** form when you are finished examining the pnoise analysis setup.

ACTION 33: When ready to perform the simulations, select the ("**Run Simulation**") button.

Following the standard PSS analysis to determine the steady state, a pnoise analysis will be performed across the specified sweep frequency range.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 34: After the simulation is complete, the ViVA window will appear as Figure 42.

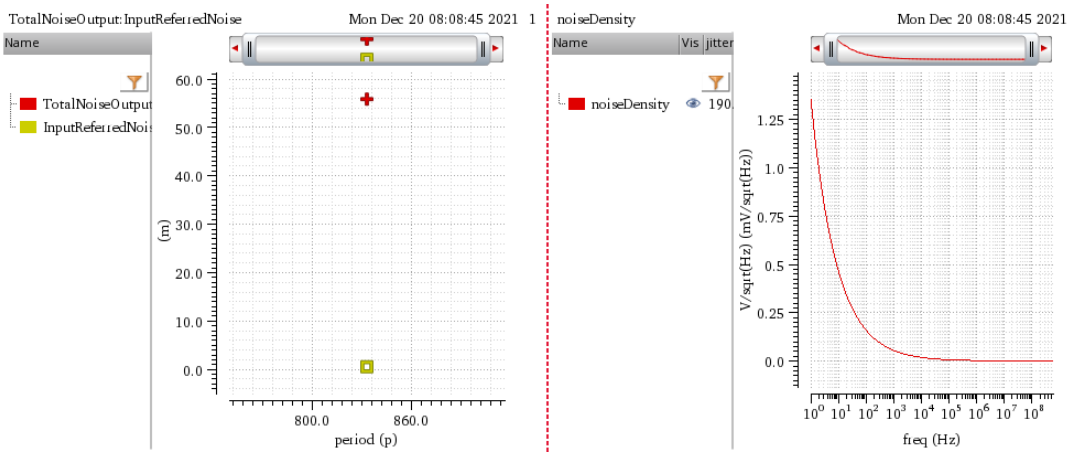


Figure 42: ViVA Window Result after Simulation

ACTION 35: In ViVA, click on the **Card** view option and select **Subwindow 2** from the pull-down menu. This is done so that only the analog output waveform is visible in the window, as shown in Figure 44.

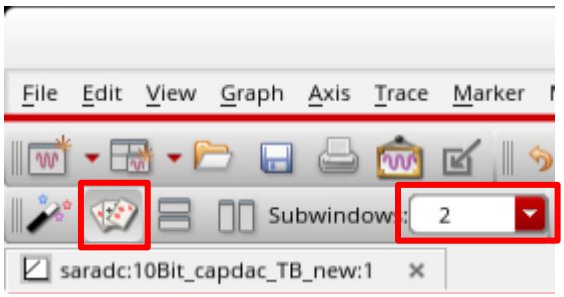


Figure 43: Card View with Subwindow 2

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

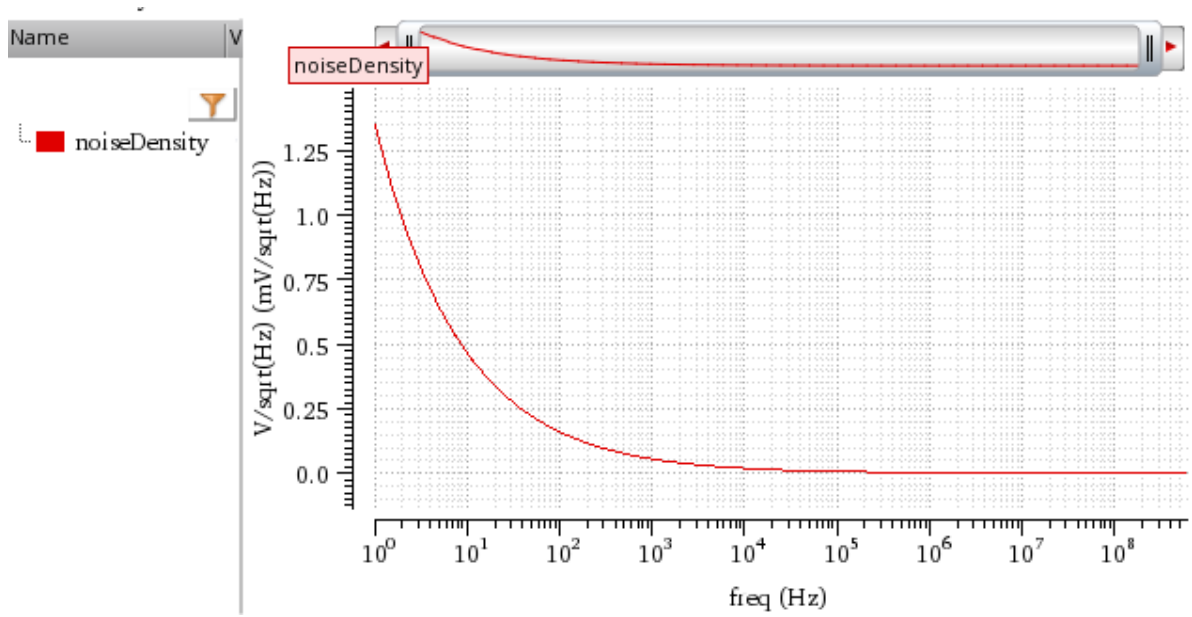


Figure 44: Total Output Noise

After the simulation completes, you should see the output noise vs. frequency, shown in Figure 44. The integrated total noise is calculated by the root-sum-square method. Noise is not correlated; so, you do not sum up the noise voltage. Instead, you sum up the noise power. The following equation shows how to get to the total integrated noise from the noise voltage:

Important

$$Total\ Integrated\ Noise = \sqrt{\int_1^{fnyquist} (noise\ voltage)^2}$$

You convert the noise voltage to noise power by:

1. Squaring the noise voltage, $(noise\ voltage)^2$, to get the noise power
2. Integrating the noise power from 1 Hz to fnyquist, $\int_1^{fnyquist} (noise\ voltage)^2$
3. Square root the results, $\sqrt{\int_1^{fnyquist} (noise\ voltage)^2}$

The final square root shown in step 3 converts the noise power back to noise voltage.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The integrated noise at the output is referred to the input by dividing the output noise by the gain of the circuit. The gain is calculated from the design variables, vout/vin. The IRN of the dynamic comparator is **~549µV$_{rms}$**.

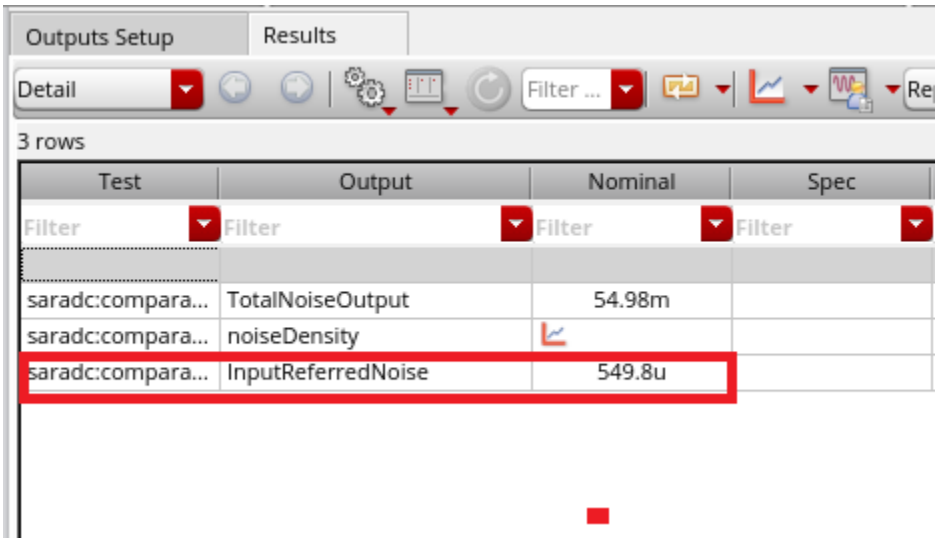


Figure 45: Output Results from Simulation

In the previous lab, you calculated the IRN noise of the dynamic comparator to be around 536µV$_{rms}$ and 528µV$_{rms}$. Comparing these results to the Transient noise analysis shows that the PSS noise analysis gives good noise results, for a fraction of the time it takes the transient noise analysis to complete. This is something for designers to keep in mind when they are characterizing their ADCs

ACTION 36: Close the ADE window by selecting **Session > Quit**. Select **No** if asked to save any changes.

Lab 3-4: Dynamic Comparator Offset Voltage Characterization

It is difficult to measure the offset voltage of a dynamic comparator since it does not have a quiescent steady state. To measure the offset voltage, you will apply a voltage staircase to the input of the dynamic comparator and latch the comparator at each step of the ramp. The input voltage level that causes the output to change is recorded. Then, the ramp direction is reversed, and the process is repeated. The offset voltage is the average of the two measurements.

To characterize the offset voltage of the comparator, the effect of global, wafer-to-wafer process variation and local, device-to-device process variation needs to be considered. You will use ADE Assembler to perform Monte Carlo simulation of the circuit.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

One important consideration is which sampling method to use when performing Monte Carlo analysis. In this lab, you will use the Low Discrepancy Sampling (LDS) method. Compared to traditional Random Sampling, LDS requires fewer Monte Carlo iterations for the statistical results to converge, therefore, making it more efficient. Compared to Latin Hypercube Sampling (LHS), LDS converges just as fast; however, LDS can be used with Monte Carlo “auto-stop”, while LHS cannot. Monte Carlo “auto-stop” is the ability to stop a Monte Carlo simulation when the results have reached a user-defined level of confidence. This is a very important tool for reducing Monte Carlo simulation time.

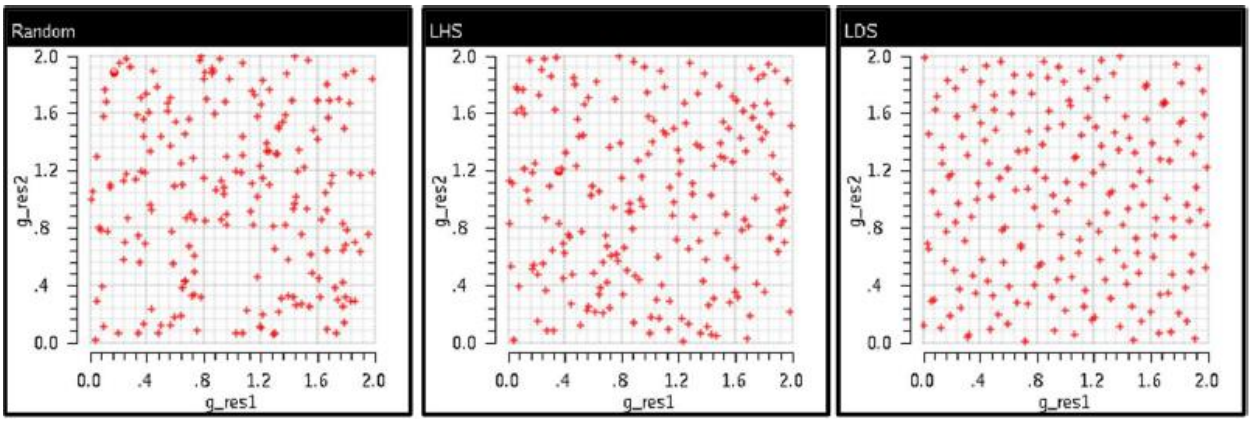


Figure 46: Distribution of Random Sampling, LHS, and LDS

Shown above is the distribution of the value of two statistical variables for each of the standard sampling methods. For example, these variables could represent the values of mobility and gate oxide used for Monte Carlo analysis. From the figure, you can see that for Random Sampling, the values are less uniform, and several samples have values close to each other. To fully explore the sample space, Random Sampling requires more iterations of Monte Carlo analysis. Using either LHS or LDS improves the uniformity of the samples, which in turn reduces the number of iterations.

One final comment–all these methods are intended for low-sigma (3σ) yield estimation. ADE Assembler provides alternative sampling methods (high-sigma yield estimation methods) when the number of failures is lower. For example, for 6σ yield estimation, the high-sigma yield estimation method can reduce the number of samples required from 10M+ for traditional Monte Carlo sampling to a few hundred samples without compromising accuracy.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The flow will be to perform the offset voltage simulation once using ADE Explorer to verify the setup and the measurement results. Then, with the same maestro cell view, an ADE Assembler instance will be created for analysis of the effect of process, operating, and environmental conditions on circuit performance.

A slow ramp, staircase, is applied as the input of the comparator, and the comparator is latched for step of the staircase. You will record the staircase value when the comparator trips. The offset voltage is measured twice (once for the ramp up and once for the ramp down) and the result is averaged. Figure 47 illustrates both the input and latch out waveforms.

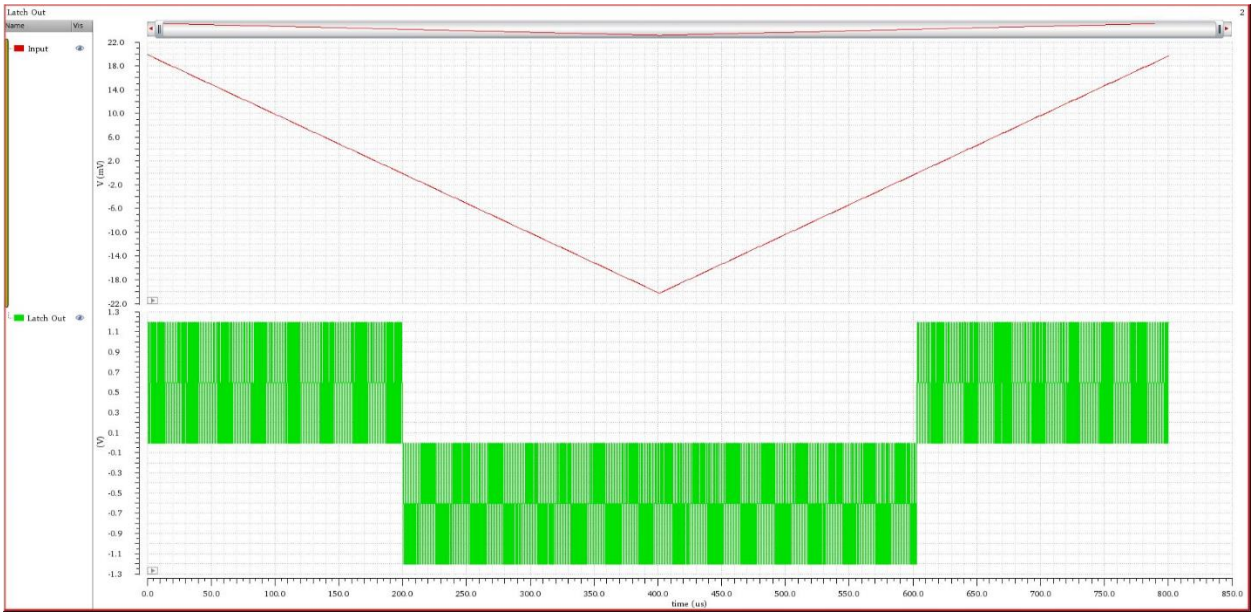


Figure 47: ViVA Window of Input and Latch Out Waveforms

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 37: In the Library Manager, open **saradc > Everything > comparator_offset_TB_new**. Double-click on **schematic** to open the schematic.

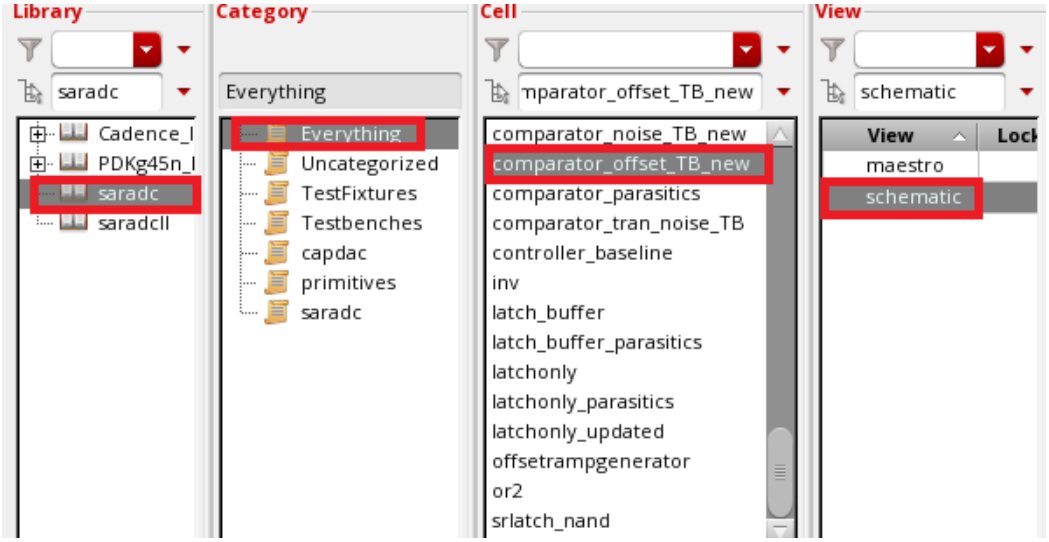


Figure 48: Library Manager

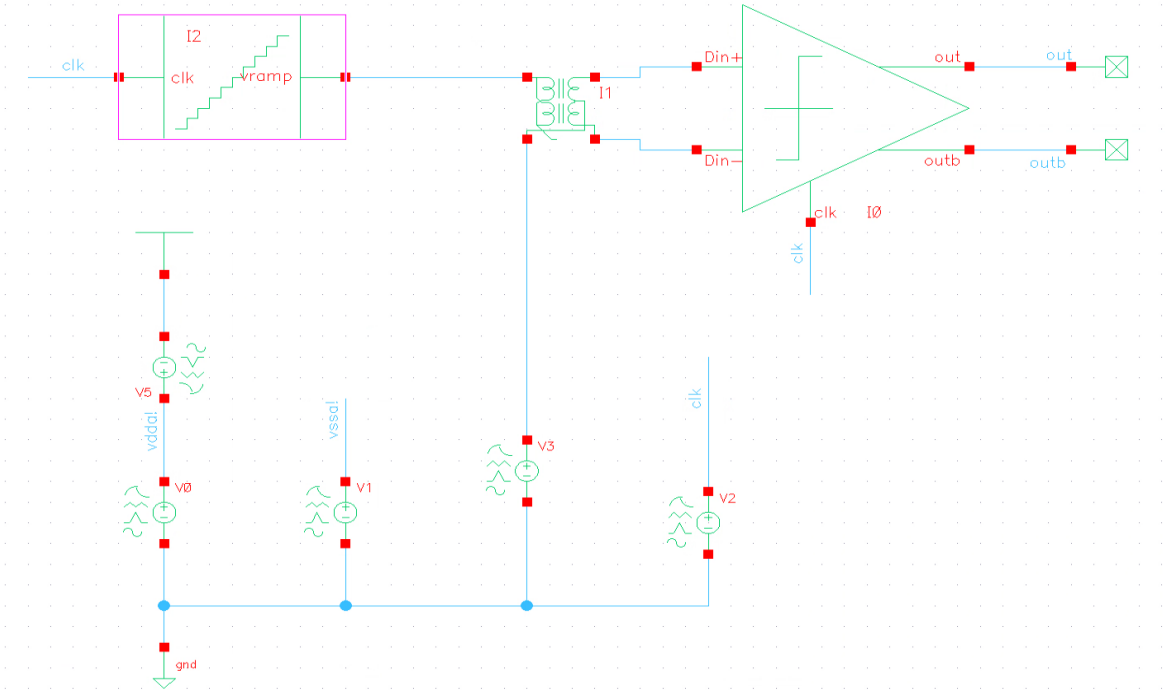


Figure 49: Comparator Offset Testbench Schematic

Note: To measure the offset voltage, a staircase will be applied. The staircase generator is implemented using the Verilog A behavioral modeling language.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 38: LMB-select the staircase generator instance **I2**. Then, do **RMB > Properties** to display the **Edit Object Properties** form as shown in Figure 50. In the **CDF Parameters of view** field, select the **veriloga** view.

Figure 50: Staircase Generator Edit Object Properties

The ramp generator is driven by the same clock signal as the comparator. When the clock goes high, the ramp is incremented after some delay time, **td**. The comparator is then latched when the ramp generator is updated to prevent changing the input level and incorrectly effecting the measurement. The design variable **tr** defines the rise time of the ramp step. You need to set the maximum expected value of the offset, **maxin**, and the resolution, **resolution**, of the measurement. These two parameters are used to calculate the number of steps in the ramp.

For the offset ramp generator, all these values are defined with design variables and the values are controlled by ADE. Using design variables allows you to configure and parameterize the testbench, the analysis, and the outputs, and control the measurement from ADE.

ACTION 39: Click **Cancel** when done checking the CDF parameters.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 40: Launch ADE Explorer. From the Schematic Editor, select **Launch > ADE Explorer**.

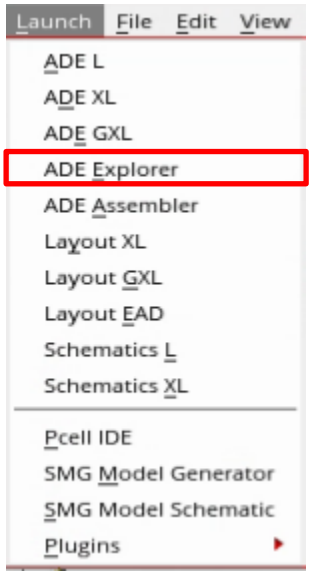


Figure 51: Launch ADE Explorer

ACTION 41: In **Launch ADE Explorer** window, select **Open Existing View**. Then, click **OK**.

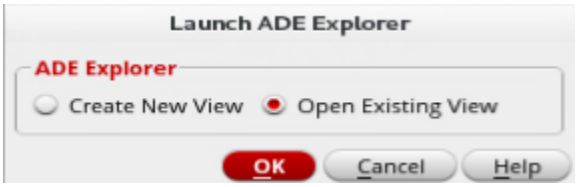


Figure 52: Open Existing View

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 42: In the **Open ADE Explorer View** window, verify that the view is set to **maestro**. Then, click **OK**.

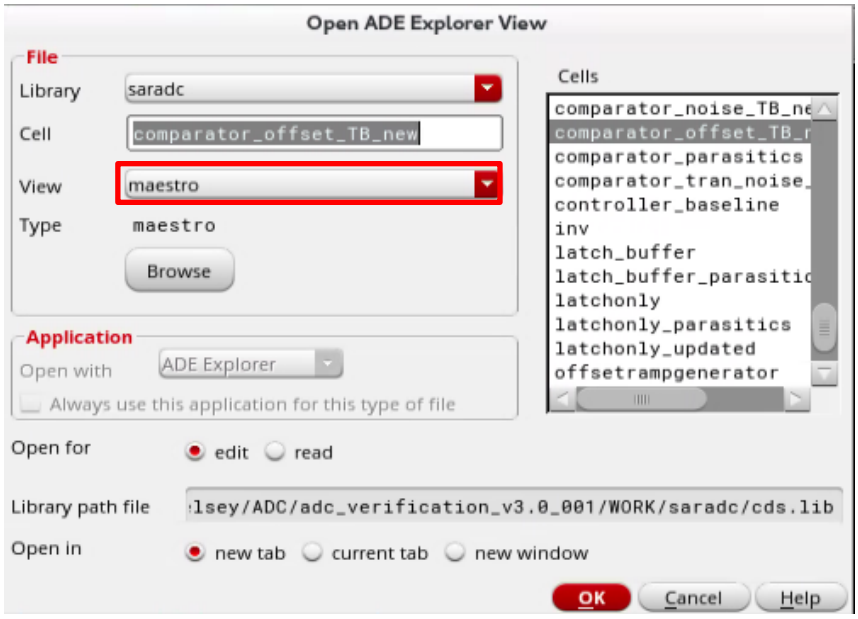


Figure 53: ADE Explorer View

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 43: Once the **maestro** tab is loaded, it should look like Figure 54.

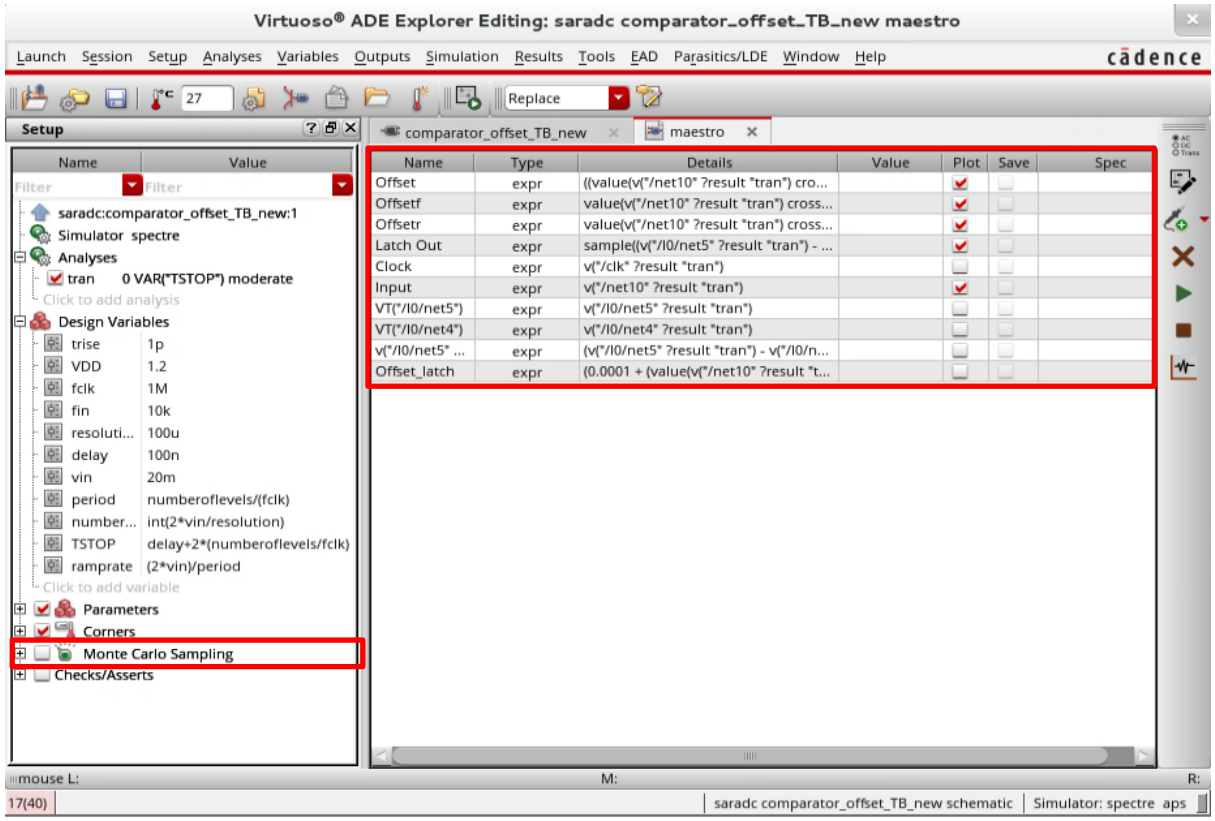


Figure 54: ADE Explorer Window

Note: Make sure that the **Monte Carlo Sampling** box is left unchecked. You will run a Monte Carlo simulation later in this lab, but not for this part.

You will now review the contents of this setup, beginning with the contents of the **Design Variables** panel on the left.

Design Variables – This panel contains a list of the design variables used in the testbench schematic. The design variables here set the maximum input level (**vin**), the power supply voltage (**VDD**), the clock frequency (**fclk**), the resolution of the offset voltage measurement (**resolution**), and the delay from latching the comparator to updating the ramp (**delay**). In addition, several design variables are derived from the user inputs including the number of steps in the staircase ramp (**numberoflevels**), the period of the ramp steps (**period**), the slew rate of the ramp (**ramprate**), and the stop time for transient analysis (**TSTOP**).

Analyses – This panel contains the list of Spectre analyses to be performed. The contents of this setup will be reviewed in the next step.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Outputs – These are lists of outputs to be measured or plotted after all analyses listed above have been completed.

ACTION 44: In the ADE Explorer window, go to the **Analysis** panel and double-click on the **tran** analysis so that you can explore the setup of this analysis.

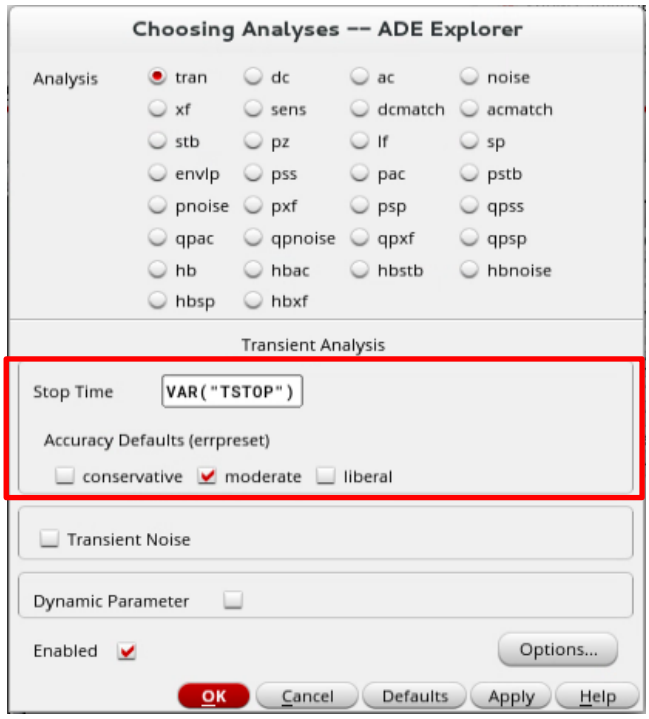


Figure 55: Choosing TRAN Analysis – ADE Explorer

Note: For this measurement, the transient stop time is derived from the measurement requirements; that is, the range and the resolution of the measurement.

ACTION 45: Click **OK** to exit the **Choosing Analyses** form.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The **Outputs** panel contains existing waveform plotting and measurement setup for plotting and calculating the comparator offset voltage.

Verify that the **Offset**, **Input**, and **Latch Out** signal, as well as the measurements, are enabled in the **Plot** column.

Name	Type	Details	Plot	Save	Spec
Offset	expr	((value(v("/net10" ?result "tran")...	✓		range -5m 5m
Offsetf	expr	value(v("/net10" ?result "tran") ...	✓		
Offsetr	expr	value(v("/net10" ?result "tran") ...	✓		
Latch Out	expr	sample((v("/I0/net5" ?result "tr...	✓		
Clock	expr	v("/clk" ?result "tran")			
Input	expr	v("/net10" ?result "tran")	✓		
VT("/I0/net5")	expr	v("/I0/net5" ?result "tran")			
VT("/I0/net4")	expr	v("/I0/net4" ?result "tran")			
v("/I0/net5" ?result "tran")-v("/I0/net4" ?result "tran")	expr	(v("/I0/net5" ?result "tran") - v("/...			
Offset_latch	expr	(0.0001 + (value(v("/net10" ?res...			

Figure 56: Plot Input and Latch Out Signals

You are now ready to begin the simulation analysis.

ACTION 46: When ready to perform the simulations, select the ("**Run Simulation**") button.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The simulation will start by performing a transient analysis. The simulation time is about 2 minutes. When completed, the following waveforms should be plotted in the following ViVA window shown in Figure 57.

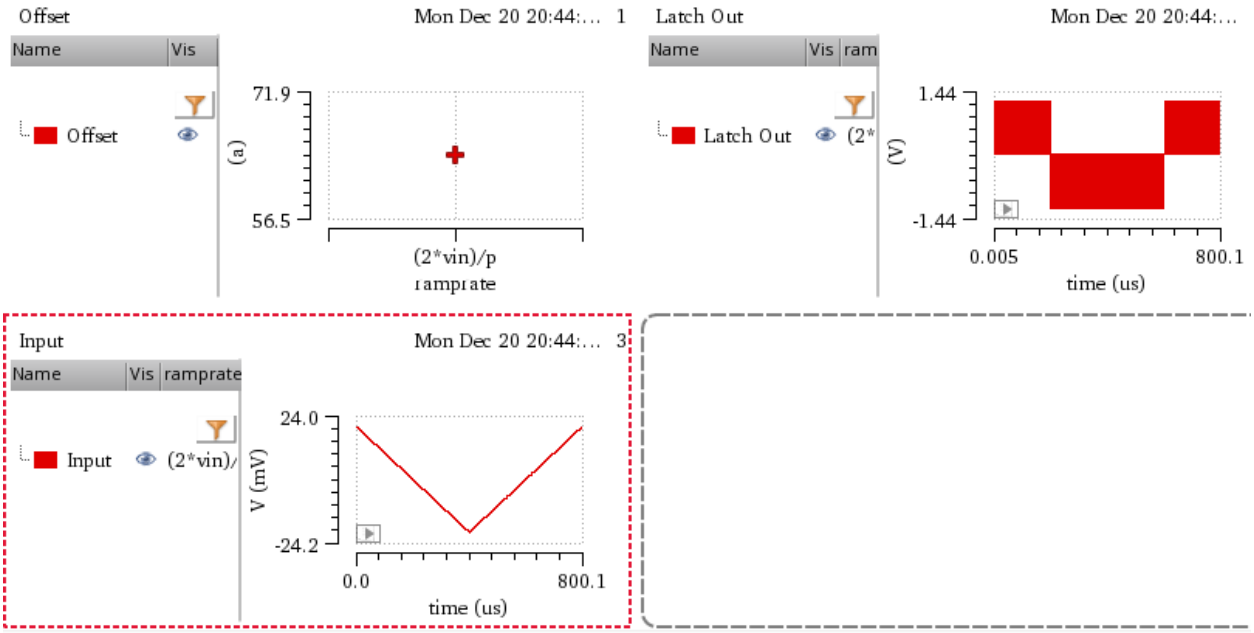


Figure 57: ViVA Window

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 47: To remove the **Offset** waveform in subwindow 1, RMB-click anywh subwindow 1 and select "**Delete**". To show only the two waveforms, click on the button. The end result should resemble Figure 59

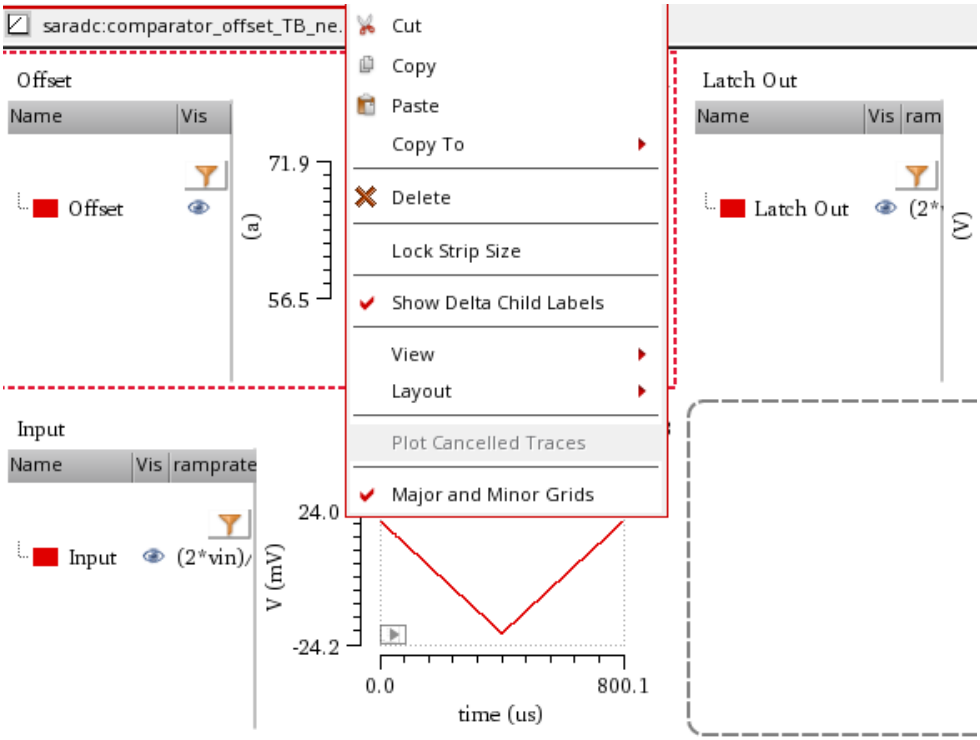


Figure 58: Delete Offset Waveform

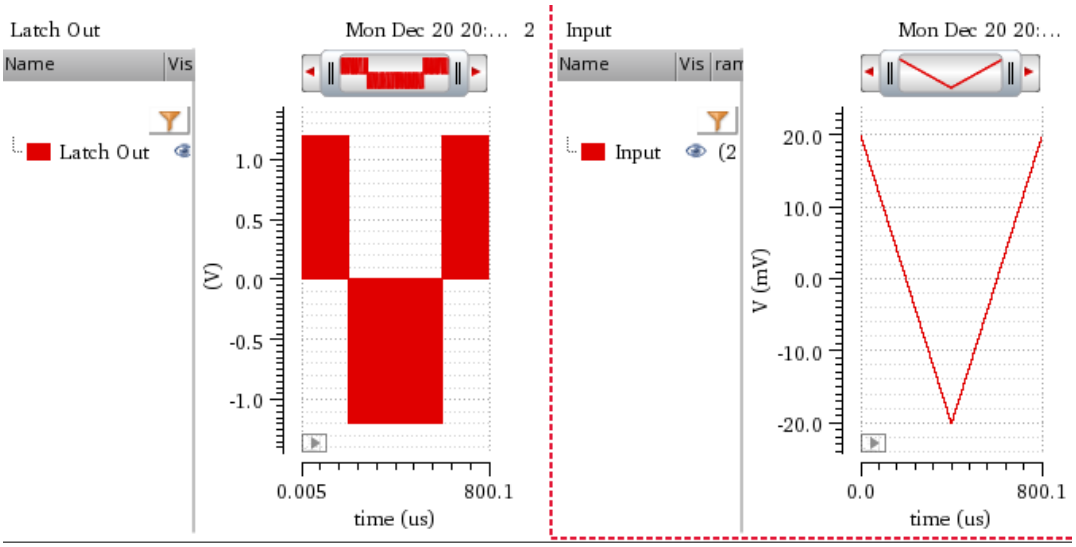


Figure 59: Latch Out and Input Signal Waveforms

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 48: Combine the two waveforms into one subwindow by dragging one waveform into the other subwindow. The two waveforms should be overlapping now. Delete the empty rectangle graph by following ACTION 47 again. To split the waveforms in the same subwindow, click on the icon to split the current strip.

ACTION 49: Figure 60 is a zoomed-in plot of the combined plot of the latch_out and input signal waveforms, where the comparator transitions from high-output level to the low-output level. This number is reported in the ADE window **Outputs** panel.

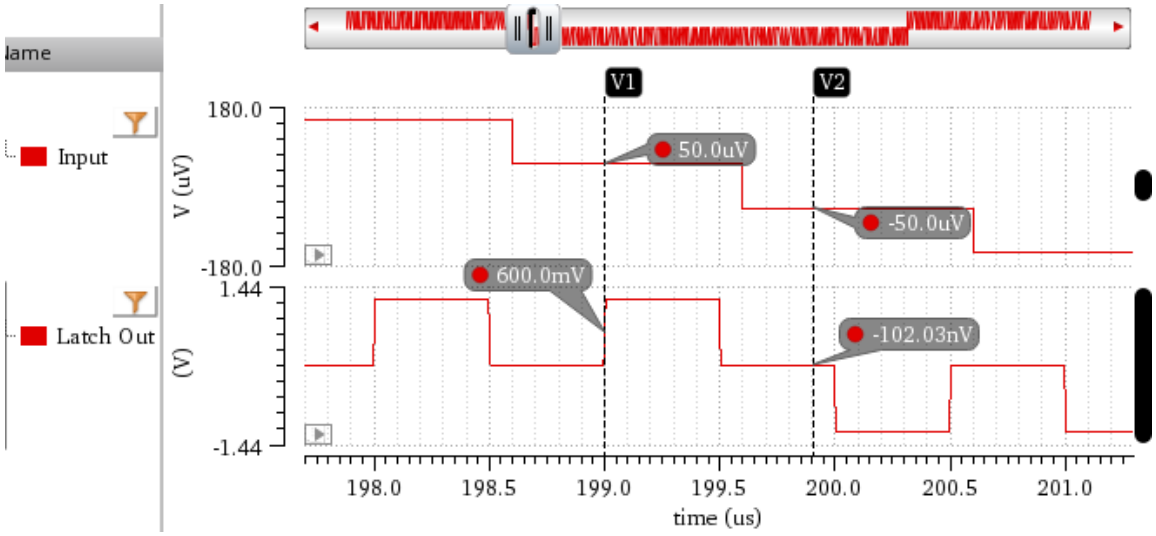


Figure 60: Zoomed-in View of Input and Output Signals

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 50: In the ADE Explorer window, click on the blue arrow to go into the ADE Assembler maestro cell view.

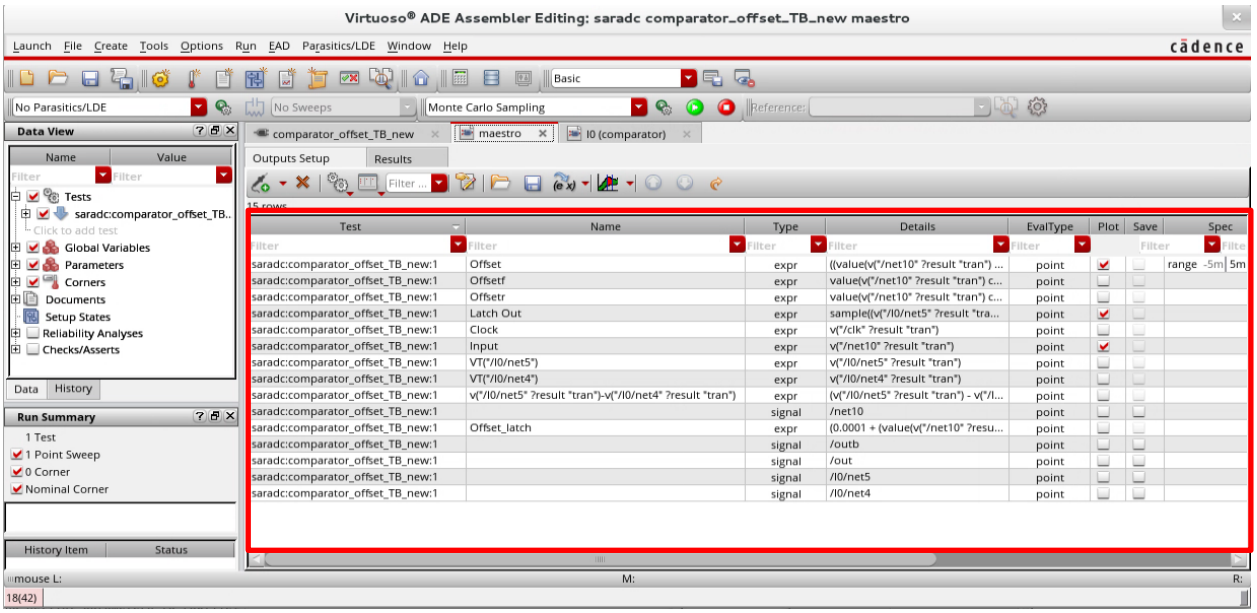


Figure 61: ADE Assembler Window

Since the Monte Carlo simulation time is quite long, you will load the results instead of performing a simulation. ADE Assembler can maintain a record of the previous 10 simulation runs, and these can be loaded to view. This feature allows designers to review the previous results and even compare the current results with previous results to track the progress of the design. If you want to run a simulation, select the ("**Run Simulation**") button.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 51: In the ADE Assembler Data View Assistant, LMB-click on the **History** tab.

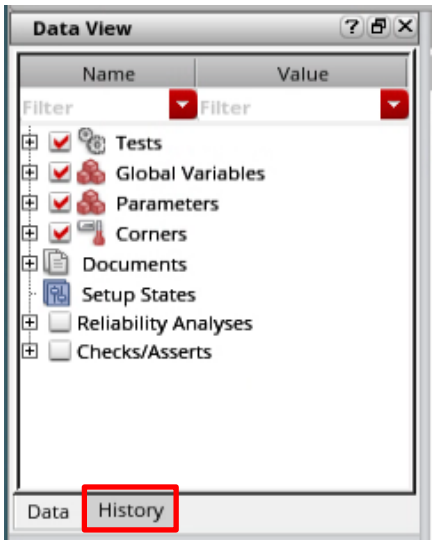


Figure 62: History Tab of Data View Assistant

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 52: In the **History** tab, RMB-click on **MonteCarlo.0** and select **Load Setup to Active** from the menu. Your maestro window should now look like Figure 64.

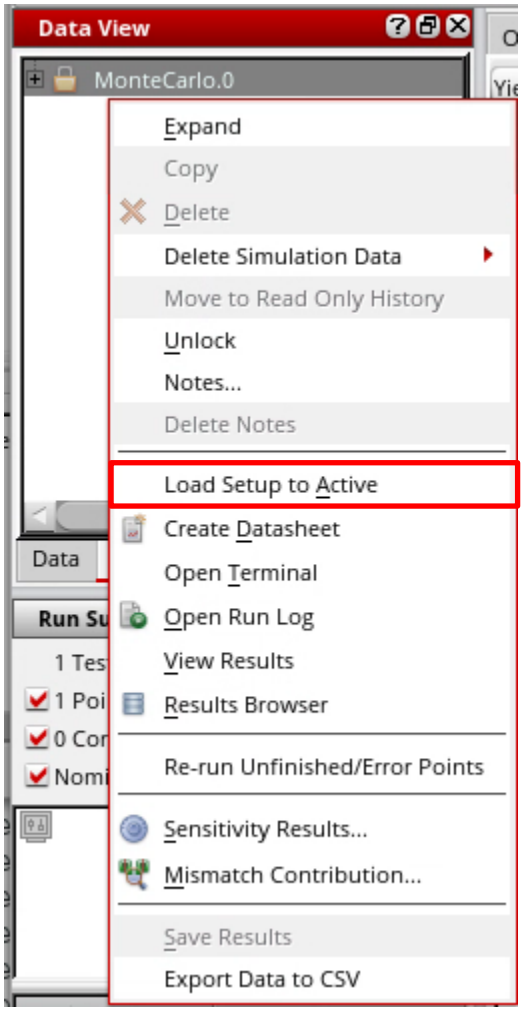


Figure 63: Load Setup to Active

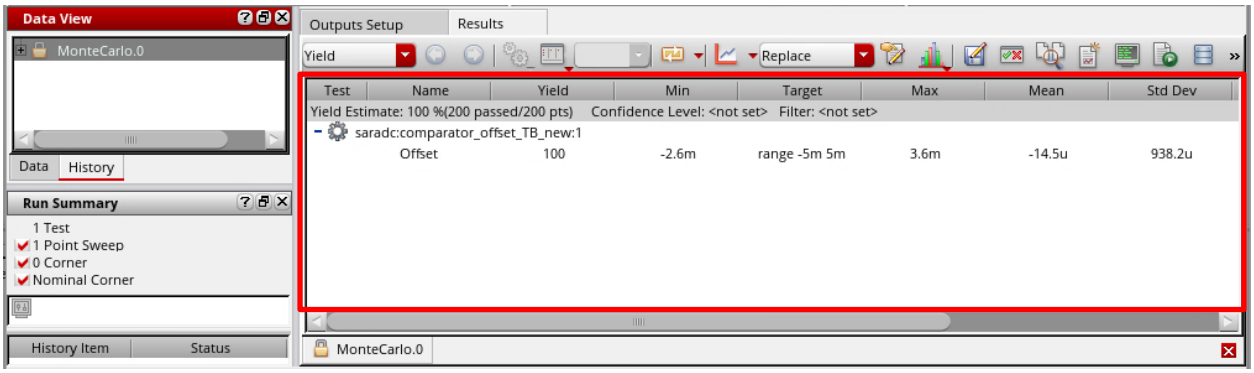


Figure 64: Maestro Window with Monte Carlo Analysis Loaded

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Note: ADE Assembler provides two models for exploring the results of statistical analysis. To take a quick look at specific hotspots, hover the mouse over the measurement and hold down the RMB. This will bring up the menu for plotting histograms and creating statistical worst-case corners.

ACTION 53: Select the Post Processing for Monte Carlo Analysis icon ( ) from the ADE Assembler Results window. Then, select the test results that you would like to see (**saradc:comparator_offset_TB_new:1**) and then choose the **Histogram** window.

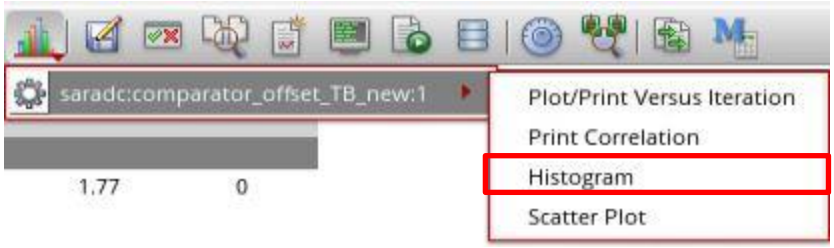


Figure 65: Plot Histogram – Monte Carlo

ACTION 54: The **Plot Histogram – MonteCarlo.0** form will open. Verify that it looks like Figure 66, and then click "**plot**".

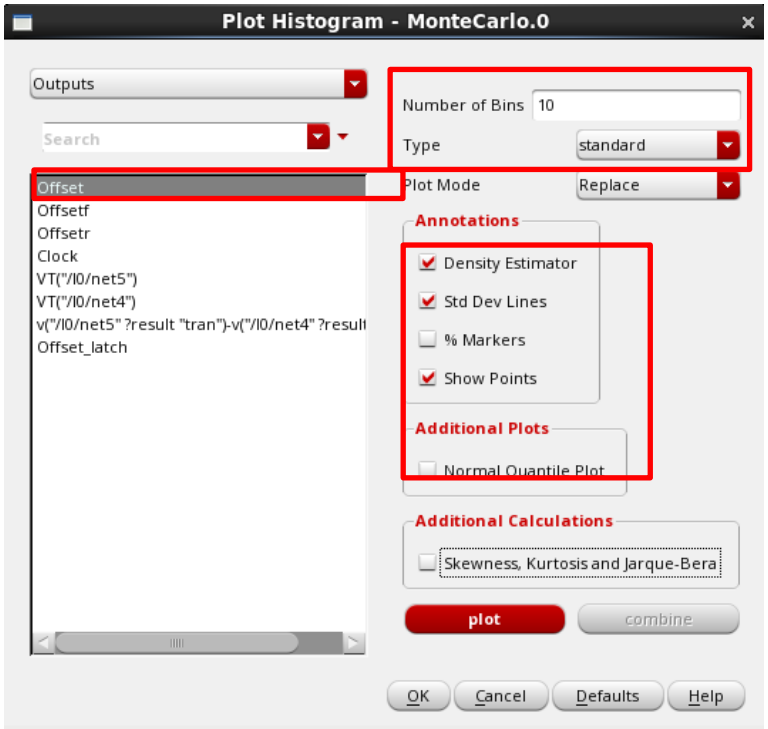


Figure 66: Plot Histogram – Monte Carlo.0 Form

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The offset voltage distribution will be plotted. As expected, the average value is near 0 and there is a deviation.

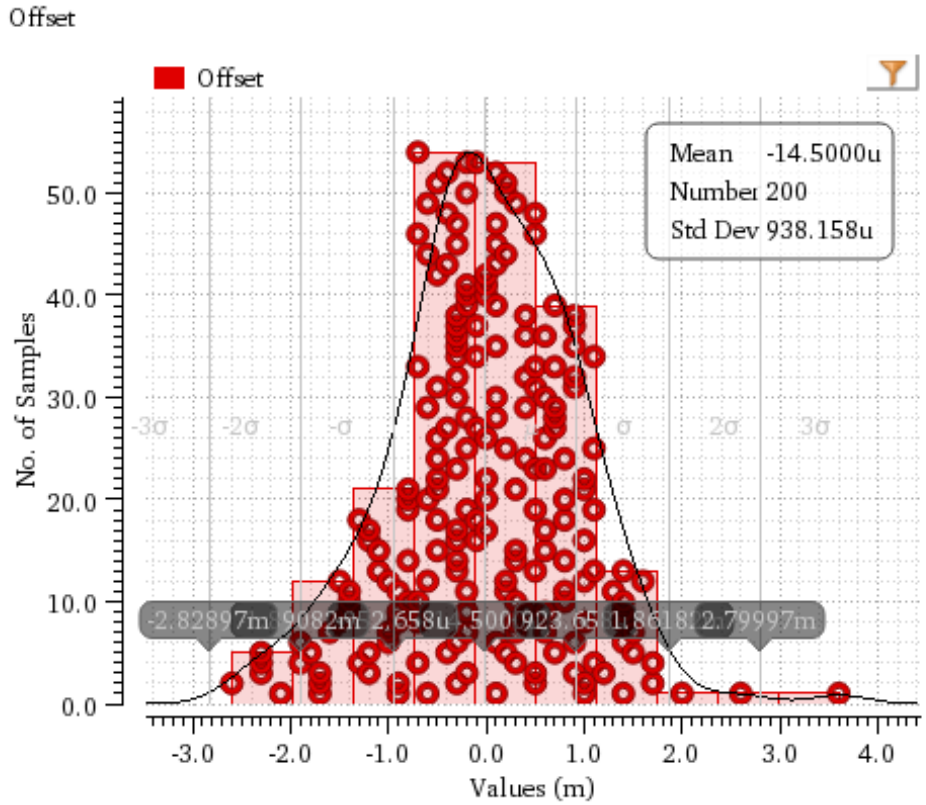


Figure 67: Histogram of Offset Voltage Distribution

The average value of the offset voltage is about 0V because the design is highly balanced. The sigma is **~938µV**. Next, you will analyze the results in more detail. First, you will re-plot the data to see if the offset voltage distribution is Gaussian or not. Then, you will look at the factors that contribute to the offset voltage and consider whether to try to improve it.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 55: Again, select the Post Processing for Monte Carlo Analysis icon from the ADE Assembler Results window. Then, select the test results that you would like to see (**saradc: comparator_offset_TB_new:1**) and choose the **Histogram** window. However, this time, also select the **Normal Quantile Plot** option under **Additional Plots** as shown in Figure 68.

