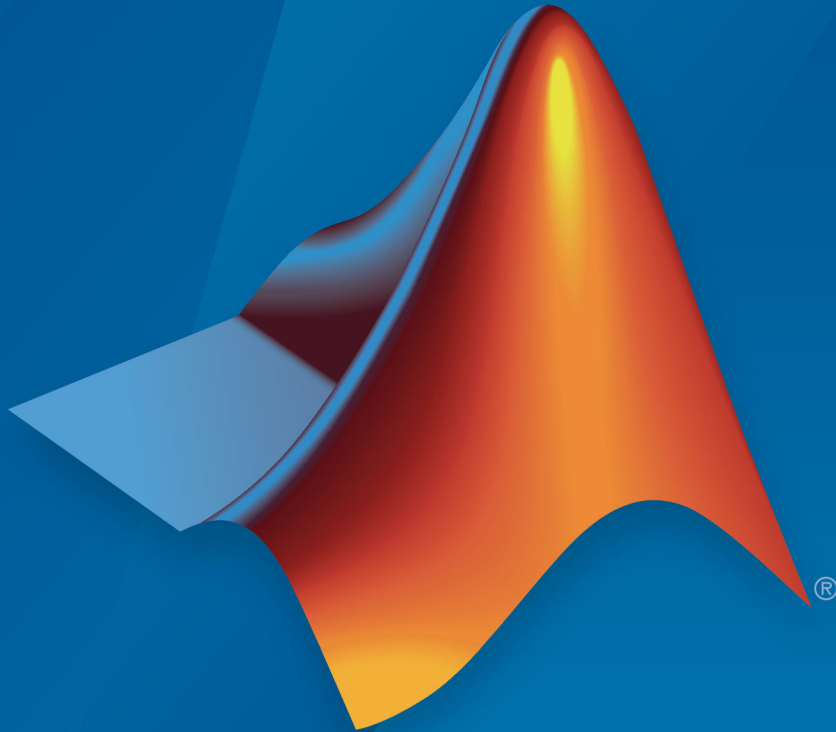


# Mixed-Signal Blockset™

## Getting Started Guide



MATLAB® & SIMULINK®

R2019b



# How to Contact MathWorks



Latest news: [www.mathworks.com](http://www.mathworks.com)  
Sales and services: [www.mathworks.com/sales\\_and\\_services](http://www.mathworks.com/sales_and_services)  
User community: [www.mathworks.com/matlabcentral](http://www.mathworks.com/matlabcentral)  
Technical support: [www.mathworks.com/support/contact\\_us](http://www.mathworks.com/support/contact_us)



Phone: 508-647-7000



The MathWorks, Inc.  
1 Apple Hill Drive  
Natick, MA 01760-2098

*Mixed-Signal Blockset™ Getting Started Guide*

© COPYRIGHT 2019 by MathWorks, Inc.

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from The MathWorks, Inc.

FEDERAL ACQUISITION: This provision applies to all acquisitions of the Program and Documentation by, for, or through the federal government of the United States. By accepting delivery of the Program or Documentation, the government hereby agrees that this software or documentation qualifies as commercial computer software or commercial computer software documentation as such terms are used or defined in FAR 12.212, DFARS Part 227.72, and DFARS 252.227-7014. Accordingly, the terms and conditions of this Agreement and only those rights specified in this Agreement, shall pertain to and govern the use, modification, reproduction, release, performance, display, and disclosure of the Program and Documentation by the federal government (or other entity acquiring for or through the federal government) and shall supersede any conflicting contractual terms or conditions. If this License fails to meet the government's needs or is inconsistent in any respect with federal procurement law, the government agrees to return the Program and Documentation, unused, to The MathWorks, Inc.

## Trademarks

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [www.mathworks.com/trademarks](http://www.mathworks.com/trademarks) for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

## Patents

MathWorks products are protected by one or more U.S. patents. Please see [www.mathworks.com/patents](http://www.mathworks.com/patents) for more information.

## Revision History

March 2019	Online only	New for Version 1.0 (Release 2019a)
September 2019	Online only	Revised for Version 1.1 (Release 2019b)

## **Introduction to Mixed-Signal Blockset**

**1**

**Mixed-Signal Blockset Product Description . . . . . 1-2**

## **Introduction to PLL**

**2**

**Design and Evaluate Simple PLL Model . . . . . 2-2**

**Phase Noise Analysis in VCO . . . . . 2-8**

## **Introduction to ADC**

**3**

**Design and Evaluate a SAR ADC . . . . . 3-2**



# Introduction to Mixed-Signal Blockset

---

## **Mixed-Signal Blockset Product Description**

### **Design and simulate analog and mixed-signal systems**

Mixed-Signal Blockset provides models of components and impairments, analysis tools, and test benches for designing and verifying mixed-signal integrated circuits (ICs).

You can model PLLs, data converters, and other systems at different levels of abstraction and explore a range of IC architectures. You can customize models to include impairments such as noise, nonlinearity, and quantization effects, and refine the system description using a top-down methodology.

Using the test benches provided, you can verify system performance and improve modeling fidelity by fitting measurement characteristics or circuit-level simulation results. Rapid system-level simulation using variable-step Simulink® solvers lets you debug the implementation and identify design flaws before simulating the IC at the transistor level.