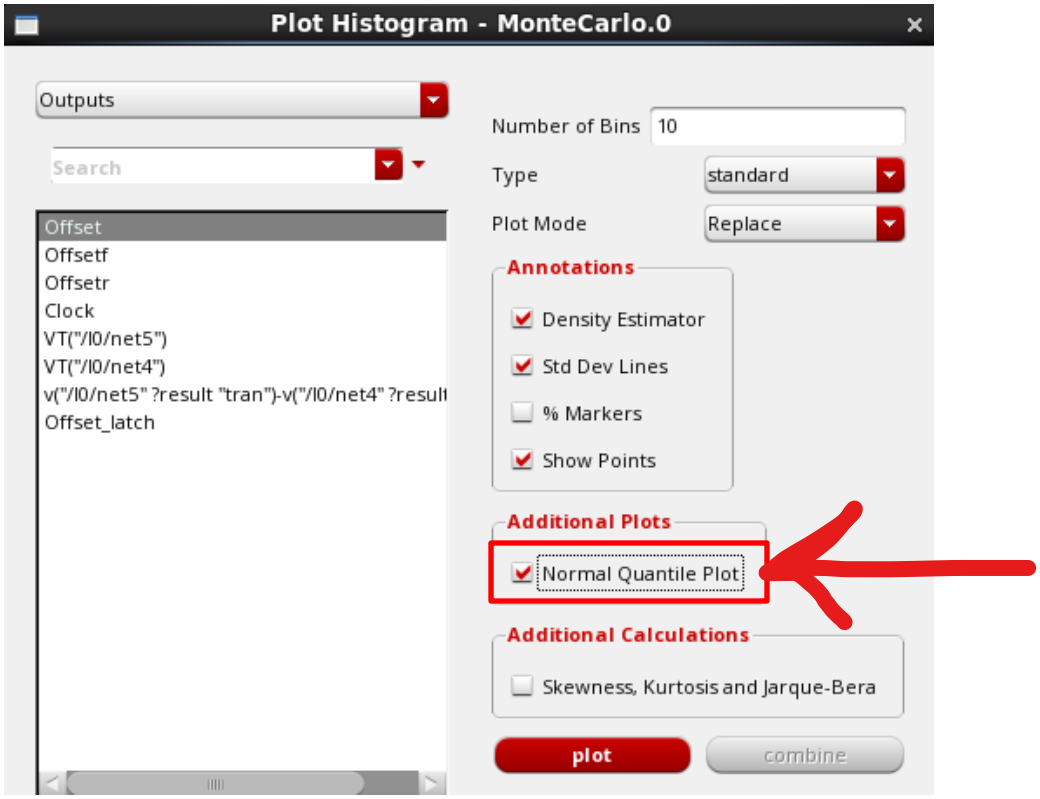


Figure 68: Plot Histogram - Monte Carlo with Normal Quantile Plot Selected

Note: Ignore the histogram and just look at the Normal Quantile plot. You can do this by selecting the "**Card**" ( ) view and making subwindow = 2.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

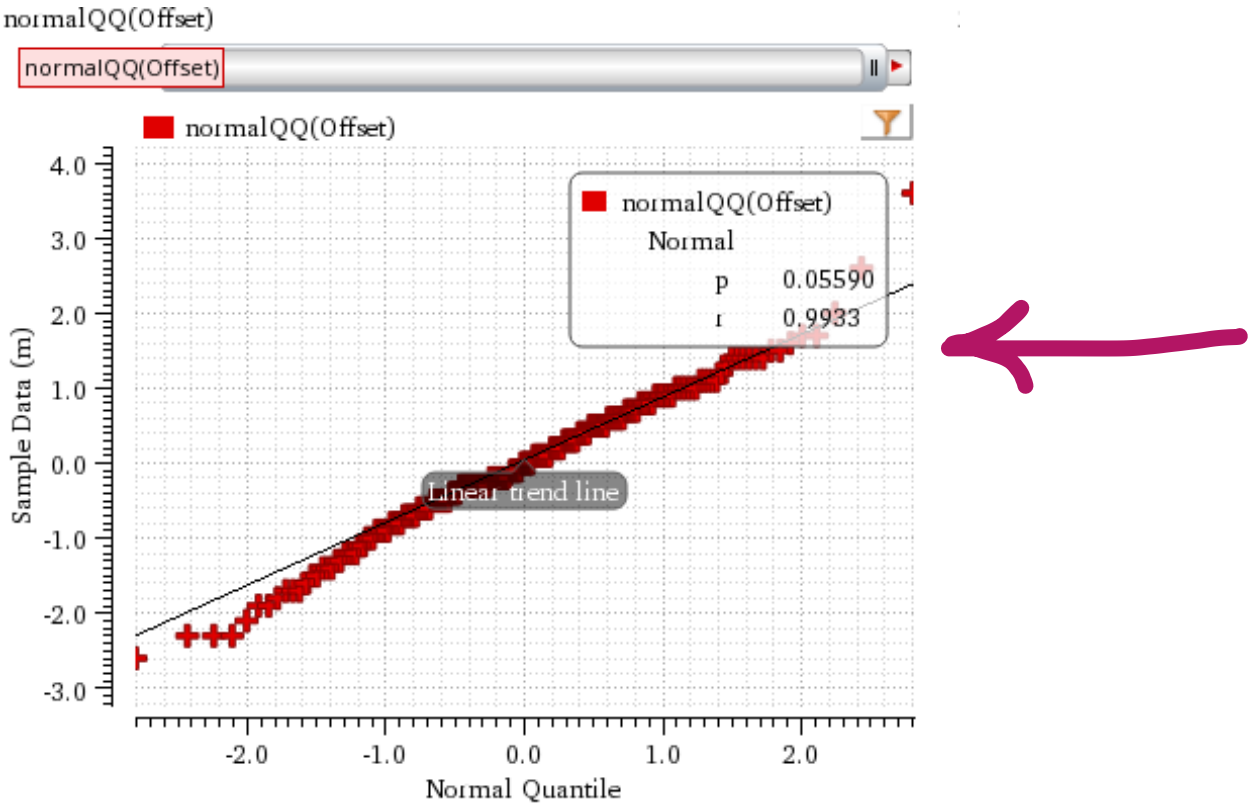


Figure 69: Normal Quantile Plot of Offset Voltage Distribution

A Normal Quantile plot is a tool used to evaluate if a distribution is Gaussian. The offset voltage data is compared to the Gaussian distribution, and the results are plotted. If the two distributions are the same, the plot will be a straight line and the offset voltage will be Gaussian in distribution.

The correlation coefficient shows that there is a good fit. Looking at the fit, there is some deviation from a straight line as sigma increases. If you want to explore the distribution at a high sigma, you can use ADE Assembler high-sigma yield estimation.

ACTION 56: Close the waveform plot and ADE windows by selecting **File > Close**. Select **No** if asked to save any changes.

Lab 3-5: Dynamic Comparator Characterization Across Corners

Now that the measurements for the key comparator parameters have been created, you will create the top-level test to use for verification. You will create a new top-level ADE Assembler for the comparator block, add the individual tests, and define the block specifications. Once the tests are defined, ADE can be used to verify the design.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 57: In the Library Manager, click on **saradc > Everything > comparator_new**. LMB-click on **maestro** to open the maestro cell view.

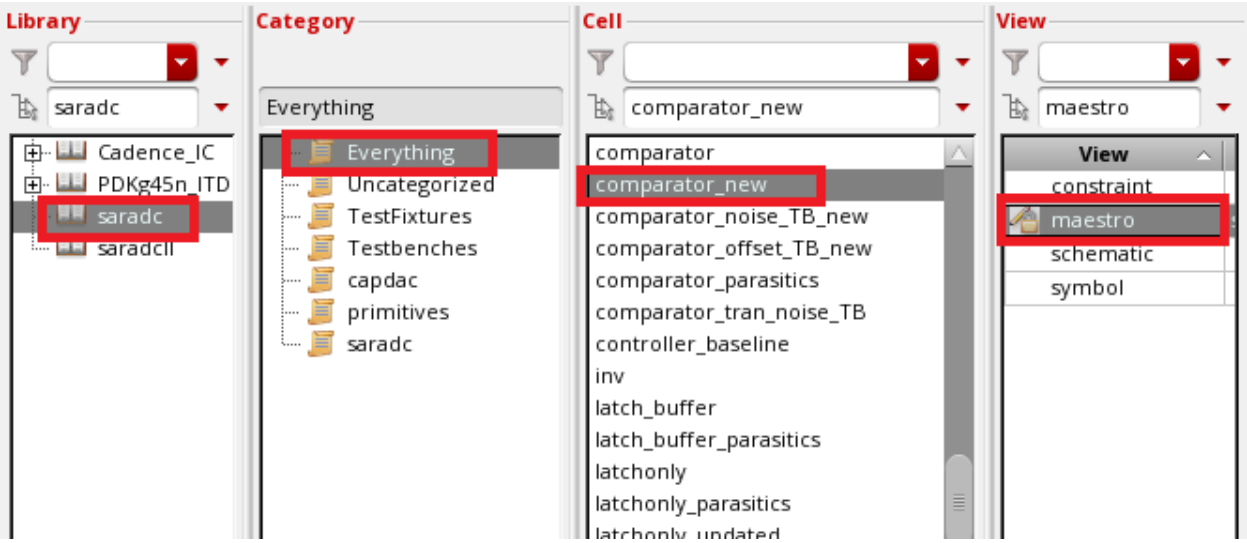


Figure 70: Library Manager

ADE Assembler allows you to create and manage in one place all the tests required to characterize the comparator. The maestro, shown below, contains all the testbenches that you have previously created: **comparator_noise_new** and **comparator_offset_new**. You will use the ADE Assembler maestro as a cockpit to perform advanced analysis.

For verification, you need to be able to manage all the tests together. Once the tests are defined, you can analyze the performance across variations in environment, operating condition, and process variation. The results are then checked for compliance with the design specification, and the results are reported. The results can then be compiled into a datasheet for use in design reviews and documentation.

ACTION 58: Click on the **History** tab in the ADE Assembler Data View Assistant.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

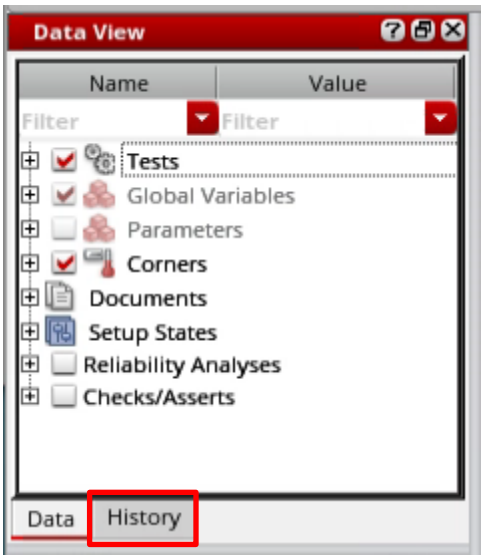


Figure 71: History Tab of Data View Assistant

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 59: Hover the mouse over the **Corner Analysis** history. Then, RMB to open the menu and select **Load Setup to Active**.

The results will be loaded into the ADE Assembler **Results** panel, as shown in Figure 72.

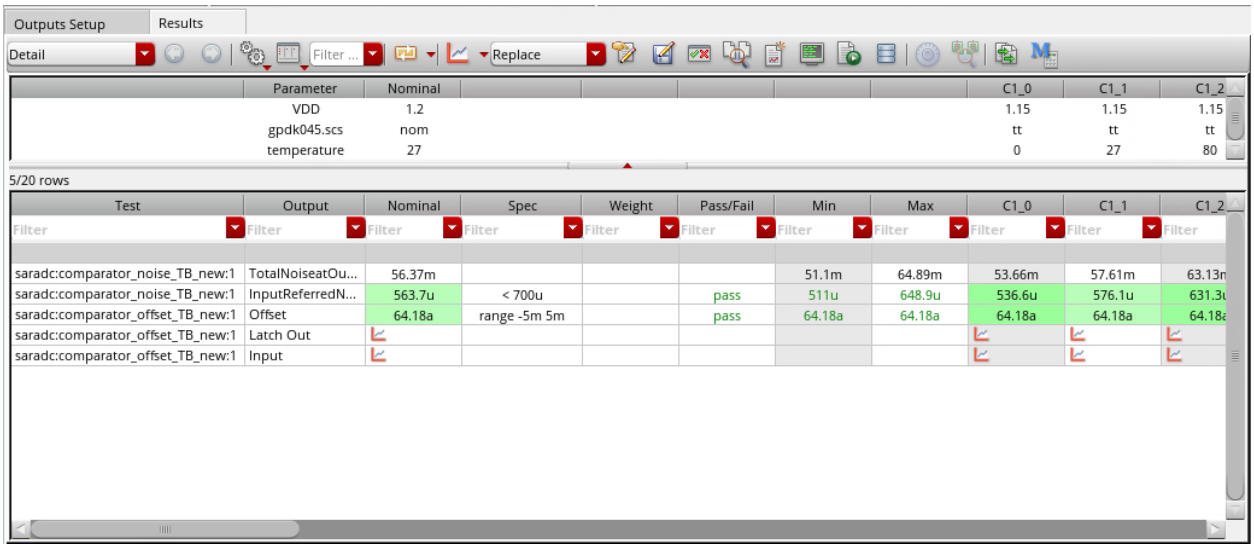


Figure 72: Load Setup to Active

ACTION 60: In the ADE Assembler Data View Assistant, open the **Data** tab and click on the "**+**" sign to open the corners.

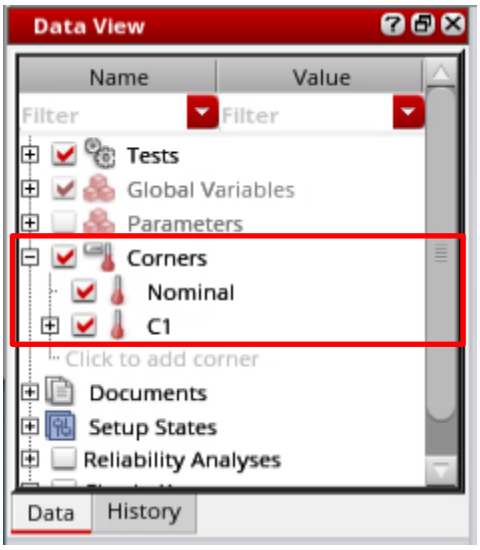


Figure 73: Corners in the Data View Assistant

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 61: Open the **Corners Setup** form by doing RMB on **C1** to open the menu and then selecting **Open Corners Setup**.

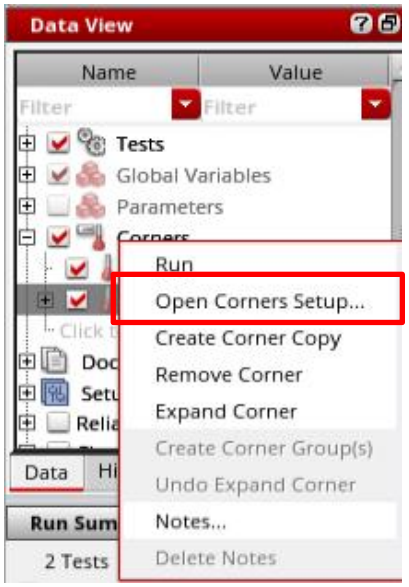


Figure 74: Open Corners Setup

ACTION 62: Review the **Corners Setup** form and click **OK** when done.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

In this case, 45 corner conditions have been identified: three VDD corners, three Temperature corners, and five process corners.

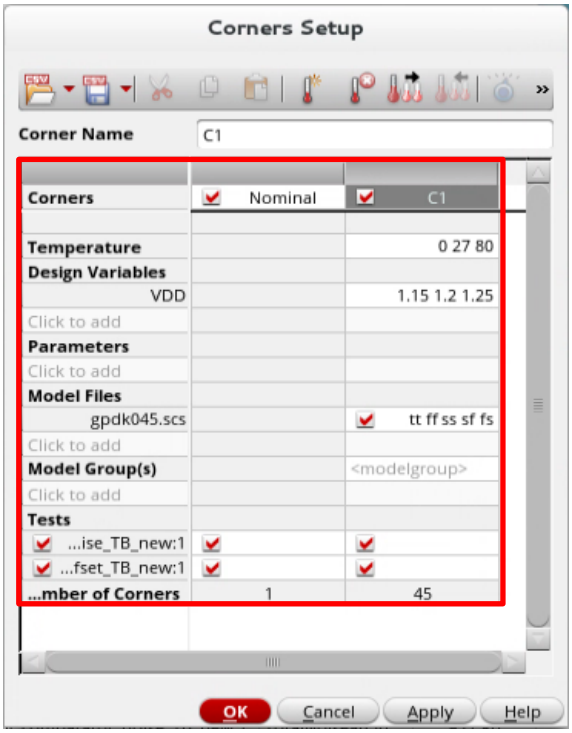


Figure 75: Corners Setup

Note: You can skip to ACTION 71, or you can practice setting corner conditions in ADE Assembler if this is new to you.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

3.4.1 [Optional] Setting Up Corner Conditions

This section is optional and is designed to help you practice creating corners. If you create your own corner, the corner number will be C2, not C1. You can delete either one when this optional lab section is complete.

ACTION 63: In the ADE Assembler Data View Assistant, click on **Click to add corner** to open the **Corners Setup** form.

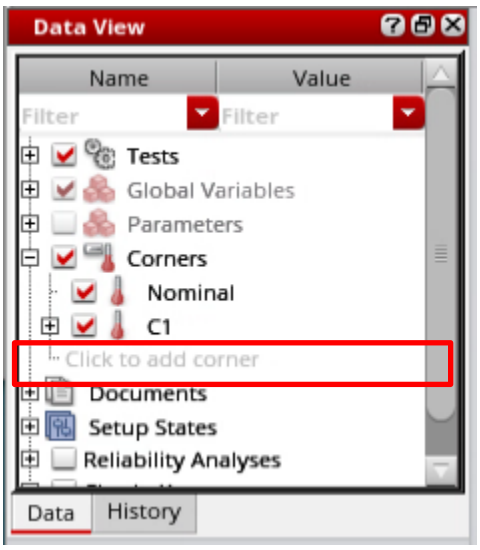
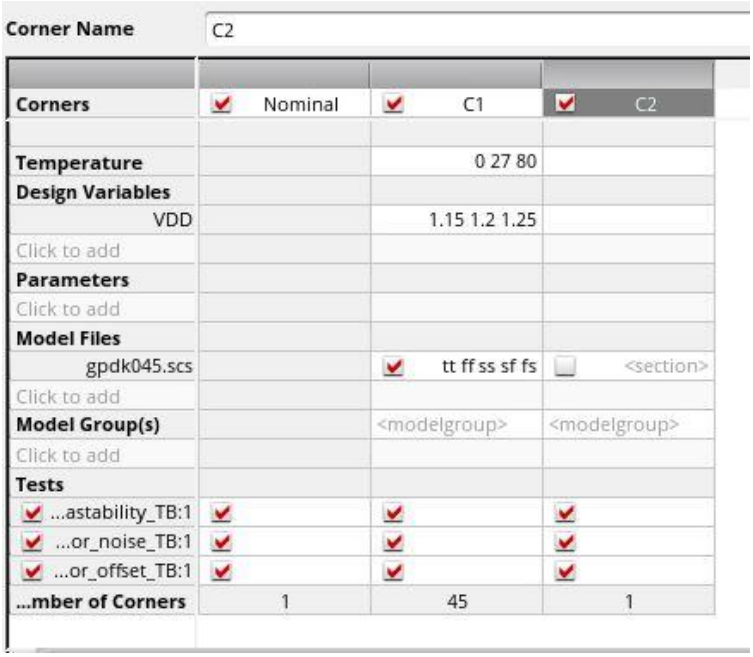


Figure 76: Add Corner

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 64: In the **Corners Setup** form, click on the **Add new corner** icon ( ) to add a corner.



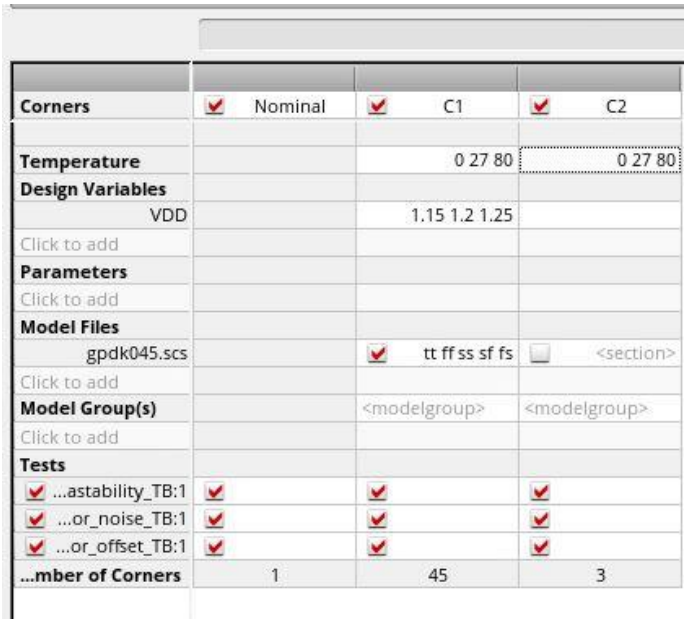
Corner Name	C2		
Corners	Nominal	C1	C2
Temperature		0 27 80	
Design Variables			
VDD		1.15 1.2 1.25	
Click to add			
Parameters			
Click to add			
Model Files			
gpdk045.scs		tt ff ss sf fs	<section>
Click to add			
Model Group(s)		<modelgroup>	<modelgroup>
Click to add			
Tests			
...astability_TB:1			
...or_noise_TB:1			
...or_offset_TB:1			
...mber of Corners	1	45	1

Figure 77: Add New Corner

Next, you will populate the corner **C2** with the corner conditions to test.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

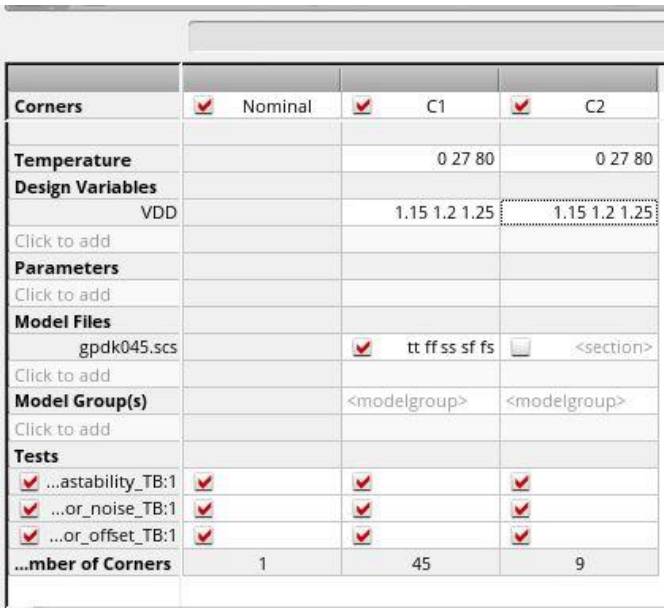
ACTION 65: For the corner **C2**, double-click in the **Temperature** field and set the corner temperatures **0 27 80**.



Corners	Nominal	C1	C2
Temperature		0 27 80	0 27 80
Design Variables			
VDD		1.15 1.2 1.25	
Click to add			
Parameters			
Click to add			
Model Files			
gpdk045.scs		tt ff ss sf fs	<section>
Click to add			
Model Group(s)		<modelgroup>	<modelgroup>
Click to add			
Tests			
...astability_TB:1			
...or_noise_TB:1			
...or_offset_TB:1			
...mber of Corners	1	45	3

Figure 78: Add Temperature Corner Values

ACTION 66: Next, under **Design Variables**, click on **Click to add** and select the design variable **VDD**. Then, under the corner **C2**, enter **1.15 1.2 1.25**.



Corners	Nominal	C1	C2
Temperature		0 27 80	0 27 80
Design Variables			
VDD		1.15 1.2 1.25	1.15 1.2 1.25
Click to add			
Parameters			
Click to add			
Model Files			
gpdk045.scs		tt ff ss sf fs	<section>
Click to add			
Model Group(s)		<modelgroup>	<modelgroup>
Click to add			
Tests			
...astability_TB:1			
...or_noise_TB:1			
...or_offset_TB:1			
...mber of Corners	1	45	9

Figure 79: Add VDD Corner Values

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 67: Next, under **Model Files**, click on **Click to add**. The **Add/Edit Model Files** form will open.

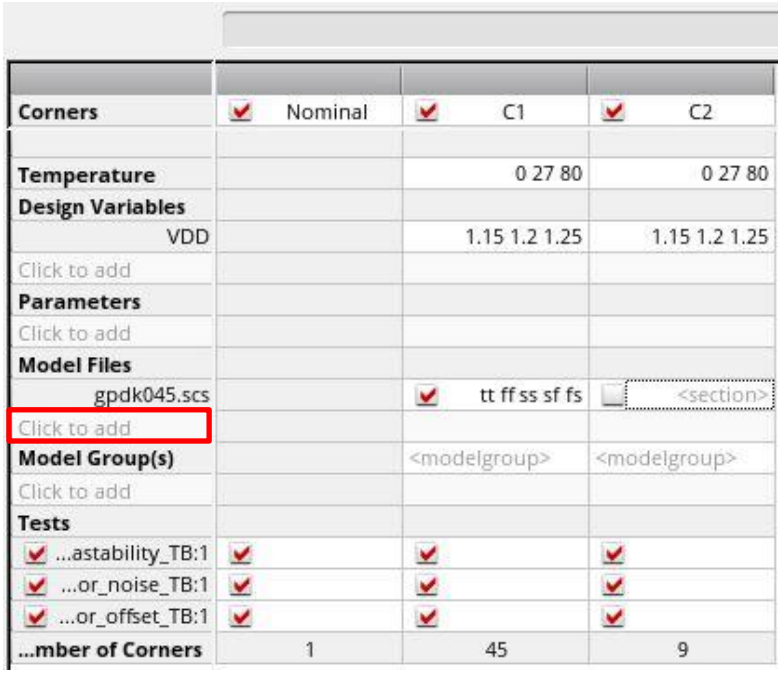


Figure 80: Click to Add Model Files

ACTION 68: Next, in the **Add/Edit Model Files** form, click on **Click to add** and define the path to the model library. Select the **Import from Tests** button and make sure that the form looks like Figure 81 below. Then, click **OK**.

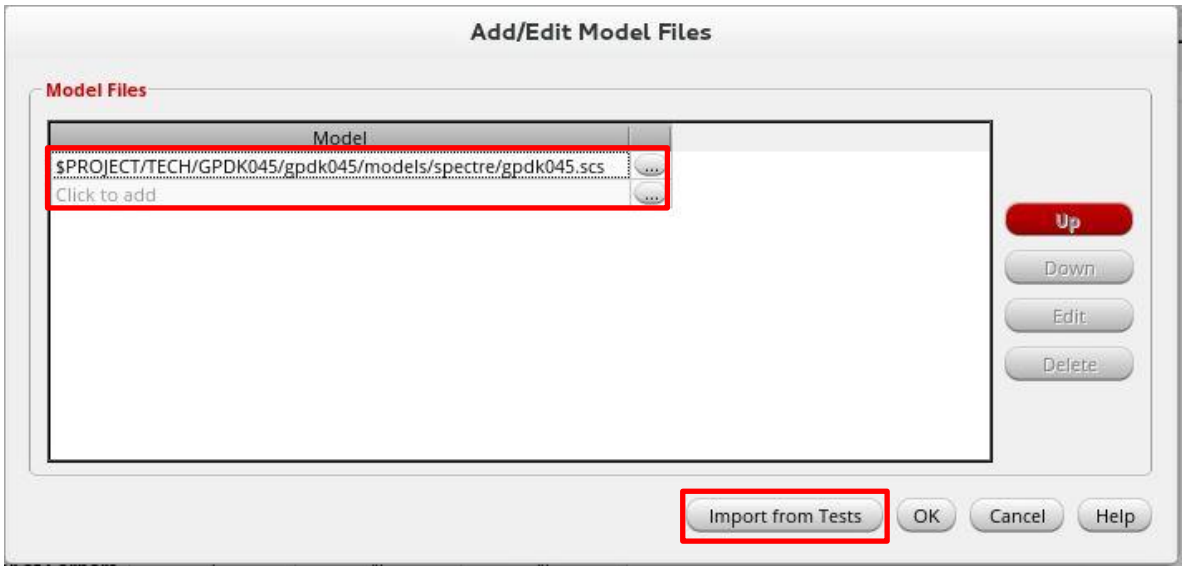


Figure 81: Add/Edit Model Files Form with Model Library Path

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 69: Click on the field of section for **Model Files** – **gpdk045.scs** and define the corner sections to use. For this section, you will only use five process conditions, **tt ff ss sf fs**, to reduce the number of corners to run.

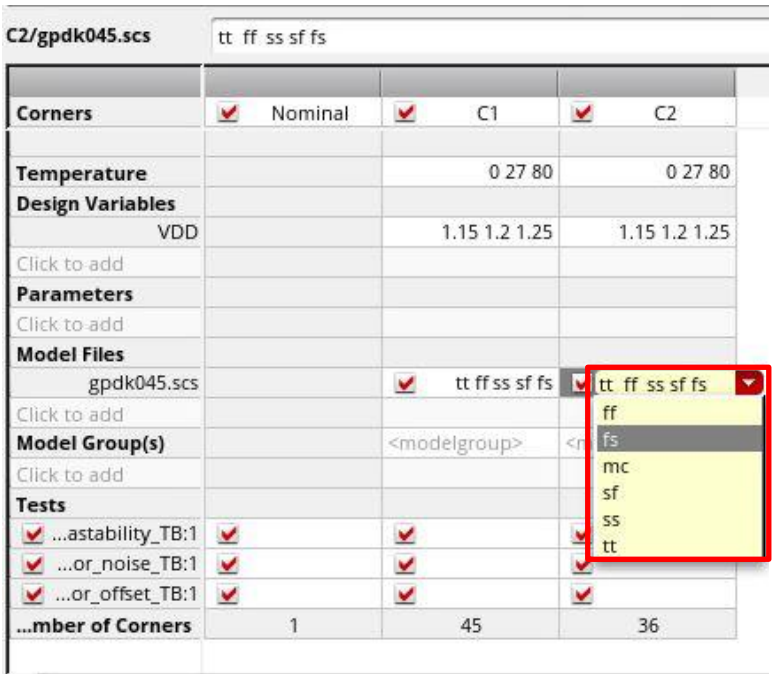


Figure 82: Add Process Corner Values

There are five sets of parameter definitions for the process corners that you will be using:

- tt : Typical model
- ff : Fast NMOS Fast PMOS model
- ss : Slow NMOS Slow PMOS model
- sf : Slow NMOS Fast PMOS model
- fs : Fast NMOS Slow PMOS model

ACTION 70: When done, verify that the Number of Corners = 45, and then click **OK** to finish the corner setup.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

3.4.2 Analyzing Corner Analysis Results

In this section, you will look at the corner analysis results and update the results by updating the specification.

ACTION 71: To display the results, select the ADE Assembler **Results** tab.

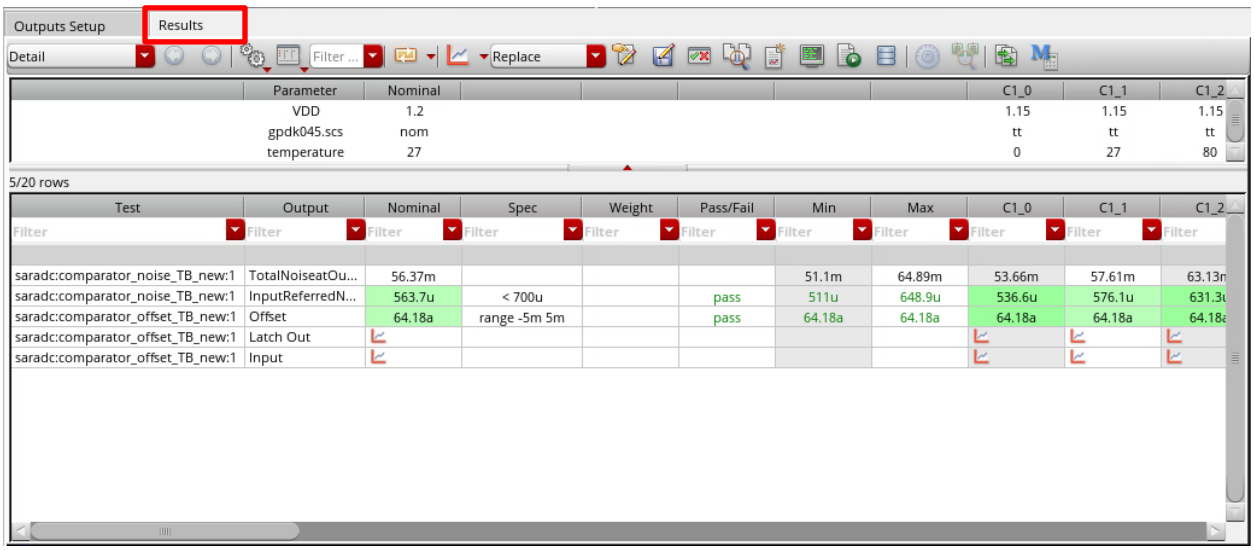


Figure 83: ADE Assembler Results

The corner analysis results are shown above. If you scroll across, you will see all the results from each corner analysis.

ACTION 72: Close the waveform plot and ADE Assembler windows by selecting **File > Close**. Select **No** if asked to save any changes.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Module 4: Capacitor D/A Converter Characterization

In this section, you will extract the contribution of device mismatch to the ADC SINAD. For a Successive Approximation ADC, the ADC mismatch is due to the Capacitor DAC (CAPDAC) mismatch. So, instead of simulating the ADC, you will only simulate the CAPDAC mismatch to speed up the simulation time. You will need to perform Monte Carlo analysis to simulate the mismatch, reducing simulation time**.**

Mismatch is important because it drives an important design trade-off of die area and yield for an ADC. As the devices of the DAC become smaller, the device-to-device matching increases and the mismatch contribution to SINAD also increases. The effect of device mismatch is reduced as device area increases. ADC linearity is determined by the DAC linearity. The largest contributor to the non-linearity of the DAC is the differences in the ratio of the capacitor values due to local process variation and mismatch. The trade-off is that scaling down the capacitor values reduces chip area and die cost, but causes a decreased yield. On the other hand, scaling up the capacitor values increases the chip area and cost, and it also increases yield. You will use Monte Carlo analysis to characterize the effect of mismatch on the ADC's linearity. To speed up the simulation, you will simulate the CAPDAC linearity directly instead of simulating the ADC. This will allow you to extract the effect of CAPDAC mismatch on ADC linearity.

The CAPDAC is shown in Figure 84. It is a differential, switched capacitor design.

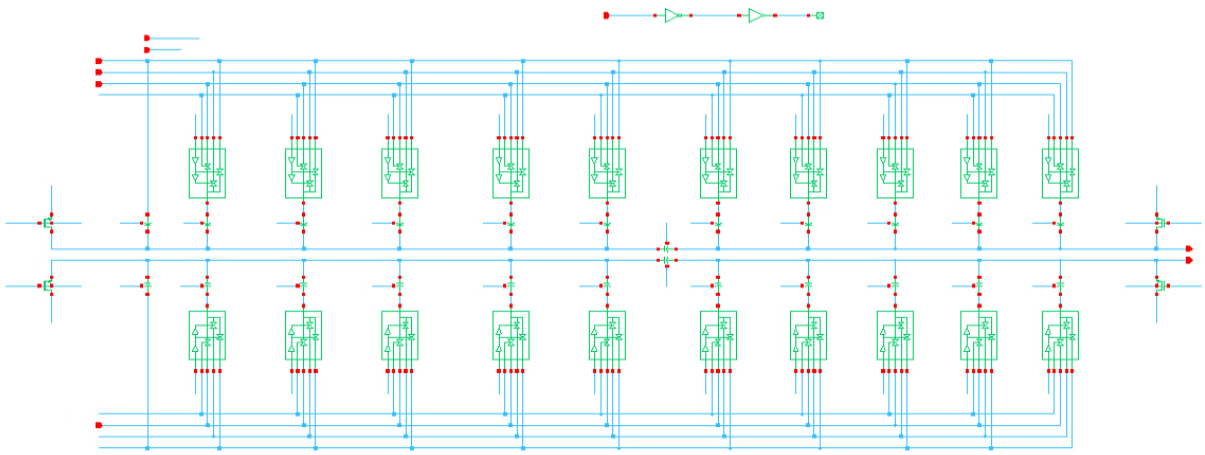


Figure 84: CAPDAC

To characterize how the capacitor mismatch degrades linearity, you will simulate the CAPDAC SINAD. To measure the SINAD, you apply a sine wave to the CAPDAC and run a transient simulation. Then, you perform an FFT on the output of the CAPDAC.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

To measure the SINAD, every lsb of the CAPDAC does not need to be tested. In this module, you chose first to analyze a 256-point FFT because it provides a reasonable compromise between performance, simulation time, and resolution. You will add a few additional conversion cycles to the simulation time. This additional time allows the circuit to settle to steady state before calculating the FFT. The testbench is parameterized to allow four clock cycles for settling the start-up.

For this RAK, you will be using a 1024-point FFT for the final ENOB and SINAD calculations. Section 4.1.2 goes into detail about setting up a 1024-point FFT and how its dynamic measurements differ from a 256-point FFT.

4.1.1 Discussion about Testbench Setup

The testbench uses the tone parameter to determine the input frequency of the testbench. The input frequency determines how much input changes from sample to sample. In this case, you will test the linearity at the maximum input frequency, close to the Nyquist frequency of the data converter. The absolute maximum input frequency is the Nyquist frequency, **fclk/2**. **fclk** is the frequency that the ADC operates at internally. In this case, the maximum input frequency can be written as **fclk (256/2=128)**, where 256 is the number of points in the FFT. So, the number of tones needs to be less than 128.

To reduce the effect of the measurement on the results, you will use synchronous, non-coherent sampling. Synchronous sampling means that an even number of periods of the input signal and clock will be used for the FFT calculation. This will reduce spectral leakage. Non-coherent sampling means that the input frequency and the clock frequency are not harmonically related. If the input and clock frequencies are harmonically related, the quantization noise of the data converter will be in tones that are harmonically related to the clock frequency. As a result, the SINAD will be correct but the distortion will be over-estimated, and the noise will be under-estimated. Typically, the input frequency is chosen to be proportional to a prime number to assure that the input frequency and the clock frequency are not harmonically related.

In this case, you will select the largest prime number less than **128, 127**. So, there will be **127** periods of the input signal and **256** clock periods used to calculate the FFT.

Once the input and clock frequencies are set up, the transient simulation conditions can be determined. In this case, an initial delay, **delay**, of a few periods of the clock is inserted to allow the data converter to settle, **3*period - period/2**. Here, **period/2** is introduced so that the FFT is performed after allowing half a clock cycle for the data converter to settle. For a Successive Approximation ADC, half the clock cycle is allocated for CAPDAC settling time, and half the clock cycle is allocated for comparator decision time. The transient stop time (**TSTOP**) is determined by the initial delay (**delay**), number of points in the FFT (**numPoints**), and the clock frequency (**period (1/fclk)**).

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Lab 4-1: Characterizing Mismatch

In this section, you will simulate the CAPDAC SINAD. You will digitize a sine wave using an ideal ADC to generate the signal to test the CAPDAC.

ACTION 73: In the Library Manager, select **saradc > Everything > 10Bit_capdac_TB_new**. Double- click on **schematic** to open the schematic.

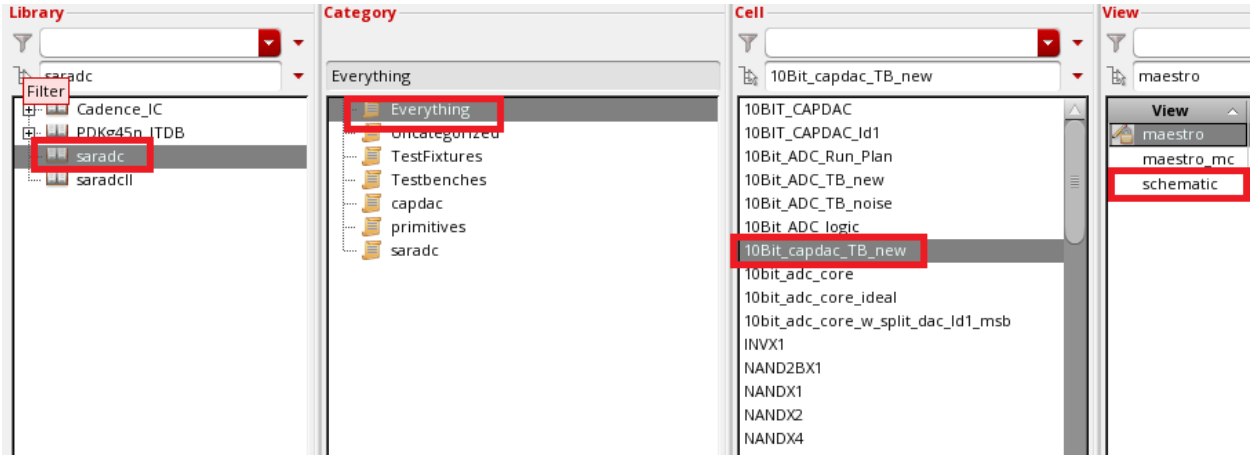


Figure 85: Library Manager

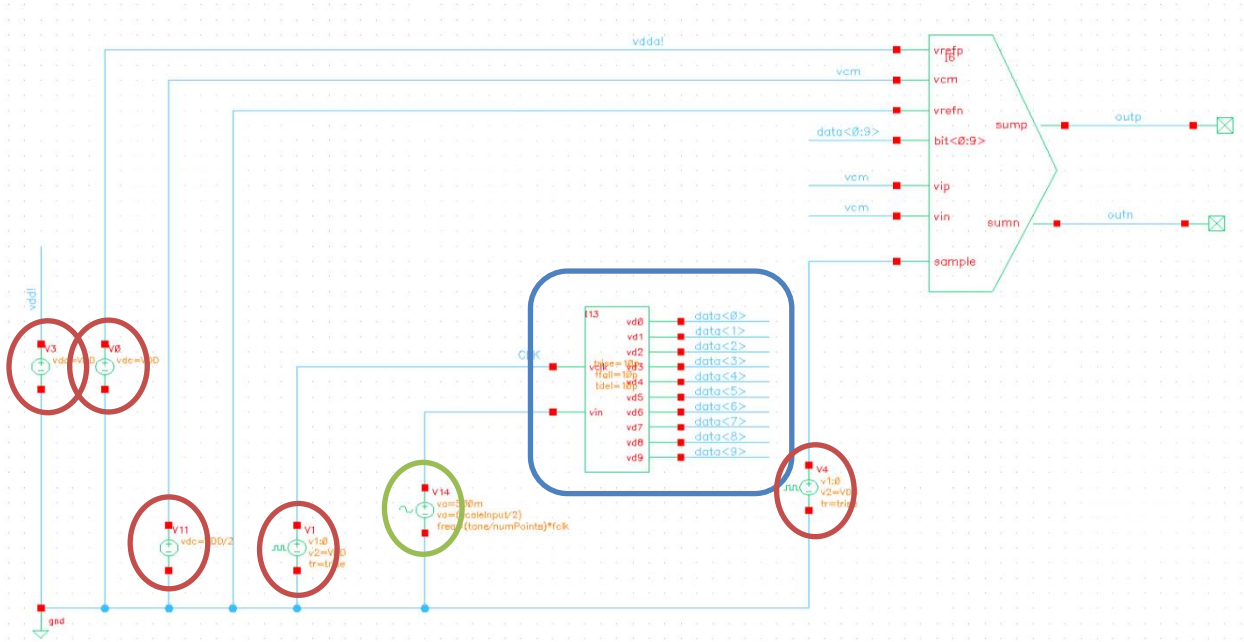


Figure 86: CAPDAC Test Schematic

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The testbench includes the CAPDAC power supplies and clock signals (red), sine wave source (green), and an ideal ADC (blue). The CAPDAC itself is in the upper-right corner.

ACTION 74: In the schematic window, select **Launch > ADE Explorer**.

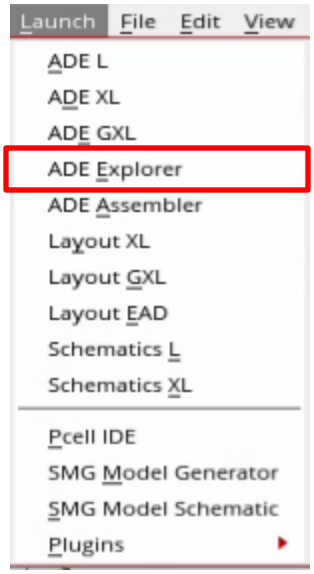


Figure 87: Launch ADE Explorer

ACTION 75: In the **Launch ADE Explorer** window, select **Open Existing View**. Then, click **OK**.

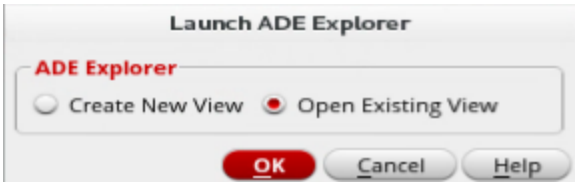


Figure 88: Open Existing View

ACTION 76: In the **Open ADE Explorer View** window, verify that the view is set to **maestro**. Then, click **OK**.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

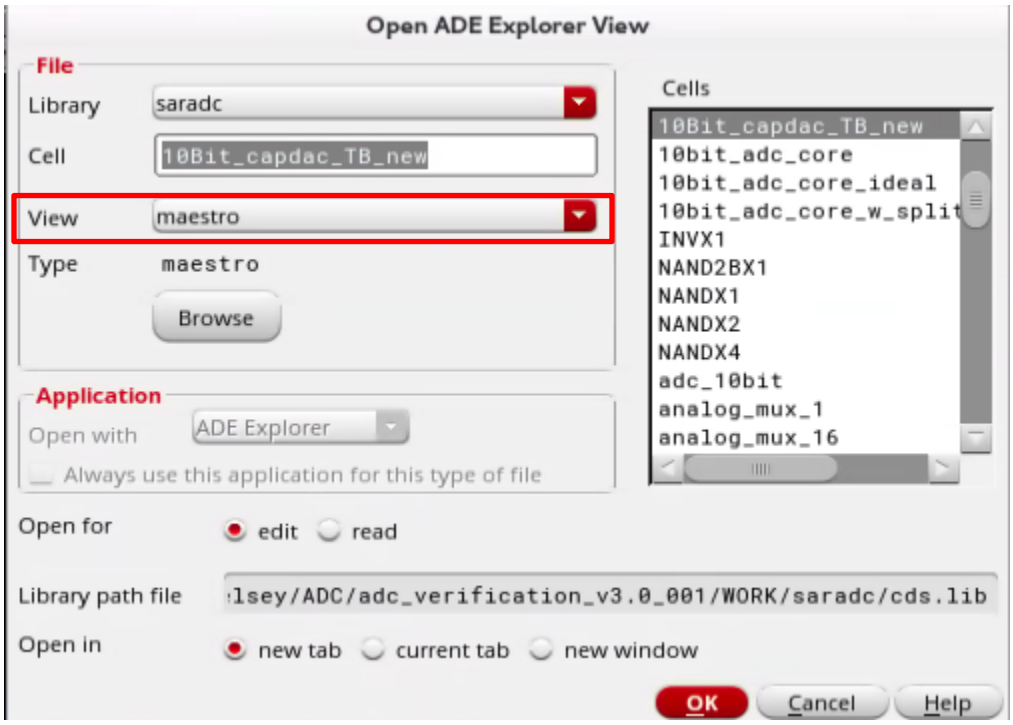


Figure 89: ADE Explorer View

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 77: Once the **maestro** tab is loaded, it should look like Figure 90.

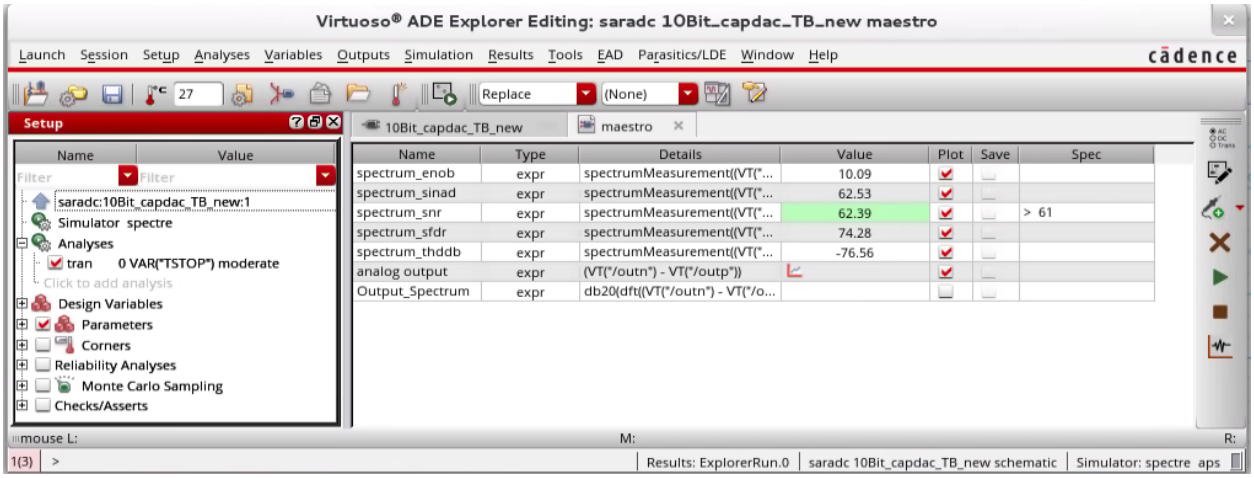


Figure 90: ADE Explorer Window

The first five output expressions correspond to the outputs of the CAPDAC. The first expression is for the THD in dB. The second expression is to calculate the SNR of the CAPDAC. The third expression is used to find the SINAD of the CAPDAC, and the fourth expression is for the SFDR. The fifth expression is the most important; it corresponds to the ENOB of the CAPDAC.

The last two remaining expressions correspond to the output frequency spectrum and the transient output voltage at node “out”. The output frequency spectrum is calculated from the transient output voltage at node “out” and is there as a reference for the calculations to come. It is to be used to check your results and verify that your values are correct. These last two expressions are used to calculate the first five equations and are used for visualization purposes.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 78: LMB-click on the "**+**" symbol next to **Analyses** to view the type of analyses set up. Double-click on **tran** to open the analysis options.

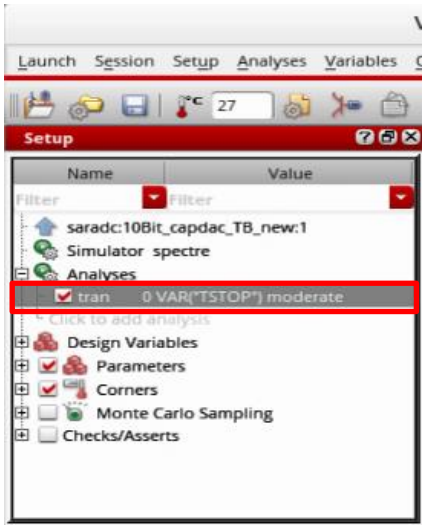


Figure 91: Transient Analysis

ACTION 79: In the **Choosing Analyses – ADE Explorer** window, verify that the transient analysis is as shown in the picture below.

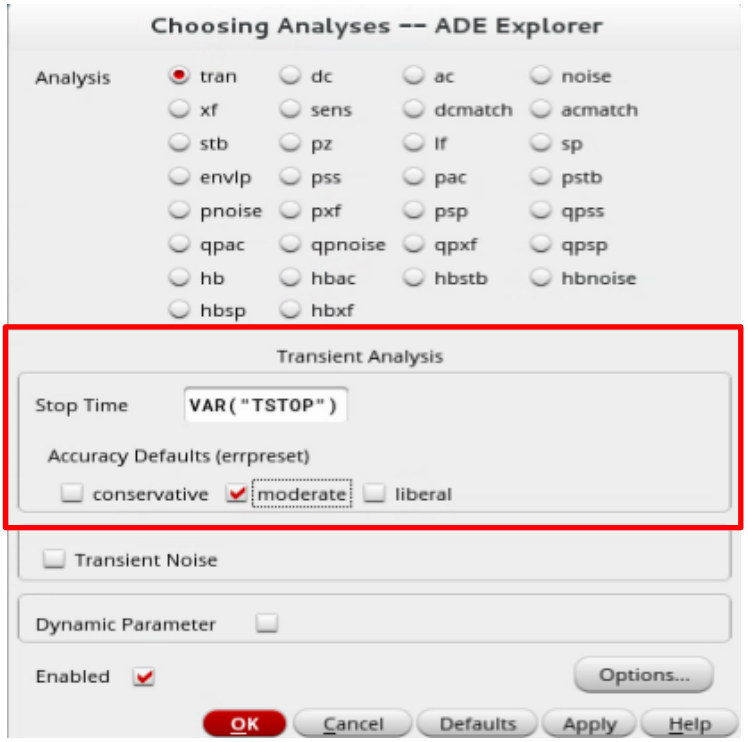


Figure 92: Transient Analysis Setup

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Make sure to verify that **Stop Time** is set to **VAR("TSTOP")** and **Accuracy Default** is set to **moderate**. The transient stop time (**TSTOP**) is determined by the initial delay (**delay**), the number of points in the FFT (**numPoints**), and the clock frequency (**period (1/fclk)**).