With Mixed-Signal Blockset you can simulate mixed-signal components together with complex DSP algorithms and control logic. As a result, both analog and digital design teams can work from the same executable specification.

# Introduction to PLL

---

- “Design and Evaluate Simple PLL Model” on page 2-2
- “Phase Noise Analysis in VCO” on page 2-8

# Design and Evaluate Simple PLL Model

This example shows how to design a simple phase-locked loop (PLL) using a reference architecture and validate it using PLL Testbench.

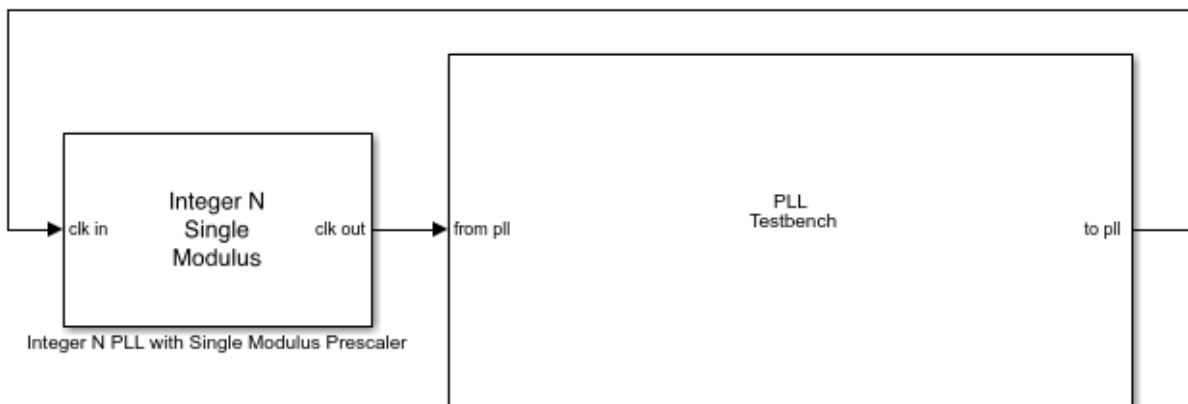
A PLL is a frequency synthesizer system that produces an output signal whose phase depends on the phase of its input signal. In the simplest form, a PLL consists of a phase/frequency detector (PFD), charge pump, loop filter, voltage controlled oscillator (VCO), and a clock divider in a feedback loop. The PFD and charge pump together produce an error signal proportional to the phase difference of its two input signals. The loop filter removes the higher-frequency components of the error signal, which drives the VCO. The output of the VCO is fed through a clock divider to the input of the PFD, producing a negative feedback loop.

Mixed-Signal Blockset™ provides reference architectures to design a simple PLL model and testbenches to verify that the designed model meets the design specifications.

### Set Up PLL Testbench Model

Open the model `simplePLL` attached to this example as a supporting file. The model consists of an Integer N PLL with Single Modulus Prescaler block and a PLL Testbench block.

```
open_system('simplePLL.slx')
```





## PLL Specifications and Impairment

Use the data sheet of Skyworks SKY73134-11 to design the PLL system to lock at 2.8 GHz.