The **errpreset** options control the accuracy of the transient simulation by selecting a collection of additional parameter values. "**moderate**" is the default setting, "**liberal**" allows for faster simulations but less accuracy, and "**conservative**" is the most accurate but takes the longest to simulate.

ACTION 80: Click on **Options** at the bottom of the **Choosing Analyses** window. The **Transient Options** window should appear. Select the **Output** tab as shown in Figure 93.

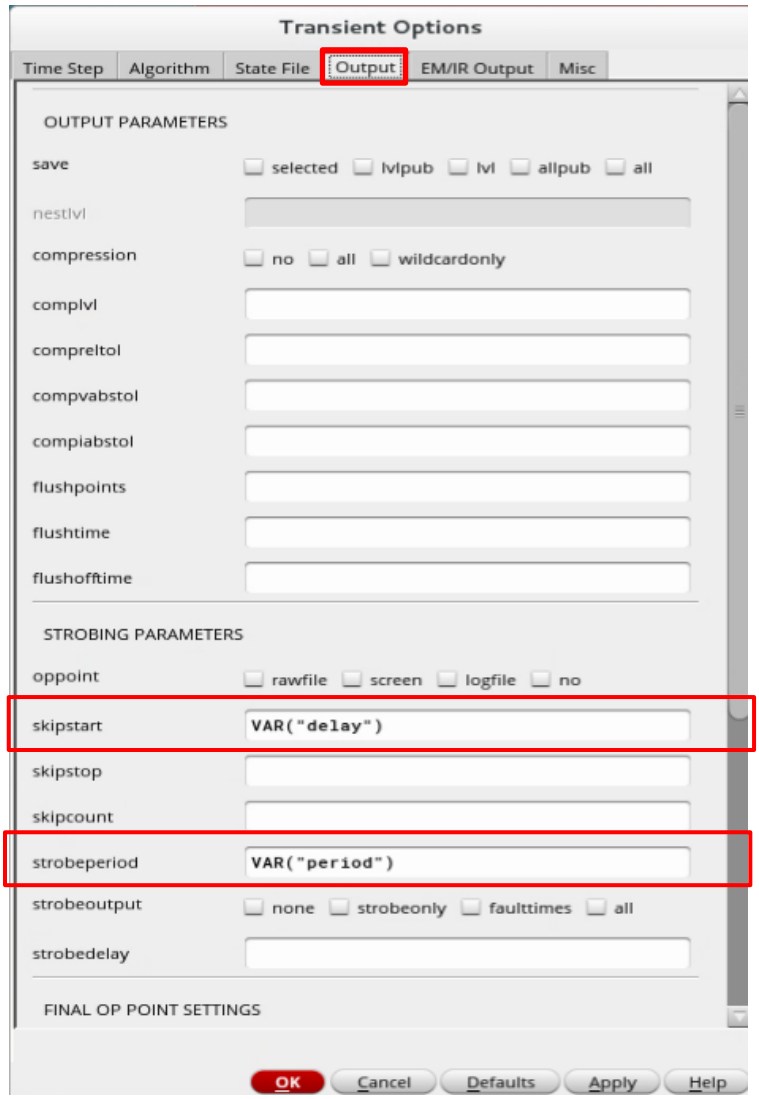


Figure 93: Transient Options

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Verify that the **skipstart** value is set to **VAR("delay")** and the **strobeperiod** value is set to **VAR("period")**. If the fields are empty, fill them so that your **Transient Options** window matches Figure 93. The reason for setting these values is explained below.

After the transient simulation is complete, an FFT will be performed to convert the time-domain results of the simulation into the frequency-domain response of the CAPDAC. When performing the FFT, an error can be introduced into the calculation due to interpolation. The simulator chooses the ideal time step to meet the accuracy and performance requirements of the simulation. However, these time points are usually different than the time points required for the FFT. Because of the difference, the output waveform is interpolated to estimate the FFT time points, and this introduces interpolation error into the FFT. The solution is to use strobing. Strobing is a feature of the simulator that forces time steps at the points required for the FFT.

To use strobing requires setting two parameters: **skipstart** and **strobeperiod**. The **skipstart** parameter defines the first strobe point, and the **strobeperiod** parameter defines how often to perform the strobe. While the simulator strobe is active, only the strobe time point data is saved. In order for this setup to be used in multiple designs, you choose the design variable, **delay**, to be used to define the first time for strobing, and **strobeperiod** is used to define how often to strobe. The value for **strobeperiod** is **(TSTOP-delay)/numPoints**, which is equal to **period** in the Design Variables.

ACTION 81: Now that you have seen how the transient analysis and options are set up, you may click the **Cancel** button to cancel the forms without making any changes.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 82: To start the simulation, select the ("**Run Simulation**") button. When the simulation completes, the CAPDAC output will be displayed in a ViVA window as shown in Figure 94.

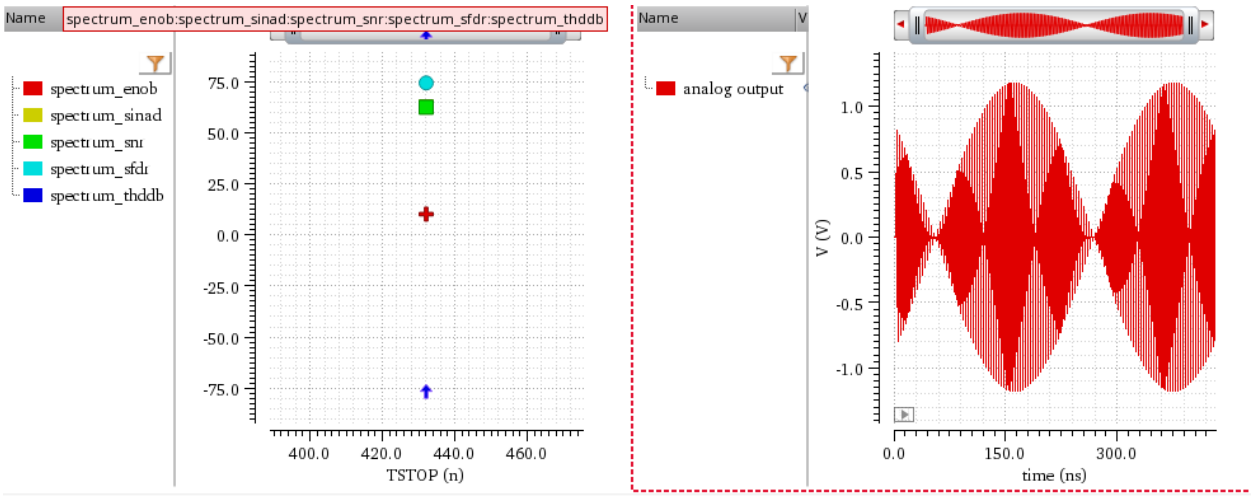


Figure 94: ViVA Window after Simulation

ACTION 83: In ViVA, click on the **Card** view option and select subwindow **2** from the pull-down menu. This is done so that only the analog output waveform is visible in the window, as shown in Figure 96.

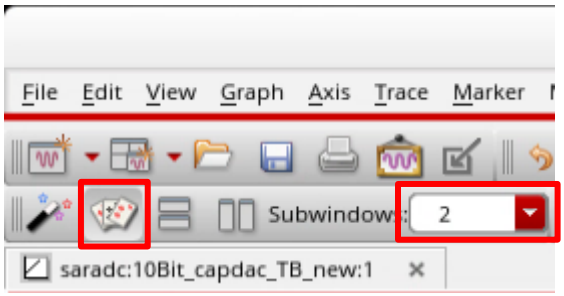


Figure 95: Card View with Subwindow 2

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

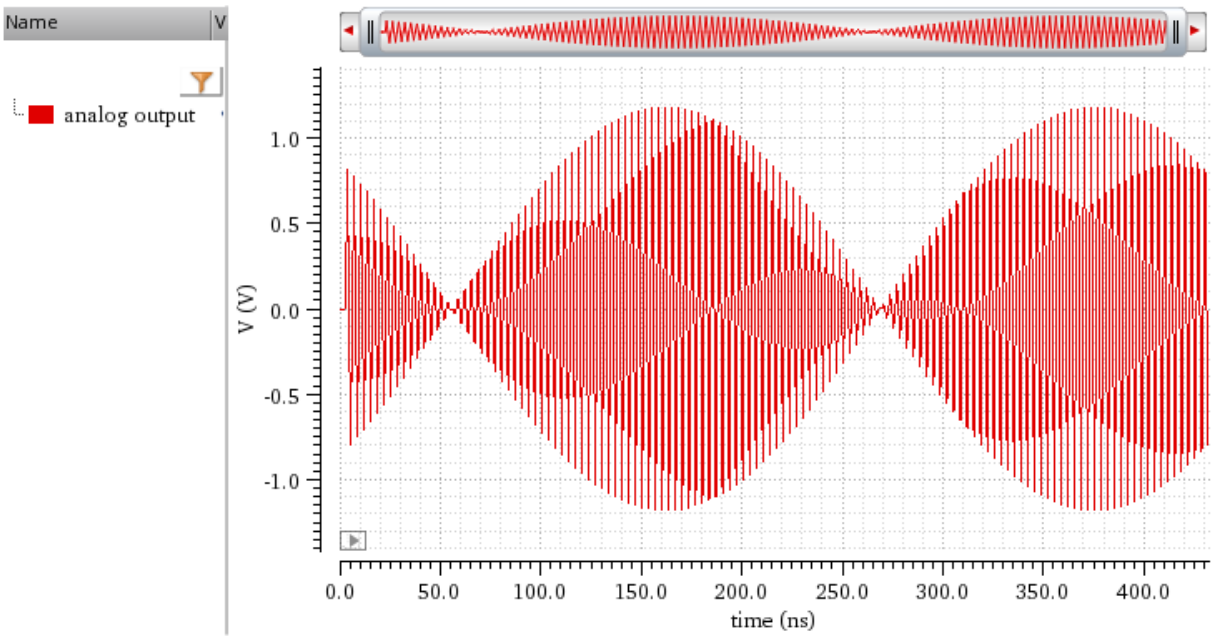


Figure 96: Analog Output of CAPDAC

From the CAPDAC output, you can calculate the dynamic characteristics of the DAC. The parameters you will calculate are ENOB, SINAD, SNR, and THD.

To calculate these parameters, you will use the ViVA Spectrum measurement. The Spectrum measurement was developed for extracting the dynamic measurement results from the spectrum of an ADC, DAC, S/H, etc. The spectrum function will perform an FFT on the time-domain data, or it can also be used with the output of the frequency spectrum output by the Spectre® Fourier Integral function. From the frequency spectrum, the Spectrum measurement calculates the standard measures for a data converter: ENOB, SINAD, SNR, and more.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 84: In the ViVA waveform window, select **Measurements > Spectrum**. This will open the Spectrum Assistant, and it will be docked to the right of the window.

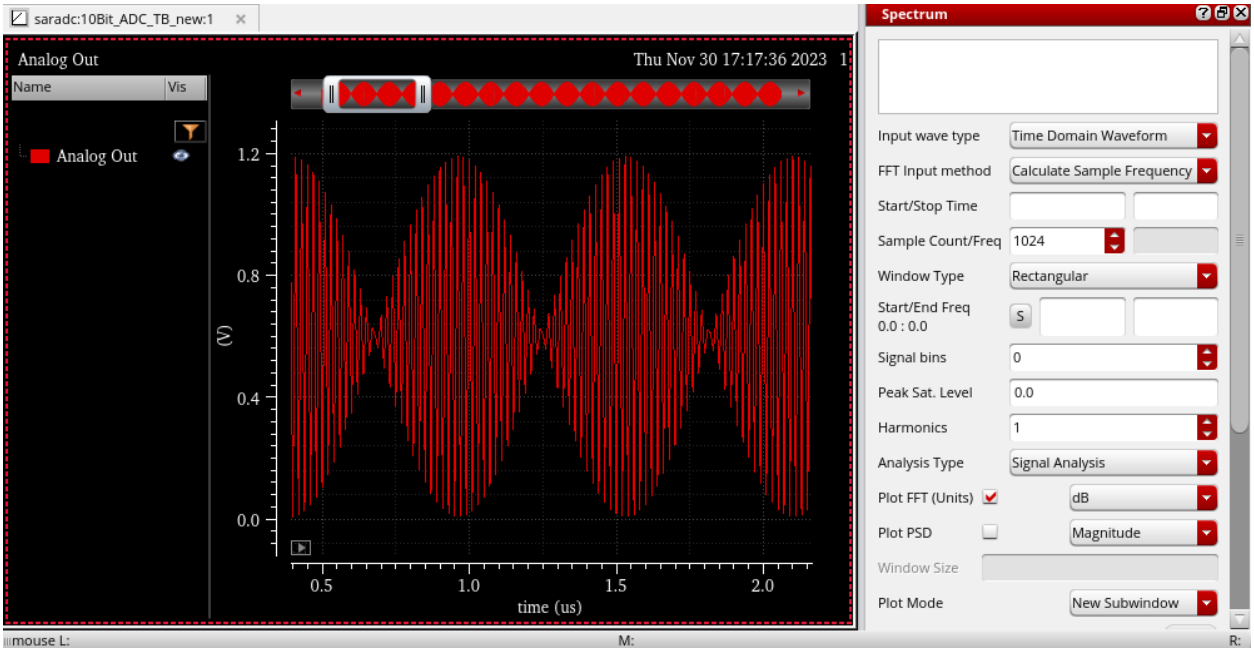


Figure 97: Spectrum Assistant

ACTION 85: Click on the **analog_output** waveform to select it. After selecting the **analog_output** waveform, its expression will appear in the **Signal/Expr Names** window of the Spectrum Assistant.

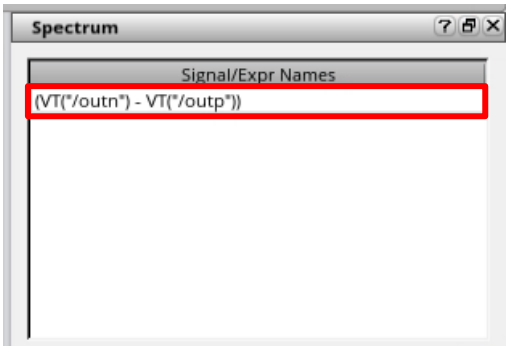


Figure 98: Signal/Expr Names Window in Spectrum Assistant

To set up the spectrum functions, there are several parameters you need to define. The spectrum function performs an FFT and then calculates the key parameters for a data converter.

ACTION 86: Set the **Start/Stop Time**. The start time is **VAR("delay")+(4*VAR("period"))**, and the stop time is **VAR("TSTOP")**.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Start Time is the first time point for the FFT. To avoid complex calculations, you will use **VAR("delay")+(4*VAR("period"))** and let ViVA read the value used by the testbench. The reason why the start time has an additional **(4*period)** added to the delay is to account for the zeroes insertion of the CAPDAC's reset. The sample for the original "delay" time is a zero; so, the actual sample is four periods later.

Stop Time is the last point for the FFT. To avoid complex calculations, you will use **VAR("TSTOP")** and let ViVA read the value used by the testbench. Additional four periods are added to **TSTOP** to account for the additional delay.

ACTION 87: Set the **Sample Count/Freq** to **256**. The spectrum function will automatically calculate the sample frequency. 256 is the number of points in the FFT and is defined by the design variable number.

SampleFreq = SampleCount/(StopTime - StartTime)

ACTION 88: Use **Window Type → Rectangular**.

ACTION 89: For the **Start/Stop Freq**, click on the icon to reset the FFT start and stop frequency.

ACTION 90: Set the **Harmonics → 10**. The number of harmonics is used to determine the number of tones to use for the THD calculation. The remaining tones will be counted in the SNR calculation.

10 is selected in this RAK because typically, after 10 harmonics, the spectrum output will be below the noise floor. This value gives very good results in simulation.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Verify that the Spectrum Assistant window looks like Figure 99 before moving on.

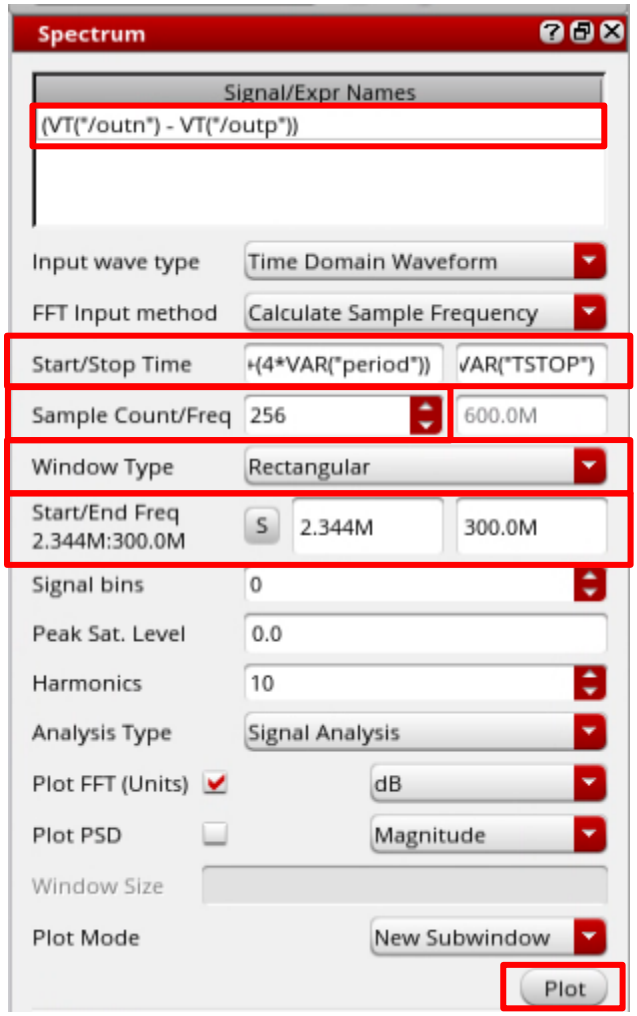


Figure 99: Spectrum Measurements Setup

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 91: Select "**Plot**" to display the Spectrum and Output measurements. The spectrum function is the frequency spectrum and the dynamic measurements. Your screen should look like Figure 100.

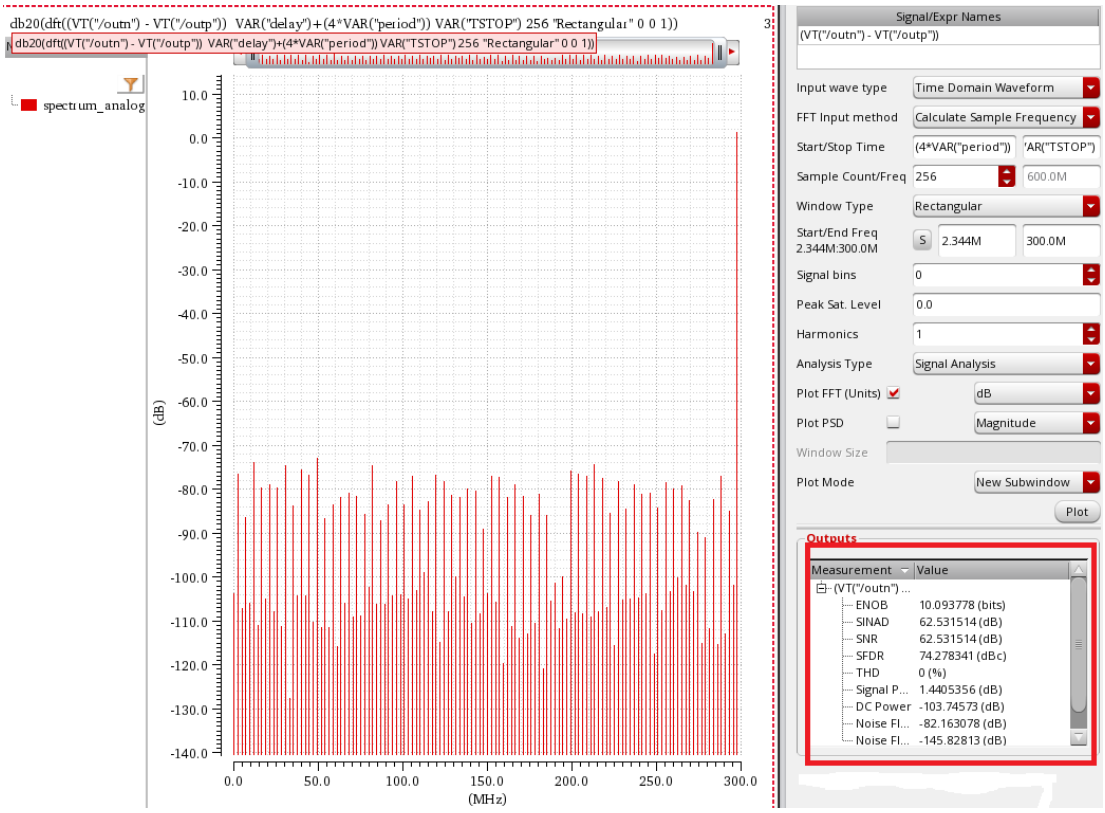


Figure 100: Frequency Spectrum and Dynamic Measurements

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 92: RMB-click on the **spectrum_analog** waveform (in red) and then select **Trace Properties** from the pop-up menu, as shown in Figure 101 below.

Figure 101: RMB-Click on Waveform Menu

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 93: Once the **Trace Properties for spectrum_analog output** window is loaded, set **Type/Style** to "**line**" from the pull-down menu. Click **OK** to update the window.

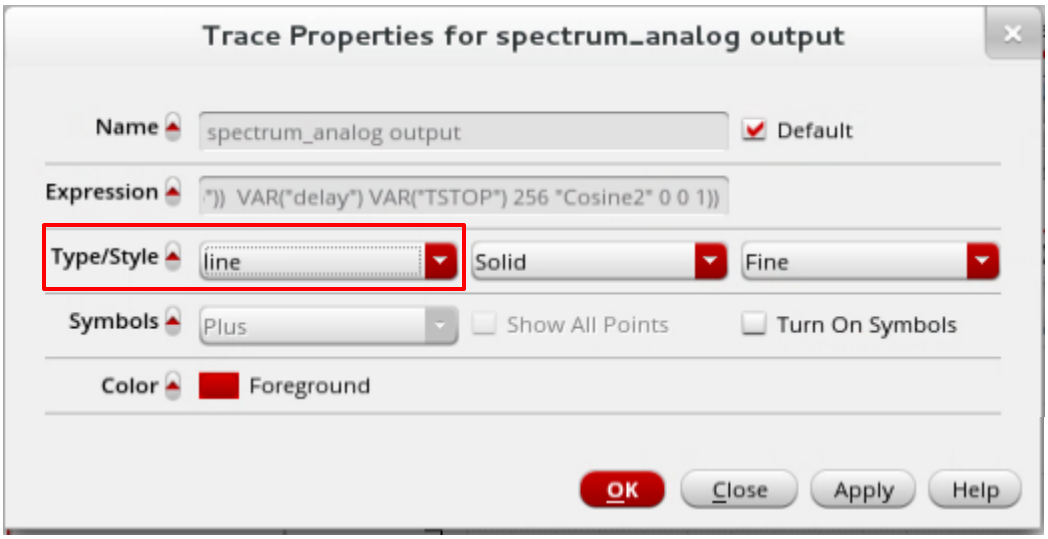


Figure 102: Trace Properties of Spectrum

ACTION 94: Verify that the **spectrum_analog output** waveform and dynamic measurements match with Figures 103 and 104.

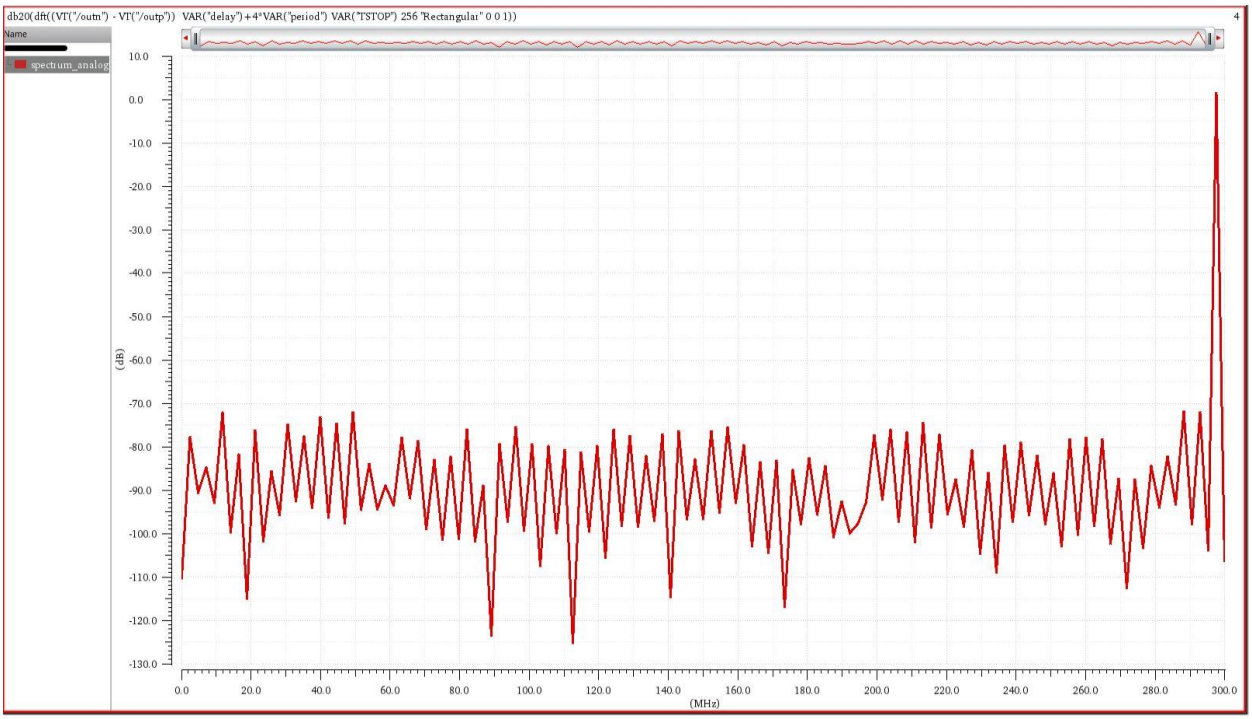
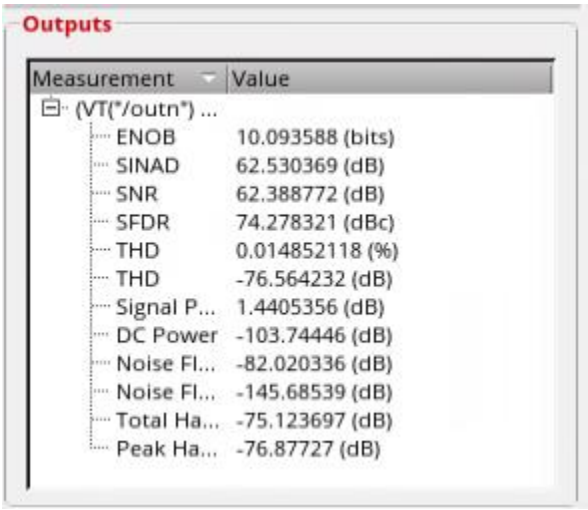


Figure 103: Line Graph of Frequency Spectrum

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

This plot illustrates the output frequency spectrum of the CAPDAC. This spectrum is used to calculate the SINAD of the CAPDAC.

Below are the corresponding dynamic measurements.



Outputs

Measurement	Value
(VT("/outn") ...	
ENOB	10.093588 (bits)
SINAD	62.530369 (dB)
SNR	62.388772 (dB)
SFDR	74.278321 (dBc)
THD	0.014852118 (%)
THD	-76.564232 (dB)
Signal P...	1.4405356 (dB)
DC Power	-103.74446 (dB)
Noise Fl...	-82.020336 (dB)
Noise Fl...	-145.68539 (dB)
Total Ha...	-75.123697 (dB)
Peak Ha...	-76.87727 (dB)

Figure 104: Dynamic Measurements

4.1.2 [Optional] Exploring the Spectrum Function, Number of FFT Points

The previous sections mentioned that a 256-point FFT will provide a good compromise between speed and resolution. In section 4.1.2, a 1024-point FFT will be performed. Let's look at the effect of increasing the number of FFT points from 256 to 1024, and how it impacts the FFT results. Since the number of points in the FFT is directly proportional to simulation time, it is important to understand how many FFT points are required to achieve accurate results. So, you will repeat the CAPDAC SINAD simulation, this time increasing the number of points in the FFT, and investigate the effect of the number of FFT points on the output spectrum.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 95: In the **Design Variables** section of the **Setup** form, set **tone = 503** and **numPoints = 2048**.

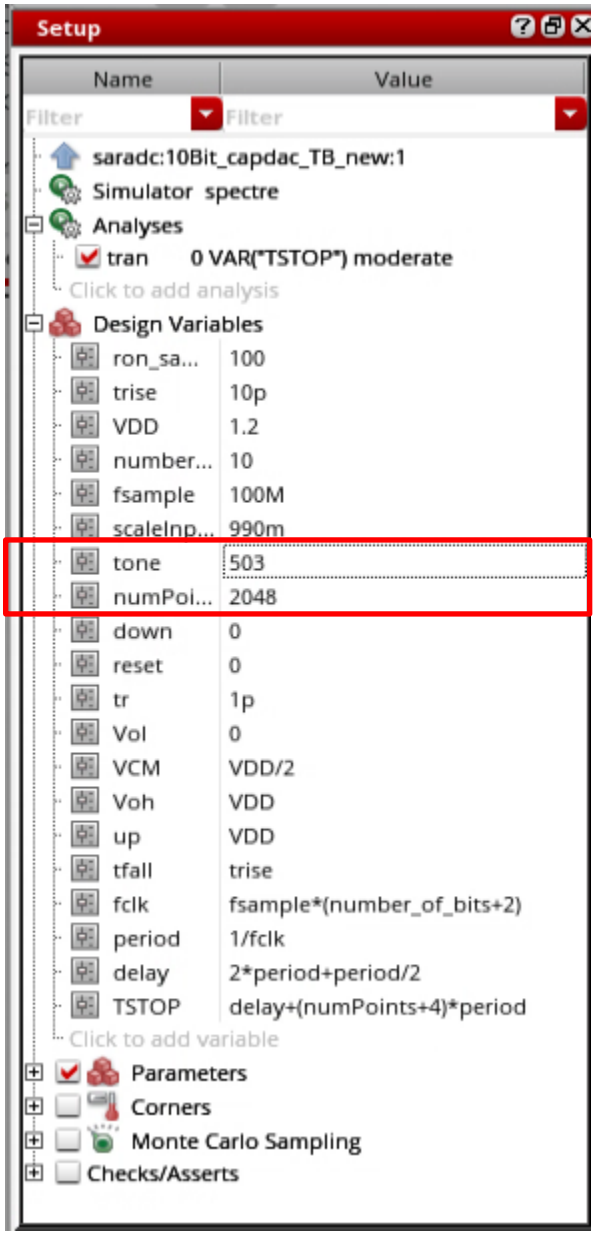


Figure 105: Tone and numPoints Design Variables

The reason for **numPoints** to be set to **2048** instead of **1048** accounts for the zeroes insertion of the CAPDAC's reset. By taking double the number of samples of the output waveform, the spectrum of the CAPDAC can be accurately measured.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 96: Click on the **Plot Options** icon ( ) and verify that **Plotting Mode** is set to "**New SubWin**". The **ADE Explorer/Assembler Plotting Options** form appears below.

Click **OK** to close the form.

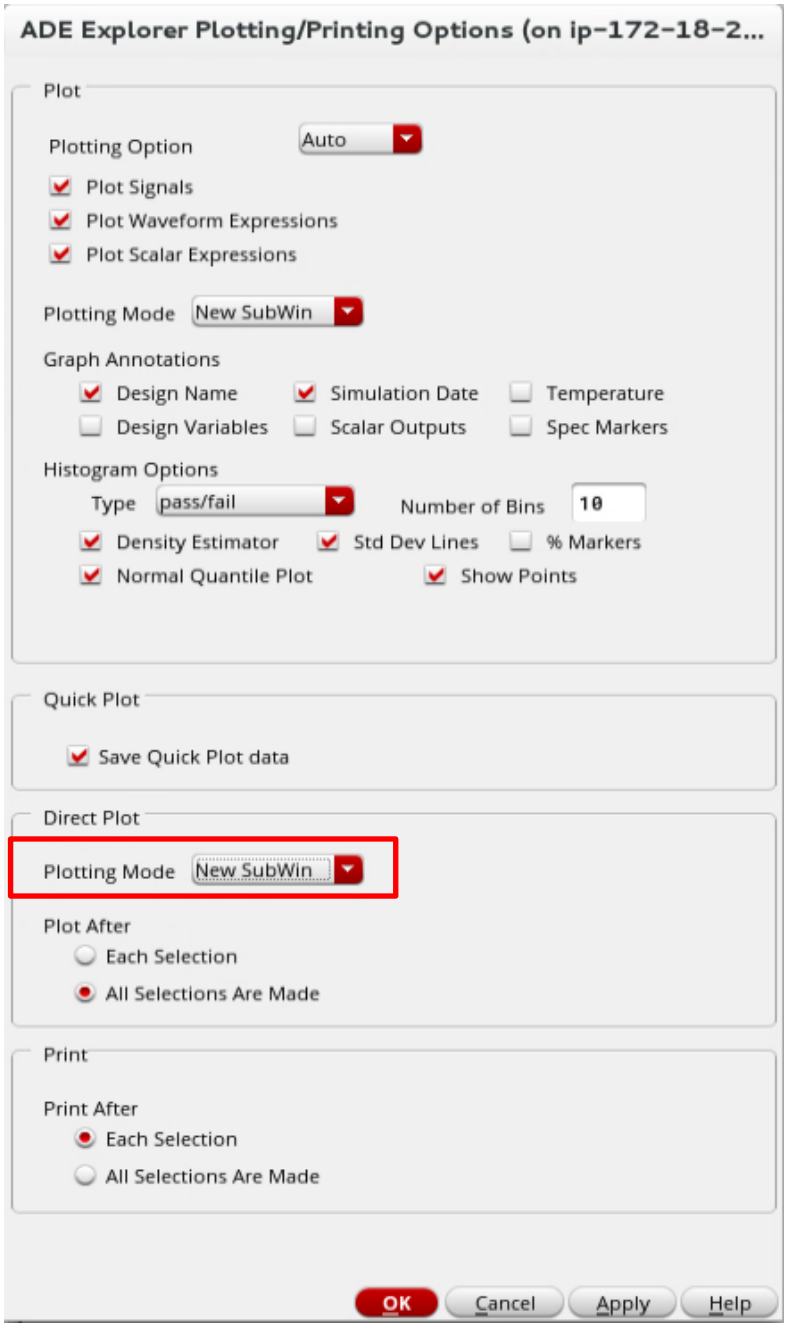


Figure 106: ADE Explorer Plotting/Printing Options

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 97: To start the simulation, select the ("**Run Simulation**") button. When the simulation completes, the CAPDAC output for the 1024-point FFT will be displayed in a ViVA window, along with the 256-point output calculated from the previous lab, as shown in Figure 107.

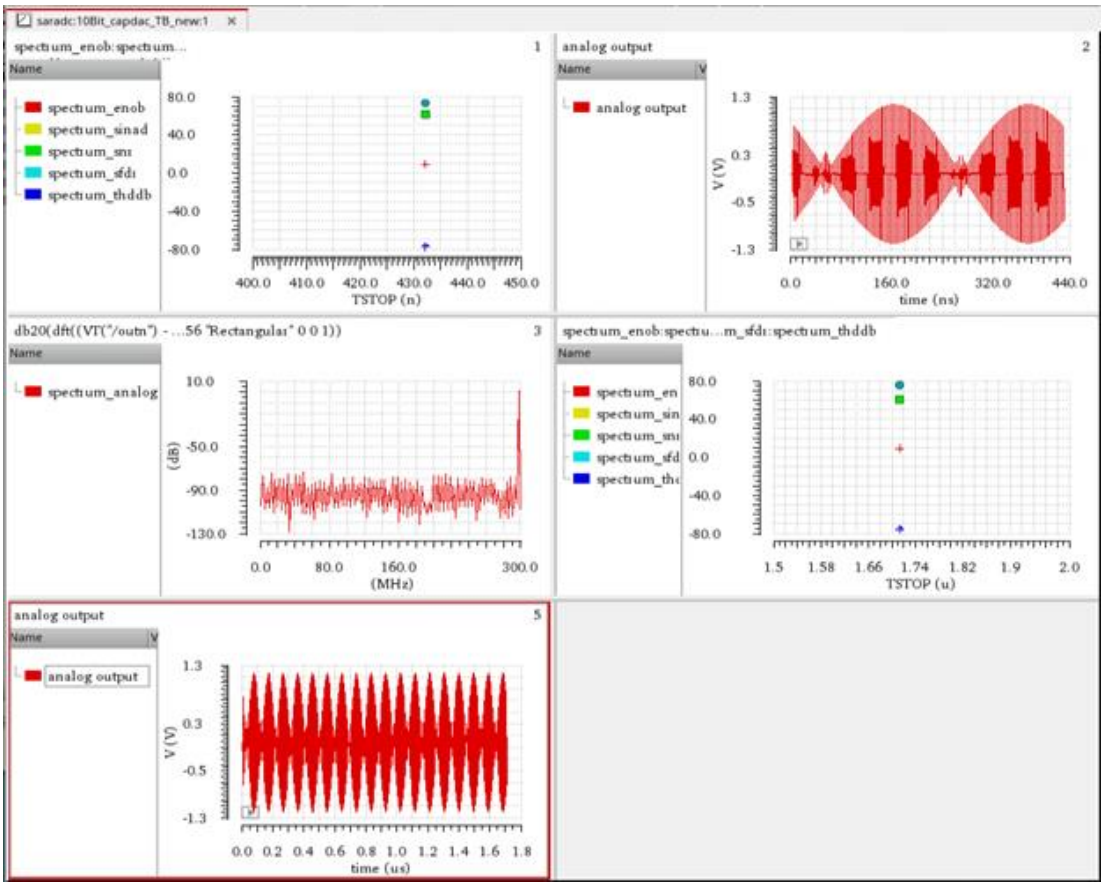


Figure 107: ViVA Window after Simulation with both 256-point and 1024-point FFTs

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 98: In ViVA, click on the **Card** view option and select subwindow **5** from the pull-down menu. This is done so that only the analog output waveform is visible in the window, as shown in Figure 109.

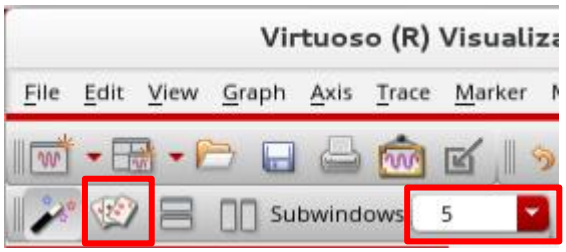


Figure 108: Card View with Subwindow 5

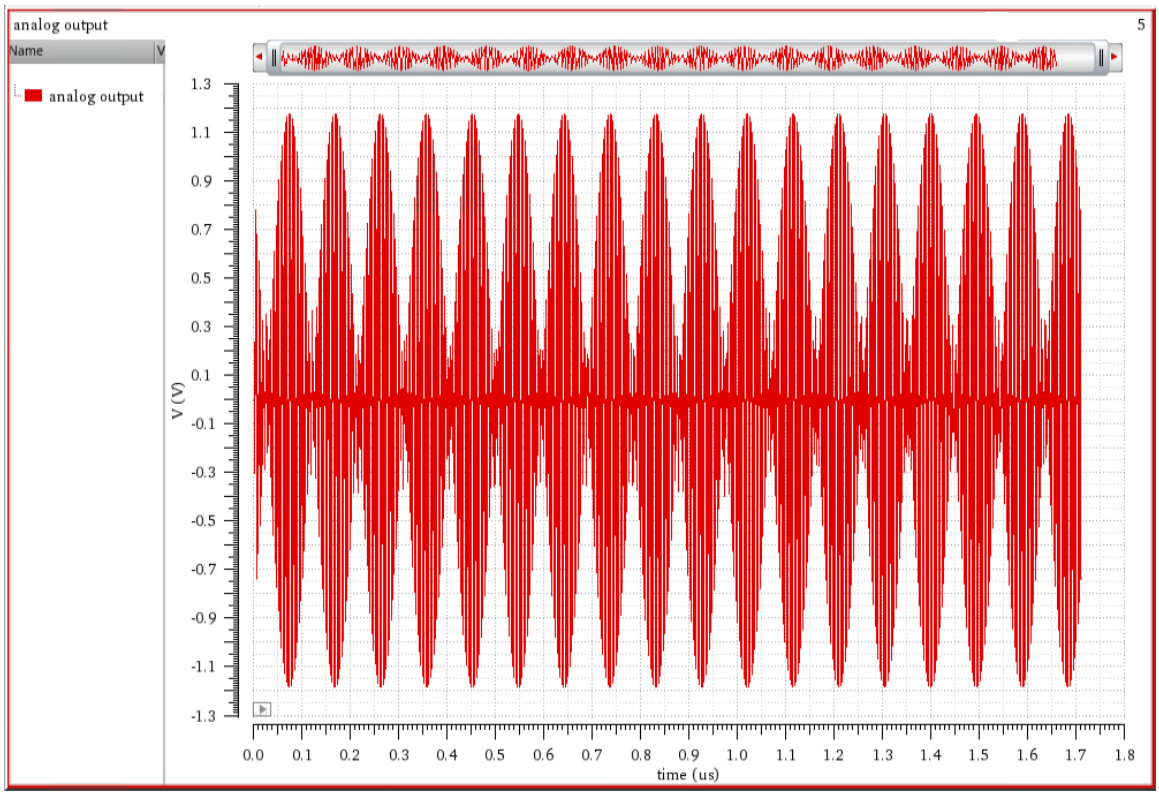


Figure 109: Analog Output of CAPDAC

From the CAPDAC output, you can calculate the dynamic characteristics of the DAC. The parameters you will calculate are ENOB, SINAD, SNR, and THD.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 99: In the ViVA waveform window, select **Measurements > Spectrum**. This will open the Spectrum Assistant, and it will be docked to the right of the window. You can disregard this step if the Spectrum Assistant is already opened from Lab 4-1.

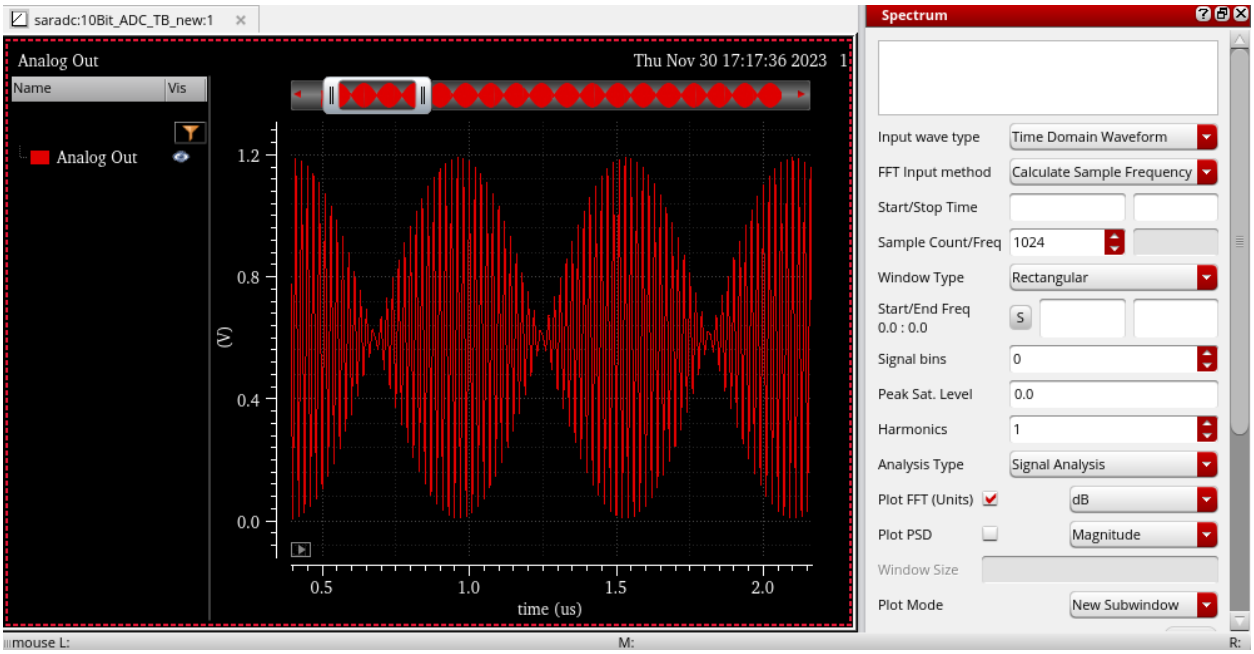


Figure 110: Spectrum Assistant

ACTION 100: Click on the **analog_output** waveform to select it. After selecting the **analog_ouput** waveform, its expression will appear in the **Signal/Expr Names** window of the Spectrum Assistant.

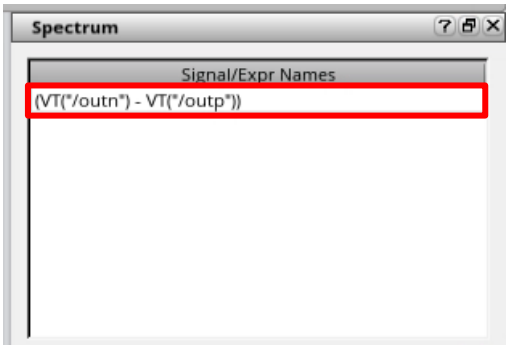


Figure 111: Signal/Expr Names Window in Spectrum Assistant

To set up the spectrum functions, there are several parameters you need to define. The spectrum function performs an FFT and then calculates the key parameters for a data converter.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 101: Set the **Start/Stop Time**. The start time is **VAR("delay")+(4*VAR("period"))**, and the stop time is **VAR("TSTOP")**.

> **Start Time** is the first time point for the FFT. To avoid complex calculations, you will use **VAR("delay")+(4*VAR("period"))** and let ViVA read the value used by the testbench. The reason why the start time has an additional **(4*period)** added to the delay is to account for the zeroes insertion of the CAPDAC's reset. The sample for the original "delay" time is a zero; so, the actual sample is four periods later.
>
> **Stop Time** is the last point for the FFT. To avoid complex calculations, you will use **VAR("TSTOP")** and let ViVA read the value used by the testbench. Additional four periods are added to **TSTOP** to account for the additional delay.

ACTION 102: Set the **Sample Count/Freq** to **1024**. The spectrum function will automatically calculate the sample frequency. 1024 is the number of points in the FFT and is defined by the design variable number.

SampleFreq = SampleCount/(StopTime - StartTime)

ACTION 103: Use **Window Type → Rectangular**.

ACTION 104: For the **Start/Stop Freq**, click on the icon to reset the FFT start and stop frequency.

ACTION 105: Set the **Harmonics → 10**. The number of harmonics is used to determine the number of tones to use for the THD calculation. The remaining tones will be counted in the SNR calculation.

10 is selected in this RAK because typically, after 10 harmonics, the spectrum output will be below the noise floor. This value gives very good results in simulation.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Verify that the Spectrum Assistant window looks like Figure 112 before moving on.

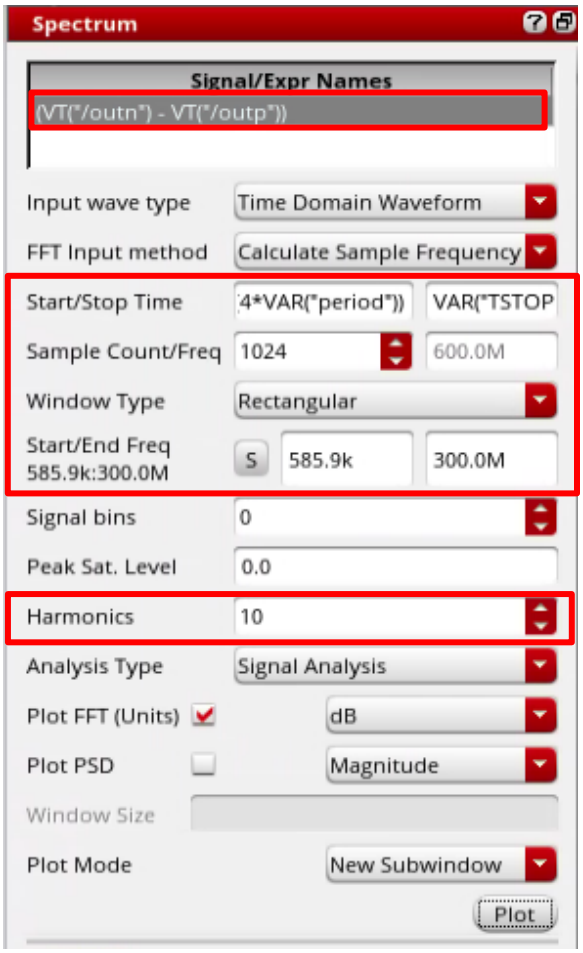


Figure 112: Spectrum Measurements Setup

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 106: The spectrum function is the frequency spectrum and the dynamic measurements. Your screen should look like the next figure.

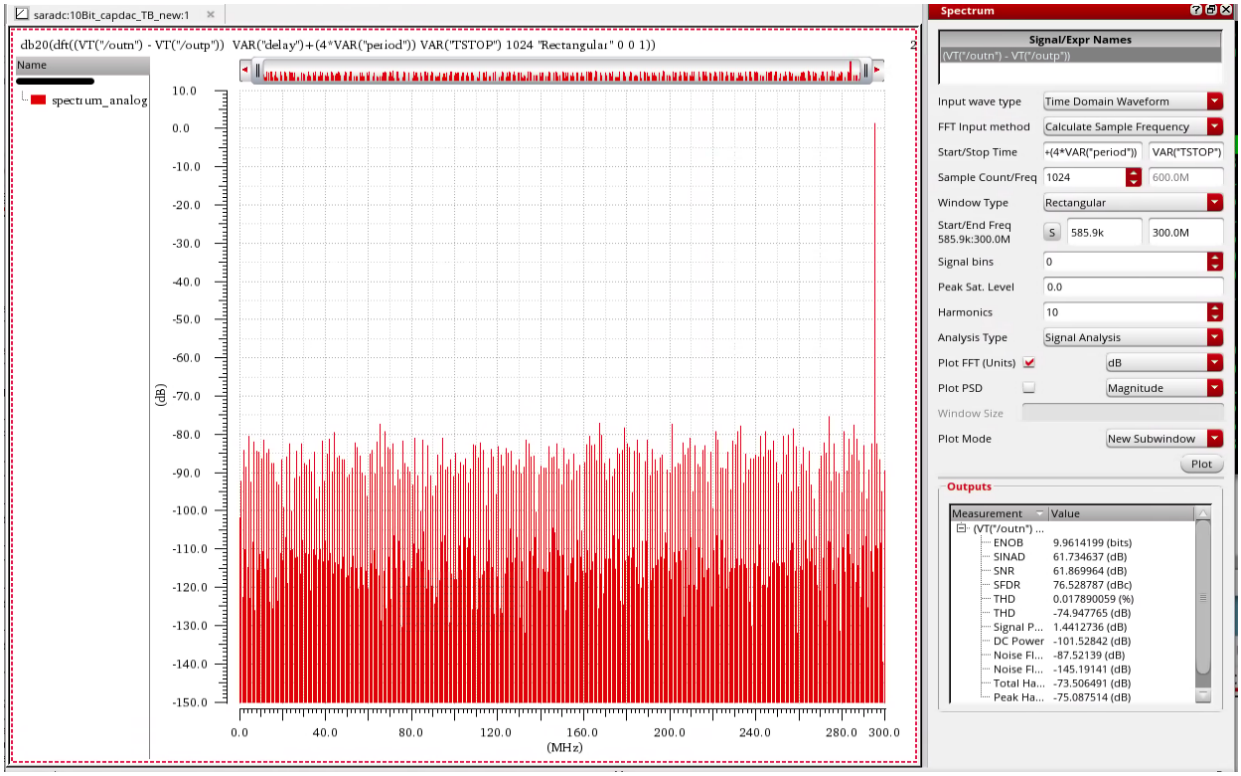


Figure 113: Frequency Spectrum and Dynamic Measurements

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 107: RMB-click on the **spectrum_analog_spectrum** waveform (in red) and then select **Trace Properties** from the pop-up menu, as shown in Figure 114 below.

Figure 114: RMB-Click on Waveform Menu

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 108: Once the **Trace Properties for spectrum_analog output** window is loaded, set **Type/Style** to "**line**" from the pull-down menu. Also, change the **Color** of the waveform to "blue" by clicking on the color square and selecting the blue color. Click **OK** to update the window.

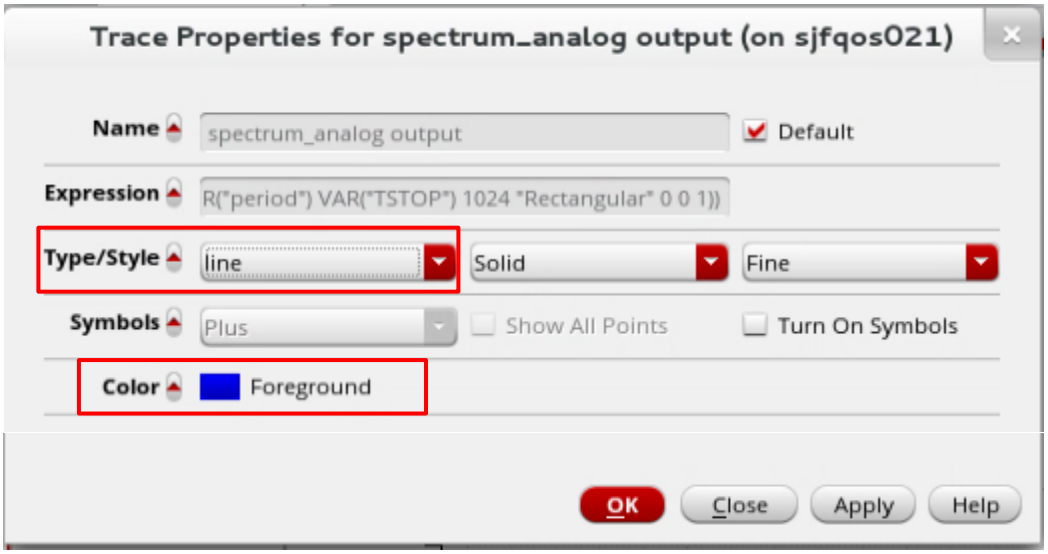


Figure 115: Trace Properties of Spectrum

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 109: Verify that the **spectrum_analog** waveform and dynamic measurements match with Figures 116 and 117.

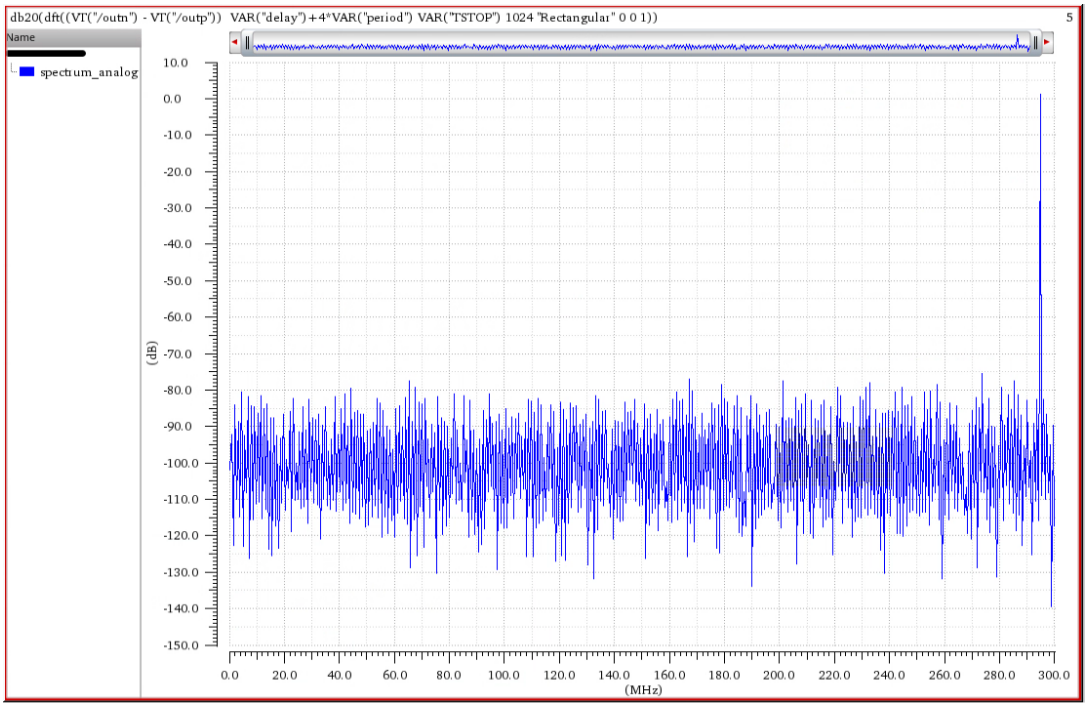
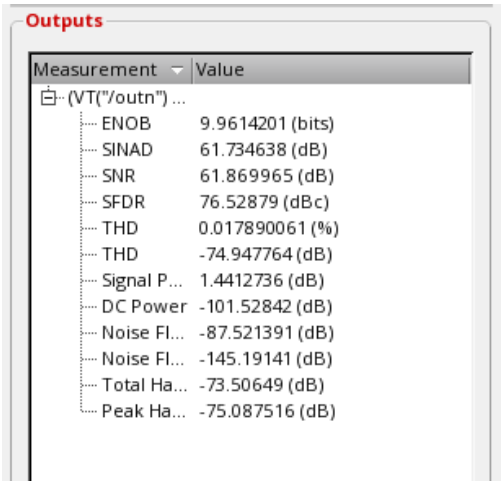


Figure 116: Line Graph of Frequency Spectrum

This plot illustrates the output frequency spectrum of the CAPDAC. This spectrum is used to calculate the SINAD of the CAPDAC.

Below are the corresponding dynamic measurements.



Outputs

Measurement	Value
(VT("/outn") ...	
ENOB	9.9614201 (bits)
SINAD	61.734638 (dB)
SNR	61.869965 (dB)
SFDR	76.52879 (dBc)
THD	0.017890061 (%)
THD	-74.947764 (dB)
Signal P...	1.4412736 (dB)
DC Power	-101.52842 (dB)
Noise Fl...	-87.521391 (dB)
Noise Fl...	-145.19141 (dB)
Total Ha...	-73.50649 (dB)
Peak Ha...	-75.087516 (dB)

Figure 117: Dynamic Measurements

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 110: Exit out of the Spectrum Assistant by clicking the button at the upper-right corner of the screen.

ACTION 111: Click on the "**Auto**" view icon ( ) to view all subwindows. Drag the 1024-FFT output spectrum (the one in blue) from its subwindow into the subwindow that contains the 256-FFT. Click on the **Card** view icon ( ) to show both output spectrums displayed on the same graph, like Figure 118.

Compared to the 256-point FFT (red), the FFT noise floor is lower for the 1024-point FFT (blue). While the total noise has not changed, the noise in the individual frequency bins is lower because each frequency bin is smaller for a 1024-point FFT than for a 256-point FFT.

Figure 118: Spectrum Comparison between 256-point and 1024-point FFT

ACTION 112: Close the ViVA and ADE Explorer windows. Move on to the next lab.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Lab 4-2: Analyze Capacitor Mismatch with Monte Carlo Analysis

In this section, you will re-use the CAPDAP testbench. The goal of this lab is to successfully measure the capacitor mismatch of the CAPDAC. Mismatch is important because it drives an important design trade-off of die area and yield for an ADC. You will use Monte Carlo analysis to characterize the effect of mismatch on the CAPDAC linearity. You will use ADE Explorer to perform a Monte Carlo simulation of the circuit.

ACTION 113: In the Library Manager, select **saradc > Everything > 10Bit_capdac_TB_new**. Double-click on **schematic** to open the schematic.

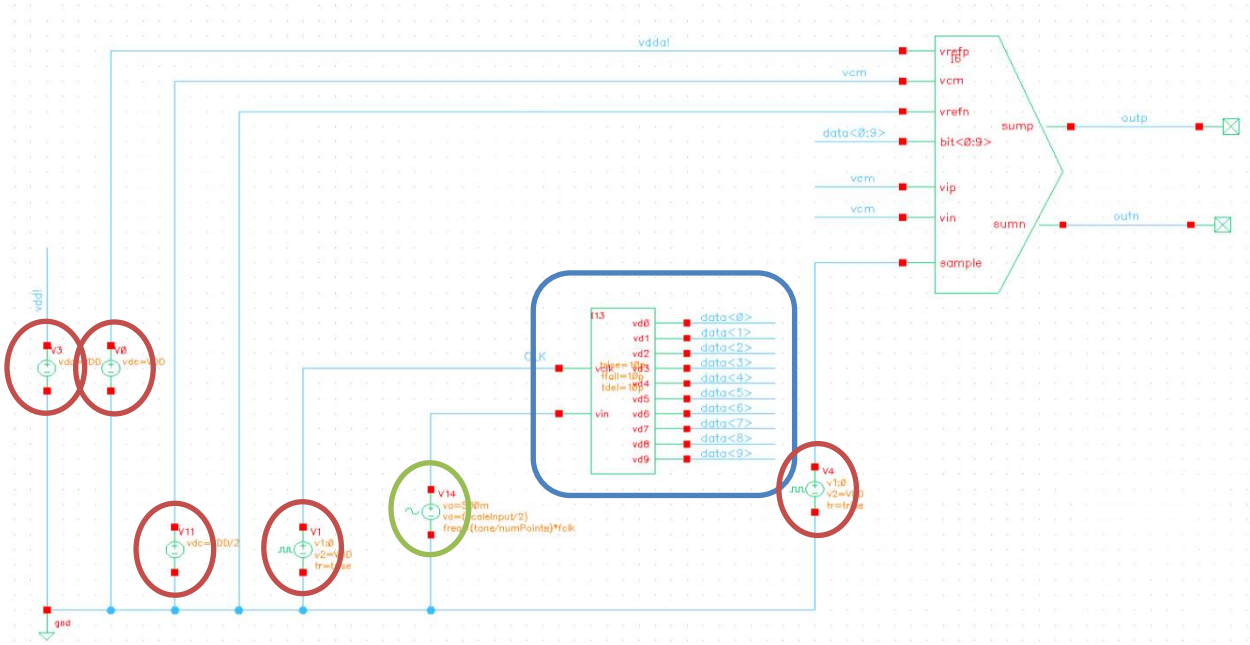


Figure 119: CAPDAC Test Schematic

The testbench includes the CAPDAC power supplies and clock signals (red), sine wave source (green), and an ideal ADC (blue). The CAPDAC itself is in the upper-right corner.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 114: In the schematic window, select **Launch > ADE Assembler**.

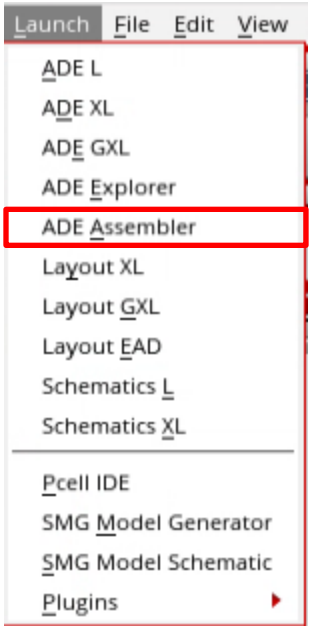


Figure 120: Launch ADE Assembler

ACTION 115: In the **Launch ADE Assembler** window, select **Open Existing View**. Then, click **OK**.

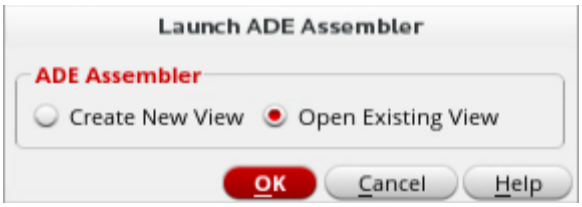


Figure 121: Open Existing Assembler View

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 116: In the **Open ADE Assembler View** window, verify that the view is set to **maestro_mc**. Then, click **OK**.

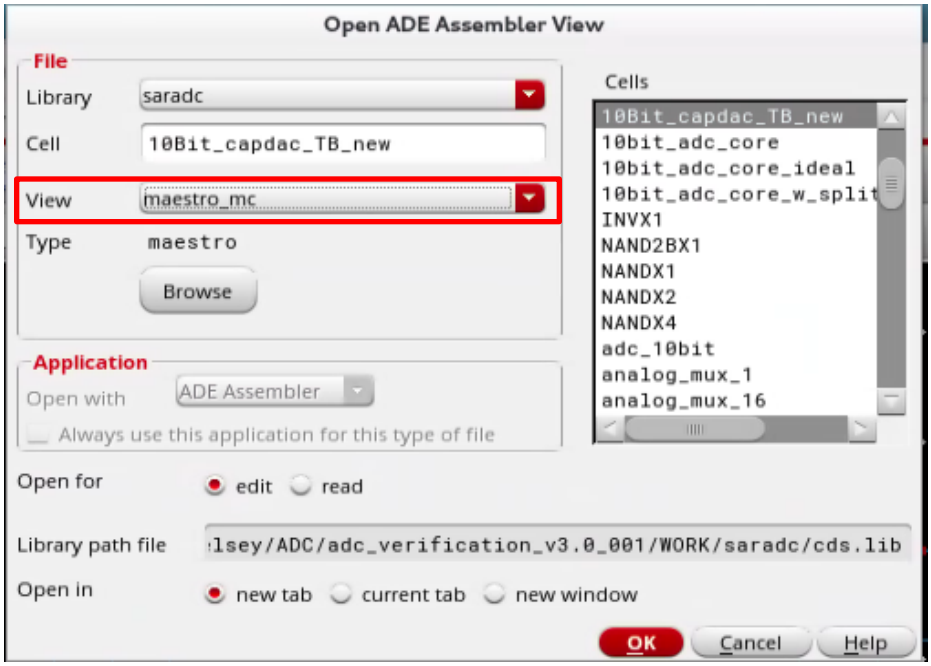


Figure 122: Open Assembler Maestro View

ACTION 117: Once the **maestro** view is loaded, verify that it looks like Figure 123.

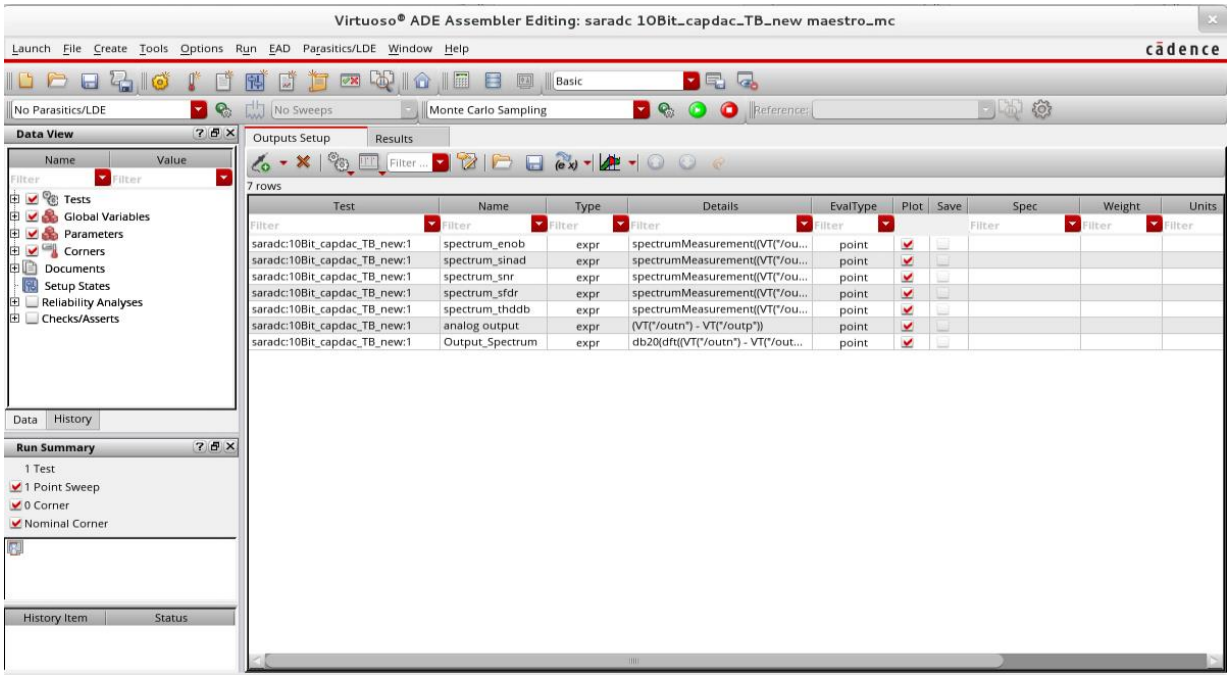


Figure 123: ADE Assembler Window

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 118: Click on the "**+**" symbol next to **Tests** in the Data View Assistant to open up all testbenches being analyzed. Click on the icon to open the Explorer View of the testbench.

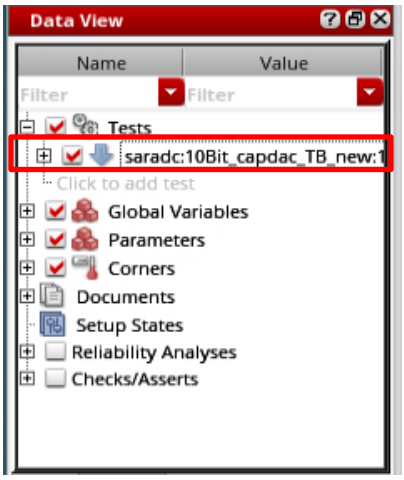


Figure 124: Data View Assistant in ADE Assembler

ACTION 119: LMB-click on the "**+**" symbol next to **Analyses** to view the type of analyses set up in the Explorer View. Double-click on **tran** to open the analysis options.

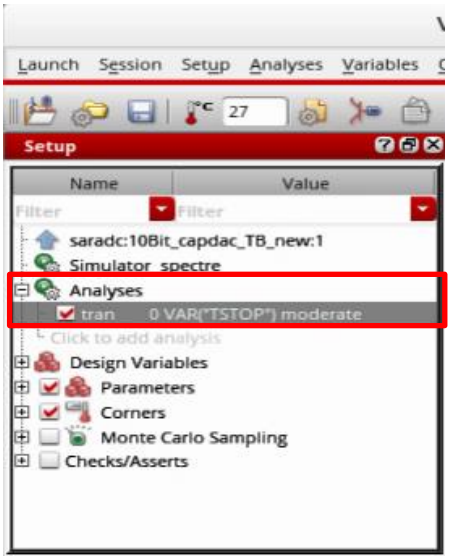


Figure 125: Data View Assistant in ADE Explorer

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 120: In the **Choosing Analyses – ADE Explorer** window, verify that the transient analysis is as shown in the picture below.

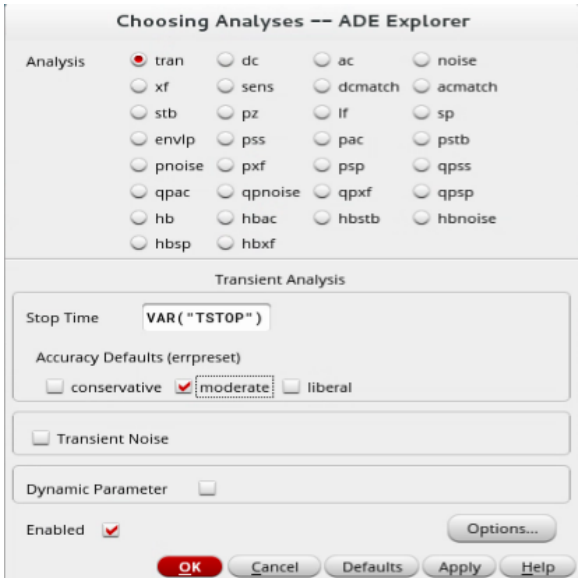


Figure 126: Transient Analysis Setup

Note: Verify that in the Design Variables, **tone = 503** and **numPoints = 2048**.

ACTION 121: Once you are done verifying the setup, click **Cancel** to close the window. Click on the icon to go back to the Assembler maestro view.

ACTION 122: In the ADE Assembler Select a Run Mode, select **Monte Carlo Sampling** from the pull-down menu and then LMB-click on (**Simulation Options**).

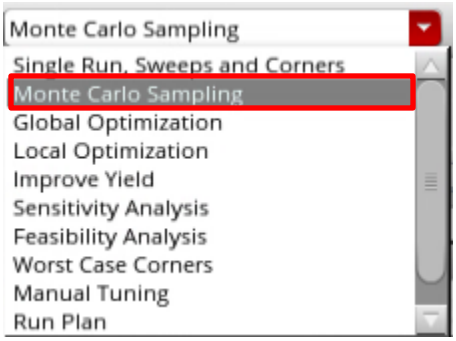


Figure 127: Monte Carlo Sampling

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 123: In the **Monte Carlo** simulation options window, set up the analysis as shown in Figure 128. Then, click **OK**.

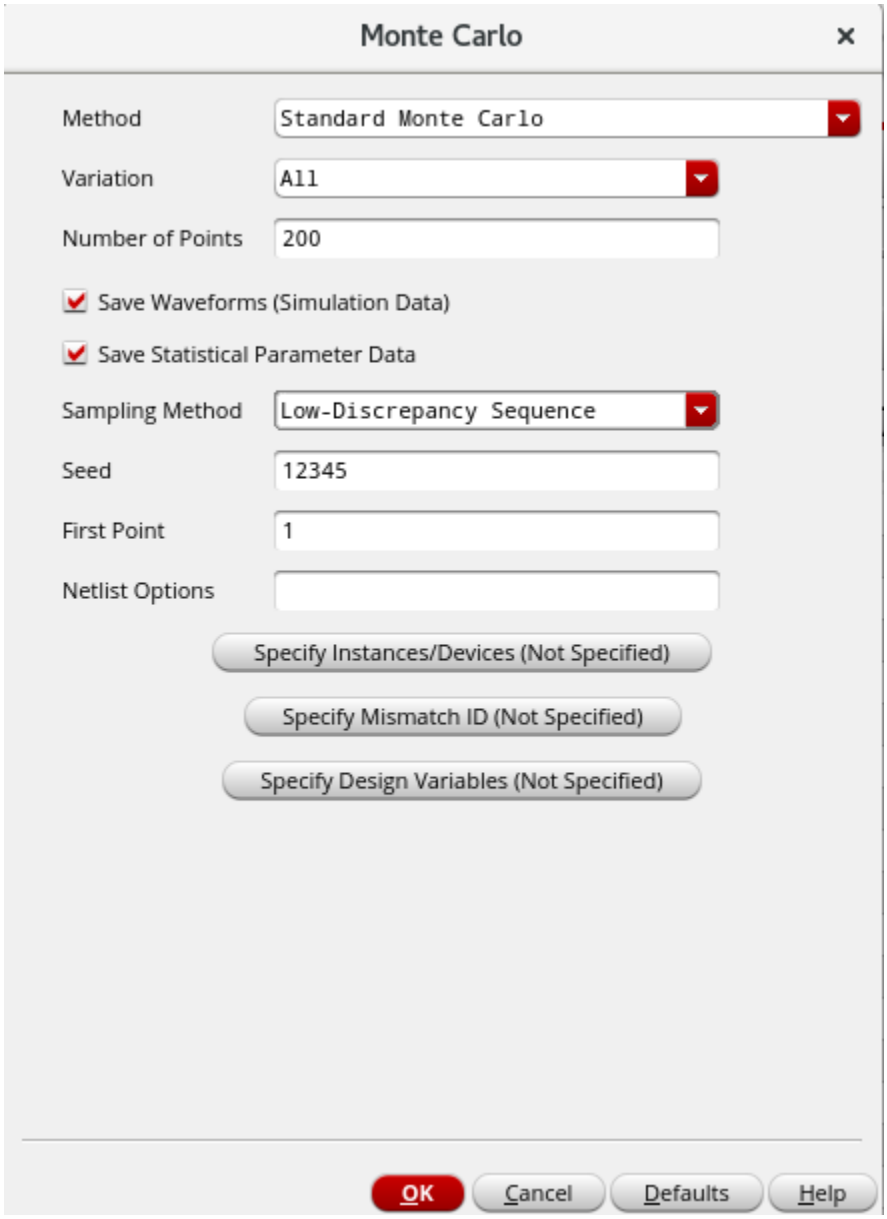


Figure 128: Monte Carlo Setup

The **Guided Mode** section provides guidance in terms of the task you need to perform using Monte Carlo. The **Advanced – Show All Options** section provides a relevant set of options to be set based on the task selected in the "**Guided Mode**" section.

You select **Variation = All** because you want to include both process and per-instance (mismatch) statistical variations.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

A Monte Carlo sampling method called Low Discrepancy Sampling (LDS) will be used. The LDS sampling combines the best features of Random Sampling, support for Monte Carlo auto-stop, and Latin Hypercube Sampling (fast convergence).

Typically, you would select a large number, say 2000 points, and set the target yield and confidence level to automatically stop analysis when it either fails or passes. In this case, you are trying to derive the target specification; so, you have chosen an intermediate number of points, 200. Saving the statistical data allows you to do post processing to analyze statistical dependence with sensitivity analysis.

Monte Carlo simulations are long; so, you will not run the simulation. Instead, you will load previous simulation results and explore them.

ACTION 124: In the ADE Assembler Data View Assistant, LMB-click on the "**History**" tab.

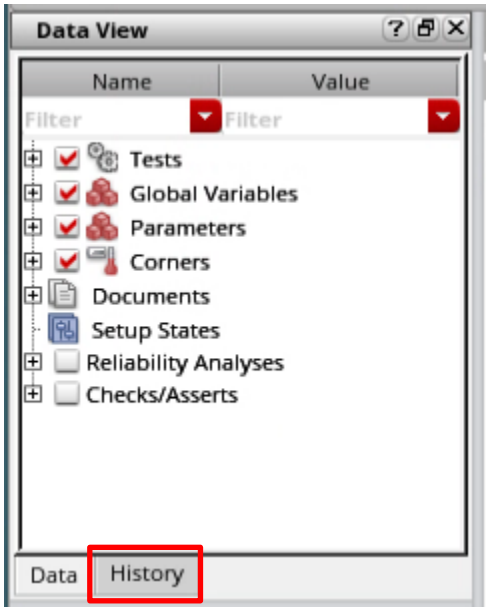


Figure 129: History Tab of Data View Assistant

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 125: In the **History** tab, RMB-click on **SINAD** and select **Load Setup to Active** from the menu.

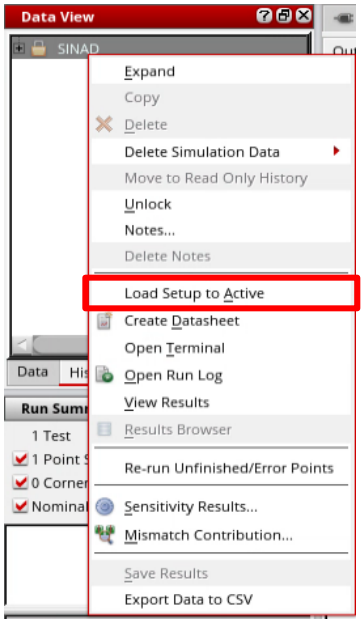


Figure 130: Load Setup to Active

ACTION 126: Verify that the setup of SINAD was loaded to the active window as shown in Figure 131.

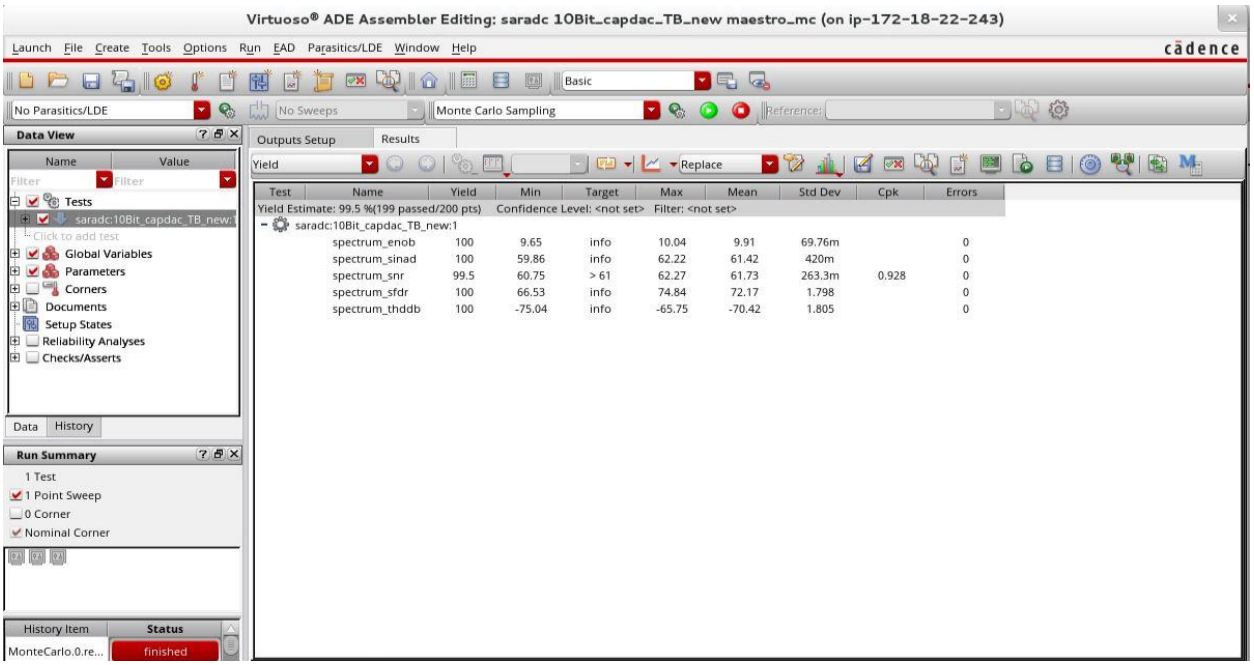


Figure 131: ADE Assembler with SINAD Loaded

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 127: Select the **Histogram** icon ( ), and then select **saradc:10Bit_capdac_TB_new:1 > Histogram** from the pull-down menu. This will open the **Plot Histogram** form.

Figure 132: Histogram of Results

ACTION 128: In the **Plot Histogram – SINAD** window, set up the histogram as shown in Figure 133 below and click **OK**.

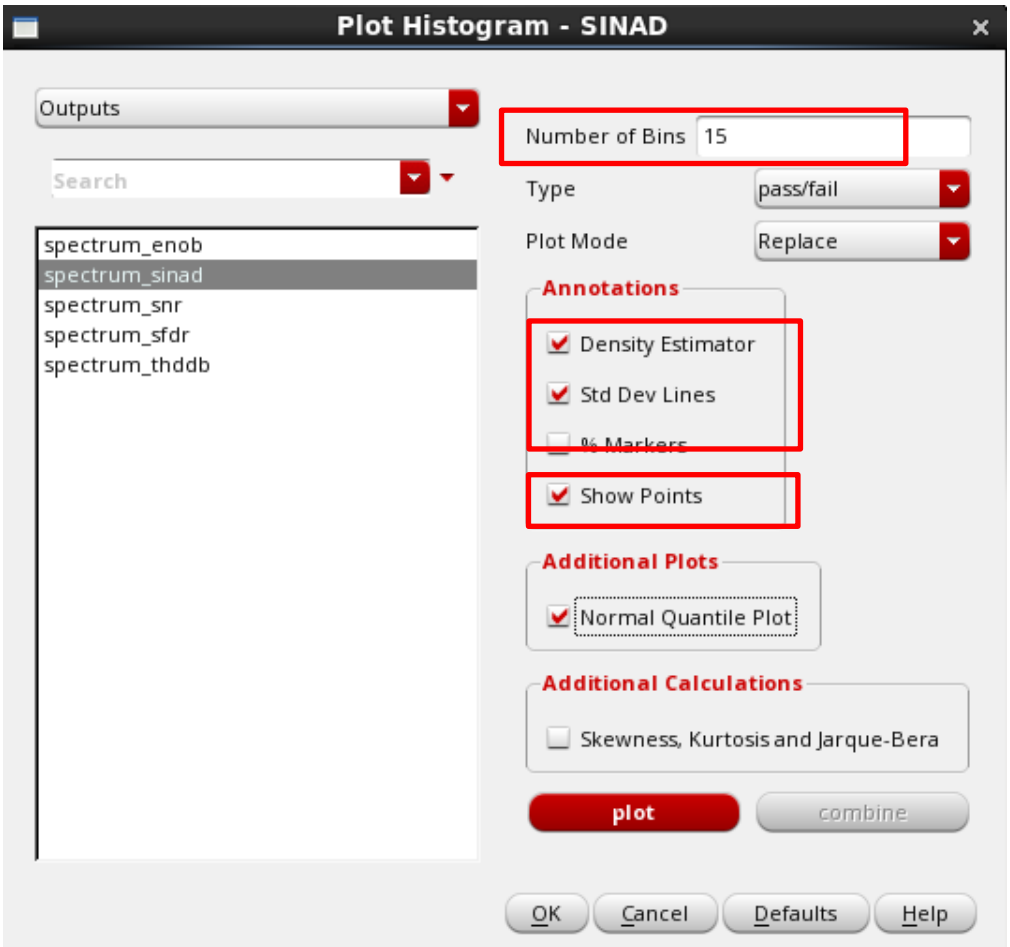


Figure 133: Plot Histogram Options

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The **Number of Bins** corresponds to the number of bars to plot in the histogram. You are using 15 bars for this histogram.

In the **Annotations** section, by selecting **Density Estimator**, the histogram plot will display a density estimator curve. By selecting **Std Dev Lines**, the plot will display the standard deviation lines. Finally, selecting **Show Points** will display each data point overlaid with the bar graph of the histogram.

By Selecting **Normal Quantile Plot**, an additional plot will be generated to show the Normal Quantile plot of the SINAD.

ACTION 129: A new ViVA window will open and display both a histogram and Normal Quantile graph for the CAPDAC SINAD. LMB-click on (**Card**) and subwindow 1 to display the histogram.

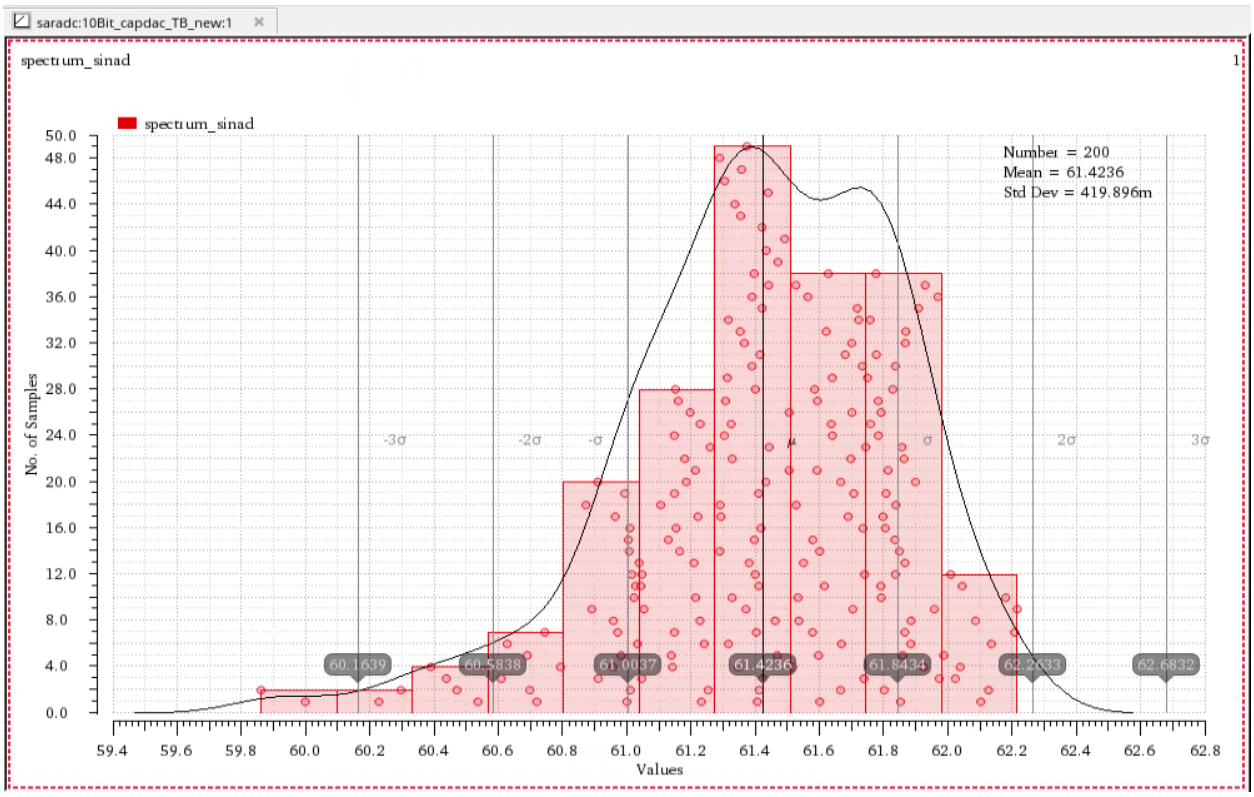


Figure 134: Histogram for the CAPDAC SINAD

The histogram for the CAPDAC SINAD is shown above. The mean value of the SINAD is **~61.4dB** and the standard deviation is **~420m dB**. You will use the standard deviation in the ADC spreadsheet to account for the effect of CAPDAC process variation on overall performance.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 130: Change the subwindow to 2 to display the Normal Quantile plot.

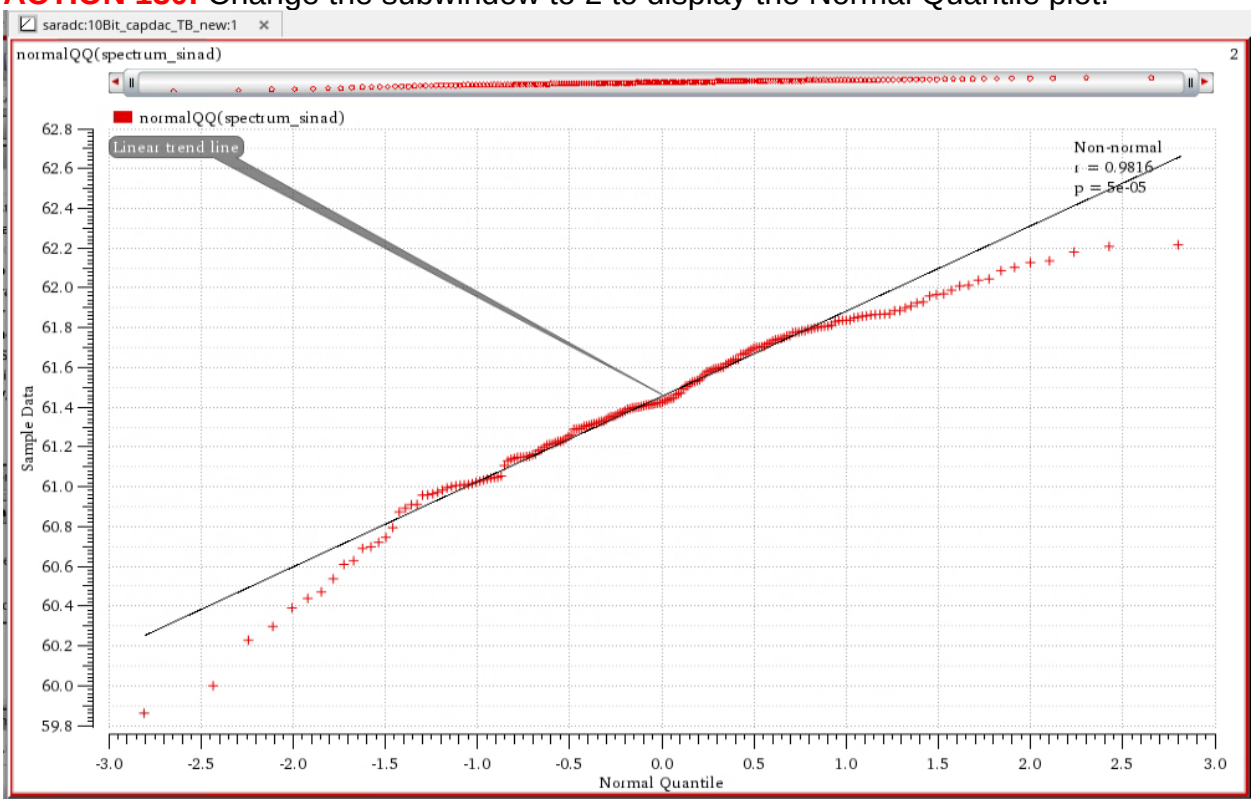


Figure 135: Normal Quantile Plot for the CAPDAC SINAD

Looking at the histogram distribution, it appears to have a long tail. You can use the Normal Quantile plot to evaluate the tails of the distribution and determine whether the distribution is Gaussian.

In a Normal Quantile plot, the X-axis is warped so that it is linear in terms of sigma (standard deviation). If the data is plotted and it is linear, the distribution is linear. The Normal Quantile plot shows that the distribution is not Gaussian. A Gaussian distribution is a straight line in a Normal Quantile plot. The result shows that the distribution is larger than what would be expected if it was Gaussian.

ACTION 131: After you are done looking at the results, close the ViVA window. Next, you will consider how the number of Monte Carlo iterations effects the results.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 132: Select the **Histogram** icon ( ), and then select **saradc:10Bit_capdac_TB_new:1 → Plot/Print Versus Iteration** from the pull-down menu. This will open the **Plot/Print Versus Iteration** form.

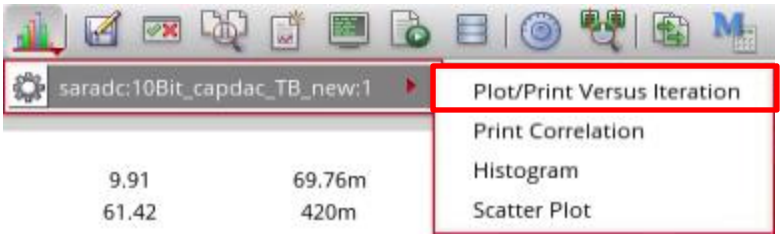


Figure 136: Plot/Print Versus Iteration

ACTION 133: In the **Plot/Print Versus Iteration – SINAD** window, set up the plot as shown in Figure 137 and click **OK**.

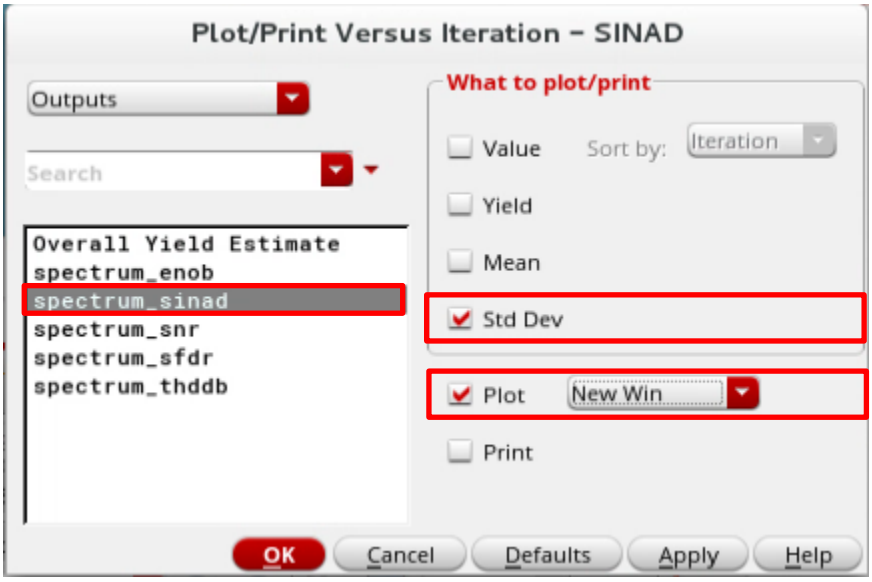


Figure 137: Plot/Print Versus Iteration – SINAD

You are trying to plot the standard deviation of the SINAD versus the Monte Carlo iteration; that is why you choose **spectrum_sinad** and **Std Dev**. **spectrum_sinad** is the expression used in this lab to calculate the SINAD of the CAPDAC. You also plot this graph on a new window because you are not comparing it to any other set of data.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 134: Verify the plot of the value of SINAD standard deviation versus iteration with Figure 138 below.

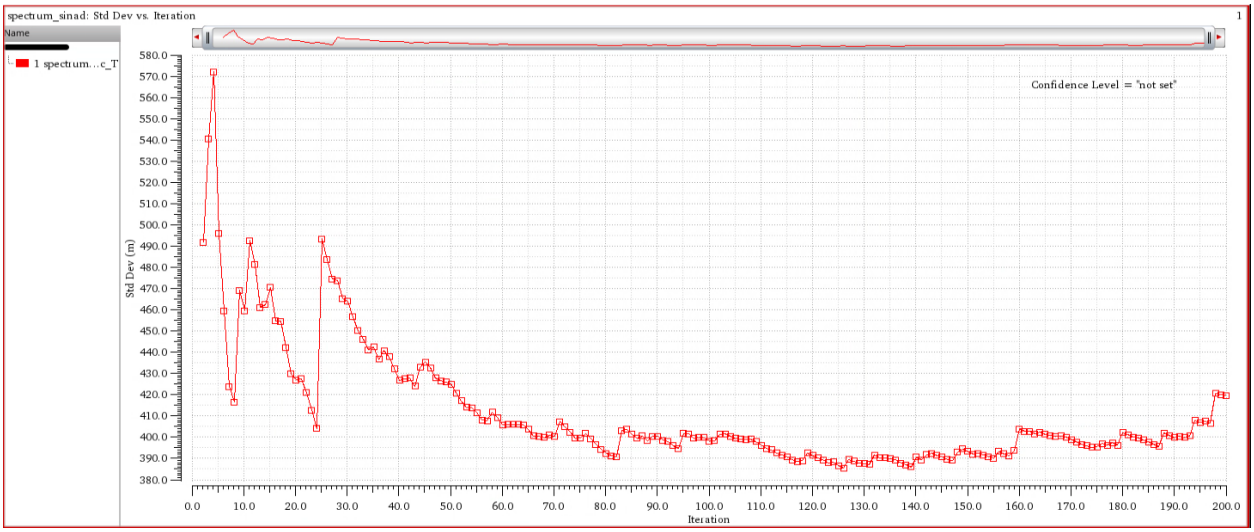


Figure 138: SINAD Standard Deviation Versus Iteration Plot

While you can eyeball the plot and say that it seems close to converged after 50 Monte Carlo iterations, the more mathematically rigorous approach would be to use Monte Carlo auto-stop. Auto-stop gives you the ability to set the interval of confidence for the yield. The analysis would then stop when it has been determined that the yield target can or cannot be satisfied based on the desired level of confidence. This approach is useful particularly when there are issues, since the Monte Carlo analysis is stopped before wasting too much time on a design with an issue. In this case, you did not use auto-stop so that you could look at how the standard deviation converges.

ACTION 135: Close the waveform plot and ADE Assembler windows, and LMB-click on the (**Exit**) icon in the upper-right corner. Select "**No**" if asked to save any changes.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Module 5: Clock Generator Characterization

In this section, you will analyze one of the most important sources of noise: clock jitter. There are three sources of jitter that you will consider:

1) Reference clock jitter: The designers need to specify the maximum clock jitter the design can tolerate. This specification is used to constrain the clock source provided by the system to the ADC.

2) Clock generator noise: The design includes a non-overlapping clock generator that generates the clock signals used by the different components in the design. The devices, transistors, in the clock generator generate noise, thermal, 1/f, etc. and this noise causes the clock jitter to increase.

3) Clock generator power supply-induced jitter: The power supply voltages used by the clock generator are noisy and produce a ripple from the switched mode power supply. The noise on the power supply changes the operating point of the devices and these changes increase the jitter of the clocks.

Jitter of the sampling clock is one of the major components of ADC noise. In the design specification for the ADC, the sampling clock jitter is specified. However, the additional jitter due to the clock phase generator also needs to be considered. There are two sources of noise that you will consider. The first phenomena you will consider is the noise that the clock phase generator introduces to the sampling clock. The second phenomena you will consider is the jitter due to noise on the power supply.

The schematic for the clock phase generator is shown in Figure 139.

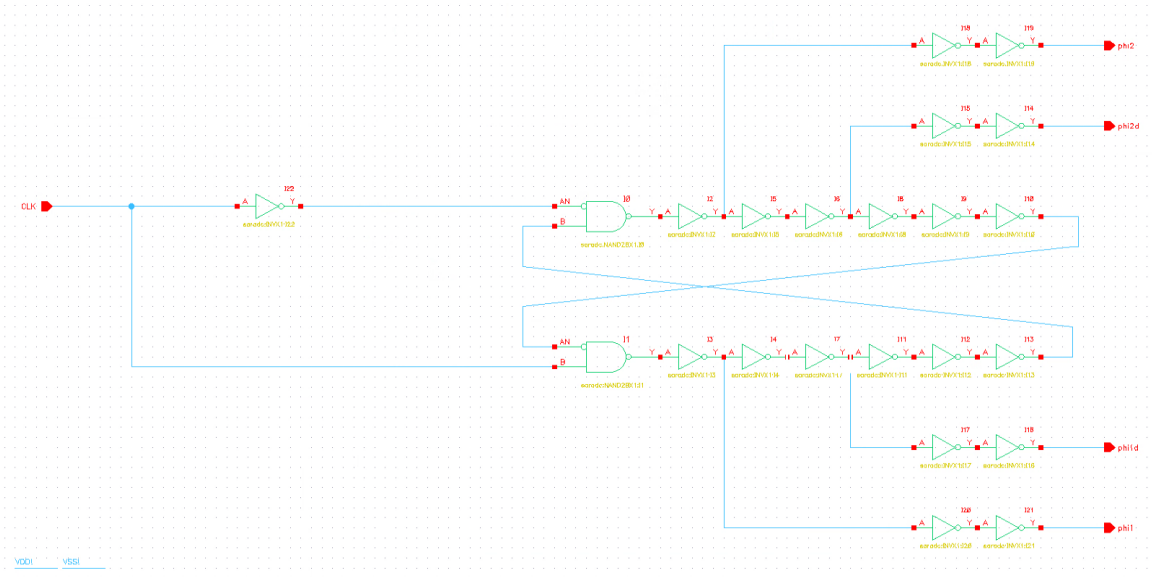


Figure 139: Clock Phase Generator

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Lab 5-1: Analyzing the Clock Generator Noise

The noise in the clock phase generator causes jitter, and you will use the same method for noise analysis as you used for analyzing the noise of a dynamic comparator, Spectre RF PNOISE jitter analysis. The testbench for clock phase generator is shown below.

ACTION 136: In the Library Manager, select **saradc > Everything > clockPhaseGenerator_TB_new**. Double-click on **schematic** to open the schematic.

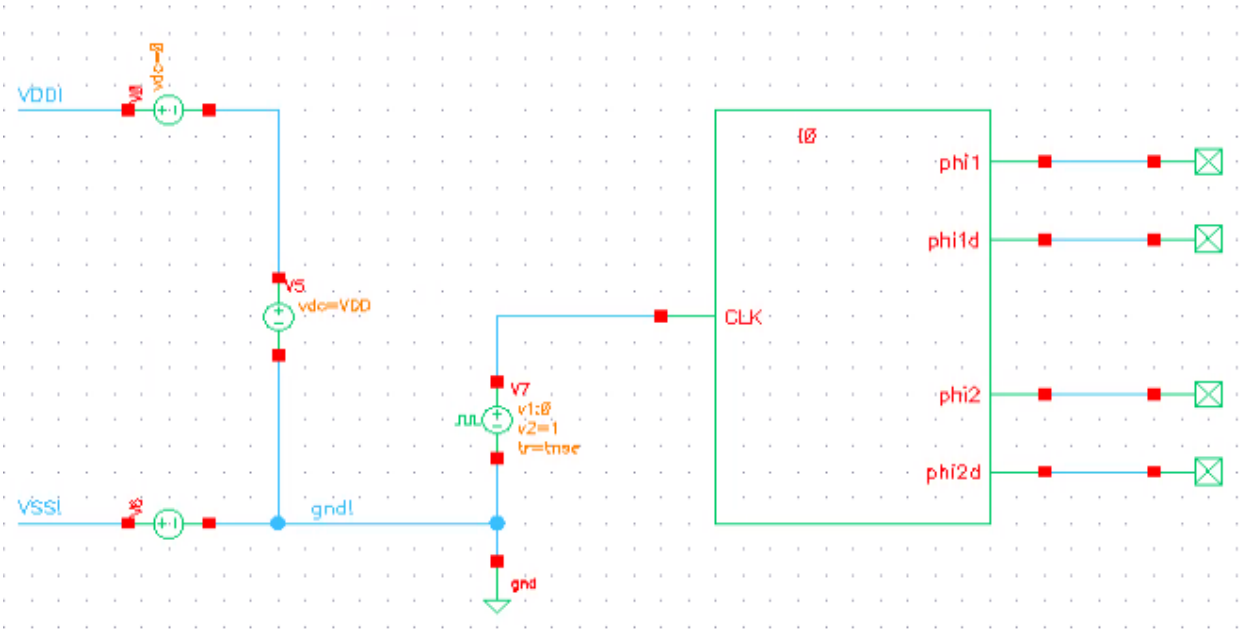


Figure 140: Clock Phase Generator Testbench

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 137: In the schematic window, select **Launch > ADE Explorer**.

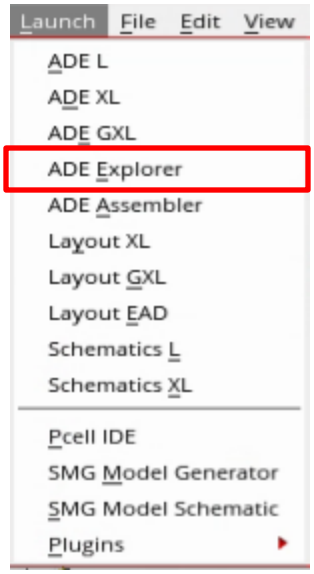


Figure 141: Launch ADE Explorer

ACTION 138: In the **Launch ADE Explorer** window, select **Open Existing View**. Then, click **OK**.