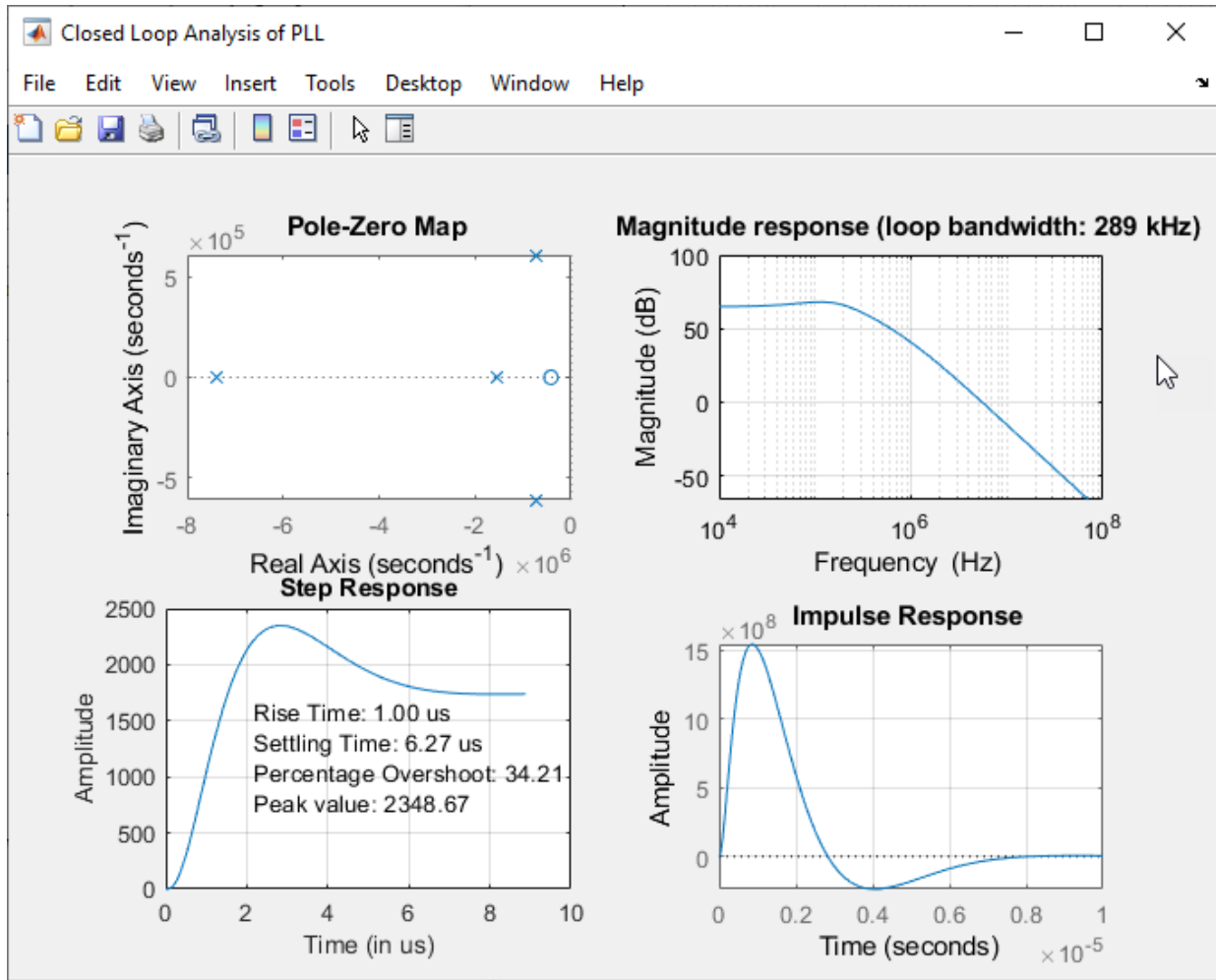
Double-click the Integer N PLL with Single Modulus Prescaler block to open the Block Parameters dialog box and verify these settings: \* Check that the impairments are disabled in the **PFD** and **Charge pump** tabs. \* In the **Charge pump** tab, the **Output current** is set to 2.7 mA. The **Deadband compensation** and **Input threshold** parameters are kept at default values.

- In the **VCO** tab, the **VCO Sensitivity** is set to 20 MHz/V. The **Free running frequency** is slightly lower than the target lock frequency and is set to 2.78 GHz. The **Phase noise frequency offset** is set to [100e3 1e6 3e6 10e6] Hz and the **Phase noise level (dBc/Hz)** is set to [-108 -134 -145 -154] dBc/Hz.
- Considering the reference input frequency to the PLL is 1.6 MHz, the **Clock divider value** and the **Min clock divider value** in the **Prescaler** tab is set to  $\frac{2.8e9}{1.6e6} = 1750$ .
- In the **Loop Filter** tab, the **Loop bandwidth** is set to 160 kHz, 1/10th of the reference input frequency. The phase margin is kept at default 45 degrees. **Filter component values** are calculated automatically.
- In the **Analysis** tab, both **Open Loop Analysis** and **Closed Loop Analysis** plots are selected.