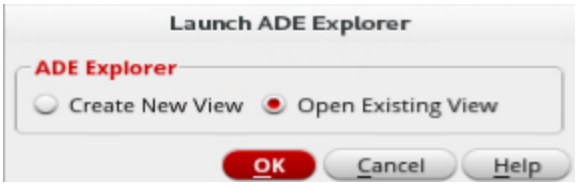


Figure 142: Open Existing Explorer View

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 139: In the **Open ADE Explorer View** window, verify that the view is set to **maestro**. Then, click **OK**.

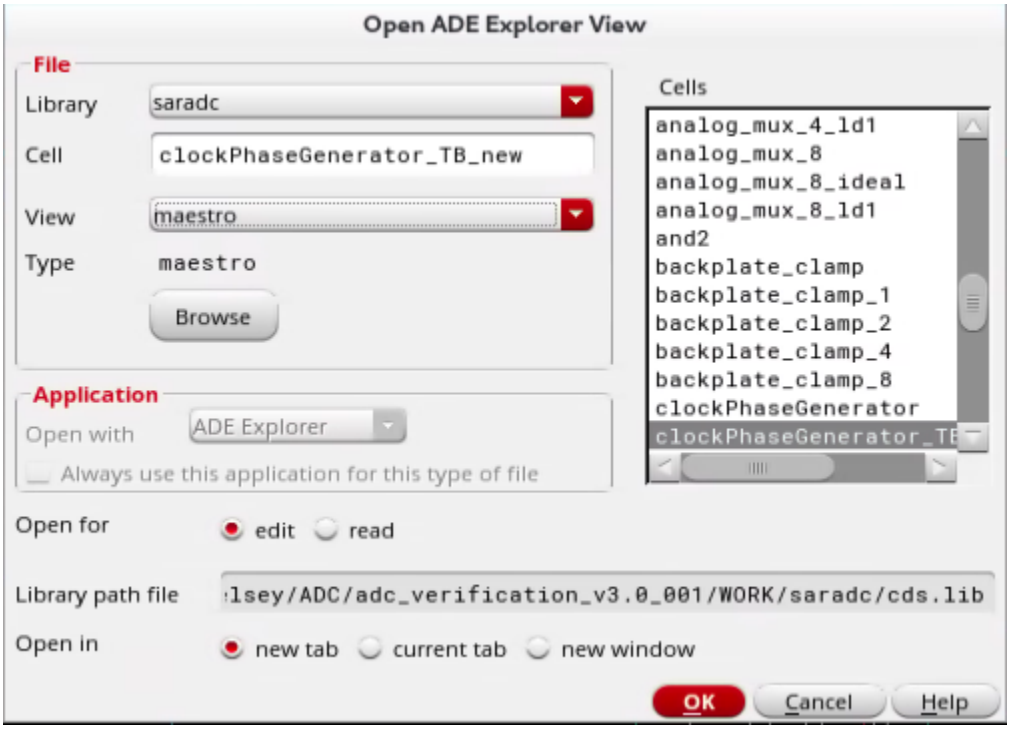


Figure 143: Open Explorer Maestro View

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The **maestro** tab should look like Figure 144 below.

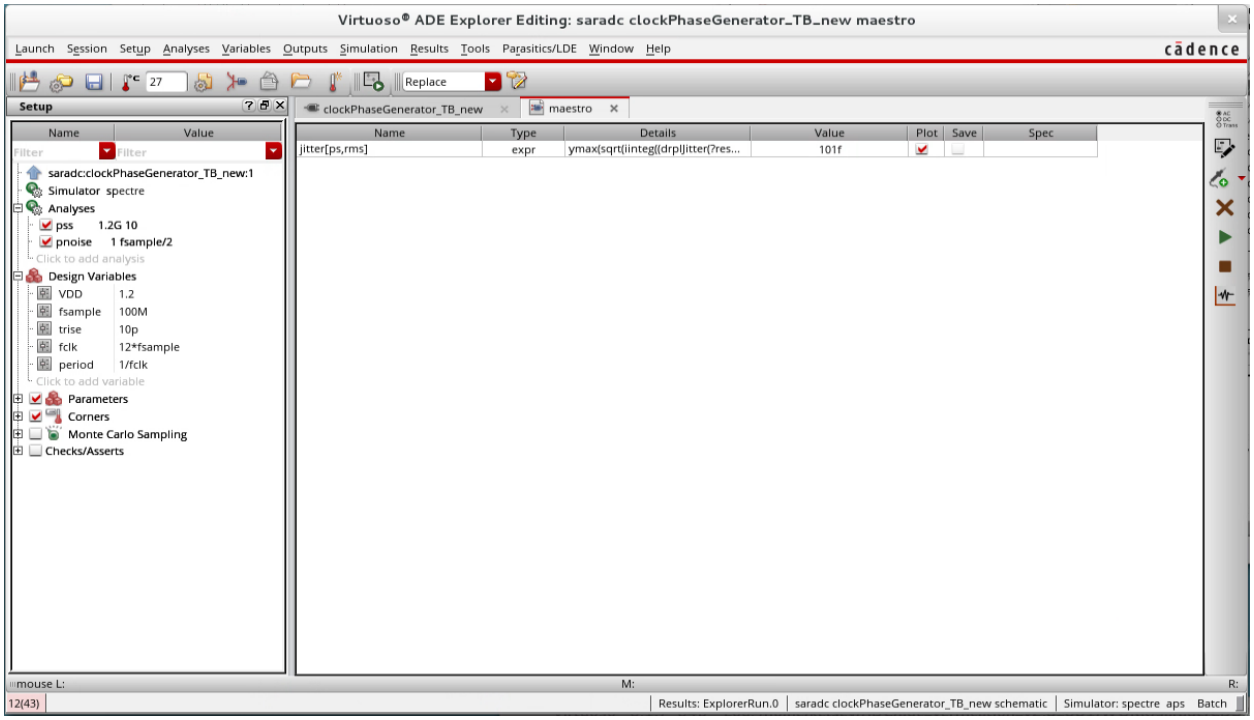


Figure 144: ADE Explorer Window

ACTION 140: RMB-click on **pss** and select **Edit** from the pop-up menu.

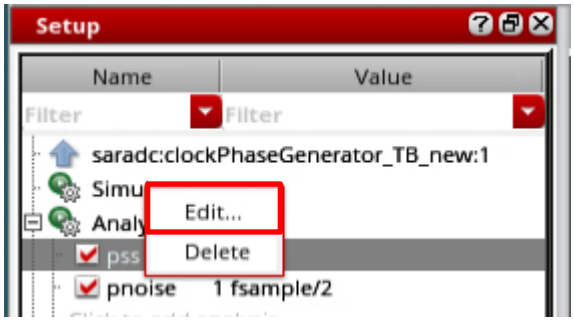


Figure 145: PSS Analysis Options

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Verify the pss analysis setup with Figure 146. Make sure that **VAR("period")** is inside the **Stop Time (tstab)** space. Click on **Apply** when done.

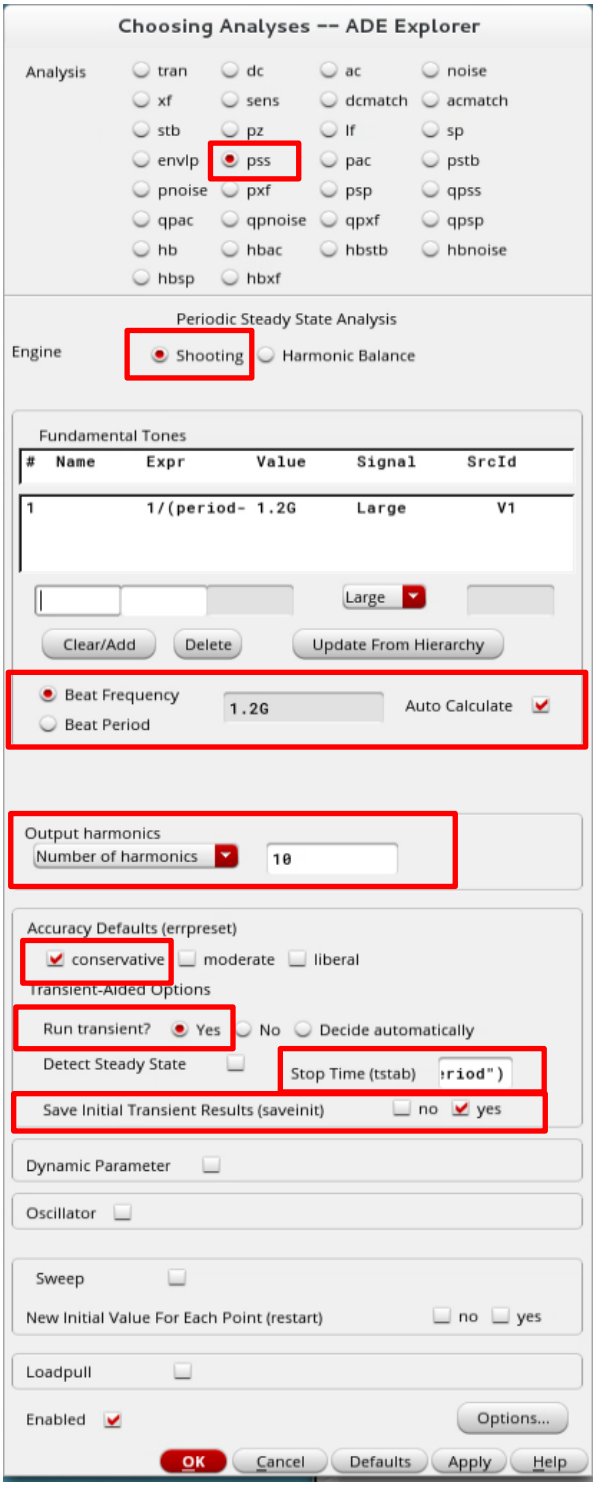


Figure 146: Choosing Analyses – ADE Explorer: PSS

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

pss – Periodic state analysis is used to calculate the response of a circuit to large input signals.

Shooting Newton – This is an iterative time-domain algorithm that is akin to solving a boundary-value problem by reducing it to the solution of an initial-value problem.

Beat Frequency – This is the periodicity of the system. When the **Auto Calculate** option is selected, ADE calculates the periodicity of the circuit and displays it in the **Beat Frequency** field.

Number of harmonics – This parameter specifies the number of harmonics to calculate in the Fourier transform that is run after the time-domain simulation to calculate the harmonics of the large-signal response. 10 is the best number because any higher number will result in more time points being forced in the pss solution to keep the accuracy of the frequency-domain results very high.

Accuracy Defaults (errpreset) – You use **conservative** because you want high-accuracy results. "**moderate**" is the general-purpose setting, and "**liberal**" is not recommended.

Run transient? – When you choose "**Yes**", the **Detect Steady State** option is enabled automatically and the simulation can process to pss without finishing the tstab simulation.

tstab – This is the user-defined stabilization time interval. You will be using the design variable "period" to define the time interval. A longer tstab allows the circuit to get closer to steady-state and thus, converge better.

You also want to save your initial transient results; so, you select "**Yes**" for this part.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 141: Select **pnoise** analysis and make the setup window resemble Figures 147 and 148. Click on "**Apply**" when done.

Note: The following two screenshots constitute the total analysis window for the pnoise.

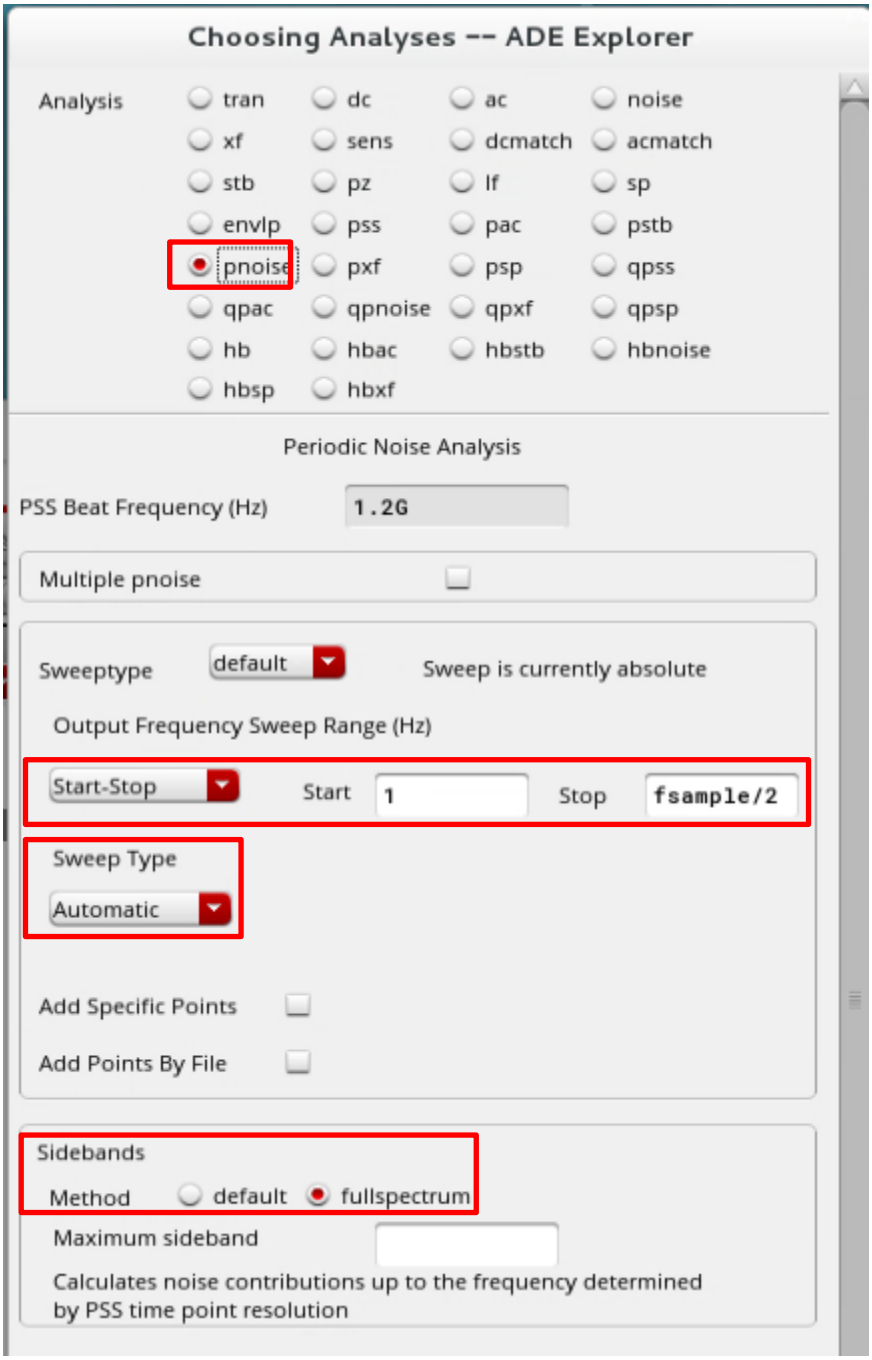


Figure 147: Upper Half of Choosing Analyses – ADE Explorer: Pnoise Window

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Since the fundamental quantity that is calculated by the noise analysis is the noise at the output, the frequency range is always the output frequency range. The output frequency range corresponds to the Nyquist rate of the ADC, **fsample/2**. fsample is the ADC conversion rate. By setting the **Stop Time** to **fsample/2**, the noise analysis will not have any effects from noise folding.

The **Sidebands Method** is set to **fullspectrum** because this method is especially useful for circuits where aliasing occurs through very high harmonics of the clock. The run-time advantages of using full spectrum are large with no loss in accuracy of the result. Using the full spectrum method setting allows pnoise to calculate the noise translations through the full spectrum of the pss analysis. Noise separation is not supported when full spectrum is selected.

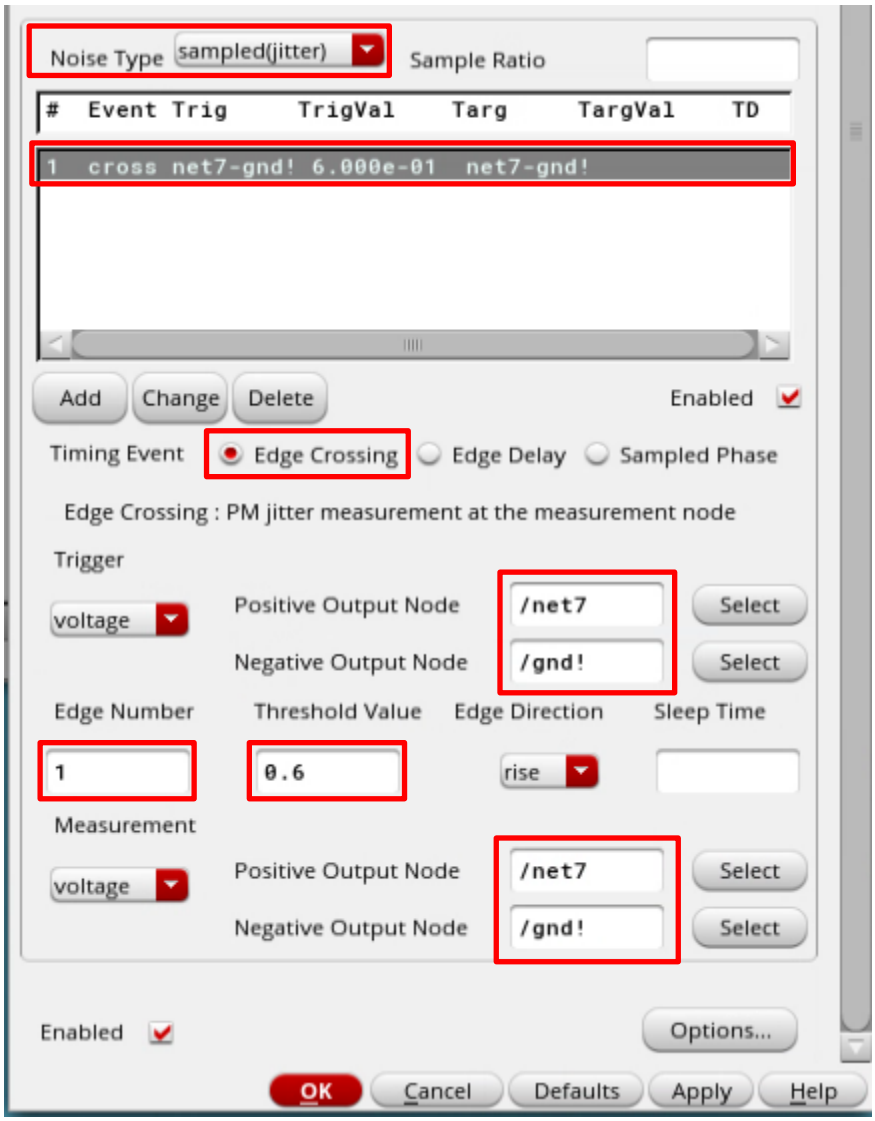


Figure 148: Lower Half of Choosing Analyses – ADE Explorer: Pnoise Window

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Sampled(jitter) – When the **noisetype** is set to **jitter**, an instantaneous measurement of the noise at the threshold crossing in the pss analysis is calculated. **Sampled (jitter)** has three modes, and the mode you will be using is **Edge Crossing**.

Edge Crossing is where you specify a threshold and direction called the trigger, which defines the time when the noise measurement is made. In many cases, the measurement net will be on the same net as the trigger, as shown in the diagram above, but the measurement can be made on another net in the circuit. This is useful when you want to make a noise measurement on the output of a switched-capacitor filter triggered by the clock waveform.

In this RAK, the trigger is the “rise” edge of the output net as it crosses VDD/2 and the noise is measured on the same net.

The threshold value is set to **0.6** because that is half as much as VDD (1.2V). The edge direction is set to “**rise**” because the signal is first low and then goes high, and you want to capture that value change when it happens. The positive output node for both the **Trigger** and **Measurement** values is set to “**/net7**” because that is the output of the CAPDAC.

ACTION 142: To populate the sampled jitter window with a trigger event, fill out the lower half of **Choosing Analyses** window as shown above and click on “**Add**” when done. This should create a trigger event as shown in Figure 148.

ACTION 143: To start the simulation, select the (“**Run Simulation**”) button.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 144: After the simulation is complete, select **Results > Direct Plot > Main Form** to open the **Direct Plot Form**.

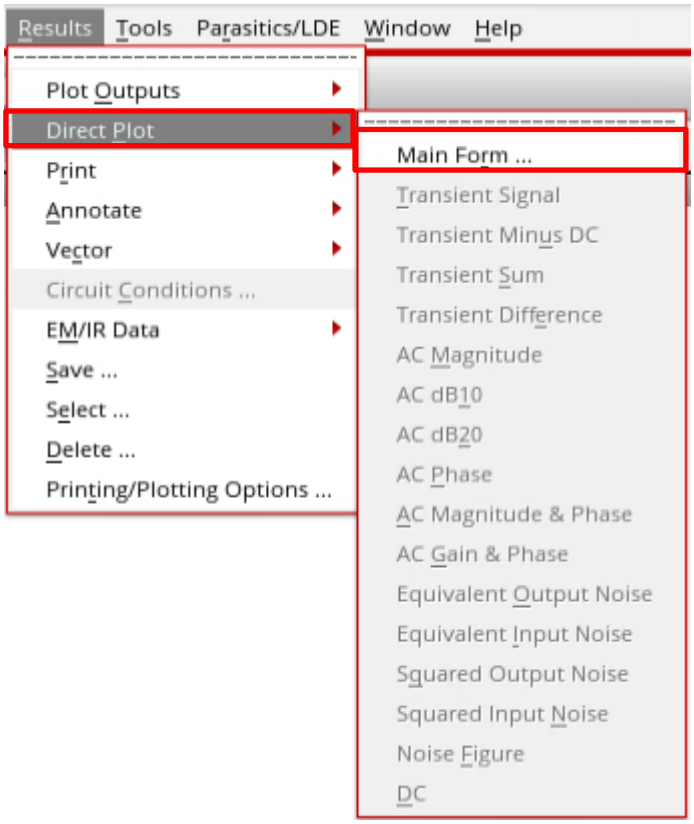


Figure 149: Direct Plot Main Form

ACTION 145: In the **Direct Plot Form**, select **pnoise sampled** and **Jee**. Also make sure that **Signal Level** is set to **rms** and **Modifier** is set to **Second**. Fill in **Integration Limits** with **Start Frequency (Hz) = 1** and **Stop Frequency (Hz) = 50M**. The reason for these integration values is that you want the whole frequency spectrum from 1Hz to fsample/2= 50 MHz. This will eliminate any chance of noise distortion from noise folding. Make sure that "**Add to Outputs**" is enabled. Verify that the window looks like Figure 150.

Note: Jee outputs the edge-to-edge jitter. Jee is a scalar.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

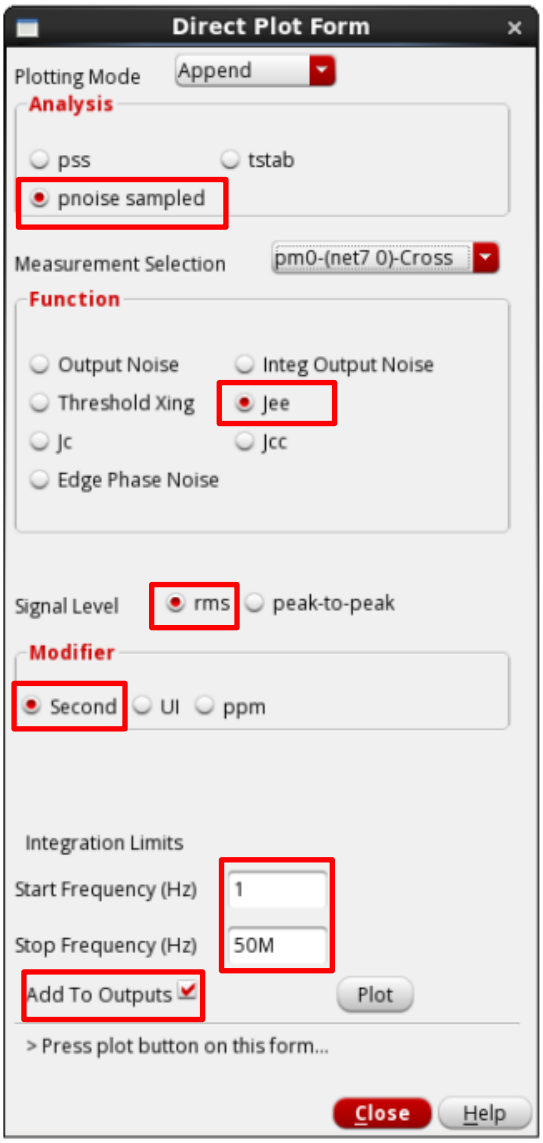


Figure 150: Direct Plot Form

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 146: Switch back to the ADE Explorer window. It should look like Figure 151 below.

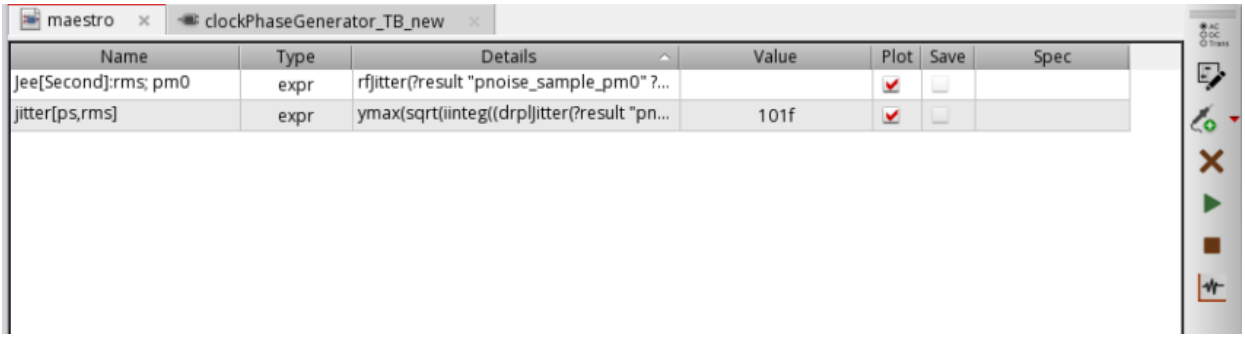


Figure 151: ViVA Calculator with Jee Expression

You will build the expression to calculate the total noise and compare the **Jee[Second]** expression result comes from the pop-up window that is created when you enable "**Add to Outputs**" button and first plot the Jee output. This expression is already in the maestro view (**jitter[ps,rms]**), but the following actions will go step by step to explain it. The total noise is the root-sum-squared of the spot noise. The equation for calculating the total integrated noise is:

$$Total_Jitter = \sqrt{\int (output_jitter)^2}$$

ACTION 147: Click on the Add Outputs button in ADE Explorer window

.

Double -click on the empty expression and click on the **Expression Builder** icon ( ). The Expression Builder should appear as shown in Figure 152.

Figure 152: Expression Builder form

ACTION 148: Add the following expression in the Expression Builder.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

drplJitter(?result "pnoise_sample_pm0" ?unit "Second" ?k 1 ?event 0)

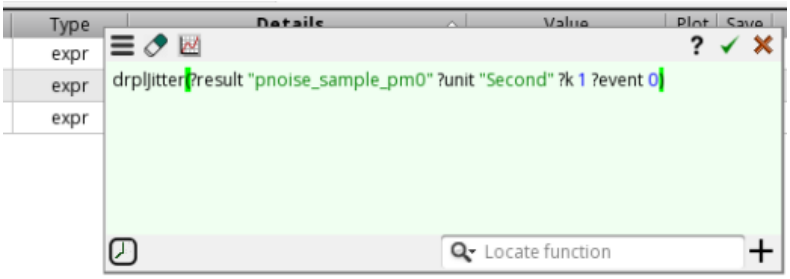


Figure 153: Expression for Total Jitter

ACTION 149: At the end of the expression, add a ****2** that corresponds to the square in the Total Noise equation. At the beginning of the expression, add the function "**iinteg**" to compute the indefinite integral of the expression.

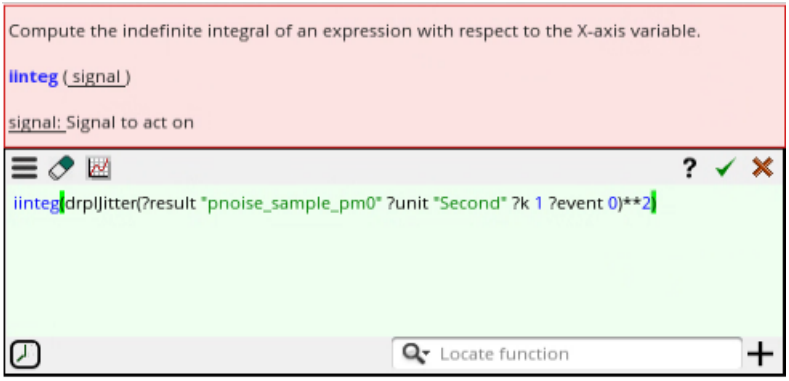


Figure 154: Expression Builder with iinteg() Function

ACTION 150: At the beginning of the new expression, add the function "**sqrt**" to compute the square root of the function.

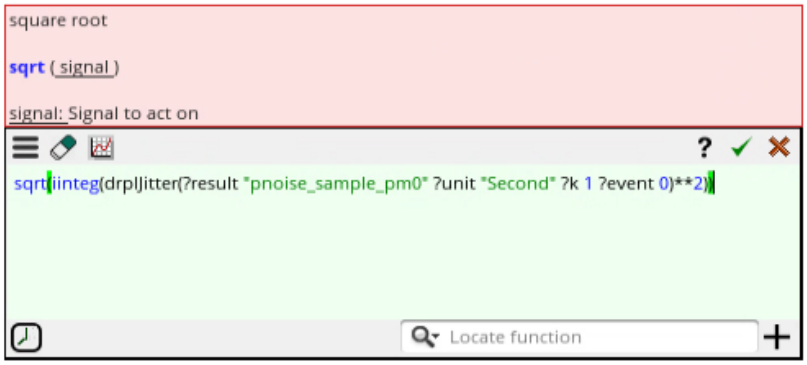


Figure 155: Expression Builder with sqrt() Function

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 151: At the beginning of the new expression, add the function "**ymax**" to return the y value at the maximum.

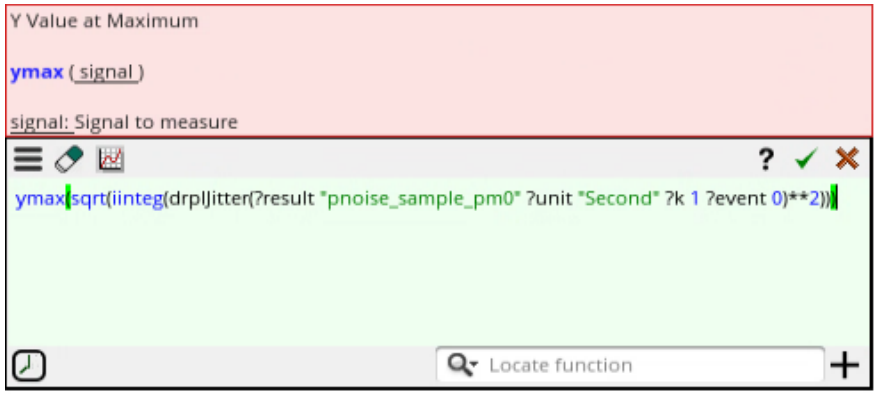
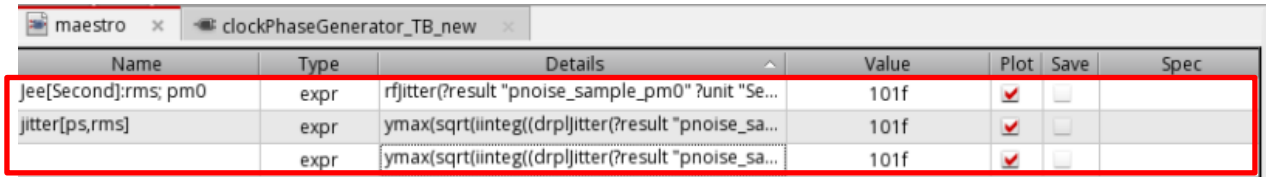


Figure 156: Expression Builder with ymax() Function

ACTION 152: Click on the little green arrow ( ) at the top-right side of the window to accept the expression.

ACTION 153: Click on the "**Plot Outputs**" ( ) button to re-evaluate the expression you just created. It should match the first expression in the **Outputs** tab of the Explorer window.

ACTION 154: After the expressions are re-evaluated, you will see how the total jitter noise was calculated for the clock generator. The jitter noise value calculated in this lab will be used to compute the ADC's ENOB and SINAD.



Name	Type	Details	Value	Plot	Save	Spec
Jee[Second]:rms; pm0	expr	rfjitter(?result "pnoise_sample_pm0" ?unit "Se...	101f	✓		
jitter[ps,rms]	expr	ymax(sqrt(iinteg((drpljitter(?result "pnoise_sa...	101f	✓		
	expr	ymax(sqrt(iinteg((drpljitter(?result "pnoise_sa...	101f	✓		

Figure 157: Jitter Noise of Clock Generator

Note: The first equation shown above comes from the pop-up window that is created when you first plot the Jee output. It comes automatically when you select "**Add to Outputs**" in the **Main Form** window.

ACTION 155: Close the waveform plot and ADE Explorer windows by clicking on the (Exit) button in the upper-right corner. Select "**No**" if asked to save any changes.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Lab 5-1.1 [Optional] Analyzing Clock Power Supply Noise Rejection with Transient Analysis

In this section, you will look at the effect of power supply noise on the clock jitter. In many systems, the power supply voltages are generated by a switched-mode power supply (SMPS) or dc-to-dc converter from the battery voltage. The ripple on the power supply voltage contributes to the clock jitter.

The testbench consists of the clock generator with a noisy power supply voltage. To model the noisy power supply voltage, you performed a transient simulation of the dc-to-dc converter and saved the results as a text file. When simulating, you will use a voltage source option that enables simulation of text files. The file is a *.csv file, in this case, generated by the ViVA Waveform Calculator.

The simulation time is a trade-off; it needs to be long enough to correctly model the ripple variation and short enough to complete in a reasonable amount of time. The dc-to-dc converter is over 1MHz; so, 10 clock periods was chosen as the transient analysis time (that is, transient stop time is 10µs). The ADC clock rate is 1.2GHz; so, the effect of power supply jitter will be averaged over ~10,000 clock periods, and this should provide a good estimate of the jitter.

To measure the jitter, you will create an eye diagram and measure the variation of the rising clock edge.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 156: In the Library Manager, select **saradc > Everything > clockPhaseGenerator_dc2dc_TB_new**. Double-click on **schematic** to open the schematic.

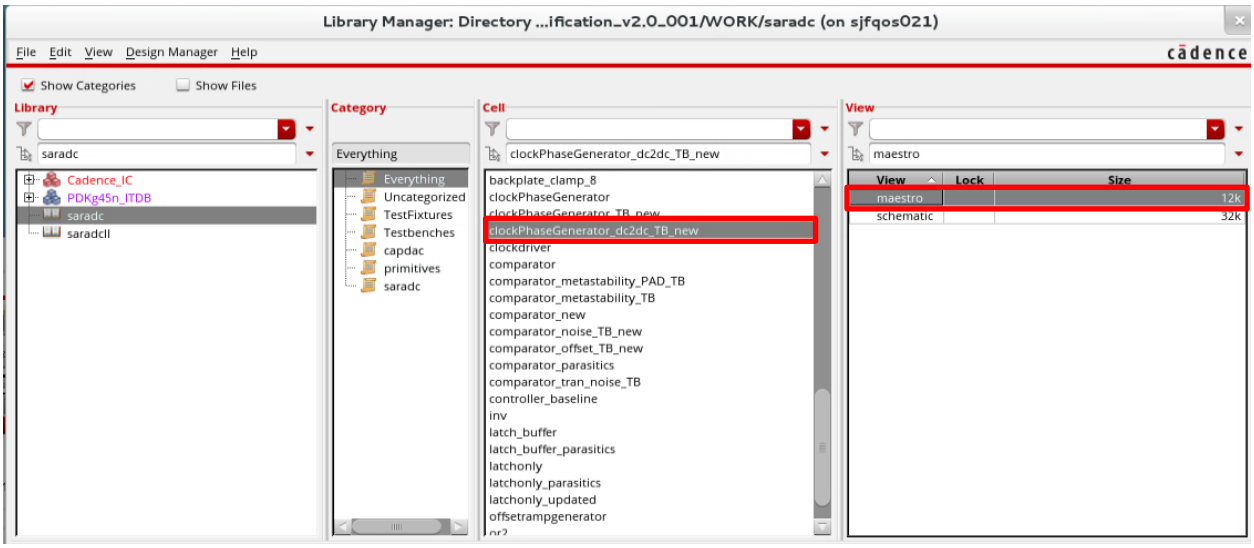


Figure 158: Library Manager

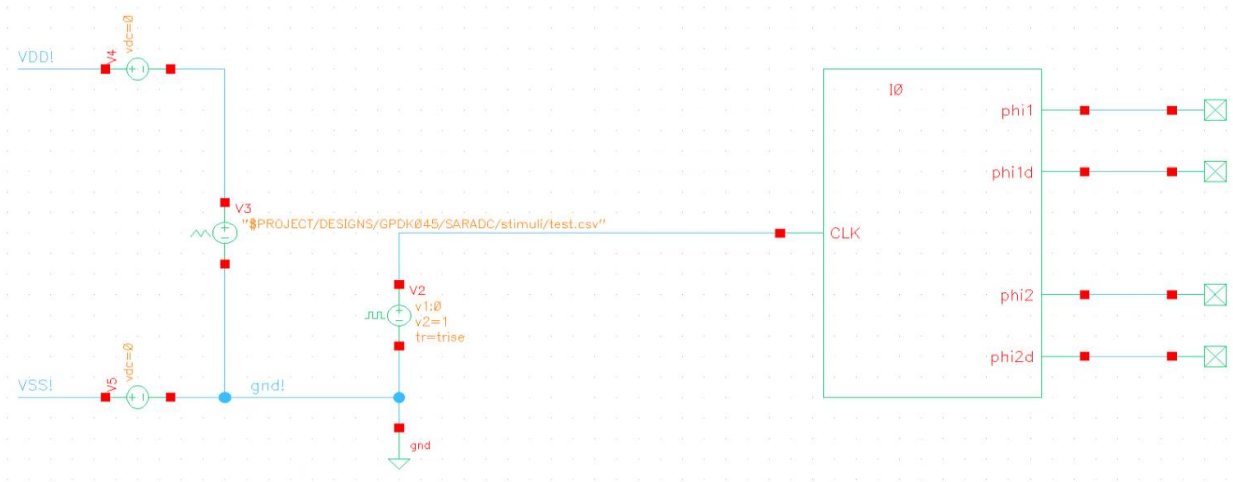


Figure 159: Clock Phase Generator Testbench with Noisy Power Supply

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 157: Click on the piece-wise linear file voltage source and use the properties hot key "**q**" to open the **Edit Object Properties** form.

The PWL file name is **$PROJECT/DESIGNS/GPDK045/SARADC/stimuli/test.csv**. This is a *.csv file, and it contains the output voltage of a dc-to-dc converter.

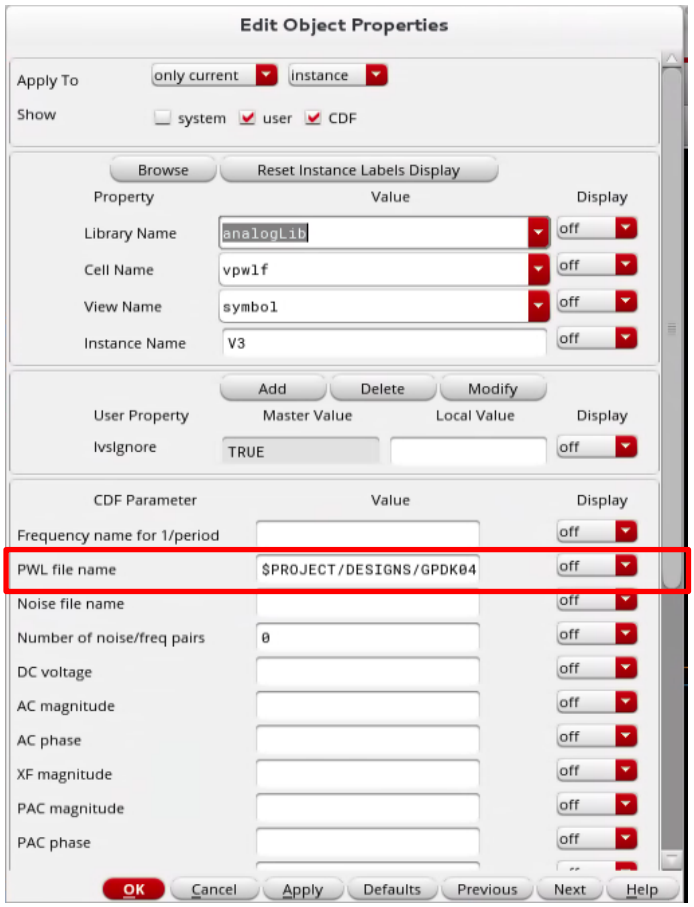


Figure 160: Edit Noisy Voltage Properties

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 158: In the schematic window, select **Launch > ADE Explorer**.

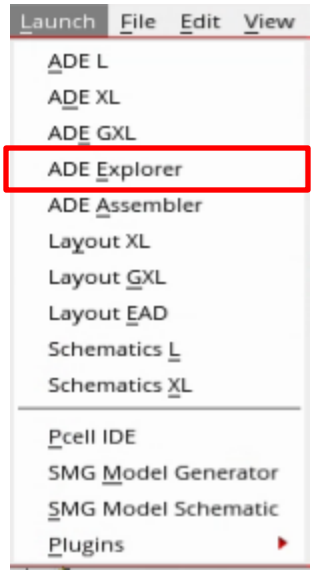


Figure 161: Launch ADE Explorer

ACTION 159: In the **Launch ADE Explorer** window, select **Open Existing View**. Then, click **OK**.

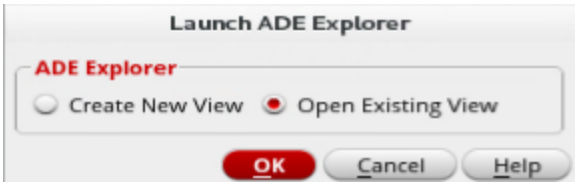


Figure 162: Open Existing Explorer View

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 160: In the **Open ADE Explorer View** window, verify that the view is set to **maestro**. Then, click **OK**.

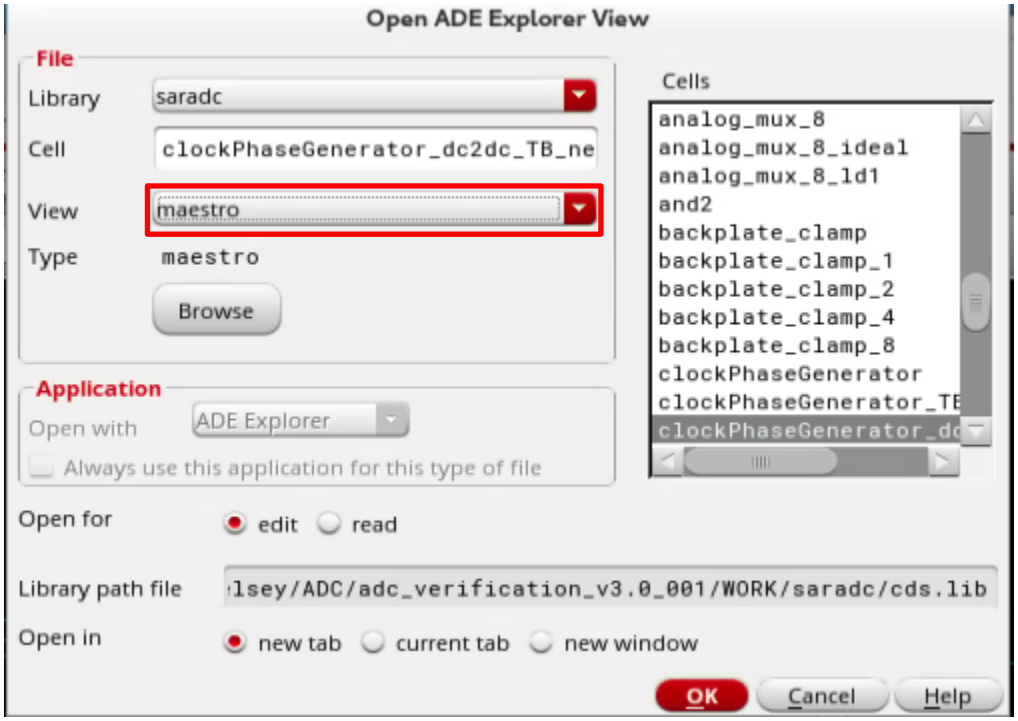


Figure 163: Open ADE Explorer Maestro View

After loading the state, the **maestro** tab should look like the picture below. To plot the eye diagram of the clock generator, you will be using the output node, ***/net7***, for your calculations.

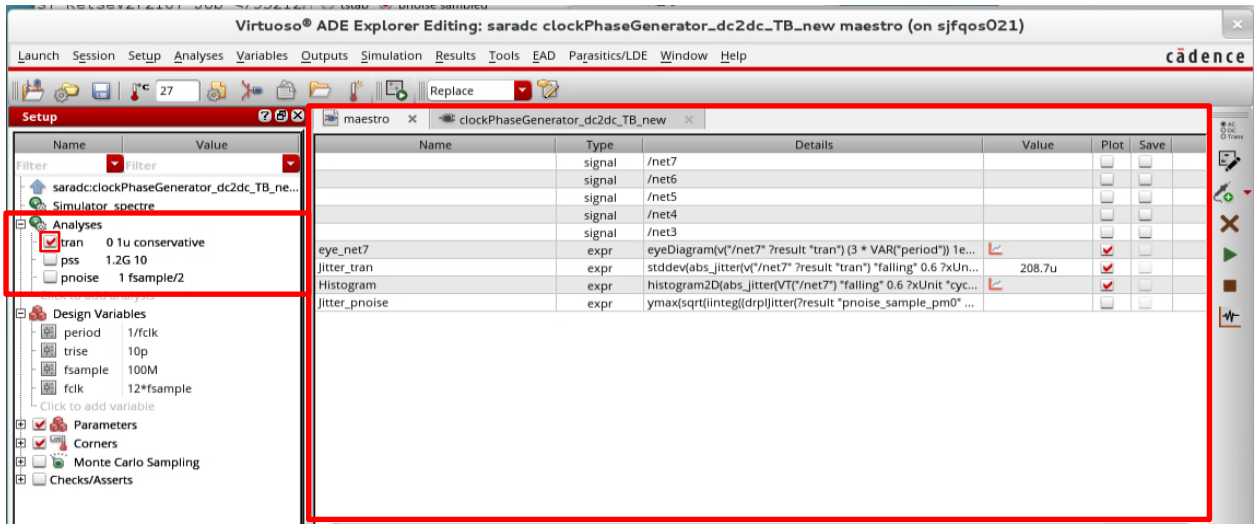


Figure 164: ADE Explorer Maestro Window

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Note: You are ignoring the PSS *and* pnoise analyses until Lab 5-2. Make sure that only the **tran** analysis is checked on. Also, make sure that the **Jitter_pnoise** expression is unselected. You will not use this expression in this lab.

The first expression in the ADE Explorer **maestro** window is the function related to calculating the eye diagram of the output of the clock generator. You will be using this as a comparison to check your results later in the RAK.

The second through sixth expressions correspond to different nets of the clock phase generator schematic. **/net7** is the output of the clock generator ("**phi1**") and is the one you are going to use to find the eye diagram. **/net6**, **/net5**, and **/net4** correspond to "**phi1d**", "**phi2**", and "**phi2d**" of the clock generator, respectively. These are additional outputs of the clock generator but are not used in this lab.

/net3 is the input to the clock generator created from noisy power supplies. It is the wire connecting **V1** to the "**CLK**" input of the clock generator.

ACTION 161: In the Setup Assistant, double-click on **tran** analysis to open its properties. Verify that the **Choosing Analyses – ADE Explorer** window looks like Figure 165 below. Then, click **OK**.

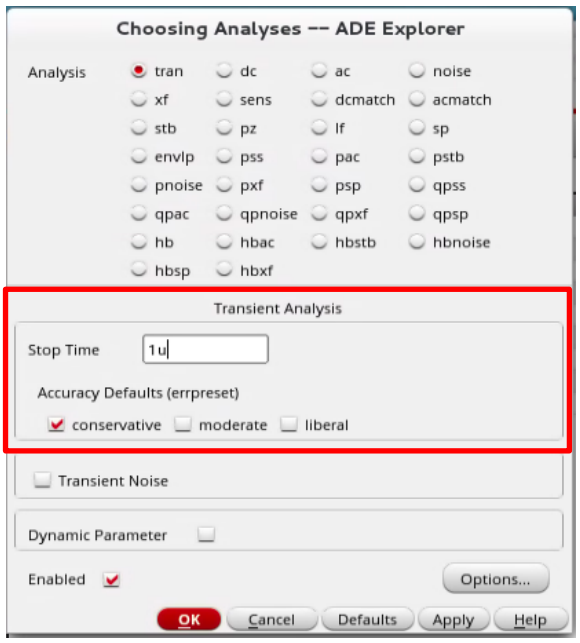


Figure 165: Choosing Analysis – ADE Explorer

errpreset controls the accuracy of the transient simulation by selecting a collection of additional parameter values. The value "**moderate**" is the default setting, "**liberal**" allows for faster simulations but less accuracy, and "**conservative**" is the most accurate

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

but takes the longest to simulate. For this module, you will use the **conservative** accuracy setting.

ACTION 162: To start the simulation select the ("**Run Simulation**") button.

ACTION 163: After the simulation is complete, click on the up arrow in the Setup Assistant . Select Results tab in the ADE Assembler window, RMB-click and select **Direct Plot > Transient Signal**.

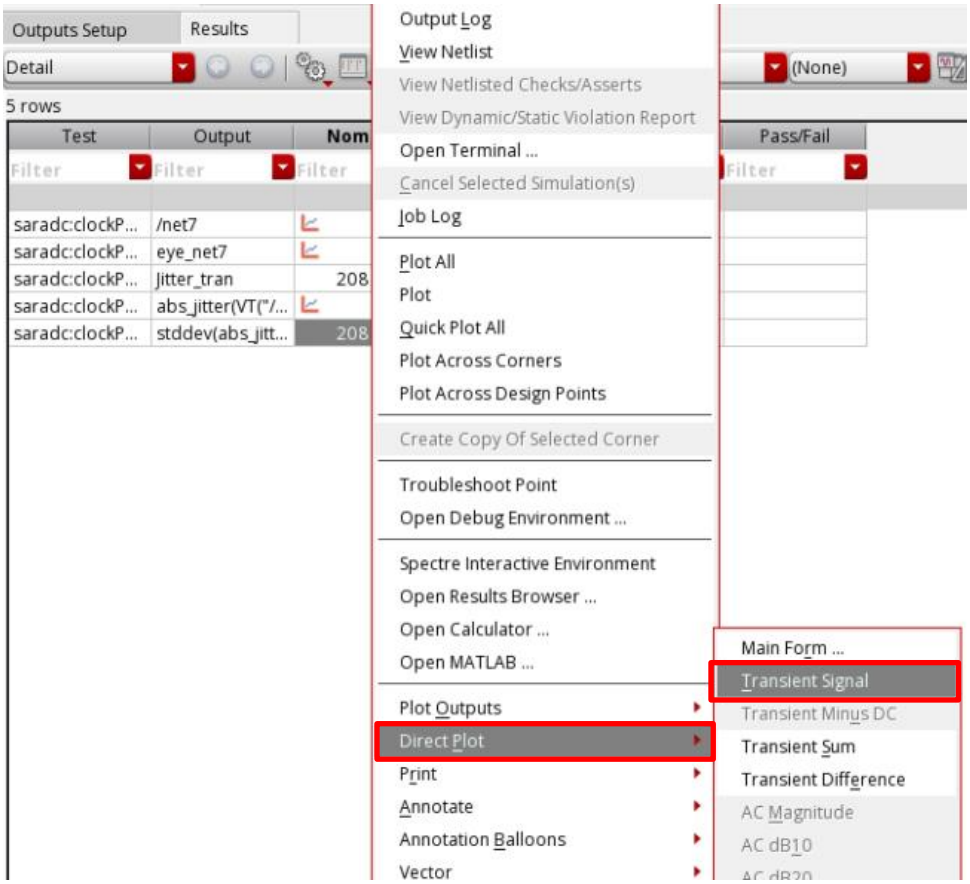


Figure 166: Direct Plot Transient Signal

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 164: In the schematic window, select the net connected to the pin labeled "**phi1**" on the right of the **clockGenerator** symbol, and then hit the "**Esc**" button to plot the output of the clock generator.

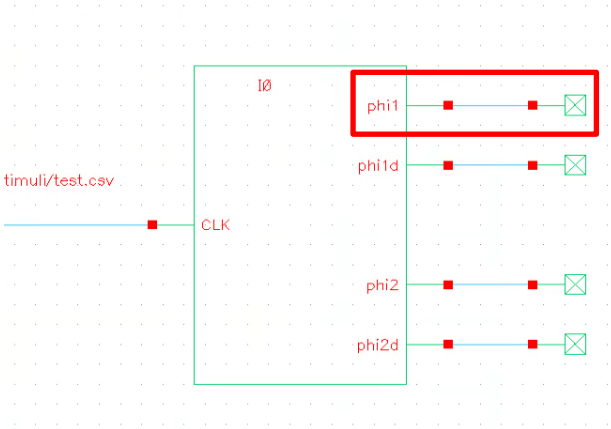


Figure 167: Schematic of "phi1" (/net7) Output of Clock Generator

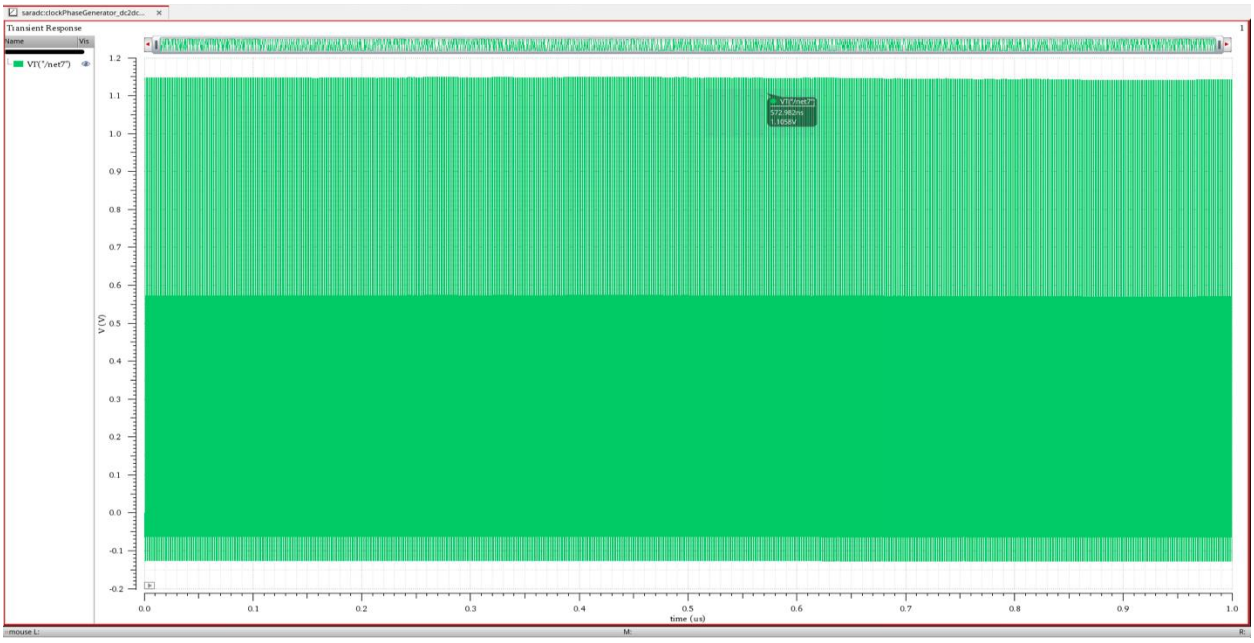


Figure 168: Output of the Clock Generator Plot

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

You will now calculate the jitter on the output waveform from the power supply noise. After generating the eye diagram for the clock signal, the jitter will be calculated.

ACTION 165: In the ViVA window, select **Measurements > Eye Diagram**.

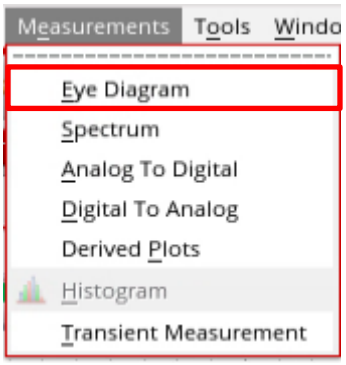


Figure 169: Eye Diagram Assistant

ACTION 166: The Eye Diagram Assistant will open and will be docked to the right of the screen. LMB-click on the waveform for **net7** to populate it in the waveform window and make sure that **Parameters** is set to **Custom**. The reason for this is that you need to be able to input a value for the **Eye Period** instead of relying on the "**Recommended**" period the Assistant creates for you.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Then, set the **Start/Stop → 0.0s**, **1µ** and set the **Period →1/VAR("fclk")**. For this example, a simulation time of 1µs is appropriate to generate good statistical results. The period is set to the inverse of the clock frequency because that is the time needed for one clock cycle. This is all you need to calculate the eye diagram. Verify that your eye diagram setup matches Figure 170, and then click on **Plot Eye** to display the eye diagram.

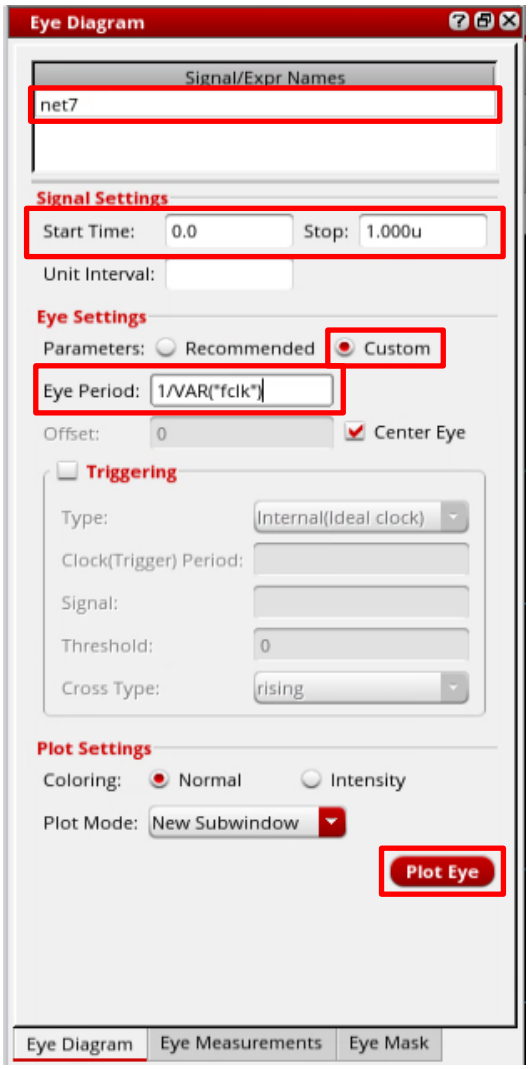


Figure 170: Eye Diagram Assistant Setup

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The eye diagram for the clock generator will be displayed as shown in Figure 171 below.

Figure 171: Eye Diagram of the Clock Generator

Next, you will calculate the jitter from the eye diagram.

Note: In this case, the input signal is a clock and the eye diagram is plotted for one period; so, the shape is a little bit different from a typical eye diagram that it is plotted from random data.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 167: Go back to the subwindow that has the **VT("/net7")** plotted. RMB-click on this waveform and then select **Send To > Calculator**. You will use this to calculate the jitter of the eye diagram.

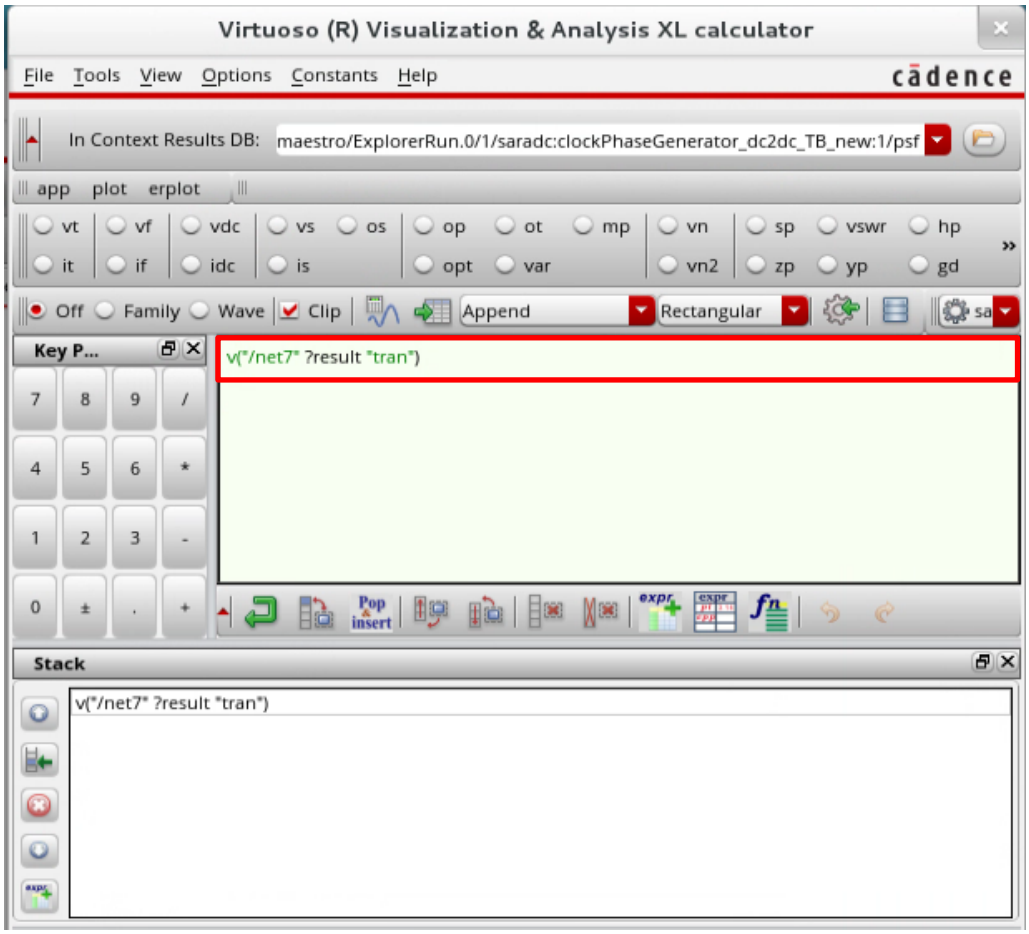


Figure 172: Calculator Window with Output of Clock Generator Expression

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 168: In the ViVA calculator, select the **abs_jitter** function from the **Functions Panel**. Populate the function as shown in Figure 173. The explanation for each value of the function is listed below Figure 173.

If the **Function Panel** window is not present, RMB-click on the **Stack Toolbar** (Stack) and select **Function Panel** from the pull-down menu.

When complete, click **OK**.

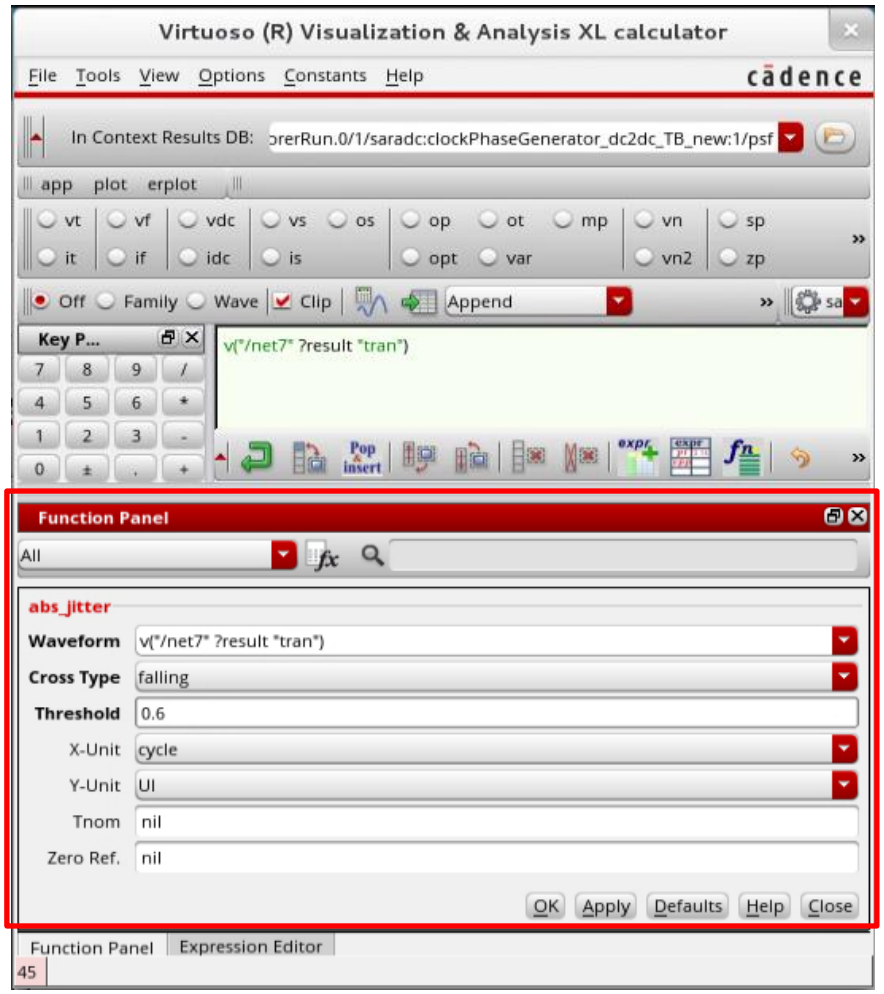


Figure 173: abs_jitter Function Setup

The function **abs_jitter** returns a waveform that contains the absolute jitter values in the input waveform for the given threshold.

Waveform -- Name of the waveform, expression, or family of waveforms.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Cross Type -- Specifies whether the jitter value can be calculated on rising (**rising**) or falling (**falling**) curves. Points at which the curves of the waveform intersect with the threshold.

Threshold -- Value at which the input waveform intersects to calculate the absolute jitter.

X-Unit -- Unit defined for X-axis of the output waveform. Specify whether you want to output the absolute jitter against time or cycle. Cycle numbers refer to the nth occurrence where the waveform crosses the given threshold.

Y-Unit -- Unit defined for Y-axis of the output waveform. Specify whether you want to calculate the phase in seconds (**s**), radians (**rad**), or unit intervals (**UI**).

Tnom -- Nominal period of the input waveform. The waveform is expected to be a periodic waveform that contains noise. If you do not enter the **Tnom** value, the **abs_jitter** function auto-calculates the approximate period of the input waveform by using the following equation:

(lastCrossing-firstCrossing) / (numCrossings-1)

where,

- Crossing times are determined by the time at which the specified threshold is crossed
- numCrossings determines the number of crossings

Zero Ref -- Defines the timing of the first edge of the ideal reference clock used for absolute jitter calculation. When this argument is nil or not specified, its default value is equal to the time of the first crossing of the input waveform so that the first value of the jitter is zero.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 169: Select the **stddev** function to calculate the RMS value of the jitter.

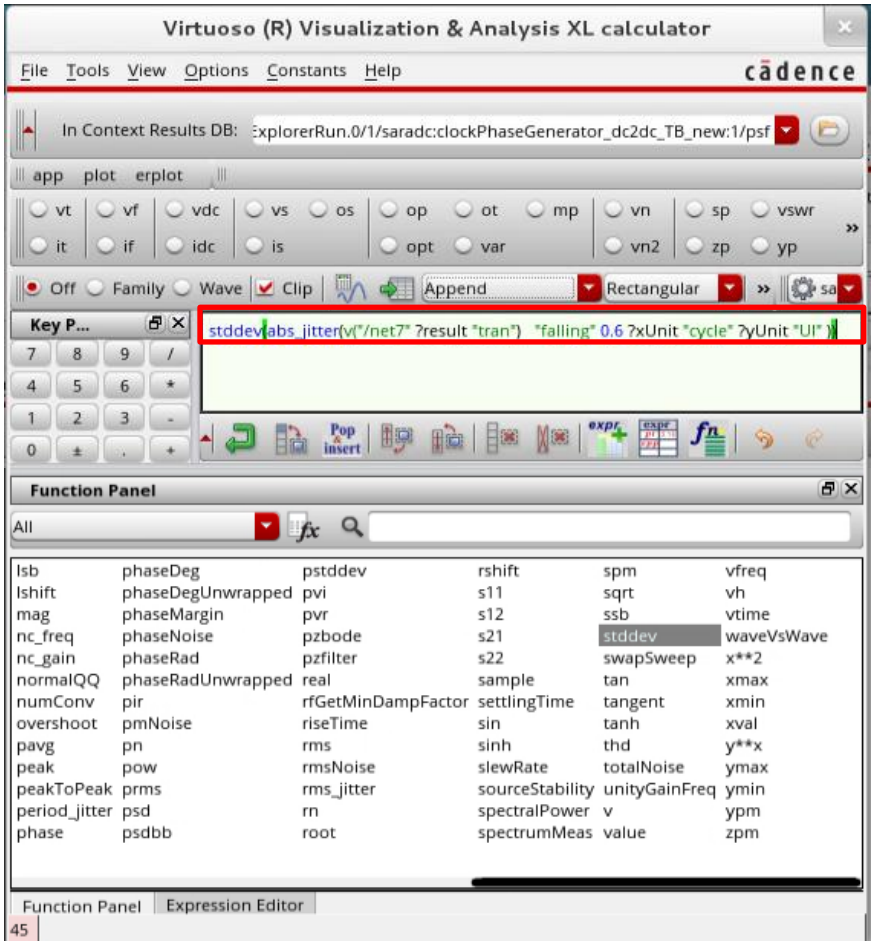


Figure 174: stddev Function

ACTION 170: Click on the (**Send to ADE**) button. This sends this expression to calculate the RMS of the jitter to the "**Outputs**" tab of the ADE Explorer window.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Next, you will create an expression for the clock phase generator's jitter.

ACTION 171: Select **v("/net7" ?result "tran")** from the **Stack** window and click to send the expression to the buffer.

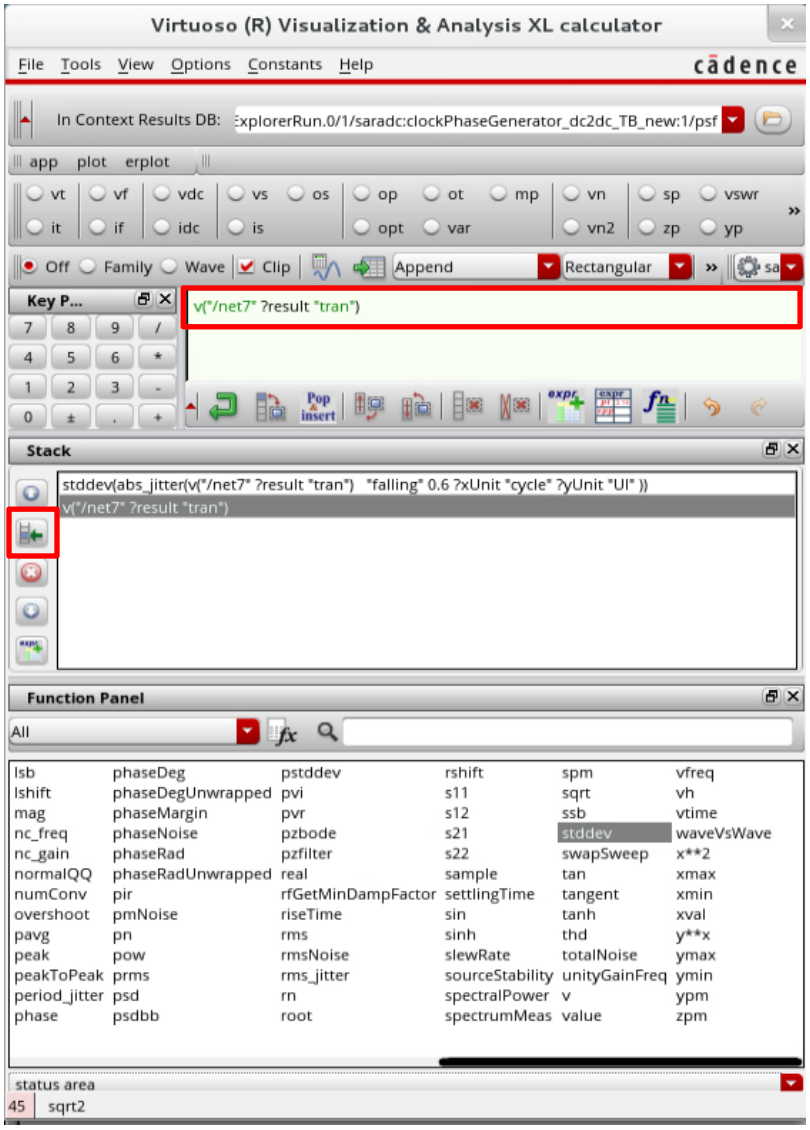


Figure 175: Calculator Window with Output of Clock Generator Expression

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 172: Select the **abs_jitter** function from the **Functions Panel**. Populate the function as shown in Figure 176. When complete, click **OK**.

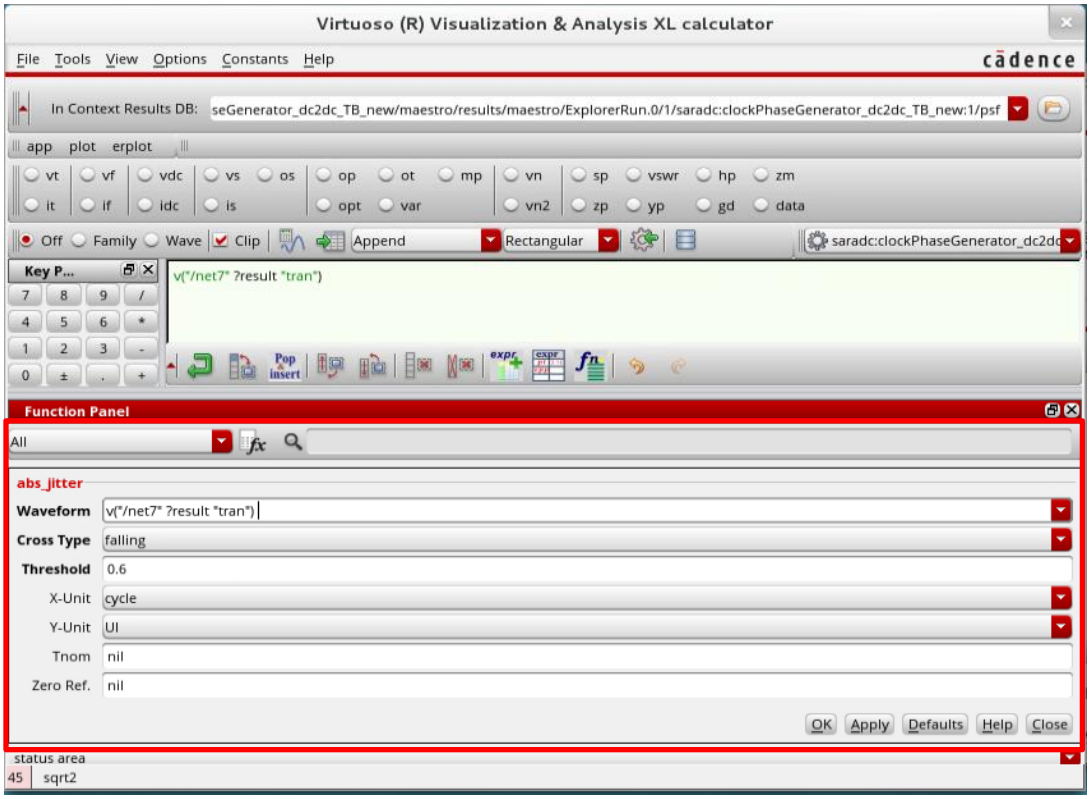


Figure 176: abs_jitter Function Setup

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 173: The **abs_jitter** function you created in ACTION 172 should be visible in the buffer. In the **Functions Panel**, click on **histogram2D** to create a histogram of the clock phase generator's jitter. Populate the function as shown in Figure 177. When complete, click **OK**.

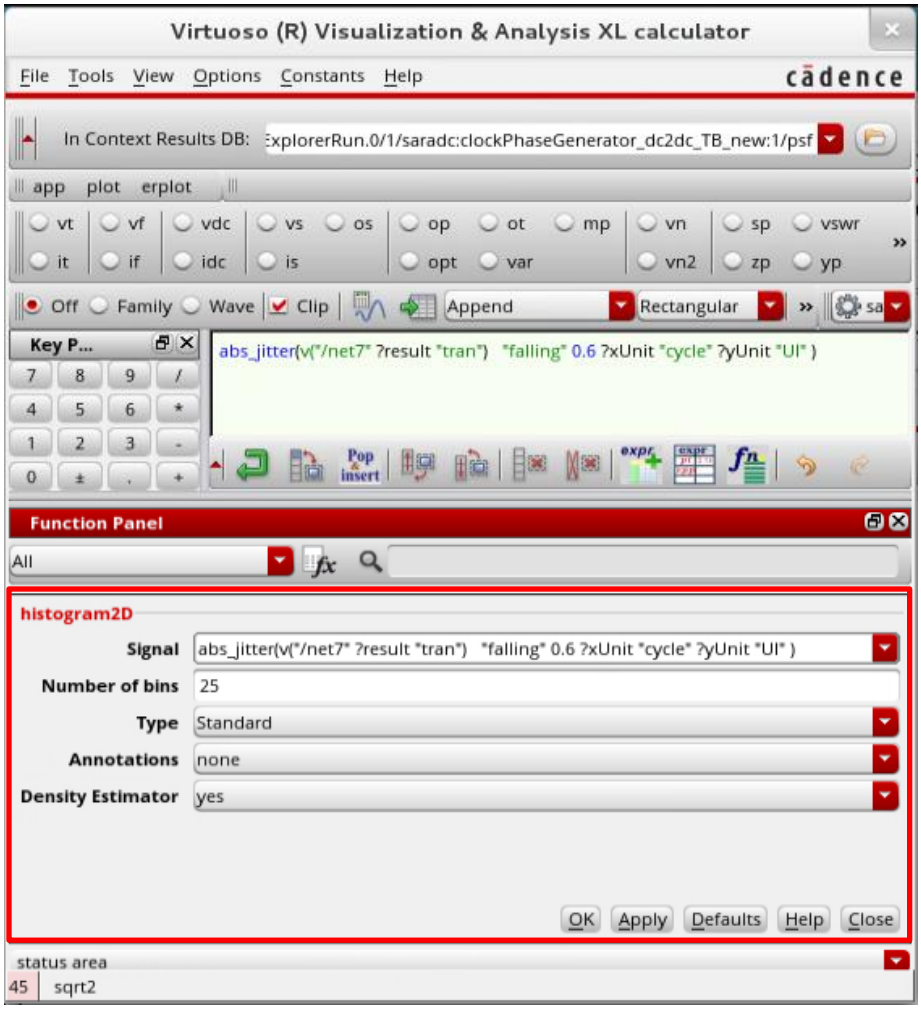
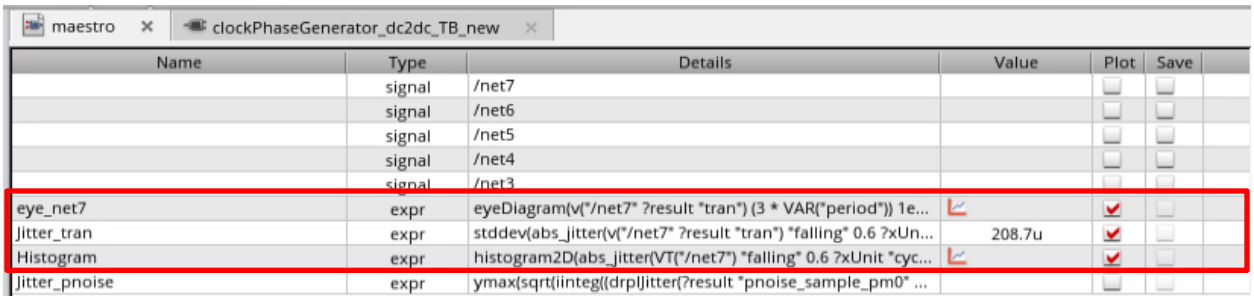


Figure 177: histogram2D Function Setup

ACTION 174: Click on the (**Send to ADE**) button. This sends this expression to the **Outputs** tab of the ADE Explorer window.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 175: Open the ADE Explorer window and examine the **Outputs** tab. Rename the two expressions created as **jitter** and **histogram**, respectfully. Make sure to have **eye_net7**, **jitter**, and **histogram** be set to "**Plot**".



Name	Type	Details	Value	Plot	Save
	signal	/net7			
	signal	/net6			
	signal	/net5			
	signal	/net4			
	signal	/net3			
eye_net7	expr	eyeDiagram(v("/net7" ?result "tran") (3 * VAR("period")) 1e...		✓	
jitter_tran	expr	stddev(abs_jitter(v("/net7" ?result "tran") "falling" 0.6 ?xUn...	208.7u	✓	
Histogram	expr	histogram2D(abs_jitter(VT("/net7") "falling" 0.6 ?xUnit "cyc...		✓	
jitter_pnoise	expr	ymax(sqrt(iinteg((drplJitter(?result "pnoise_sample_pm0" ...			

Figure 178: ADE Explorer View with Added Expressions

ACTION 176: Click on to plot the outputs. You should see that the new equations that you made match the pre-made ones. Also, you should see a new ViVA window open with the histogram of the clock phase generator's jitter.

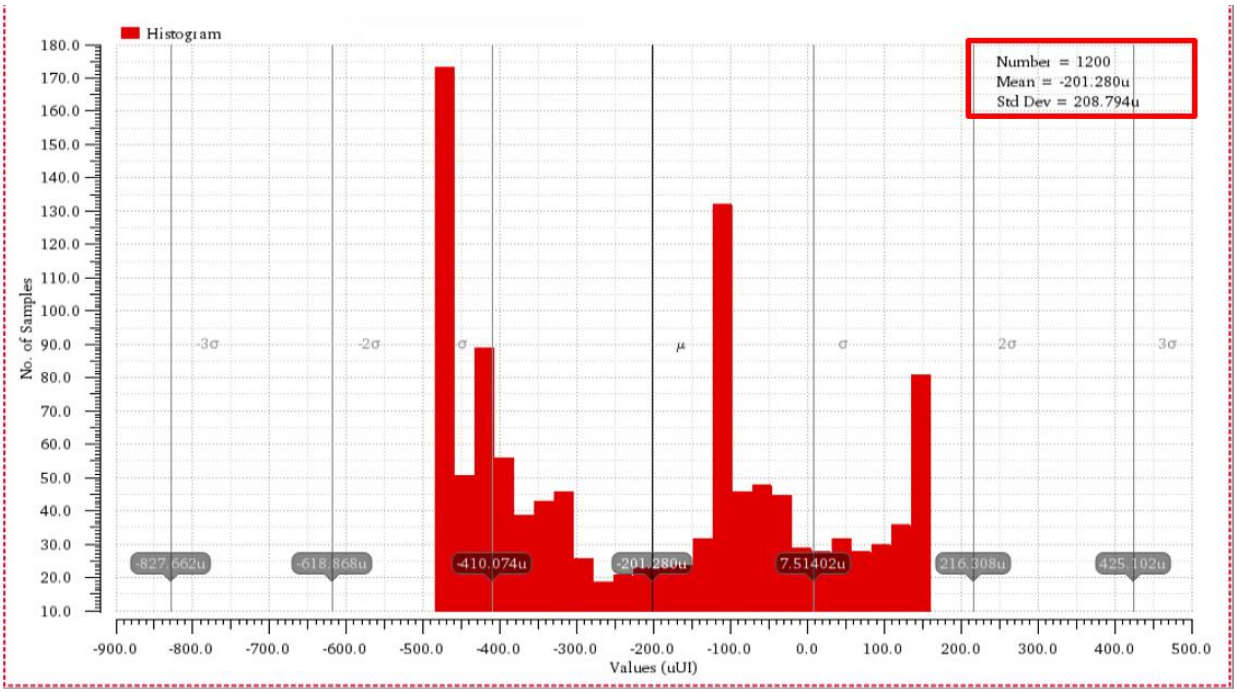


Figure 179: Histogram of Clock Phase Generator's Jitter

ACTION 177: Close the ViVA and ADE Explorer windows by clicking on the (**Exit**) button in the upper-right corner. Select "**No**" if asked to save any changes.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Lab 5-2: Analyzing Clock Power Supply Noise Rejection with PSS/Pnoise Analysis

This lab is a continuation of the previous lab, but you will analyze the clock jitter using PSS analysis instead of transient analysis.

ACTION 178: In the Library Manager, select **saradc > Everything > clockPhaseGenerator_dc2dc_TB_new**. Double-click on **schematic** to open the schematic.

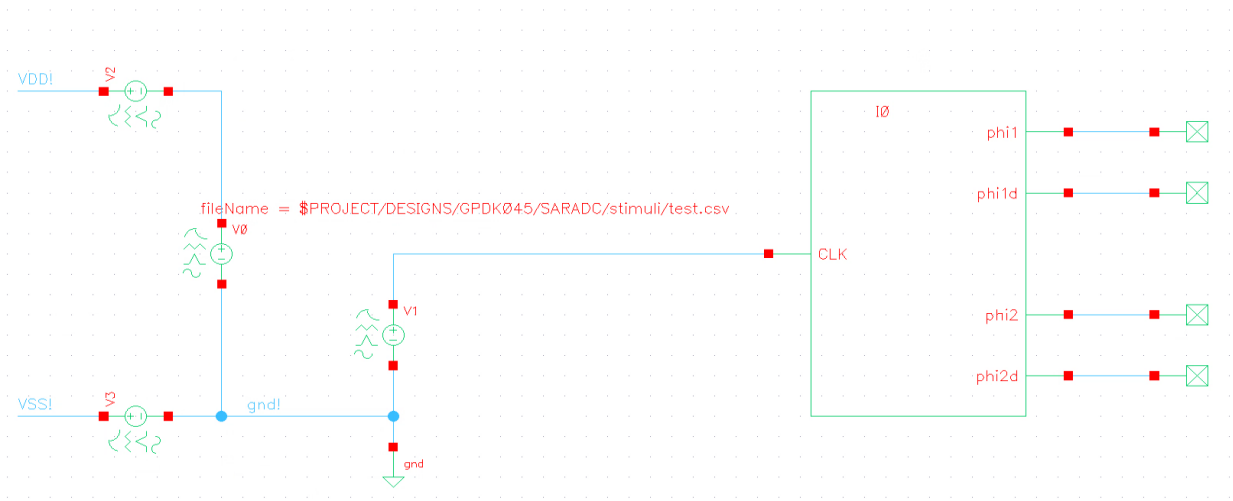


Figure 180: Clock Phase Generator Testbench with Noisy Power Supply

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 179: Click on the piece-wise linear file voltage source and use the properties hot key "**q**" to open the **Edit Object Properties** form.

The PWL file name is **$PROJECT/DESIGNS/GPDK045/SARADC/stimuli/test.csv**. This is a *.csv file that contains the output voltage of a dc-to-dc converter.

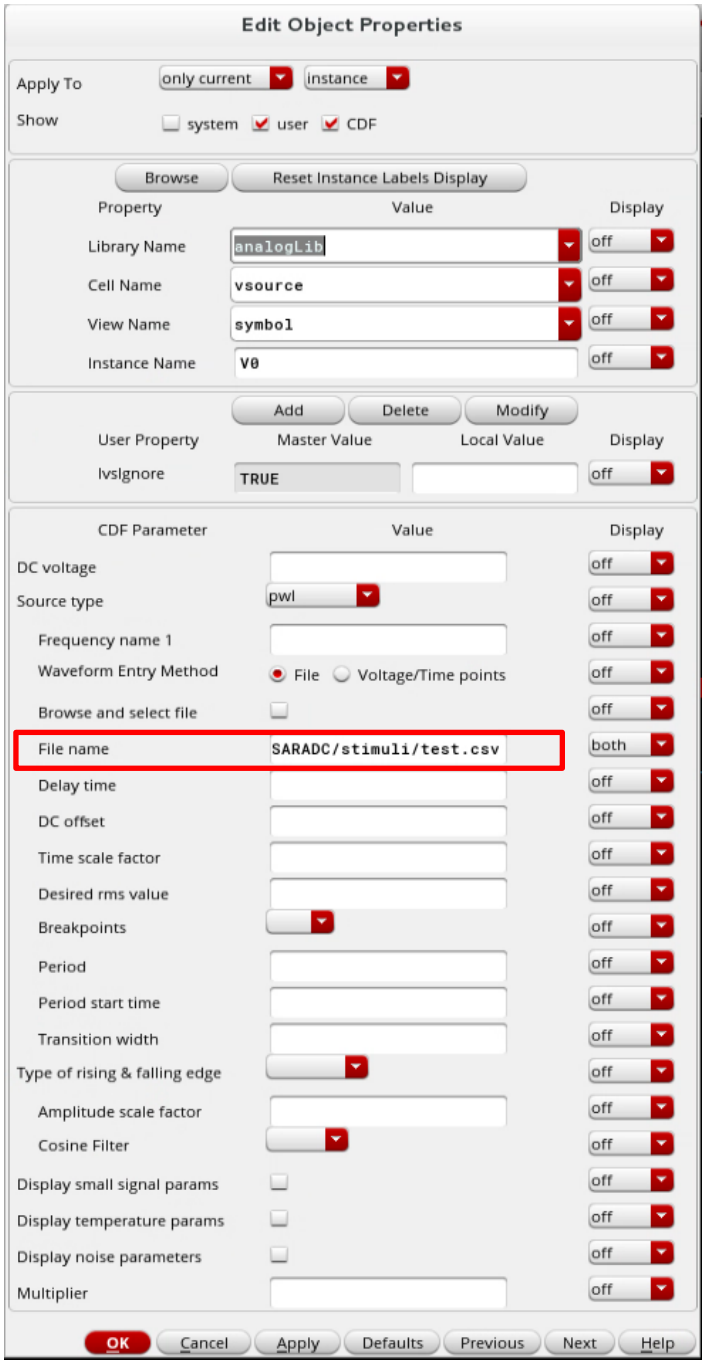


Figure 181: Edit Noisy Voltage Properties

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 180: In the schematic window, select **Launch > ADE Explorer**.

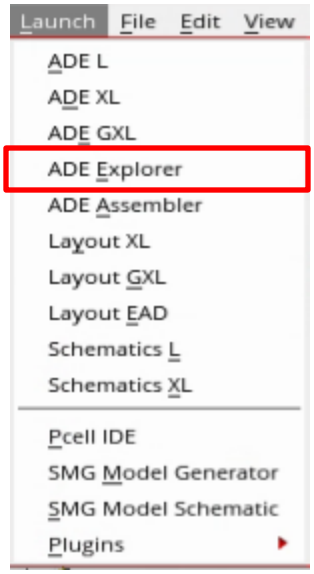


Figure 182: Launch ADE Explorer

ACTION 181: In the **Launch ADE Explorer** window, select **Open Existing View**. Then, click **OK**.

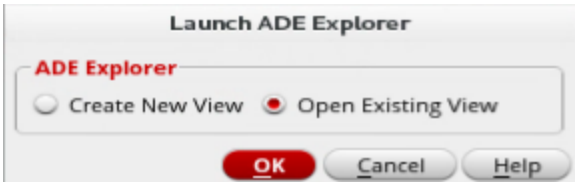


Figure 183: Open Existing Explorer View

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 182: In the **Open ADE Explorer View** window, verify that the view is set to **maestro**. Then, click **OK**.

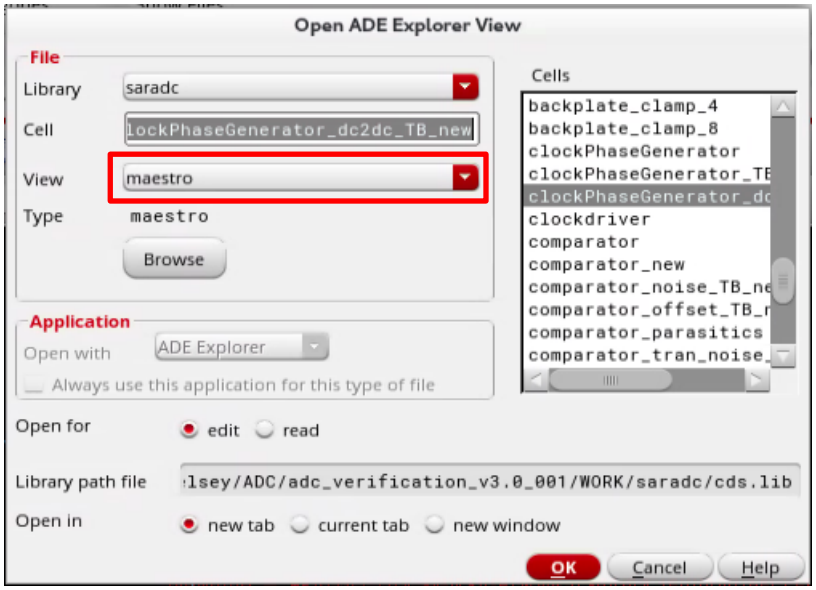


Figure 184: Open ADE Explorer Maestro View

After loading the state, the **maestro** tab should look like the picture below. To plot the eye diagram of the clock generator, you will be using the output node, **/net7**, for your calculations.

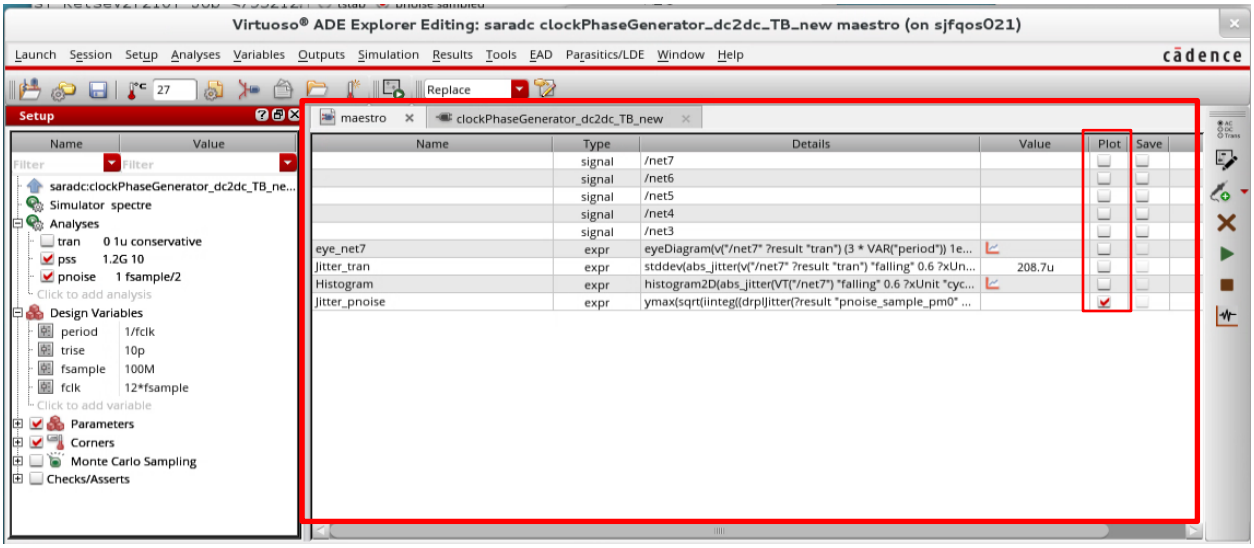


Figure 185: ADE Explorer Maestro Window

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The first expression in the ADE Explorer **maestro** window is the function related to calculating the eye diagram of the output of the clock generator. You will be using this as a comparison to check your results later in the RAK.

The second through sixth expressions correspond to different nets of the clock phase generator schematic. **/net7** is the output of the clock generator ("**phi1**") and is the one you are going to use to find the eye diagram. **/net6**, **/net5**, and **/net4** correspond to "**phi1d**", "**phi2**", and "**phi2d**" of the clock generator, respectively. These are additional outputs of the clock generator but are not used in this lab.

/net3 is the input to the clock generator created from noisy power supplies. It is the wire connecting **V1** to the "**CLK**" input of the clock generator.

Note: Make sure to unselect **eye_net7**, **Histogram**, and **Jitter_tran** since you will not be using these expressions for this lab.

ACTION 183: Make sure to unselect the **tran** analysis before moving on. RMB-click on **pss** and select **Edit** from the pop-up menu.

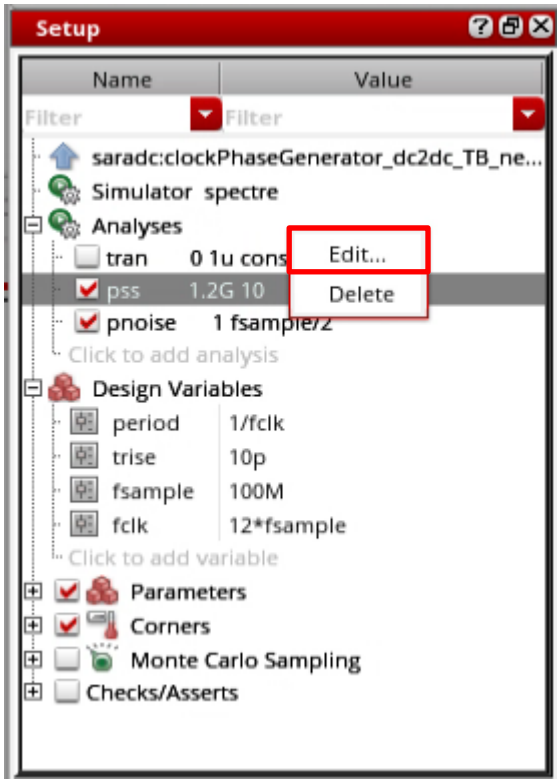


Figure 186: PSS Analysis Options

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Verify the pss analysis setup with Figure 187. Make sure that **VAR("period")** is inside the **Stop Time (tstab)** space. Click on **Apply** when done.

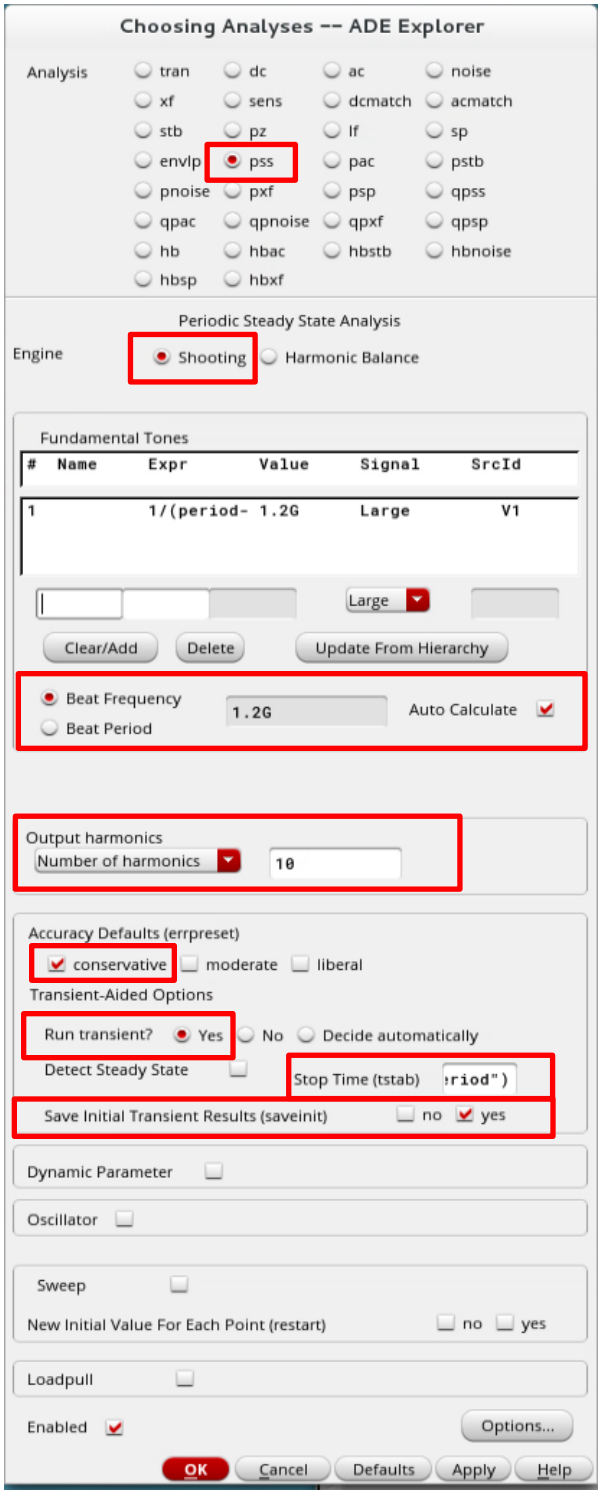


Figure 187: Choosing Analyses – ADE Explorer: PSS

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

pss – Periodic state analysis is used to calculate the response of a circuit to large input signals.

Shooting Newton – This is an iterative time-domain algorithm that is akin to solving a boundary-value problem by reducing it to the solution of an initial-value problem.

Beat Frequency – This is the periodicity of the system. When the **Auto Calculate** option is selected, ADE calculates the periodicity of the circuit and displays it in the **Beat Frequency** field.

Number of harmonics – This parameter specifies the number of harmonics to calculate in the Fourier transform that is run after the time-domain simulation to calculate the harmonics of the large-signal response. 10 is the best number because any higher number will result in more time points being forced in the pss solution to keep the accuracy of the frequency-domain results very high.

Accuracy Defaults (errpreset) – You use **conservative** because you want high-accuracy results. "**moderate**" is the general-purpose setting, and "**liberal**" is not recommended.

Run transient? – When you choose "**Yes**", the **Detect Steady State** option is enabled automatically and the simulation can process to pss without finishing the tstab simulation.

tstab – This is the user-defined stabilization time interval. You will be using the design variable "period" to define the time interval. A longer tstab allows the circuit to get closer to steady-state and thus, converge better.

You also want to save your initial transient results; so, you select "**Yes**" for this part.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 184: Select **pnoise** analysis and make the setup window resemble Figures 188 and 189. Click on "**Apply**" when done.

Note: The following two screenshots constitute the total analysis window for the pnoise.

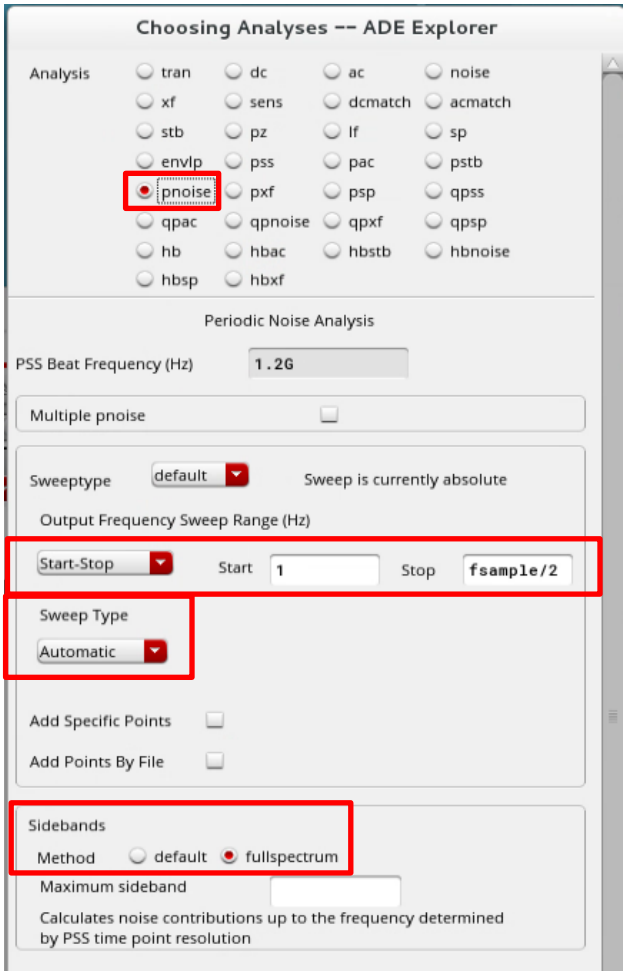


Figure 188: Upper Half of Choosing Analyses – ADE Explorer: Pnoise Window

Since the fundamental quantity that is calculated by the noise analysis is the noise at the output, the frequency range is always the output frequency range. The output frequency range corresponds to the Nyquist rate of the ADC, **fsample/2**. fsample is the ADC conversion rate. By setting the **Stop Time** to **fsample/2**, the noise analysis will not have any effects from noise folding.

The **Sidebands Method** is set to **fullspectrum** because this method is especially useful for circuits where aliasing occurs through very high harmonics of the clock. The run-time advantages of using full spectrum are large with no loss in accuracy of the result. Using the full spectrum method setting allows pnoise to calculate the noise

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

translations through the full spectrum of the pss analysis. Noise separation is not supported when full spectrum is selected.

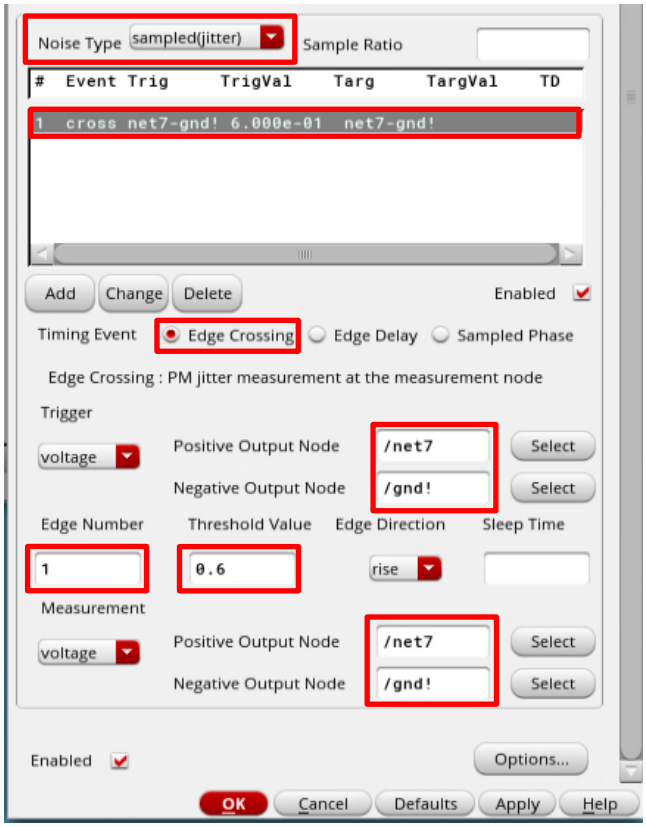


Figure 189: Lower Half of Choosing Analyses – ADE Explorer: Pnoise Window

Sampled(jitter) – When the **noisetype** is set to **jitter**, an instantaneous measurement of the noise at the threshold crossing in the pss analysis is calculated. **Sampled (jitter)** has three modes, and the mode you will be using is **Edge Crossing**.