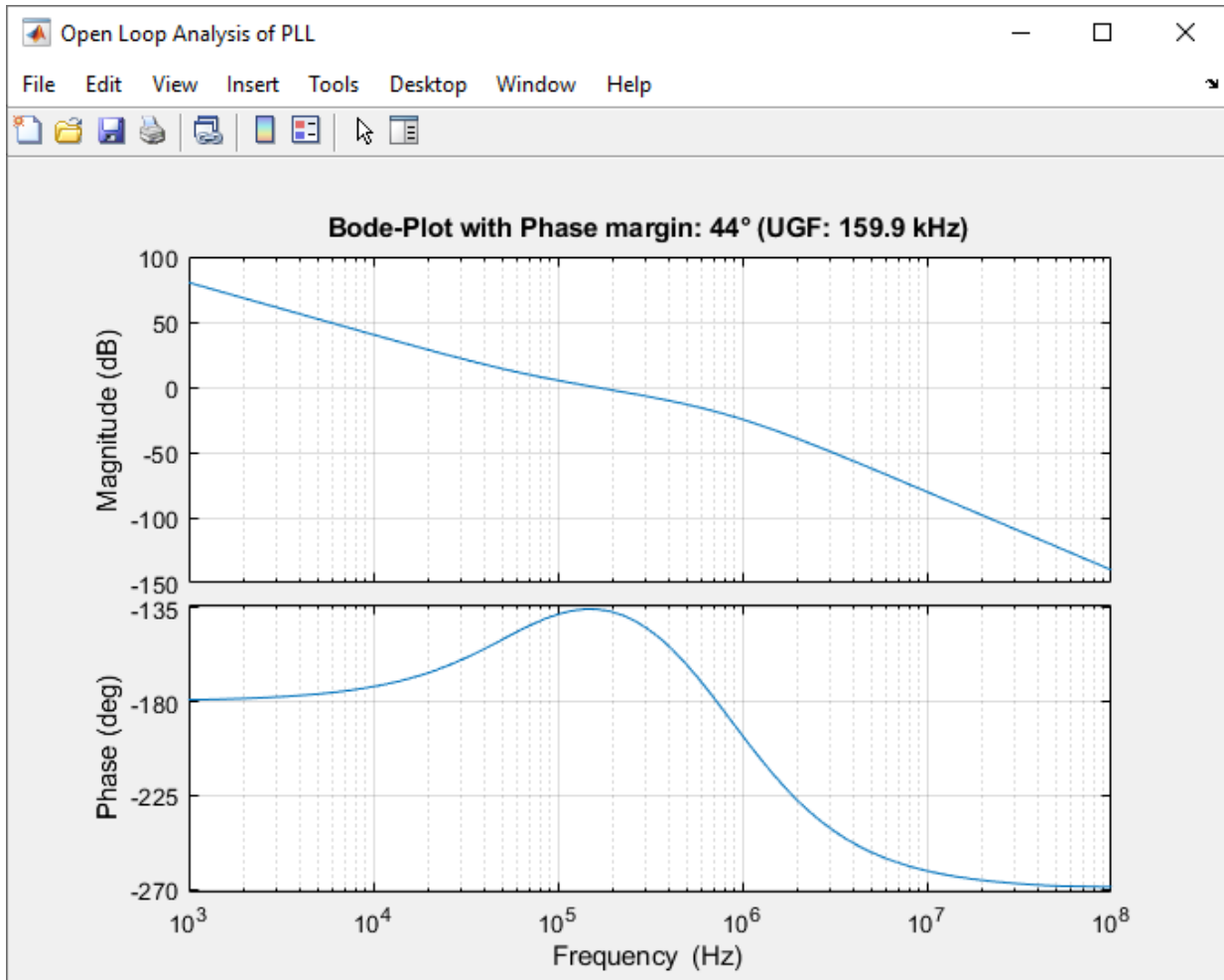
## Plot Presimulation PLL Loop Dynamics

Click the **Plot Loop Dynamics** button to view the presimulation results and assess the stability of the system.

The closed loop analysis consists of the Pole-Zero Map, Magnitude Response, Step Response, and Impulse Response. The loop bandwidth of the system is 289 kHz. The system is stable.



The open loop analysis consists of Bode plots of the PLL system. The phase margin is 44 degrees.



### Modify PLL Testbench for Phase Noise Measurement

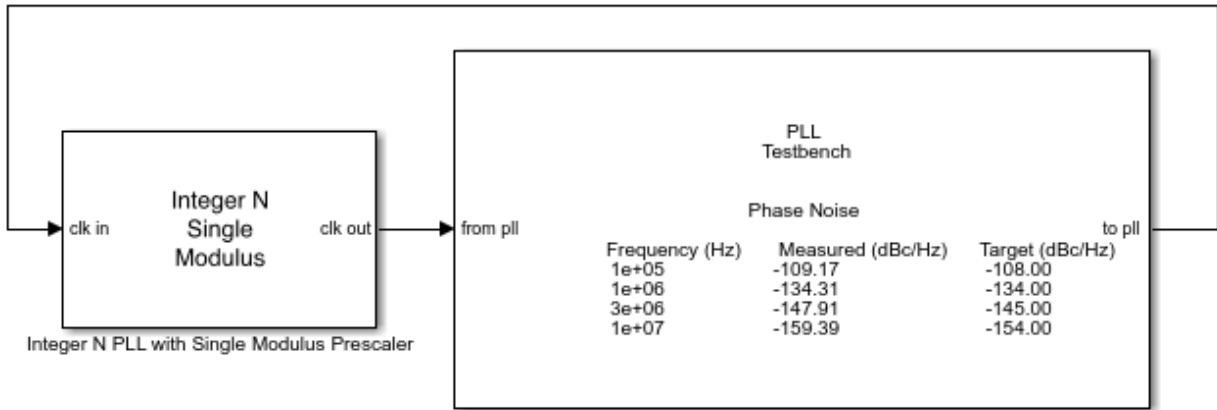
Double-click the PLL Testbench to open the Block Parameters dialog box and verify these settings: \* In the **Stimulus** tab, the input signal to the PLL is defined as a square wave of 1.6 MHz.

- In the **Setup** tab, check that the **Phase noise** measurement option is selected. **Frequency of operation** and **Lock time** measurement options are deselected. Set the **Resolution bandwidth** to 50 kHz and **No. of spectral averages** to 4.

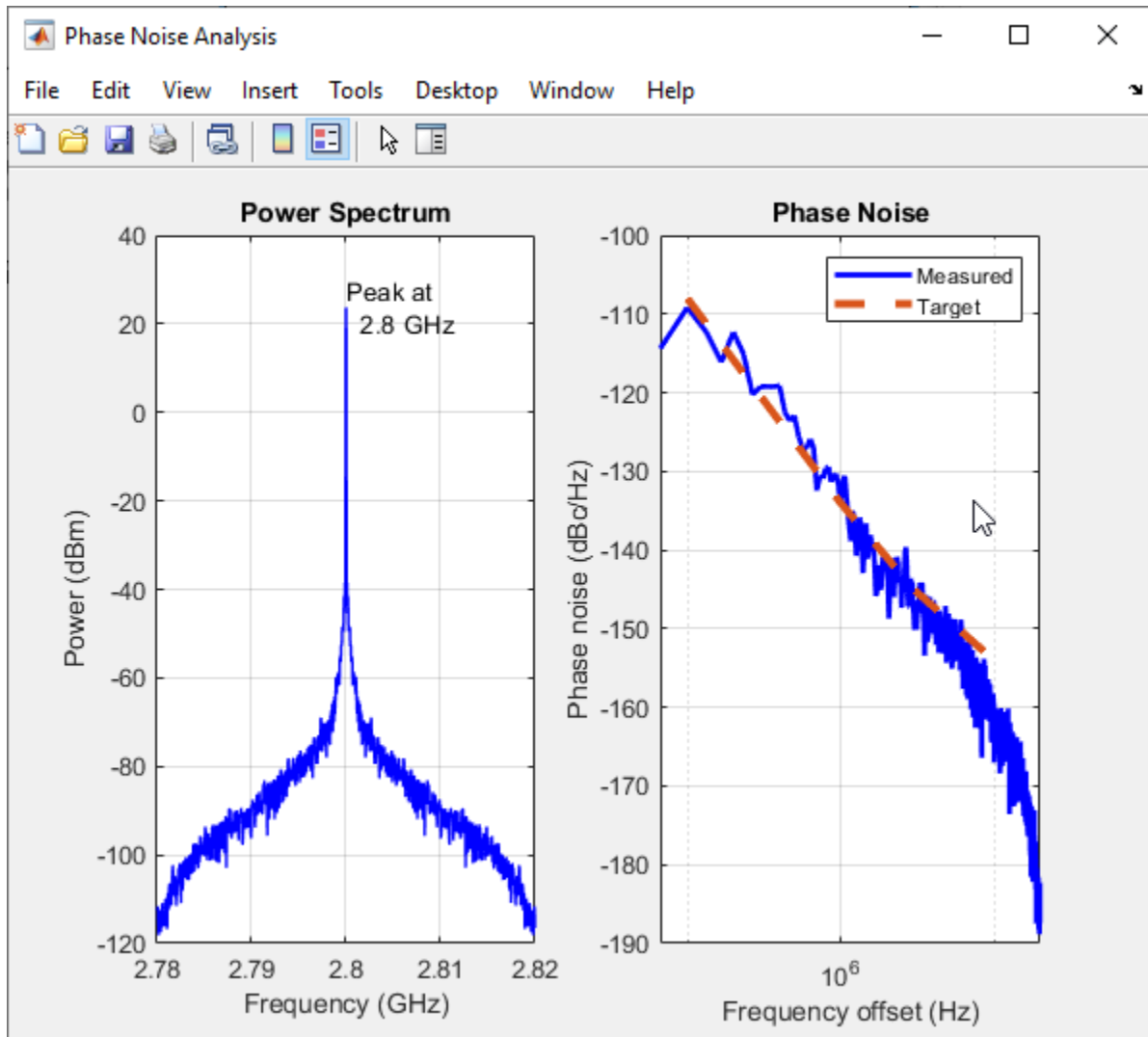
- In the **Target Metrics** tab, set the **Phase noise (dBc/Hz)** to  $[-108 \ -134 \ -145 \ -154]$ , the same as the PLL phase noise profile.

**Plot PLL Phase Noise Profile**

Run the simulation for  $3.6 \times 10^{-4}$  s. The simulation results are displayed on the icon of the PLL Testbench. The measured phase noise levels at specific frequency offsets are consistent with their target values.



Double-click the PLL Testbench block to open the Block Parameters dialog box. Click the **Plot phase noise profile** button. The PLL operating frequency is 2.8 GHz, and the measured phase noise profile matches the target profile.



### Reference

1. Skyworks SKY73134-11

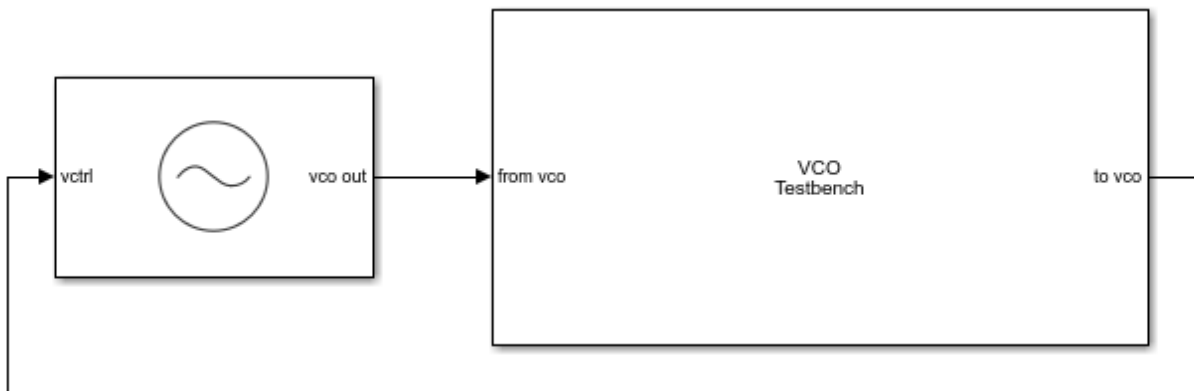
## Phase Noise Analysis in VCO

This example shows how to measure and analyze the effect of phase noise in a voltage controlled oscillator (VCO). Using a VCO block and VCO testbench, this example defines the typical phase noise levels from datasheet specifications and validates the VCO.