Edge Crossing is where you specify a threshold and direction called the trigger, which defines the time when the noise measurement is made. In many cases, the measurement net will be on the same net as the trigger, as shown in the diagram above, but the measurement can be made on another net in the circuit. This is useful when you want to make a noise measurement on the output of a switched-capacitor filter triggered by the clock waveform.

In this RAK, the trigger is the “rise” edge of the output net as it crosses VDD/2 and the noise is measured on the same net.

The threshold value is set to **0.6** because that is half as much as VDD (1.2V). The edge direction is set to “**rise**” because the signal is first low and then goes high, and you want to capture that value change when it happens. The positive output node for both the

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Trigger and **Measurement** values is set to "**/net7**" because that is the output of the CAPDAC.

ACTION 185: To populate the sampled jitter window with a trigger event, fill out the window as shown and click on "**Add**" when done. This should create a trigger event as shown in Figure 189.

ACTION 186: To start the simulation, select the ("**Run Simulation**") button.

ACTION 187: After the simulation is complete, select **Results > Direct Plot > Main Form** to open the **Direct Plot Form**.

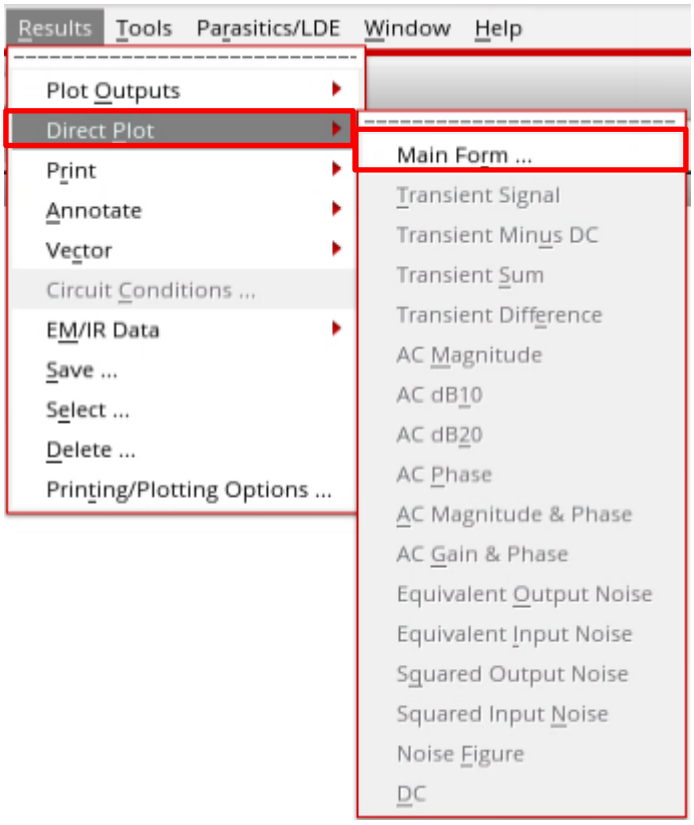


Figure 190: Direct Plot Main Form

ACTION 188: In the **Direct Plot Form**, select **pnoise sampled** and **Jee**. Also, make sure that **Signal Level** is set to **rms** and **Modifier** is set to **Second**. Fill in **Integration Limits** with **Start Frequency (Hz) = 1** and **Stop Frequency (Hz) = 50M**. The reason for these integration values is that you want the whole frequency spectrum from 1Hz to fsample/2= 50 MHz. This will eliminate any chance of noise distortion from noise folding. Make sure that "**Add to Outputs**" is enabled. Verify that the window looks like Figure 191.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Note: Jee outputs the edge-to-edge jitter.

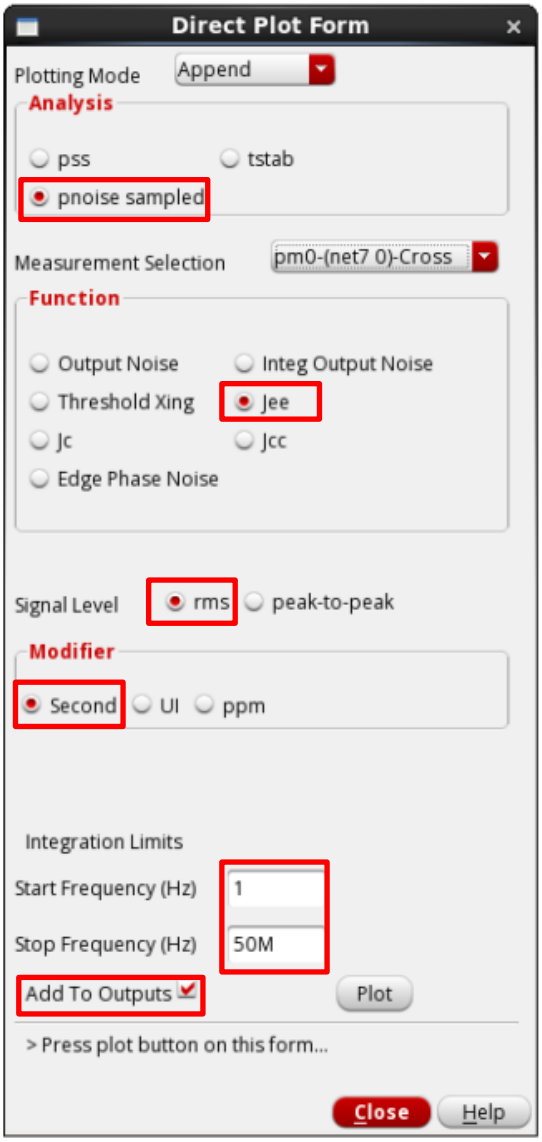


Figure 191: Direct Plot Form

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 189: Switch back to the ADE Explorer window. It should look like Figure 192 below.

Name	Type	Details	Value	Plot	Save	Spec
	signal	/net7				
	signal	/net6				
	signal	/net5				
	signal	/net4				
	signal	/net3				
eye_net7	expr	eyeDiagram(v("/net7" ?result "tran") (3 * VAR("period")) 1e-05 ...				
Jitter_tran	expr	stddev(abs_jitter(v("/net7" ?result "tran") "falling" 0.6 ?xUnit "cy...				
Histogram	expr	histogram2D(abs_jitter(VT("/net7") "falling" 0.6 ?xUnit "cycle" ?...				
Jitter_pnoise	expr	ymax(sqrt(iinteg((drplJitter(?result "pnoise_sample_pm0" ?unit...	149.1f			
Jee[Second...	expr	rfJitter(?result "pnoise_sample_pm0" ?unit "Second" ?from 1 ?t...				

Figure 192: Explorer Maestro View with Jee Outputs

You will build the expression to calculate the total noise and compare the **Jee[Second]** expression result comes from the pop-up window that is created when you enable "**Add to Outputs**" button and first plot the Jee output. This expression is already in the maestro view (**jitter_pnoise**), but the following actions will go step by step to explain it.

The total noise is the root-sum-squared of the spot noise. The equation for calculating the total integrated noise is:

$$Total_Jitter = \sqrt{\int (output_jitter)^2}$$

ACTION 190: Click on the Add Outputs button in ADE Explorer window

.

Double -click on the empty expression and click on the **Expression Builder** icon ( ). The Expression Builder should appear as shown in Figure 153.

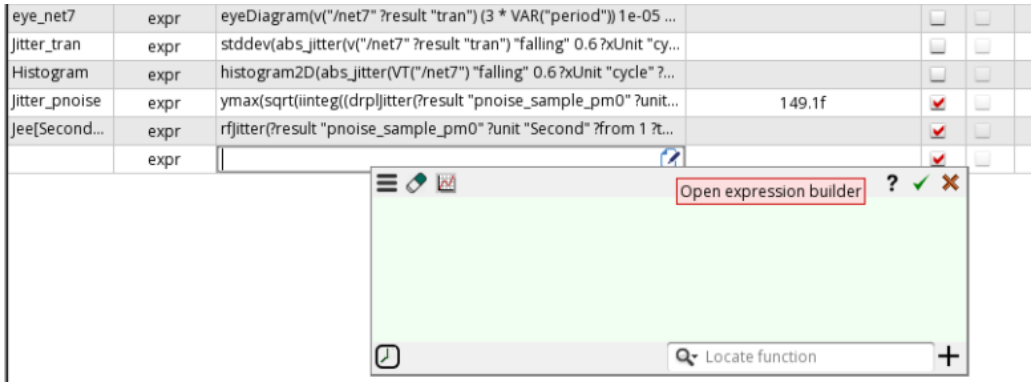


Figure 193: Expression Builder form

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 191: Add the following expression in the Expression Builder.

drplJitter(?result "pnoise_sample_pm0" ?unit "Second" ?k 1 ?event 0)

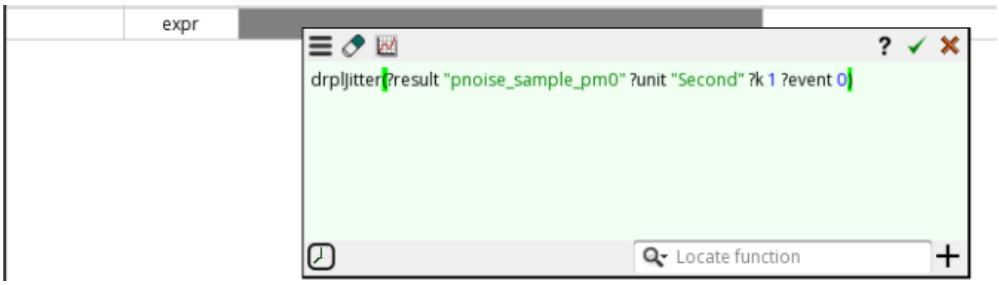


Figure 194: Expression for Total Jitter

ACTION 192: At the end of the expression, add a ****2** that corresponds to the square in the Total Noise equation. At the beginning of the expression, add the function "**iinteg**" to compute the indefinite integral of the expression.

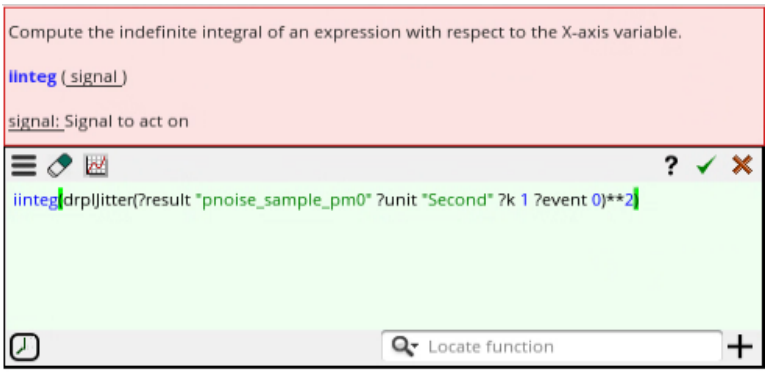


Figure 195: Expression Builder with iinteg() Function

ACTION 193: At the beginning of the new expression, add the function "**sqrt**" to compute the square root of the function.

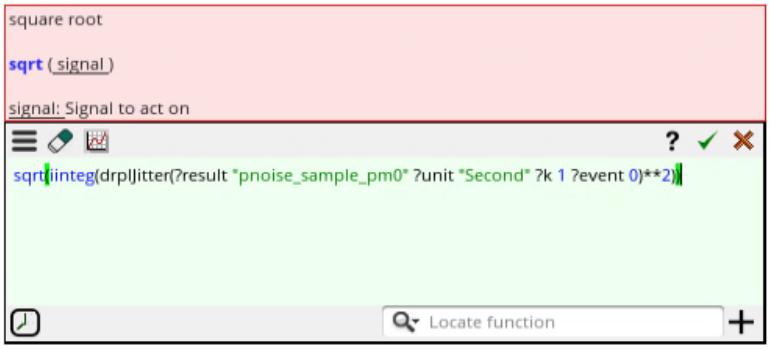


Figure 196: Expression Builder with sqrt() Function

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 194: At the beginning of the new expression, add the function "**ymax**" to return the y value at the maximum.

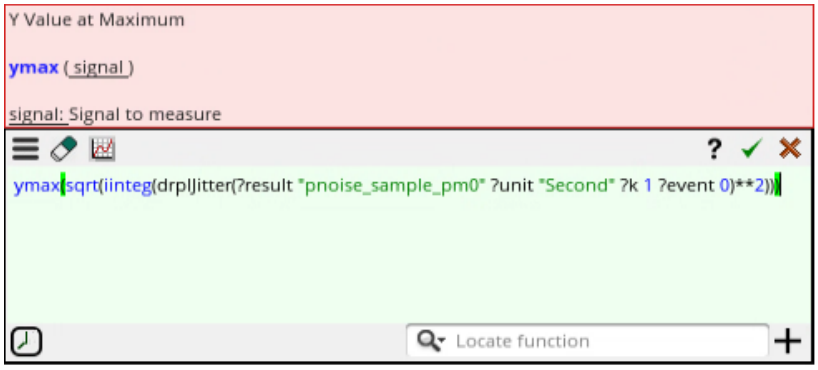


Figure 197: Expression Builder with ymax() Function

ACTION 195: Click on the little green arrow ( ) at the top-right side of the window to accept the expression.

ACTION 196: Click on the "**Plot Outputs**" ( ) button to re-evaluate the expression you just created. It should match the first expression in the **Outputs** tab of the Explorer window.

ACTION 197: After the expressions are re-evaluated, you will see how the total jitter noise was calculated for the clock generator. The jitter noise value calculated in this lab will be used to compute the ADC's ENOB and SINAD.

Name	Type	Details	Value	Plot	Save	Spec
	signal	/net7		☐	☑	
	signal	/net6		☐	☐	
	signal	/net5		☐	☐	
	signal	/net4		☐	☐	
	signal	/net3		☐	☐	
eye_net7	expr	eyeDiagram(v("/net7" ?result "tran") (3 * VAR("period")) 1e-05 ...		☐	☐	
Jitter_tran	expr	stddev(abs_jitter(v("/net7" ?result "tran") "falling" 0.6 ?xUnit "cy...		☐	☐	
Histogram	expr	histogram2D(abs_jitter(VT("/net7") "falling" 0.6 ?xUnit "cycle" ?...		☐	☐	
Jitter_pnoise	expr	ymax(sqrt(iinteg((drplJitter(?result "pnoise_sample_pm0" ?unit...	149.1f	☑	☐	
Jee[Second...	expr	rfJitter(?result "pnoise_sample_pm0" ?unit "Second" ?from 1 ?t...	149.1f	☑	☐	
	expr	ymax(sqrt(iinteg((drplJitter(?result "pnoise_sample_pm0" ?unit...	149.1f	☑	☐	

Figure 198: Total Jitter of Clock Generator with a Noisy Supply

Note: The equation name **Jee[Second]** shown comes from the pop-up window that is created when you first plot the Jee output. It comes automatically when you select "**Add to Outputs**" in the **Main Form** window.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 198: Close the waveform plot and ADE Assembler windows by clicking on the (**Exit**) button in the upper-right corner. Select "**No**" if asked to save any changes.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Module 6: Top-Level ADC Calculations

Lab 6-1: Top-Level ADC Testbench

In this final module of the RAK, you will test the ADC's Large Signal Performance by applying a sine wave into the input and then observing the digital code generated at the output.

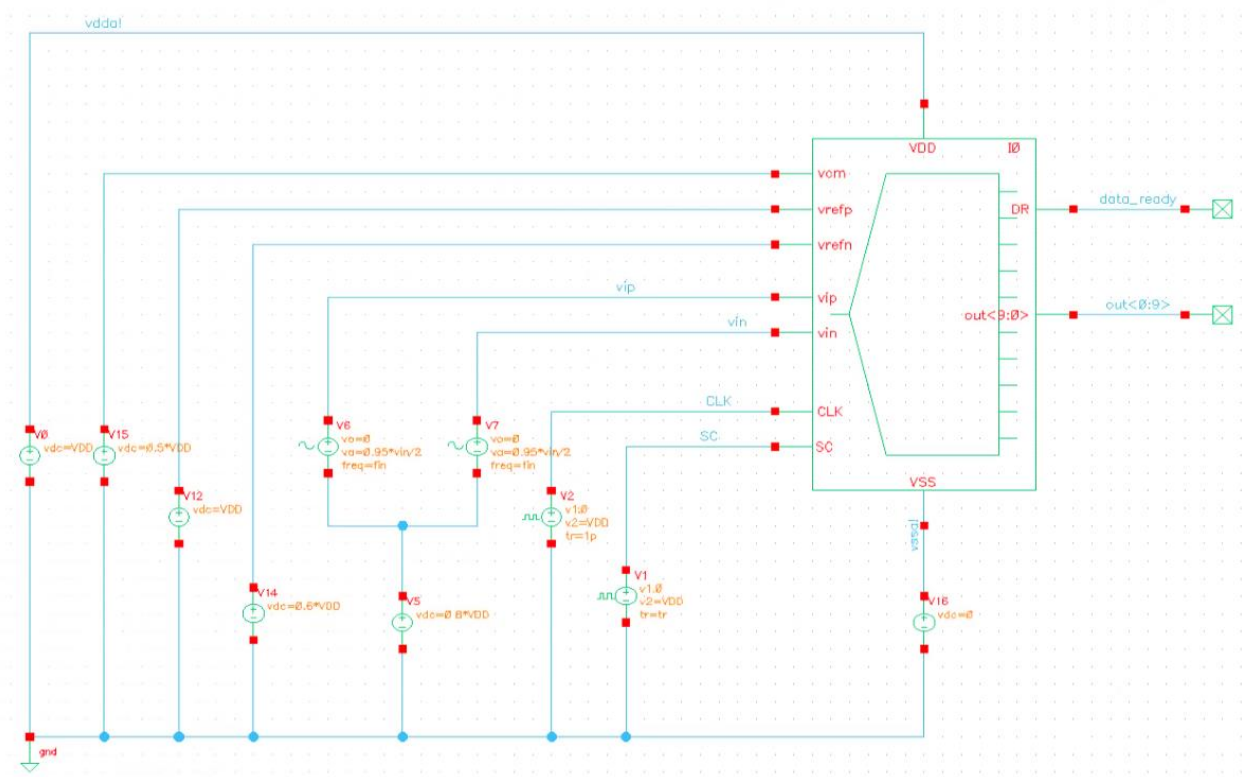


Figure 199: Top-Level ADC Testbench Schematic

Figure 199 shows the schematic of the top-level ADC testbench. **V0** is the voltage source corresponding to the value of VDD. This is the main power supply for the ADC.

V15 is the common mode voltage for the comparator. The value for this voltage source is set to VDD/2.

V12 and **V14** correspond to the positive and negative voltage reference sources for the DAC, respectfully.

V5 is the voltage source that controls the input voltage level swing which, in this example, is set to **0.8*VDD**.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

V6 and **V7** are the input positive and negative sine wave voltage sources, respectfully. The two sources differ from each other by a phase of 180º.

V2 is a pulse waveform voltage source that corresponds to the clock input into the ADC. It has a high value of VDD and a low value of 0. It has a period that is the inverse of the clock frequency.

V1 is the voltage source that controls the “Start Convert” of the DAC. It tells the ADC to start and reset between input values.

Finally, **V16** is a DC voltage source with value equaling 0. This source is placed so that the negative current of the ADC can be measured.

Lab 6-2: Top-Level ADC Transient Analysis

In this section, you will be running a transient analysis on the top-level ADC to calculate the ADC’s SINAD. This value will be used to verify that all the values calculated in the sections above add up to the correct value of the ADC’s SINAD.

ACTION 199: In the Library Manager, select **saradc > Everything > 10Bit_ADC_TB_new**. Double-click on **schematic** to open the schematic.

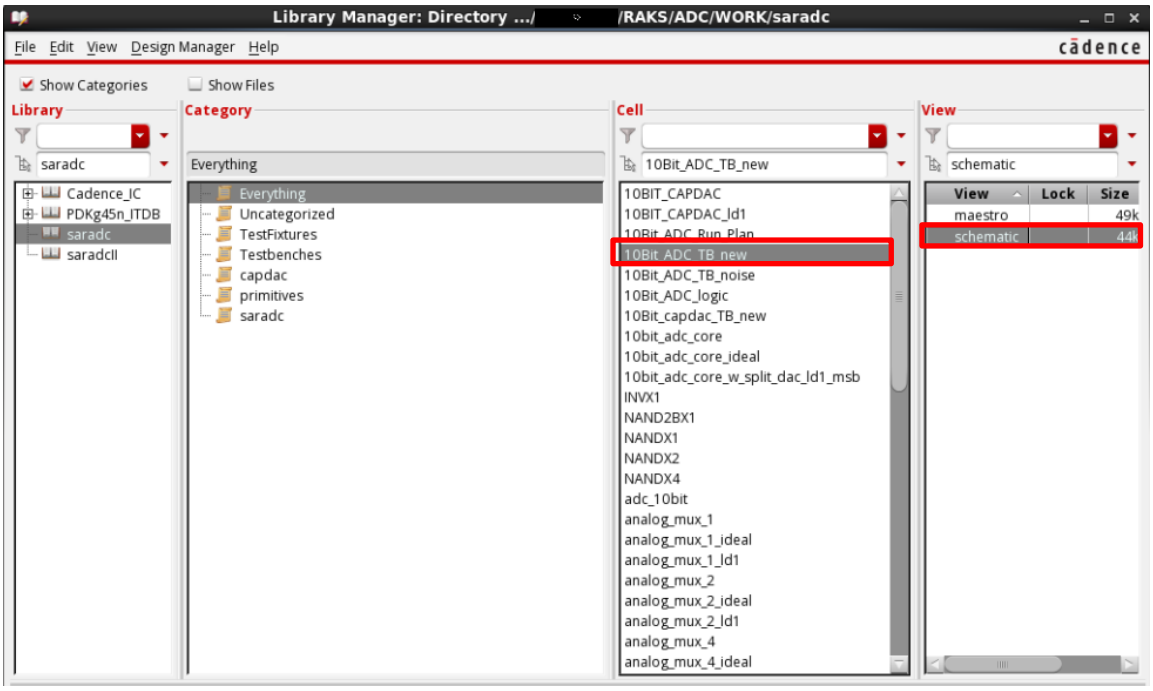


Figure 200: Library Manager

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 200: In the schematic window, select **Launch > ADE Explorer**.

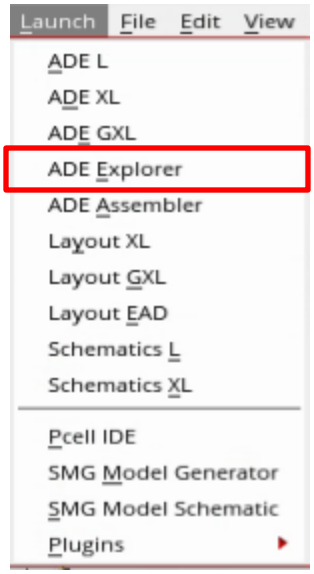


Figure 201: Launch ADE Explorer

ACTION 201: In the **Launch ADE Explorer** window, select **Open Existing View**. Then, click **OK**.

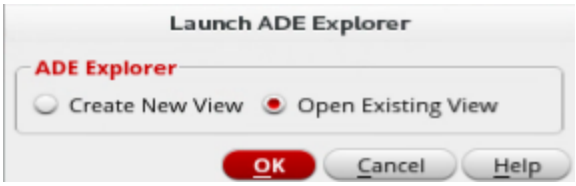


Figure 202: Open Existing Explorer View

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 202: In the **Open ADE Explorer View** window, verify that the view is set to **maestro**. Then, click **OK**.

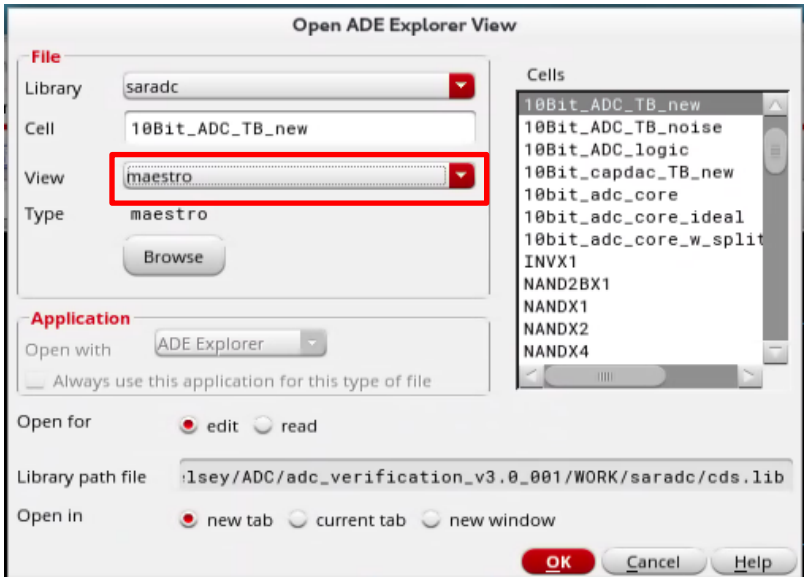


Figure 203: Open Explorer Maestro View

The **maestro** view should look like the Figure 204 below.

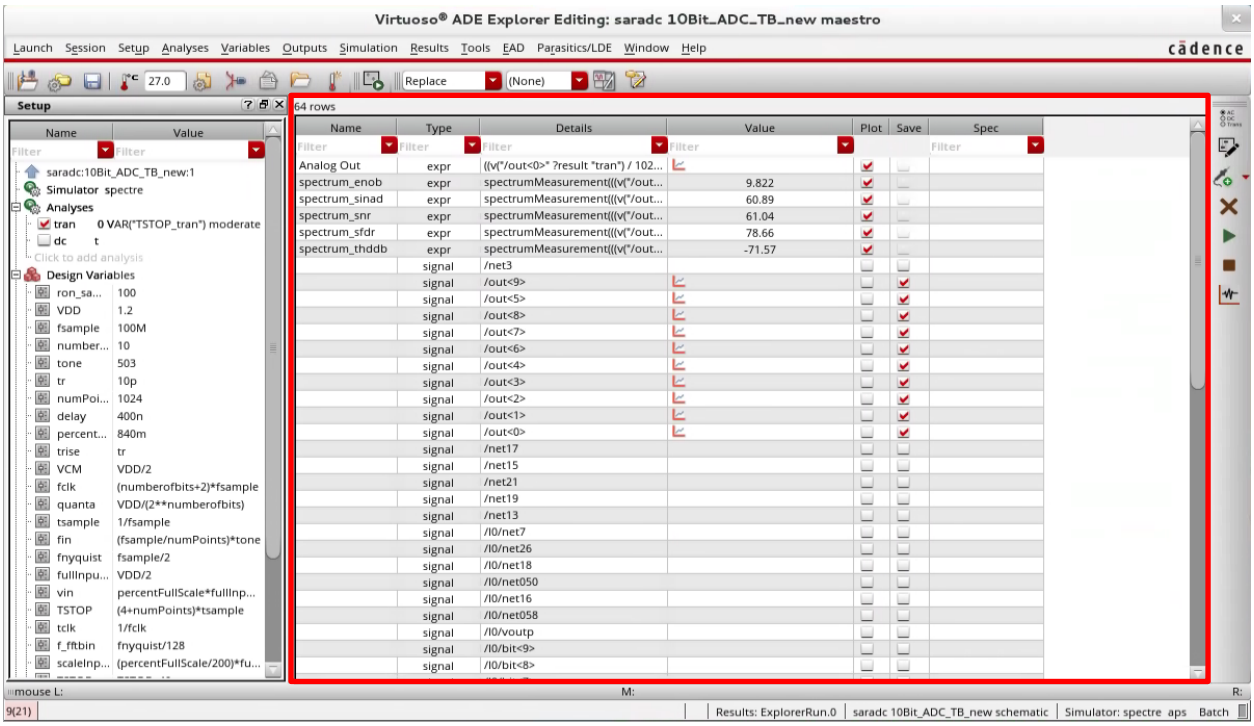


Figure 204: ADE Explorer Window

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 203: Double-click on the **tran** analysis in the **Analyses** section of the Setup Assistant.

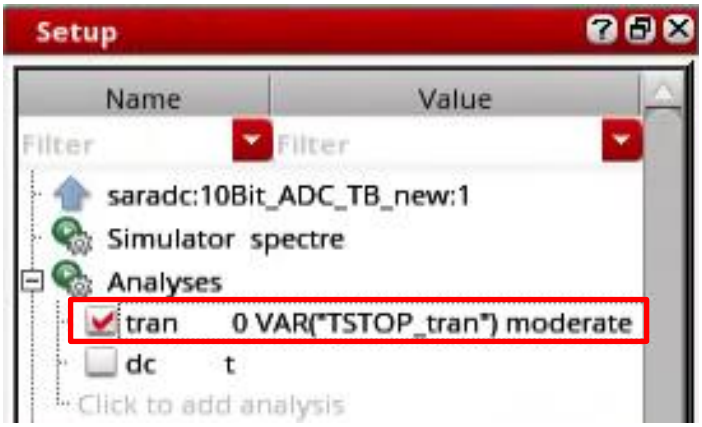


Figure 205: Setup Assistant for Analyses

ACTION 204: The **Choosing Analyses — ADE Explorer** window should appear as shown in Figure 206. After looking this over, click **OK** to exit the window.

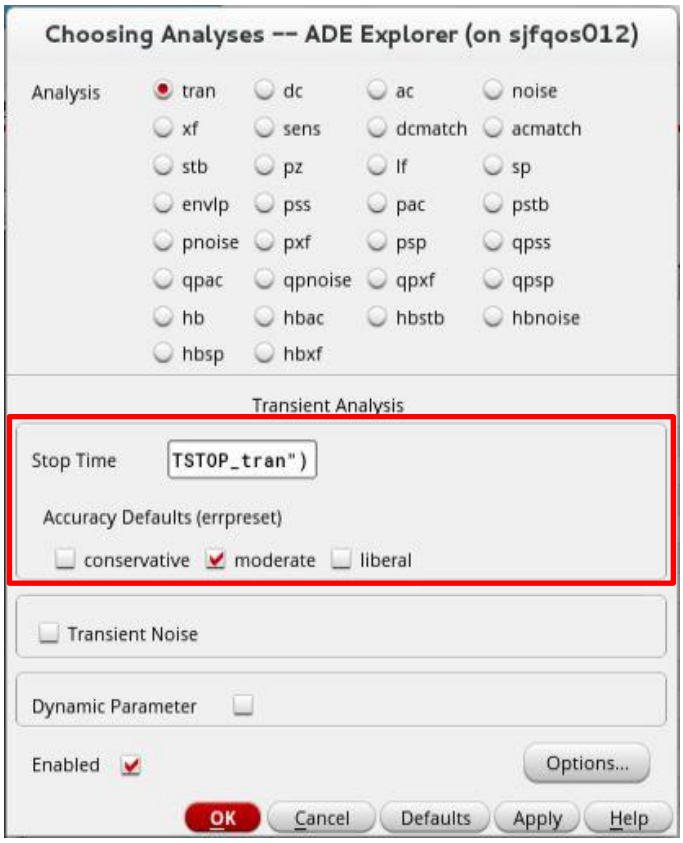


Figure 206 : Choosing Analyses -- ADE Explorer Window

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 205: To start the simulation, select the ("**Run Simulation**") button. This simulation will take some time to complete. Make sure that you have enough memory space to store the results of the simulation.

Note: You can skip running the simulation by loading the results of a previous simulation in the Assembler **maestro** view of the **10Bit_ADC_TB_new** testbench. Click on the icon to get to the Assembler **maestro** view, and then click on the **History** tab. RMB-click on **Interactive.1** and then select **View Results** from the pull-down menu to see the previous data.

ACTION 206: After the simulation is complete, a ViVA window will appear with the analog output of the ADC and the different spectrum calculations.

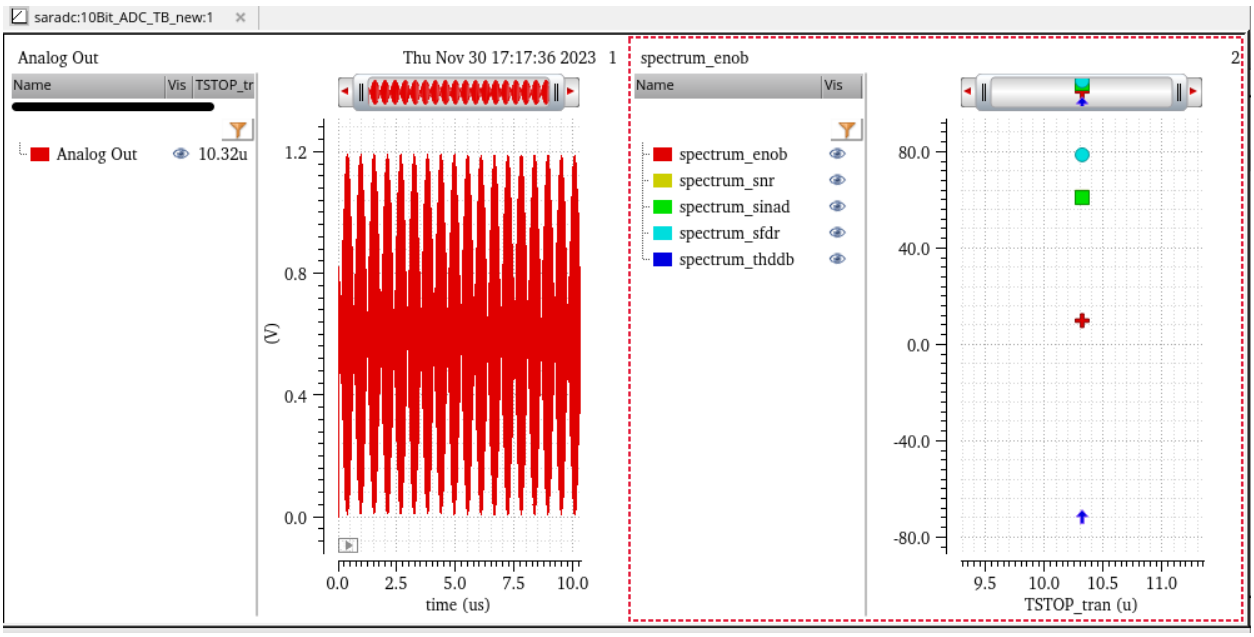


Figure 207: Analog Output and Spectrum Calculations of ADC

ACTION 207: Minimize the ViVA window and look at the Explorer window for the values of the ENOB, SINAD, SNR, SFD and THD.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

10Bit_ADC_TB_new | 10Bit_ADC_TB_new

64 rows

Name	Type	Details	Value	Plot	Save	Spec
Filter	Filter	Filter	Filter			Filter
Analog Out	expr	((v("/out<0>" ?result "tran") / 102...		✓		
spectrum_enob	expr	spectrumMeasurement(((v("/out...	9.822	✓		
spectrum_sinad	expr	spectrumMeasurement(((v("/out...	60.89	✓		
spectrum_snr	expr	spectrumMeasurement(((v("/out...	61.04	✓		
spectrum_sfdr	expr	spectrumMeasurement(((v("/out...	78.66	✓		
spectrum_thddb	expr	spectrumMeasurement(((v("/out...	-71.57	✓		
	signal	/net3				

Figure 208: ADC Transient Analysis Results

These results will be used in the spreadsheet to validate and compare the results of this RAK. If the RAK's calculation of the ADC's SINAD is comparable to the ADC Transient SINAD, this proves the RAK's accuracy and purpose.

ACTION 208: Close the ViVA and ADE Explorer windows by clicking on the (**Exit**) button in the upper-right corner. Select "**No**" if asked to save any changes.

Lab 6-3: Top-Level ADC Transient Noise Analysis

In this section, you will run a transient noise analysis on the top-level ADC testbench to see how the SINAD and ENOB change with applied noise. You will use the value and compare it to the SINAD simulated in Lab 6-2.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 209: In the Library Manager, select **saradc > Everything > 10Bit_ADC_TB_noise**. Double-click on **schematic** to open the schematic.

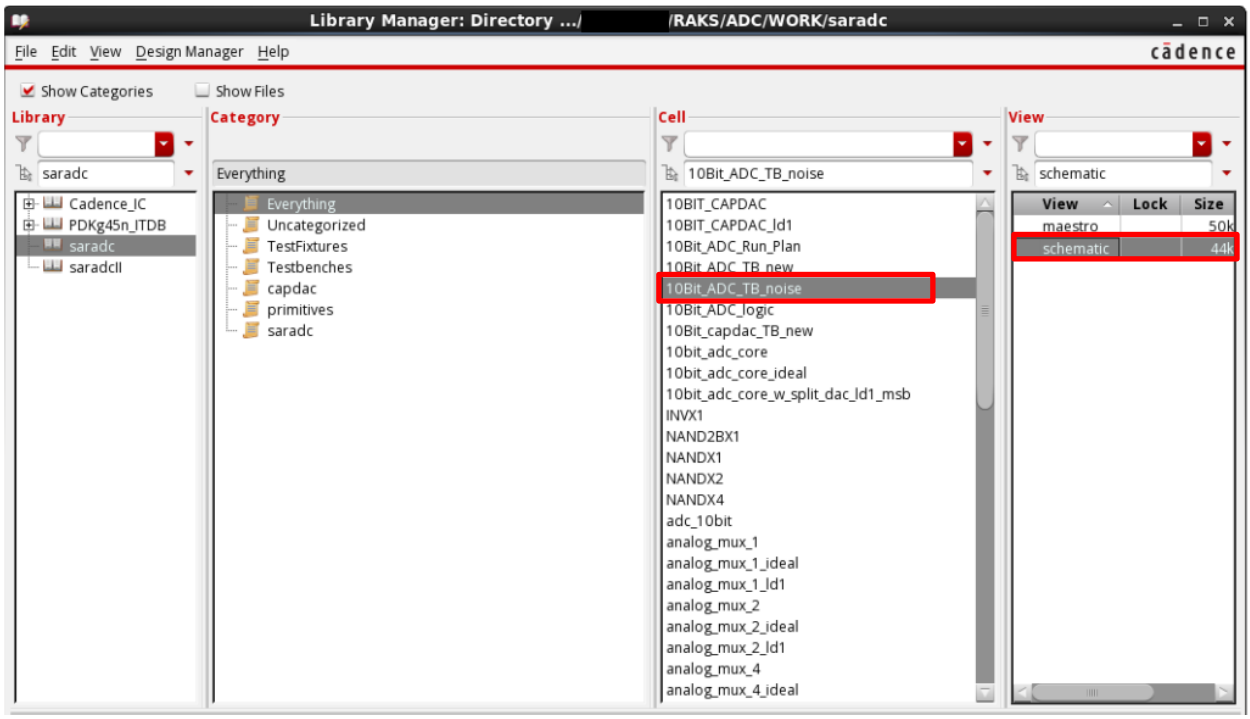


Figure 209: Library Manager

ACTION 210: In the schematic window, select **Launch > ADE Explorer**.

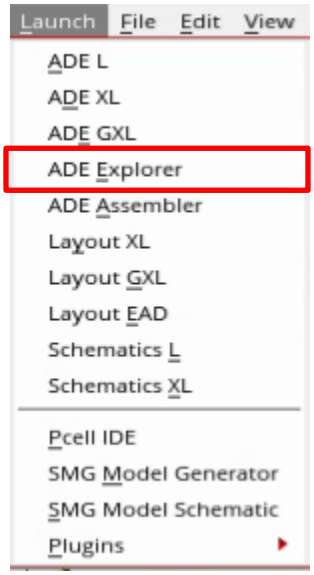


Figure 210: Launch ADE Explorer

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 211: In the **Launch ADE Explorer** window, select **Open Existing View**. Then, click **OK**.

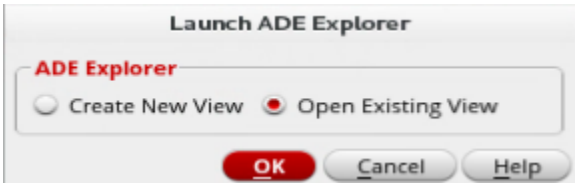


Figure 211: Open Existing Explorer View

ACTION 212: In the **Open ADE Explorer View** window, verify that the view is set to **maestro**. Then, click **OK**.

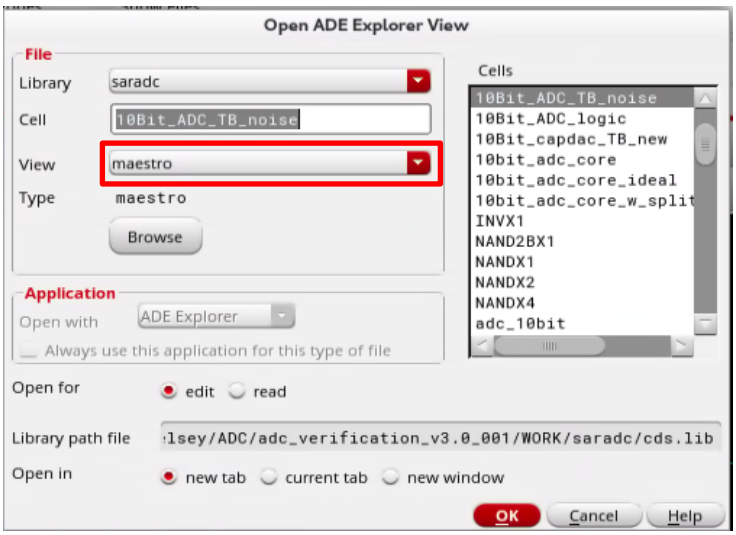


Figure 212: Open Explorer Maestro View

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

The **maestro** tab should look like Figure 213 below.

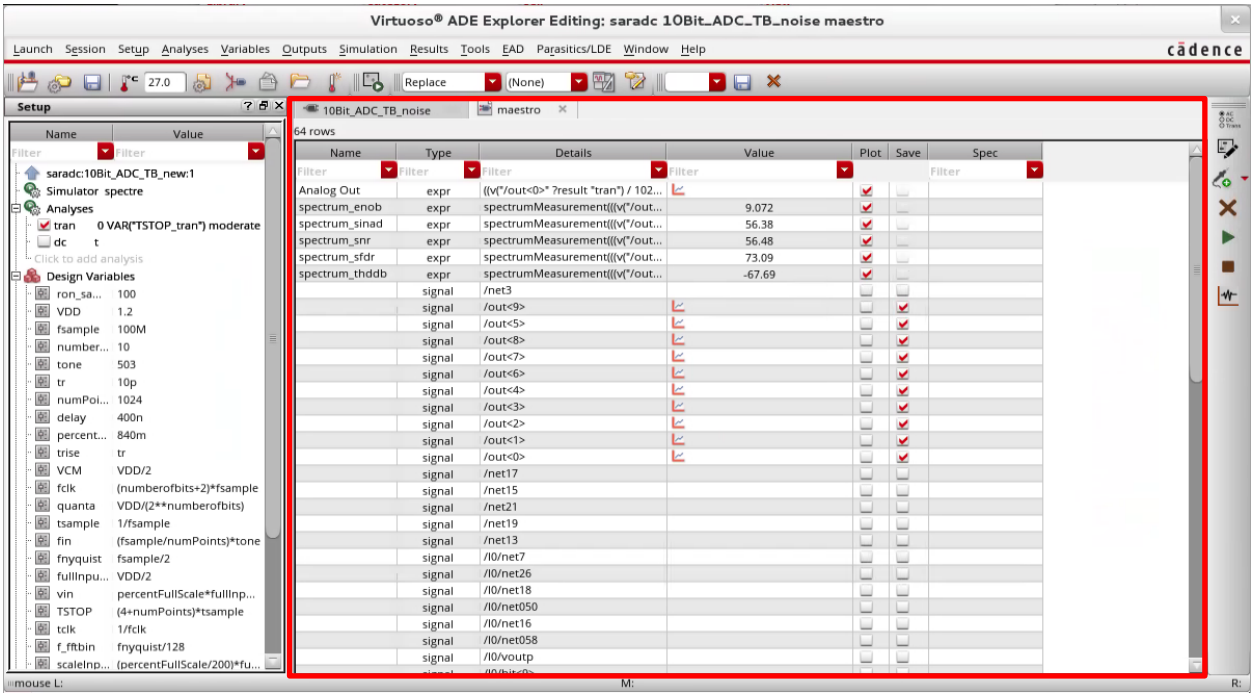


Figure 213: ADE Explorer Window

ACTION 213: Double-click on the **tran** analysis in the **Analyses** section of the Setup Assistant.

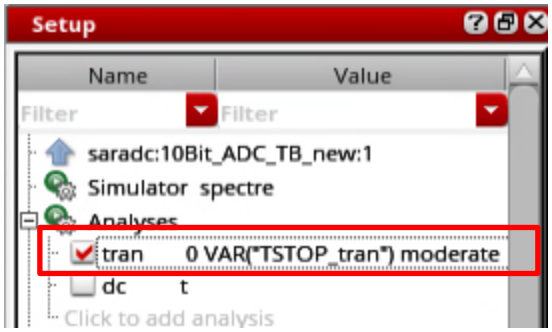


Figure 214: Setup Assistant for Analyses

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 214: The **Choosing Analyses — ADE Explorer** window should appear as shown in Figure 215. After looking this over, click **OK** to exit the window.

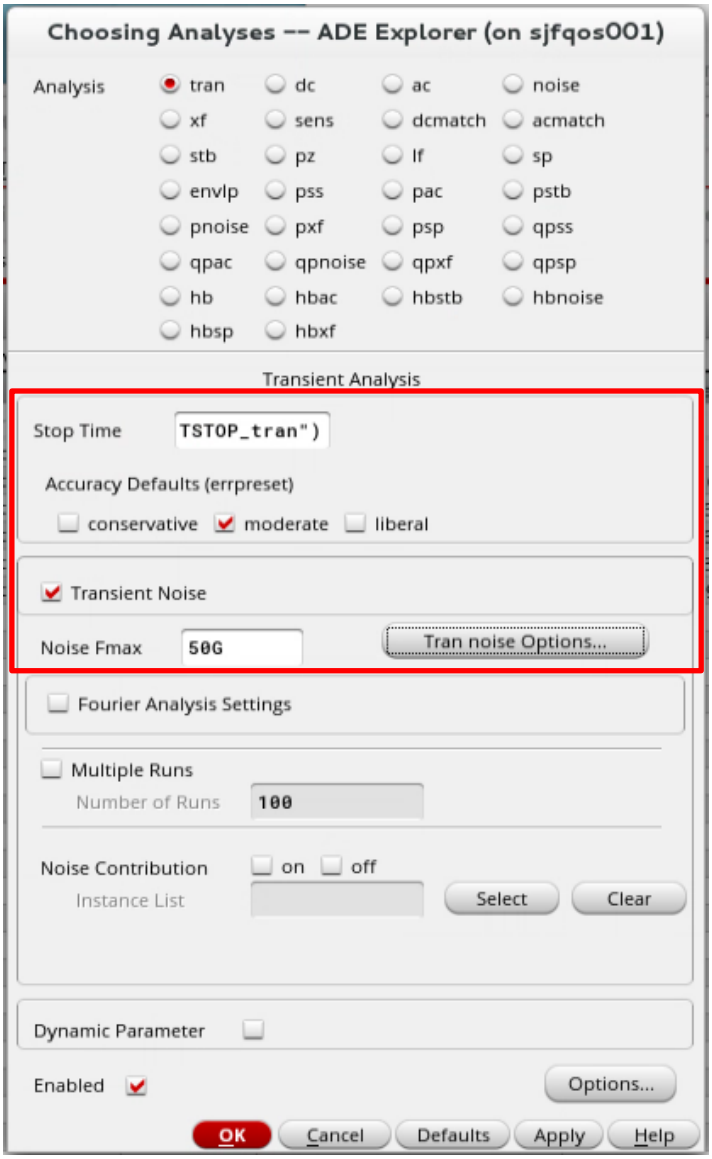


Figure 215: Choosing Analyses Window for ADC Transient Noise Analysis

TSTOP_tran is simply the value TSTOP plus 40ns. TSTOP is determined by 4 plus the number of points in the FFT (for you, it is 1024) times the period of the sample frequency.

Make sure the **Transient Noise** option is selected. The **Noise Fmax** value should be set to **50G**. Noise Fmax sets the bandwidth of the random noise sources that are injected at each time point in the transient analysis. Since your ADC's clock frequency is

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

set to about 1.2GHz, you want your maximum noise frequency to be about 40 times the clocked frequency of the ADC.

ACTION 215: To start the simulation, select the ("**Run Simulation**") button. This simulation will take some time to complete. Make sure that you have enough memory space to store the results of the simulation.

Note: You can skip running the simulation by loading the results of a previous simulation in the Assembler **maestro** view of the **10Bit_ADC_TB_noise** testbench. Click on the icon to get to the Assembler **maestro** view, and then click on the **History** tab. RMB-click on **Interactive.0** and then select **View Results** from the pull-down menu to see the previous data.

ACTION 216: After the simulation is complete, a ViVA window will appear with the analog output of the ADC and the different spectrum calculations.

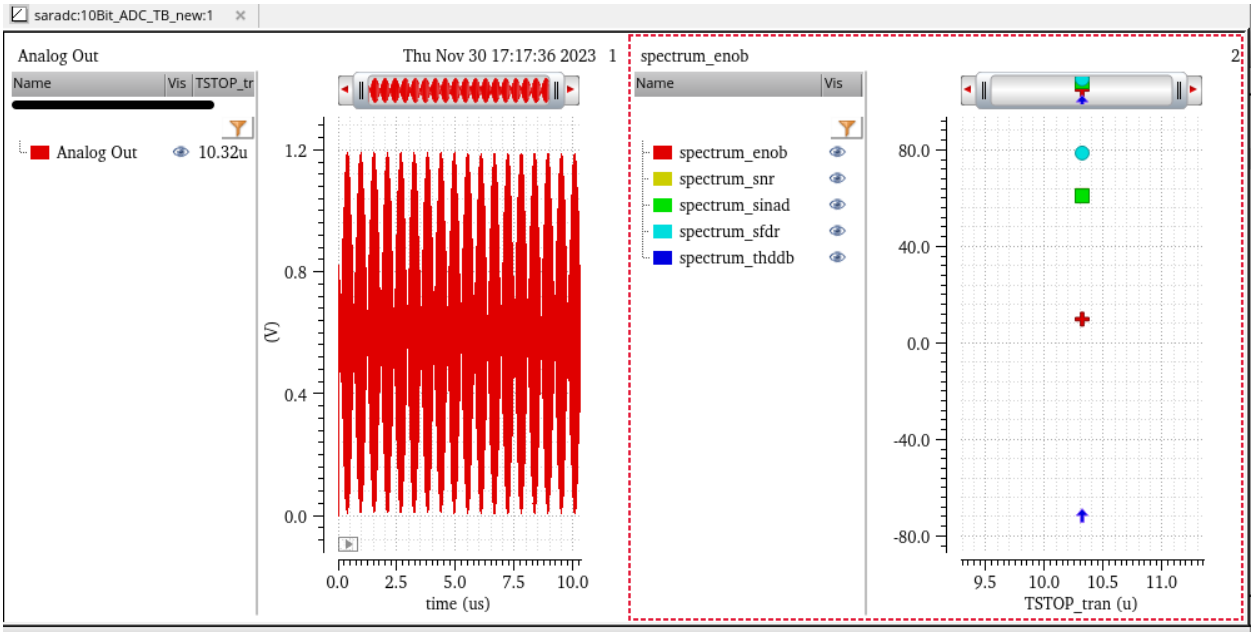
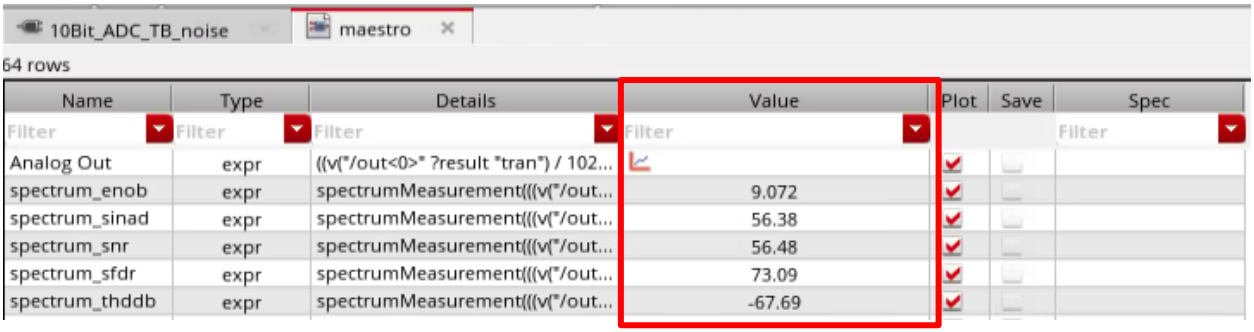


Figure 216: Analog Output and Spectrum Calculations of ADC

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 217: Minimize the ViVA window and look at the Explorer window for the values of the ENOB, SINAD, SNR, SFD, and THD.



Name	Type	Details	Value	Plot	Save	Spec
Filter	Filter	Filter	Filter			Filter
Analog Out	expr	((v("/out<0>" ?result "tran") / 102...		✓		
spectrum_enob	expr	spectrumMeasurement(((v("/out...	9.072	✓		
spectrum_sinad	expr	spectrumMeasurement(((v("/out...	56.38	✓		
spectrum_snr	expr	spectrumMeasurement(((v("/out...	56.48	✓		
spectrum_sfdr	expr	spectrumMeasurement(((v("/out...	73.09	✓		
spectrum_thddb	expr	spectrumMeasurement(((v("/out...	-67.69	✓		

Figure 217: ADC Transient Noise Analysis Results

The SINAD value calculated in this lab will be used to check the SINAD and ENOB calculated in the spreadsheet. You would like your final calculations discussed in the next section to be similar to the ADC transient noise analysis results shown in Figure 217.