### Set Up VCO Testbench Model

Open the model `vcoPhaseNoiseAnalysis`. The model consists of a VCO block and a VCO Testbench.

```
open_system('vcoPhaseNoiseAnalysis.slx')
```



### VCO Specifications and Phase Noise Impairments

Double click the VCO block to open the **Block Parameters** dialog box. In the **Configuration** tab, the **Voltage Sensitivity (Hz/V)** is set to  $125e6$ . In the **Impairment** tab, check that the **Add phase noise** option is enabled. **Phase noise frequency offset (Hz)** and **Phase noise level (dBc/Hz)** parameters represent a typical phase noise profile in a VCO.

### Modify VCO Testbench According to VCO Specifications

Double click the VCO Testbench block to open the **Block Parameters** dialog box. In the **Stimulus** tab, **Control voltage (V)** is set to 4.

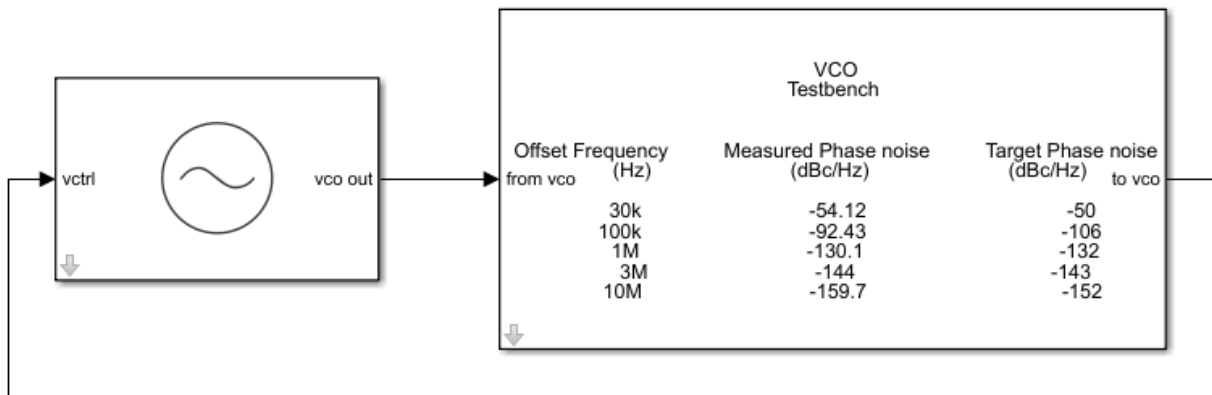
Since the VCO has a sensitivity of 1 MHz/V and the free running frequency of 2.5 GHz, the operating frequency of the VCO is 3 GHz. In the **Setup** tab, the **Sampling frequency**

**(Hz)** is set to  $24\text{e}9$ , which is eight times the target operating frequency. Also check that in the **Target Metric** tab, the **Phase noise frequency offset (Hz)** and **Phase noise level (dBc/Hz)** parameters match the values set in the VCO block.

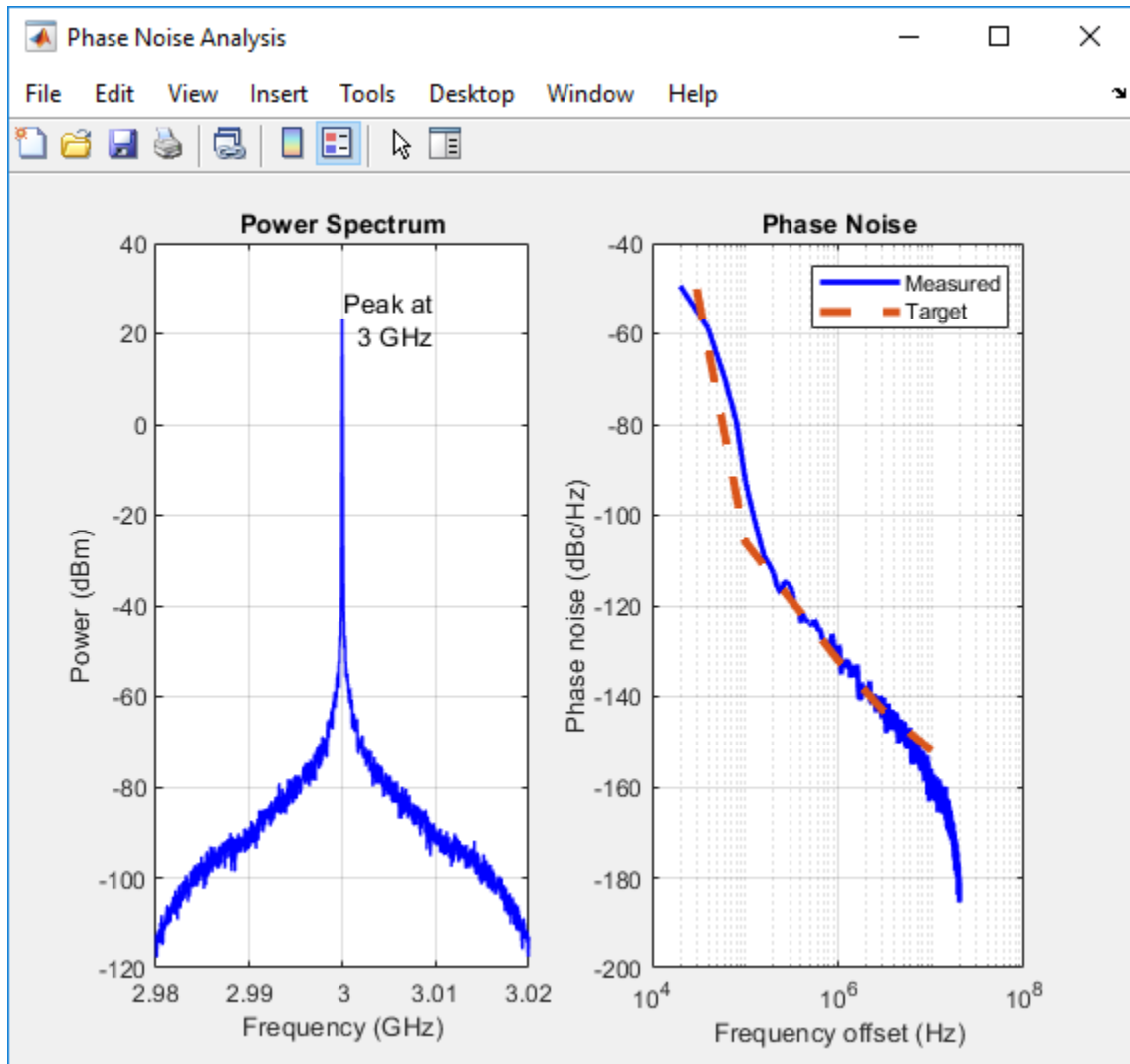
### Plot Phase Noise Profile

Run the simulation for 1.2 ms, according to the **Recommended min. simulation stop time (s)** in the **Block Parameters** dialog box of VCO Testbench.

Once the simulation is complete, the phase noise profile is displayed on the icon of the VCO Testbench. The measured phase noise is comparable to target phase noise.



In the **Block Parameters** dialog box of VCO, click the **Plot measurement** button to plot the phase noise profile of the VCO. Notice that the VCO operating frequency is 3 GHz, and that the measured and targeted phase noise profiles match.





# Introduction to ADC

---

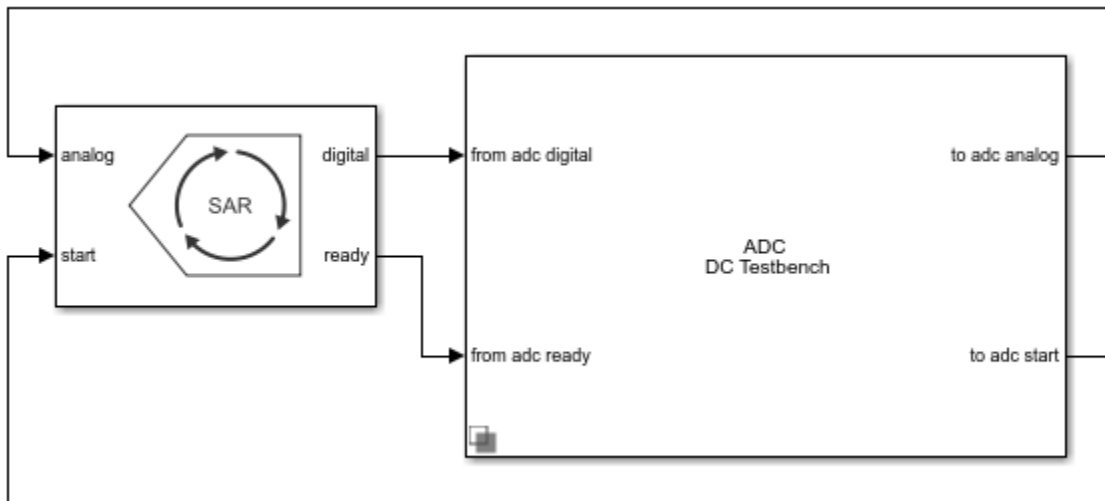
## Design and Evaluate a SAR ADC

This example shows how to design a SAR ADC using reference architecture and validate the ADC using ADC Testbench.

### Set UP SAR ADC Testbench Model

Open the model SAR\_ADC attached to this example as a supporting file. The model consists of a SAR ADC block and an ADC Testbench.

```
open_system('SAR_ADC.slx')
```



### ADC Specifications and Impairments

Double click the SAR ADC block to open the Block Parameters dialog box. The **Number of bits** is set to 8, and the **SAR Frequency** is  $2e7$  Hz. Check that in the **Impairments** tab, impairments are enabled.