ACTION 218: Close the ViVA and ADE Explorer windows by clicking on the (**Exit**) button in the upper-right corner. Select "**No**" if asked to save any changes.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Lab 6-4: Top-Level ADC Run Plan

In this section, you will be going over a Run Plan to enable you to run all the simulations discussed in the previous labs, all in one ADE Assembler window. The results of this Run Plan will make it easier to calculate the ADC's SINAD and ENOB without referencing the Excel spreadsheet.

ACTION 219: In the Library Manager, select **saradc > Everything > 10Bit_ADC_Run_Plan**. Double-click on **maestro** to open the maestro view.

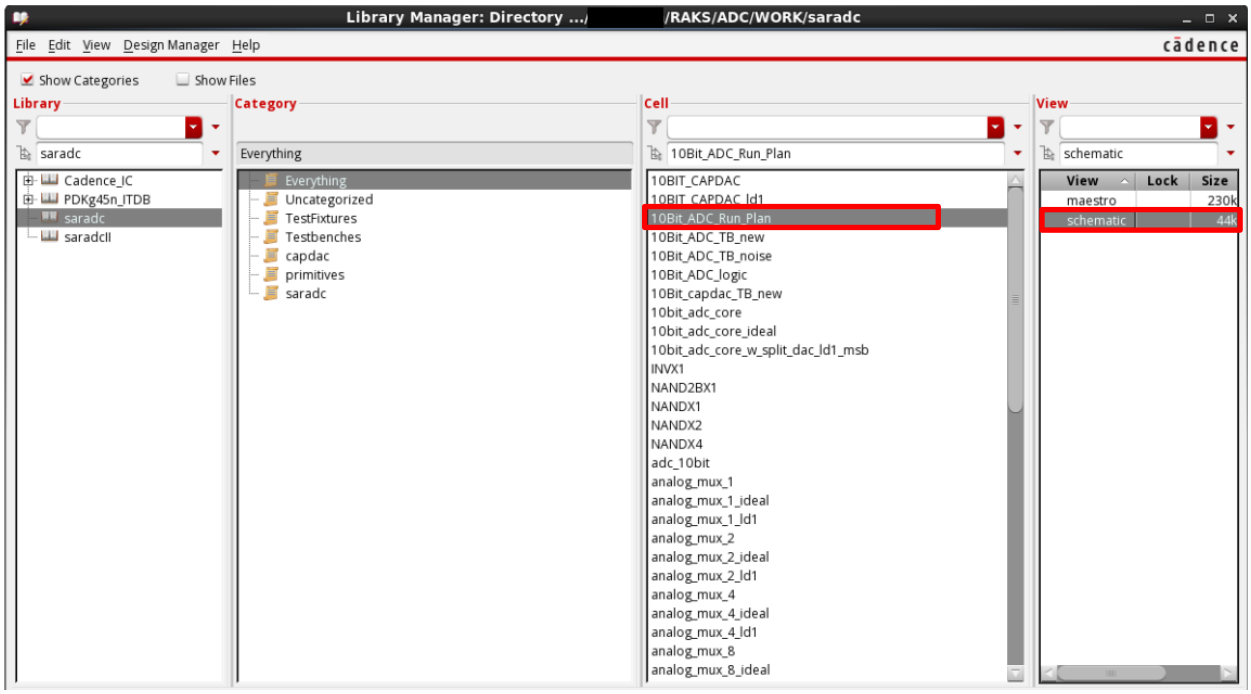


Figure 218: Library Manager

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 220: The **maestro** view will open an ADE Assembler window containing all the tests you have completed in Modules 2 through 5.

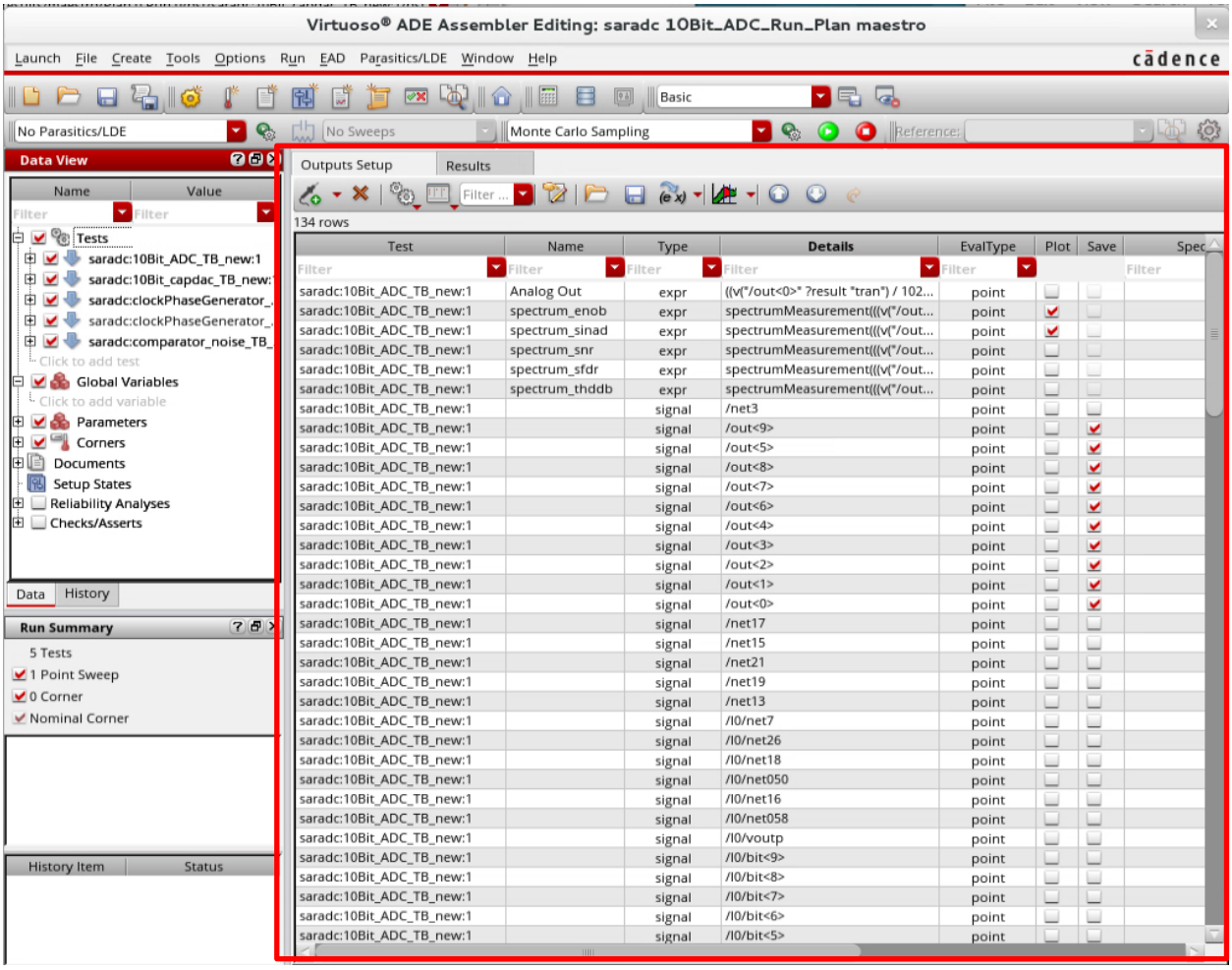


Figure 219: ADE Assembler Window for Run Plan

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 221: Click on **Window > Assistants > Run Plan** to display the **Run Plan** window to the right of the screen, as shown in Figure 220 and Figure 221.

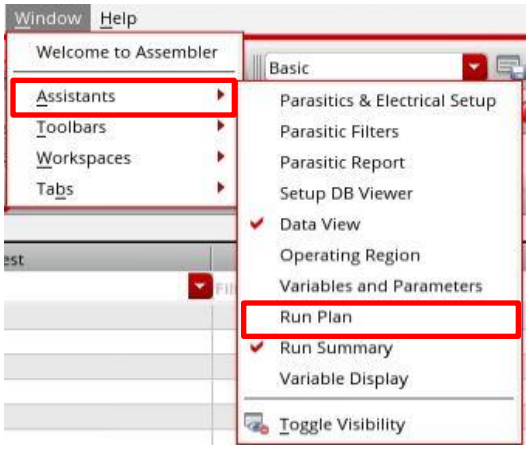


Figure 220: Enable Run Plan Assistant

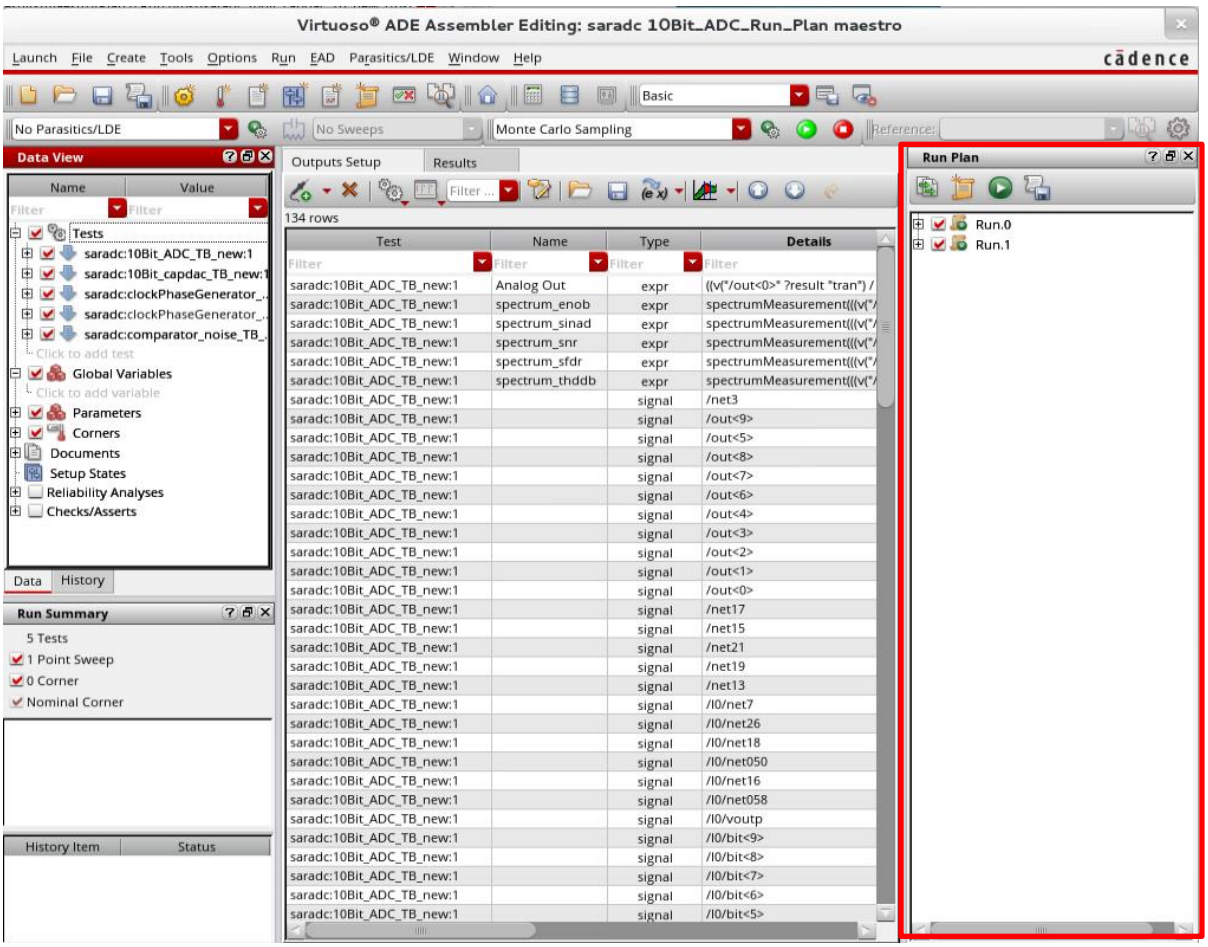


Figure 221: ADE Assembler Window for Run Plan with Run Plan Assistant

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 222: First, you will examine the content of the first Run Plan, **Run.0**, shown in Figure 222.

Figure 222: Run.0 Corresponding to Transient and PSS Analyses on Different ADC Blocks

The following tests are selected:

- 10Bit_ADC_TB_new
- 10Bit_capdac_TB_new
- clockPhaseGenerator_TB_new
- clockPhaseGenerator_dc2dc_TB_new
- comparator_noise_TB_new

These tests are selected because they all implement a transient analysis to find the values required to calculate the ADC's ENOB and SINAD.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 223: Next, you will examine the content of the second Run Plan, **Run.1**, shown in Figure 223.

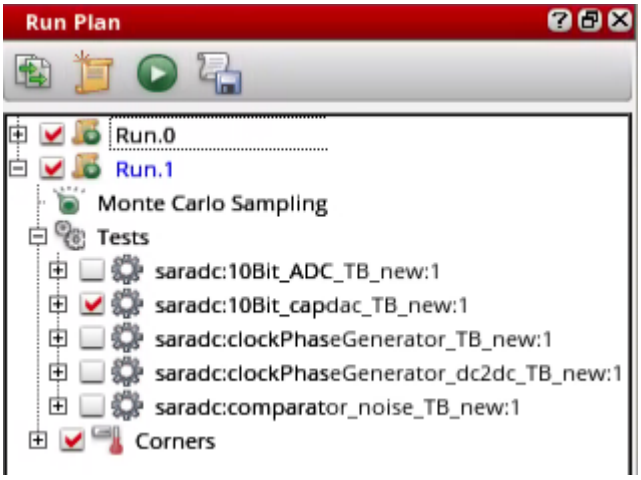


Figure 223: Run.1 Corresponding to Monte Carlo Analyses Related to the Dynamic Comparator and CAPDAC

The following test is selected:

10Bit_capdac_TB_new

This test is selected because it implements a Monte Carlo analysis to find the values required to calculate the ADC's ENOB and SINAD. You will use the mean value of the Monte Carlo's yield as the matching CAPDAC SINAD value in the upcoming calculations.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 224: **Run.1** is the only Run Plan that has run conditions attached to it. RMB-click on **Run.1** and then click on **Edit Run Conditions** from the drop-down menu. The **Add/Edit Run Conditions – Run.1** window should open and resemble Figure 225.

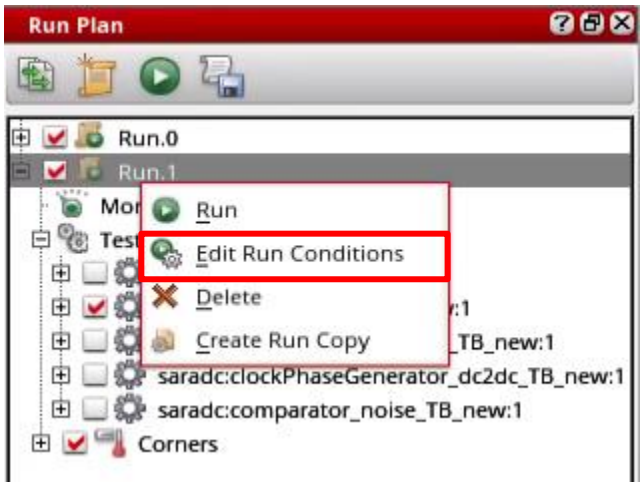


Figure 224: Edit Run Conditions

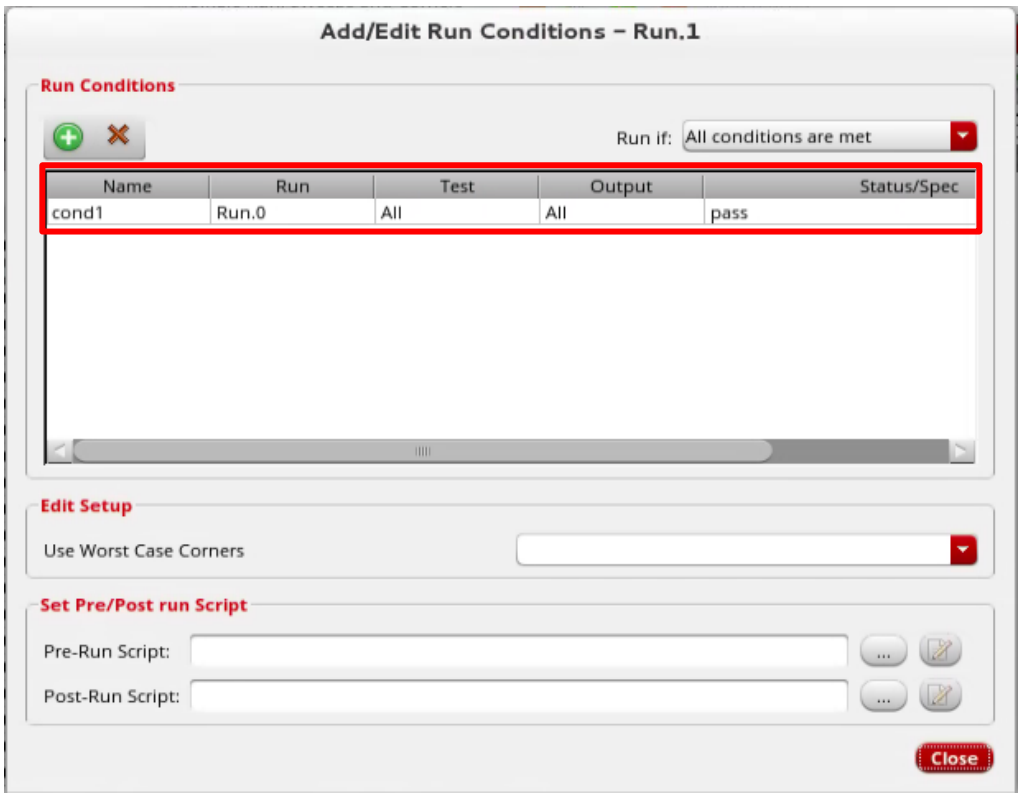


Figure 225: Add/Edit Run Conditions – Run.1 Window

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 225: The only condition applied to **Run.1** is for **Run.0** to complete successfully. If **Run.0** does not pass, **Run.1** will not run. This saves a lot of time because **Run.1** consists of the Monte Carlo analysis, which is quite lengthy to run.

Click on **Close** when you are done analyzing the run conditions on **Run.1**.

ACTION 226: After looking at each Run Plan, start the simulation by clicking on the ("**Run Simulation**") button. This simulation will take about a day to complete, depending on your machine's performance. Make sure that you have enough memory space to store the results of the simulation.

Note: You can load previous data by clicking the **History** tab and then doing an RMB-click on **Plan.1 → Load Setup to Active**. You can also load the previous results by doing an RMB-click on **Plan.1 → View Results**.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 227: Once each Run Plan has successfully completed, click on the **Results** tab to view the results of each test. They should match everything that was done in the Modules before this, plus more.

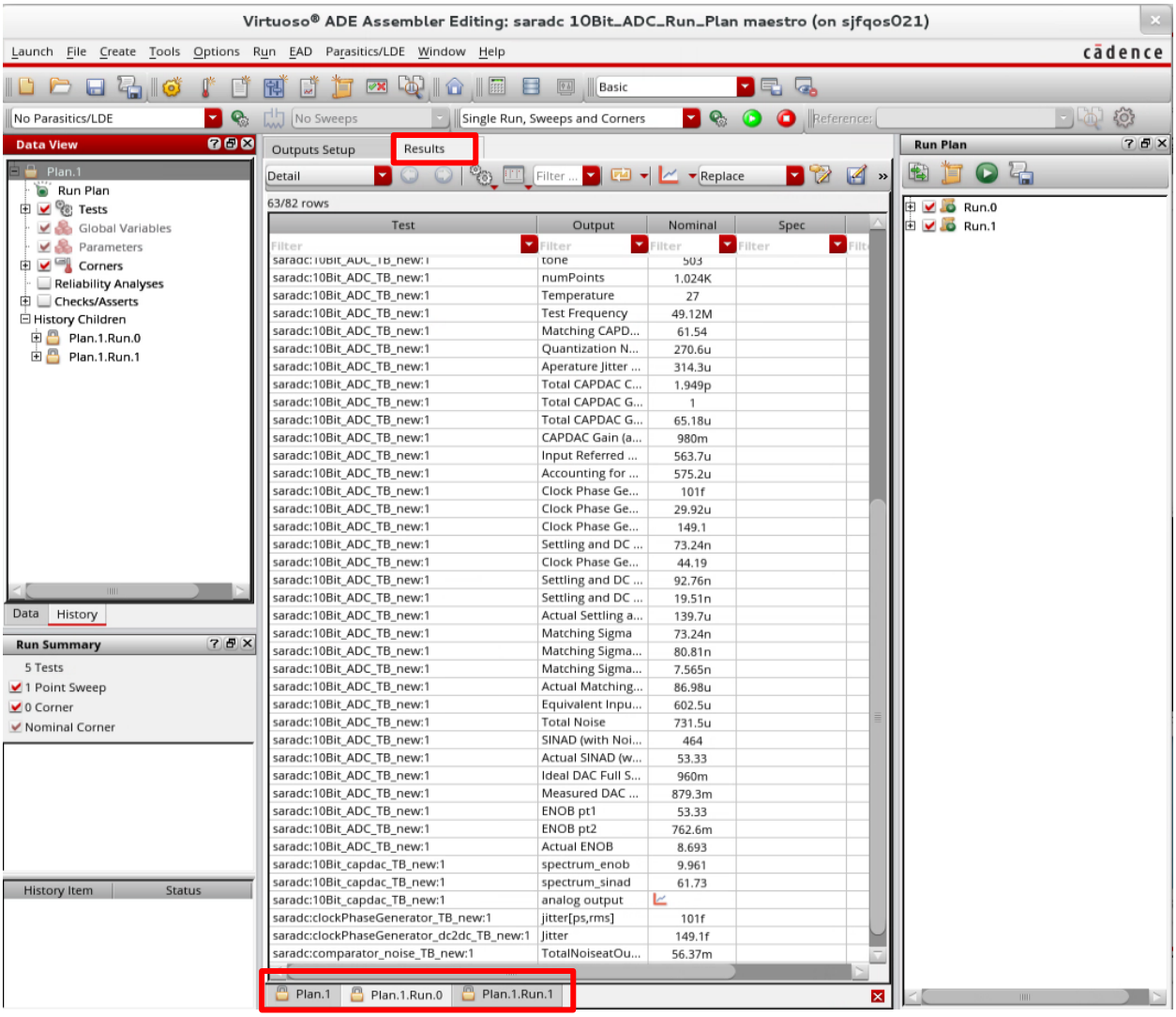


Figure 226: Run Plan Results

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Note: If your **Run.1** crashes or does not finish, you can always RMB-click on the run in the **History** tab and select **Re-run Unfinished/Error Points**. It will create a new run to finish the remining unfinished Monte Carlo iterations and will result in a new run called **Plan.0.Run.1.rerun.0**. This can be seen if you load the results that come with the RAK.

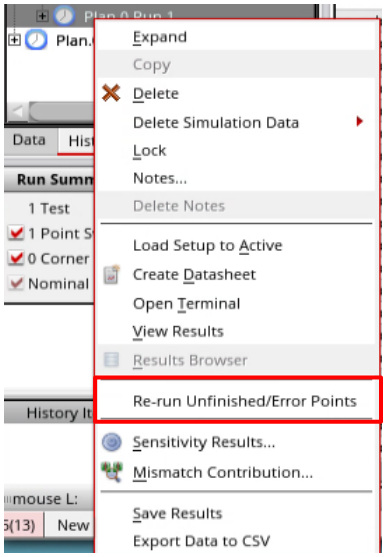


Figure 227: Re-run Unfinished/Error Points

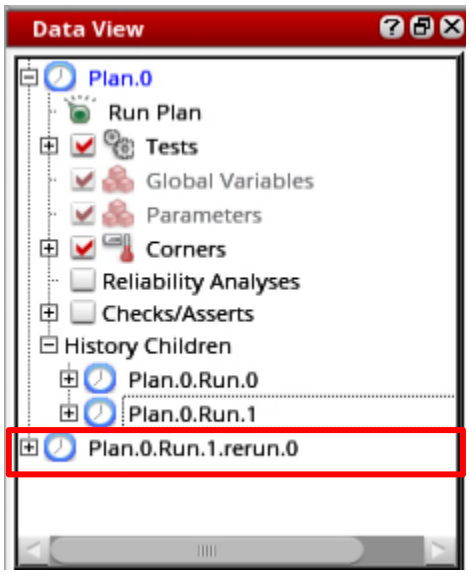


Figure 228: Data View with Plan.0.Run.1.rerun.0

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 228: Click on the **Plan.1.Run.0 Results** tab at the bottom of the window. This will display all the values used to calculate the ENOB and SINAD of the ADC instead of using the Excel worksheet in the Summary Section below. The value for **Actual ENOB** should match the value calculated in the Excel worksheet.

Outputs Setup | Results

Detail | Filter ... | Replace

63/82 rows

Test	Output	Nominal	Spec	Weigh
Filter	Filter	Filter	Filter	Filter
saradc:10Bit_ADC_TB_new:1	/I0/sump			
saradc:10Bit_ADC_TB_new:1	/I0/sumn			
saradc:10Bit_ADC_TB_new:1	Number of Bits	10		
saradc:10Bit_ADC_TB_new:1	Full Scale Input	960m		
saradc:10Bit_ADC_TB_new:1	Full Scale Input (rms)	339.4m		
saradc:10Bit_ADC_TB_new:1	Quanta size	937.5u		
saradc:10Bit_ADC_TB_new:1	Aperature Jitter	3p		
saradc:10Bit_ADC_TB_new:1	Sample Frequency	100M		
saradc:10Bit_ADC_TB_new:1	tone	503		
saradc:10Bit_ADC_TB_new:1	numPoints	1.024K		
saradc:10Bit_ADC_TB_new:1	Temperature	27		
saradc:10Bit_ADC_TB_new:1	Test Frequency	49.12M		
saradc:10Bit_ADC_TB_new:1	Matching CAPDAC SINAD	61.54		
saradc:10Bit_ADC_TB_new:1	Quantization Noise	270.6u		
saradc:10Bit_ADC_TB_new:1	Aperature Jitter Noise	314.3u		
saradc:10Bit_ADC_TB_new:1	Total CAPDAC Capacitance	1.949p		
saradc:10Bit_ADC_TB_new:1	Total CAPDAC Gain (or Attenua...	1		
saradc:10Bit_ADC_TB_new:1	Total CAPDAC Gain Input Refer...	65.18u		
saradc:10Bit_ADC_TB_new:1	CAPDAC Gain (attenuation fro...	980m		
saradc:10Bit_ADC_TB_new:1	Input Referred PSS Measured C...	563.7u		
saradc:10Bit_ADC_TB_new:1	Accounting for CAPDAC attenu...	575.2u		
saradc:10Bit_ADC_TB_new:1	Clock Phase Generator Jitter	101f		
saradc:10Bit_ADC_TB_new:1	Clock Phase Generator Noise (I...	29.92u		
saradc:10Bit_ADC_TB_new:1	Clock Phase Generator Jitter No...	149.1f		
saradc:10Bit_ADC_TB_new:1	Settling and DC Accuracy	73.24n		
saradc:10Bit_ADC_TB_new:1	Clock Phase Generator Noise (I...	44.19u		
saradc:10Bit_ADC_TB_new:1	Settling and DC Accuracy pt1	92.76n		
saradc:10Bit_ADC_TB_new:1	Settling and DC Accuracy pt2	19.51n		
saradc:10Bit_ADC_TB_new:1	Actual Settling and DC Accuracy	139.7u		
saradc:10Bit_ADC_TB_new:1	Matching Sigma	73.24n		
saradc:10Bit_ADC_TB_new:1	Matching Sigma pt1	80.81n		
saradc:10Bit_ADC_TB_new:1	Matching Sigma pt2	7.565n		
saradc:10Bit_ADC_TB_new:1	Actual Matching Sigma	86.98u		
saradc:10Bit_ADC_TB_new:1	Equivalent Input Referred Ther...	604.2u		
saradc:10Bit_ADC_TB_new:1	Total Noise	732.8u		
saradc:10Bit_ADC_TB_new:1	SINAD (with Noise)	463.2		
saradc:10Bit_ADC_TB_new:1	Actual SINAD (with Noise)	53.31		
saradc:10Bit_ADC_TB_new:1	Ideal DAC Full Scale Output	960m		
saradc:10Bit_ADC_TB_new:1	Measured DAC Output	879.3m		
saradc:10Bit_ADC_TB_new:1	ENOB pt1	53.31		
saradc:10Bit_ADC_TB_new:1	ENOB pt2	762.6m		
saradc:10Bit_ADC_TB_new:1	Actual ENOB	8.691		
saradc:10Bit_capdac_TB_new:1	spectrum_enob	9.961		
saradc:10Bit_capdac_TB_new:1	spectrum_sinad	61.73		
saradc:10Bit_capdac_TB_new:1	analog output			
saradc:clockPhaseGenerator_TB_new:1	jitter[ps,rms]	101f		
saradc:clockPhaseGenerator_dc2dc_TB_new:1	Jitter	149.1f		
saradc:comparator_noise_TB_new:1	TotalNoiseatOutput	56.37m		
saradc:comparator_noise_TB_new:1	InputReferredNoise	563.7u		

Plan.1 | Plan.1.Run.0 | Plan.1.Run.1

Figure 229: Results Tab Showing Expression Results of Run Plan

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 229: Click on the **Data** tab in the Data View Assistant and then click on the "**+**" symbol by **Tests**. Click on the icon next to **saradc:10Bit_ADC_TB_new:1** to open the Explorer View of the test. The Explorer View should appear like Figure 230.

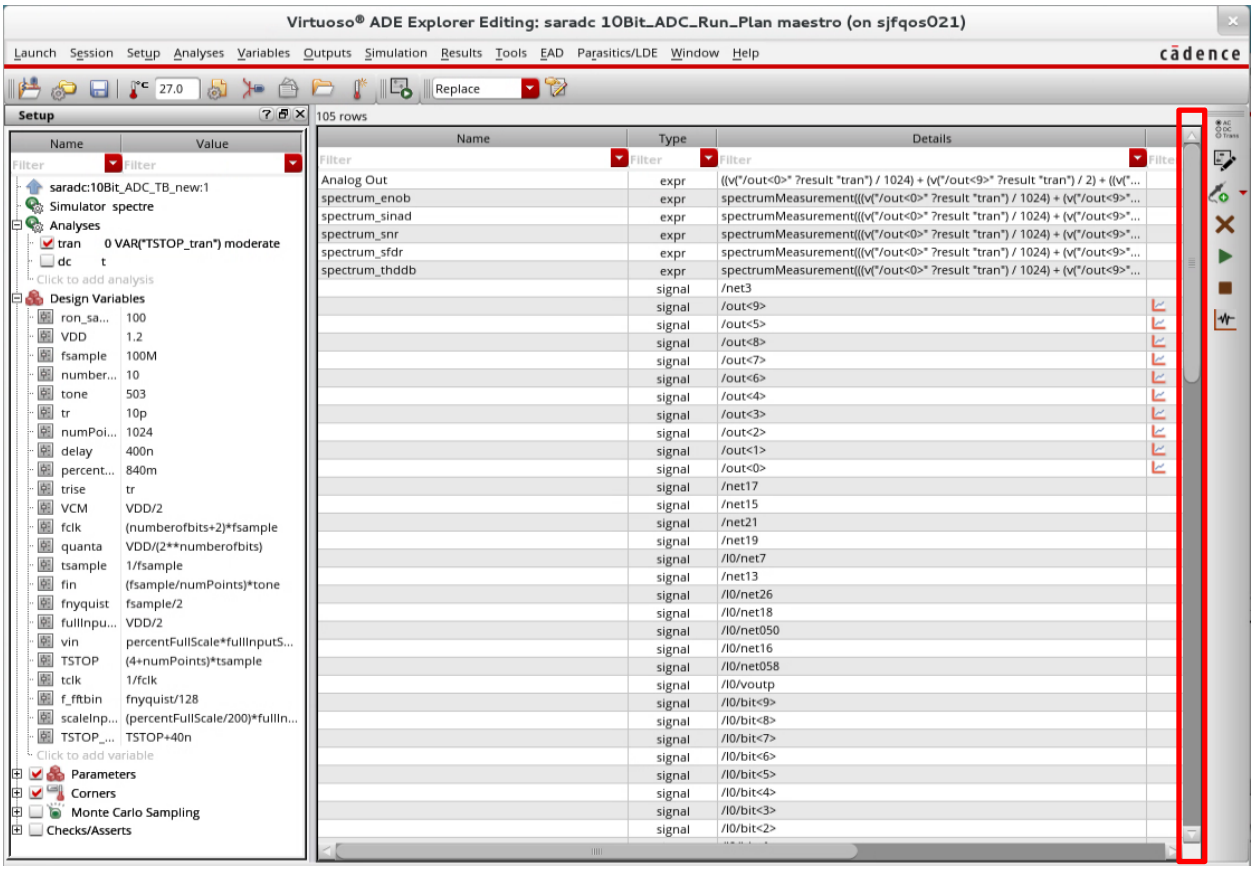


Figure 230: Explorer Maestro View of the 10Bit_ADC_TB_new Testbench

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

ACTION 230: Scroll down the **Outputs** plane view to see all the equations used to calculate the ADC's ENOB and SINAD. Appendix A contains the formulas used for each of the expressions shown in Figure 231.

104 rows

Name	Type	Details	Value
Number of Bits	expr	10	10
Full Scale Input	expr	0.96	960m
Full Scale Input (rms)	expr	(calcVal("Full Scale Input" "saradc:10Bit_ADC_TB_new:1") / (2 * sqrt(2)))	339.4m
Quanta size	expr	(calcVal("Full Scale Input" "saradc:10Bit_ADC_TB_new:1") / pow(2 calcVal("Number of Bits" "saradc:10Bit_ADC_TB_new:1")))	937.5u
Aperature Jitter	expr	3e-12	3p
Sample Frequency	expr	100000000	100M
tone	expr	503	503
numPoints	expr	1024	1.024K
Temperature	expr	27	27
Test Frequency	expr	((calcVal("Sample Frequency" "saradc:10Bit_ADC_TB_new:1") / calcVal("numPoints" "saradc:10Bit_ADC_TB_new:1")) * calcVal("t...	49.12M
Matching CAPDAC SINAD	expr	61.54	61.54
Quantization Noise	expr	(calcVal("Quanta size " "saradc:10Bit_ADC_TB_new:1") / sqrt(12))	270.6u
Aperature Jitter Noise	expr	(calcVal("Full Scale Input (rms)" "saradc:10Bit_ADC_TB_new:1") * 2 * 3.14159 * calcVal("Test Frequency" "saradc:10Bit_ADC_TB_...	314.3u
Total CAPDAC Capacitance	expr	(2 * (2 * (16 + 8 + 4 + 2 + 1 + 1)) * 1.52265e-14)	1.949p
Total CAPDAC Gain (or Attenuation)	expr	1	1
Total CAPDAC Gain Input Referred Noise	expr	(sqrt(2) * sqrt(((1.38e-23 * (calcVal("Temperature" "saradc:10Bit_ADC_TB_new:1") + 273)) / (calcVal("Total CAPDAC Gain (or Atte...	65.18u
CAPDAC Gain (attenuation from parasitic cap)	expr	0.98	980m
Input Referred PSS Measured Comparator Noise	expr	calcVal("InputReferredNoise" "saradc:comparator_noise_TB_new:1")	563.7u
Accounting for CAPDAC attenuation	expr	(calcVal("Input Referred PSS Measured Comparator Noise" "saradc:10Bit_ADC_TB_new:1") / calcVal("CAPDAC Gain (attenuatio...	575.2u
Clock Phase Generator Jitter	expr	calcVal("jitter[ps,rms]" "saradc:clockPhaseGenerator_TB_new:1")	101f
Clock Phase Generator Noise (Input Referred)	expr	(calcVal("Clock Phase Generator Jitter" "saradc:10Bit_ADC_TB_new:1") * 2 * 3.14159 * calcVal("Full Scale Input" "saradc:10Bit_A...	29.92u
Clock Phase Generator Jitter Noisy	expr	calcVal("jitter" "saradc:clockPhaseGenerator_dc2dc_TB_new:1")	149.1f
Settling and DC Accuracy	expr	pow(calcVal("Quantization Noise" "saradc:10Bit_ADC_TB_new:1") 2)	73.24n
Clock Phase Generator Noise (Input Referred) Noisy	expr	(calcVal("Clock Phase Generator Jitter Noisy" "saradc:10Bit_ADC_TB_new:1") * 2 * 3.14159 * calcVal("Full Scale Input" "saradc:1...	44.19u
Settling and DC Accuracy pt1	expr	pow((calcVal("Full Scale Input (rms)" "saradc:10Bit_ADC_TB_new:1") / pow(10 (calcVal("spectrum_sinad" "saradc:10Bit_ADC_TB...	92.76n
Settling and DC Accuracy pt2	expr	(calcVal("Settling and DC Accuracy pt1" "saradc:10Bit_ADC_TB_new:1") - calcVal("Settling and DC Accuracy " "saradc:10Bit_ADC...	19.51n
Actual Settling and DC Accuracy	expr	sqrt(calcVal("Settling and DC Accuracy pt2" "saradc:10Bit_ADC_TB_new:1"))	139.7u
Matching Sigma	expr	pow(calcVal("Quantization Noise" "saradc:10Bit_ADC_TB_new:1") 2)	73.24n
Matching Sigma pt1	expr	pow((calcVal("Full Scale Input (rms)" "saradc:10Bit_ADC_TB_new:1") / pow(10 (calcVal("Matching CAPDAC SINAD" "saradc:10Bit...	80.81n
Matching Sigma pt2	expr	(calcVal("Matching Sigma pt1 " "saradc:10Bit_ADC_TB_new:1") - calcVal("Matching Sigma" "saradc:10Bit_ADC_TB_new:1"))	7.565n
Actual Matching Sigma	expr	sqrt(calcVal("Matching Sigma pt2" "saradc:10Bit_ADC_TB_new:1"))	86.98u
Equivalent Input Referred Thermal Noise	expr	sqrt((pow(calcVal("Total CAPDAC Gain Input Referred Noise" "saradc:10Bit_ADC_TB_new:1") 2) + pow(calcVal("Accounting for ...	604.2u
Total Noise	expr	sqrt((pow(calcVal("Quantization Noise" "saradc:10Bit_ADC_TB_new:1") 2) + pow(calcVal("Aperature Jitter Noise" "saradc:10Bit_...	732.8u
SINAD (with Noise)	expr	(calcVal("Full Scale Input (rms)" "saradc:10Bit_ADC_TB_new:1") / calcVal("Total Noise" "saradc:10Bit_ADC_TB_new:1"))	463.2
Actual SINAD (with Noise)	expr	(20 * log10(calcVal("SINAD (with Noise)" "saradc:10Bit_ADC_TB_new:1")))	53.31
Ideal DAC Full Scale Output	expr	calcVal("Full Scale Input" "saradc:10Bit_ADC_TB_new:1")	960m
Measured DAC Output	expr	(peakToPeak((VT("/I0/sump") - VT("/I0/sumn"))) * 2 * 1.2)	879.3m
ENOB pt1	expr	calcVal("Actual SINAD (with Noise)" "saradc:10Bit_ADC_TB_new:1")	53.31
ENOB pt2	expr	(20 * log10((calcVal("Ideal DAC Full Scale Output" "saradc:10Bit_ADC_TB_new:1") / calcVal("Measured DAC Output" "saradc:10...	762.6m
Actual ENOB	expr	(((calcVal("ENOB pt1" "saradc:10Bit_ADC_TB_new:1") - 1.76) + calcVal("ENOB pt2" "saradc:10Bit_ADC_TB_new:1")) / 6.02)	8.691

Figure 231: Expressions Used to Calculate the ADC's ENOB and SINAD

Note: Run Plan currently does not have the capabilities to import the Monte Carlo mean value using the **calVal()** expression; so, you will have to manually fill in the value of Matching CAPDAC SINAD with the mean value of the Monte Carlo run (**Run.1**).

Note: Similarly, Run Plan currently does not have the ability to extract values from design variables in each test; so, you will need to manually input the user-defined values of your ADC for:

Number of Bits = 10

Full Scale Input = 0.96

Aperture Jitter = 3e-12

Sample Frequency = 100 MHz

tone = 503

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

numPoints = 1024

Temperature = 27

ACTION 231: Every time you make a change to the expressions in the **Output** tab, make sure to click on the "**Plot Outputs**" button ( ) to refresh the **maestro** view.

ACTION 232: After reviewing and making sure that all values match the previous simulations, click on **File > Close** to close the ADE Assembler window.

Note: All expressions used in the Run Plan were taken from the Excel spreadsheet. You can use either the Run Plan or the Excel spreadsheet to calculate your ADC's ENOB and SINAD.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Summary

ADC SINAD Calculation

In this section, you will fill in the values into the Excel spreadsheet and calculate the ADC SINAD and ENOB. The Excel spreadsheet can be found in **$PROJECT/WORK/saradc/SARADC_ENOB_calculation.xlsx**.

	A	B	C	D	E
1	# of Bits	=	10.00		Data Converter Specification
2	Full Scale Input	=	0.96	V(pk-pk) Diff	Data Converter Specification
3	Full Scale Input		339.41	mV (rms)	Calculated mathematically from C2
4	Quanta Size (q)	=	0.94	mV	Calculated mathematically from C1 and C3
5	Aperture Jitter	=	3.00	ps (rms)	Data Converter Specification; this is the maximum allowed jitter for the reference clock
6	Sampling Frequency	=	100.00	MHz	Data Converter Specification
7	Test Frequency	=	49.12	MHz	Data Converter Specification; user-determined for the SINAD target**
8	Temperature	=	27.00	C	Data Converter Specification; user-determined for the SINAD target**
					** These parameters are specific to a test condition, or corner, and these parameters will change every time the analysis conditions change. Each new page of the spreadsheet is a new test condition, and these parameters may need to be updated. The worst-case Test Frequency is the Nyquist ADC, fsample/2.
9	**IDEAL SINAD**		61.96	dB	Calculated mathematically using the standard assumption of quantization noise floor = (1/sqrt(12))*lsb and C1
10	ADC Transient SINAD	=	60.89	dB	Measured from a SINAD Transient Simulation, Lab 6-2
11	Matching SINAD	=	61.42	dB	Measured from CAPDAC Monte Carlo Transient Simulation, Lab 4-2
12	Ideal DAC Full Scale Output	=	0.96	V (p-p)	Same value as Full-Scale Input (C2)
13	Measured Ideal DAC Output	=	0.88	V (p-p)	Measured from CAPDAC output of ADC, Lab 6-2
14	**Quantization Noise**				

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

15	Quantization Noise	=	270.63	µV (rms)	Calculated mathematically using the standard assumption of quantization noise floor = (1/sqrt(12))*lsb, C1, and C4
16	**Jitter Noise**				
17	Aperture Jitter Noise	=	314.26	µV (rms)	Calculated mathematically; jitter noise voltage is slew rate (C3/C7) * jitter noise, time (C5)
18	**CAPDAC Noise**				
19	Total CAPDAC Capacitance		1.95	pF	Calculated mathematically from value of the individual capacitors in the CAPDAC. The CAPDAC is a differential DAC, dual DACs. Each CAPDAC consists of two 5-bit, fully segmented capacitors + 1 series capacitor to scale the lower CAPDAC values. Total Capacitance = 2 * (2 * (16 + 8 + 4 + 2 + 1 + 1))) * Cunit, the series capacitor is treated as the reference cap for the msb DAC.
20	Total CAPDAC Gain (or Attenuation)		1.00		Data Converter Specification, determined by circuit architecture
21	Total CAPDAC Gain Input Referred Noise	x	92.20	µV (rms)	Calculated mathematically, kTC noise from Temperature(C8), capacitance (C19), and noise gain (C20)
22	CAPDAC Gain (attenuation from parasitic cap)		0.98		Measured from a CAPDAC Transient Simulation, not shown. The parasitic backplate capacitance of the device capacitors in the CAPDAC results in attenuation of signal from the ADC, CAPDAC, input to the comparator input, CAPDAC output. This phenomenon is equivalent to noise gain from the CAPDAC output to the CAPDAC input. Since the noise is with reference to the ADC input, it needs to be scaled up by the CAPDAC attenuation.
23	**Strong – Capacitor Noise**				
24	Input Referred PSS Measured Comparator Noise		563.7	µV (rms)	Measured from Pnoise, Lab 3-3
25	Accounting for CAPDAC attenuation	x	575.20	µV (rms)	Calculated mathematically, from noise (C24) and noise gain, CAPDAC attenuation (C22)
26	**Jitter from Phase Clock Generator Noise**				

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

27	Clock Phase Generator Jitter		101	fs (rms)	Measured from Pnoise, Lab 5-1
28	Clock Phase Generator Noise (Input Referred)	=	29.93	µV (rms)	Calculated mathematically from input slew rate(2*PI*C2*C8) and jitter, timing(C27) converted from Volts to µVolts
29	**Jitter from Clock Phase Generator Supply**				
30	Clock Phase Generator Jitter		149.1	fs (rms)	Measured from Pnoise, Lab 5-2
31	Clock Phase Generator Noise (Input Referred)	=	44.18	µV (rms)	Calculated mathematically from input slew rate(2*PI*C3*C7) and jitter from power supply noise, timing(C30) converted from Volts to µVolts
32	**SINAD Transient "Noise"**				
33	Settling + DC Accuracy	=	143.57	µV (rms)	Extracted from above Transient SINAD (C10). Subtract the quantization noise (C15)
34	**CAPDAC Mismatch (1 Sigma)**				
35	Matching (1 Sigma)	=	98.85	µV (rms)	Extracted from the matching SINAD (C11). Subtract the quantization noise (C15) from the matching SINAD (C11)
36					
37	**Equivalent Input referred Thermal Noise**	=	610.40	µV (rms)	Calculated mathematically from the individual non-idealities contributions, yellow boxes: C21, C25, C28, C31, C33, C35
38	**Total Noise**	=	**737.97**	µV (rms)	Calculated mathematically from the quantization noise (C15), clock jitter (C17), and the non-idealities (C37)
39	**SINAD (with Noise)**	=	53.25	dB	Calculated mathematically from input level(C3) and equivalent input referred thermal noise (C37)
40	**ENOB**	=	8.68	bits	Convert SINAD to ENOB

Figure 232: ADC ENOB Calculation

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Now that the first pass through the design is complete, you can assess the status of the design and identify the areas that need more focus. Your analysis shows the ADC ENOB of 8.68 effective bits and, when compared to the ADC Transient noise ENOB (=9.072), your analysis considers more noise and can also show where the noise is generated from. Although the transient noise analysis might give more favorable results, designers will consider your break down of the noise far more useful than the single-value results that the transient noise analysis returns.

Looking at the individual contributions to the ADC's ENOB, you can see that the largest contributor to the ENOB is the ADC Transient SINAD: Settling + DC Accuracy contributor. The ADC Transient SINAD contribution is dependent on the signal level and the input frequency. So, for lower signal levels and lower operating frequency, the ADC will be able to perform close to the target specification. One option to explore is trading off ADC large-signal performance by decreasing the unit capacitor size against the CAPDAC contribution from mismatch.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

References

"Comparator Metastability Analysis", William Evans, Eric Naviasky, Hao Tang, Bryan Allison, and John Matsuzaki, Designers Guide

Support

Cadence Support Portal provides access to support resources, including an extensive knowledge base, access to software updates for Cadence products, and the ability to interact with Cadence Customer Support. Visit https://support.cadence.com.

Feedback

Email comments, questions, and suggestions to content_feedback@cadence.com.

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Appendix A

The following table explains the methodology of each expression used in the Run Plan testbench.

Number of Bits	**ADC's Number of Bits**
Full Scale Input	$\text{Peak to Peak Input Voltage to CAPDAC} \;=\; 2 \times (vrefp - vrefn)$
Full Scale Input (rms)	$Full\ Scale\ Input \div 2\sqrt{2}$
Quanta size	$Full\ Scale\ Input \div 2^{\ Number\ of\ Bits}$
Aperture Jitter	$Maximum\ allowed\ jitter\ of\ reference\ clock = 3ps$
Sampling Frequency	$User\ Specified = 100MHz$
tone	$Used\ to\ determine\ the\ input\ frequnecy\ of\ test\ bench.\ Must\ be\ less\ than\ the\ Nyquist\ frequnecy\ = 503$
numPoints	$Number\ of\ Points\ in\ FFT = 1024$
Temperature	$Nominal\ Temperature = 27°C$
Test Frequency	$(Sample\ Frequnecy\ \div numPoints)\ \times tone = 49.12MHz$
Matching CAPDAC SINAD	$Mean\ Value\ of\ Monte\ Carlo\ Yield = 61.54\ dB$
Quantization Noise	$Quanta\ size \div \sqrt{12}$
Aperture Jitter Noise	$Full\ Scale\ Input(rms) \times 2 \times \pi \times Test\ Frequency \times Aperature\ Jitter$
Total CAPDAC Capacitance	$2 \times 2 \times (16 + 8 + 4 + 2 + 1 + 1) \times (1.522\ pF)\ =\ 1.95pF$

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Total CAPDAC Gain (or Attenuation)	$Determined\ by\ Circuit\ Architechture = 1$
Total CAPDAC Gain Input Referred Noise	$2 \times \sqrt{\frac{1.38 \times 10^{-23} \times Temperature + 273}{Total\ CAPDAC\ Gain\ (or\ Attenuation) \times Total\ CAPDAC\ Capacitance}}$
CAPDAC Gain (attenuation from parasitic cap)	$Measured\ from\ CAPDAC\ Transient\ Simulation, not\ shown = 0.98$
Input Referred PSS Measured Comparator Noise	$Input\ Referred\ Noise\ calculated\ from\ PNOISE, Lab\ 3 - 2 = 563.7\mu V$
Accounting for CAPDAC attenuation	$\frac{Input\ Referred\ PSS\ Measured\ Comparator\ Noise}{CAPDAC\ Gain\ (attenuation\ from\ parasitic\ cap)}$
Clock Phase Generator Jitter	$Jitter\ calculated\ from\ PNOISE, Lab\ 5 - 1 = 101fs$
Clock Phase Generator Noise (Input Referred)	$Clock\ Phase\ Generator\ Jitter \times 2 \times \pi \times Full\ Scale\ Input \times Test\ Frequency$
Clock Phase Generator Jitter Noisy	$Jitter\ calculated\ from\ PNOISE, Lab\ 5 - 2 = 149.1fs$
Clock Phase Generator Noise (Input Referred) Noisy	$Clock\ Phase\ Generator\ Jitter\ Noisy \times 2 \times \pi \times Full\ Scale\ Input \times Test\ Frequency$

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

Settling and DC Accuracy	$Quantization\ Noise^2$
Settling and DC Accuracy pt1	$\left(\frac{Full\ Scale\ Input\ (rms)}{10^{\ (ADC\ Transient\ SINAD \div 20)}}\right)^2$
Settling and DC Accuracy pt2	$(Settling\ and\ DC\ Accuracy\ pt1) - (Settling\ and\ DC\ Accuracy)$
Actual Settling and DC Accuracy	$\sqrt{(Settling\ and\ DC\ Accuracy\ pt2)}$
Matching Sigma	$Quantization\ Noise^2$
Matching Sigma pt1	$\left(\frac{Full\ Scale\ Input\ (rms)}{10^{\ (Matching\ CAPDAC\ SINAD \div 20)}}\right)^2$
Matching Sigma pt2	$Matching\ Sigma\ pt1 - Matching\ Sigma$
Actual Matching Sigma	$\sqrt{Matching\ Sigma\ pt2}$
Equivalent Input Referred Thermal Noise	$\sqrt{\begin{array}{c}(Total\ CAPDAC\ Gain\ Input\ Referrred\ Noise)^2 \\ +(Accounting\ for\ CAPDAC\ attenuation)^2 \\ +\left(Clock\ Phase\ Generator\ Noise\ (Input\ Referred)\right)^2 \\ +(Clock\ Phase\ Generator\ Noise\ (Input\ Referred)\ Noisy)^2 \\ +(Actual\ Settling\ and\ DC\ Accuracy)^2 + (Actual\ Matching\ Sigma)^2\end{array}}$
Total Noise	$\sqrt{\begin{array}{c}(Quantization\ Noise)^2 + (Aperature\ Jitter\ Noise)^2 \\ +(Total\ CAPDAC\ Gain\ Input\ Referrred\ Noise)^2 \\ +(Accounting\ for\ CAPDAC\ attenuation)^2 \\ +\left(Clock\ Phase\ Generator\ Noise\ (Input\ Referred)\right)^2 \\ +(Clock\ Phase\ Generator\ Noise\ (Input\ Referred)\ Noisy)^2 \\ +(Actual\ Settling\ and\ DC\ Accuracy)^2 + (Actual\ Matching\ Sigma)^2\end{array}}$

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.

SINAD (with Noise)	$\frac{Full\ Scale\ Input\ (rms)}{Total\ Noise}$
Actual SINAD (with Noise)	$20 \times log_{10}(SINAD\ (with\ Noise))$
Ideal DAC Full Scale Output	$Full\ Scale\ Input$
Measured DAC Output	$Peak\ to\ peak\ Output\ Voltage\ of\ CAPDAC = 2 \times (I0/sump - I0/sumn)$
ENOB pt1	Actual SINAD (with Noise)
ENOB pt2	$20 \times log_{10}\left(\frac{Ideal\ DAC\ Full\ Scale\ Output}{Measured\ DAC\ Output}\right)$
Actual ENOB	$\frac{(ENOB\ pt1) - 1.76 + (ENOB\ pt2)}{6.02}$

© 2023 Cadence Design Systems, Inc. All rights reserved worldwide.