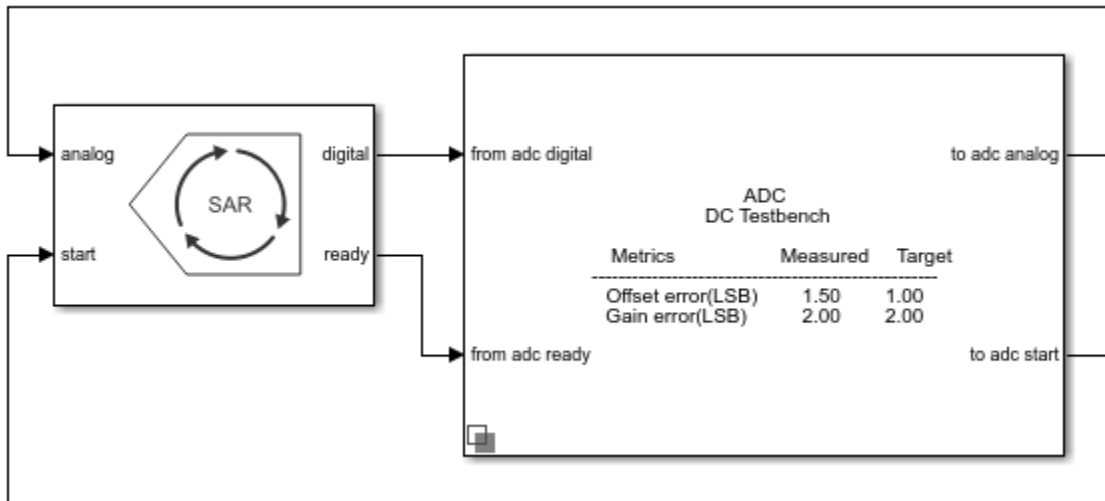
### Modify ADC Testbench According to ADC Specification

Double click the ADC Testbench block to open the Block Parameters dialog box. The **Measurement** option is selected as DC. In the **Setup** tab, click the **Autofil setup parameters** button to automatically propagate the ADC parameters to the testbench. In

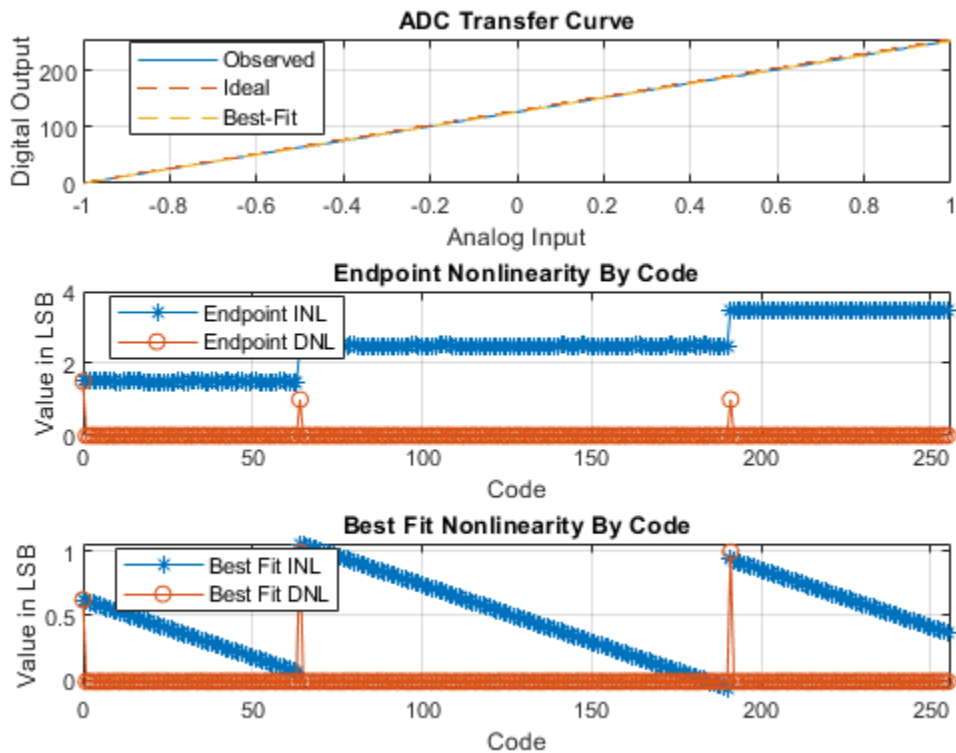
the **Target Metric** tab, click the **Autofill target metric** button to automatically propagate the ADC target metrics to the testbench. Save the changes.

### Plot DC Analysis Results

Run the simulation for 0.00512 s. The measured and target values of offset error and gain error are displayed on the icon of the ADC Testbench block.



Double click the ADC Testbench block to open the Block Parameters dialog box. Click the **Plot DC analysis results** button to view the ADC transfer curve, endpoint nonlinearity and best fit nonlinearity.



### Perform AC Analysis

Double click the ADC Testbench block to open the Block Parameters dialog box. Set the \*Measurement option to AC and save the change.

Run the simulation for 0.009 s. The conversion delay, SINAD, SFDR, SNR, ENOB and Noise floor are displayed on the icon of the ADC Testbench.

