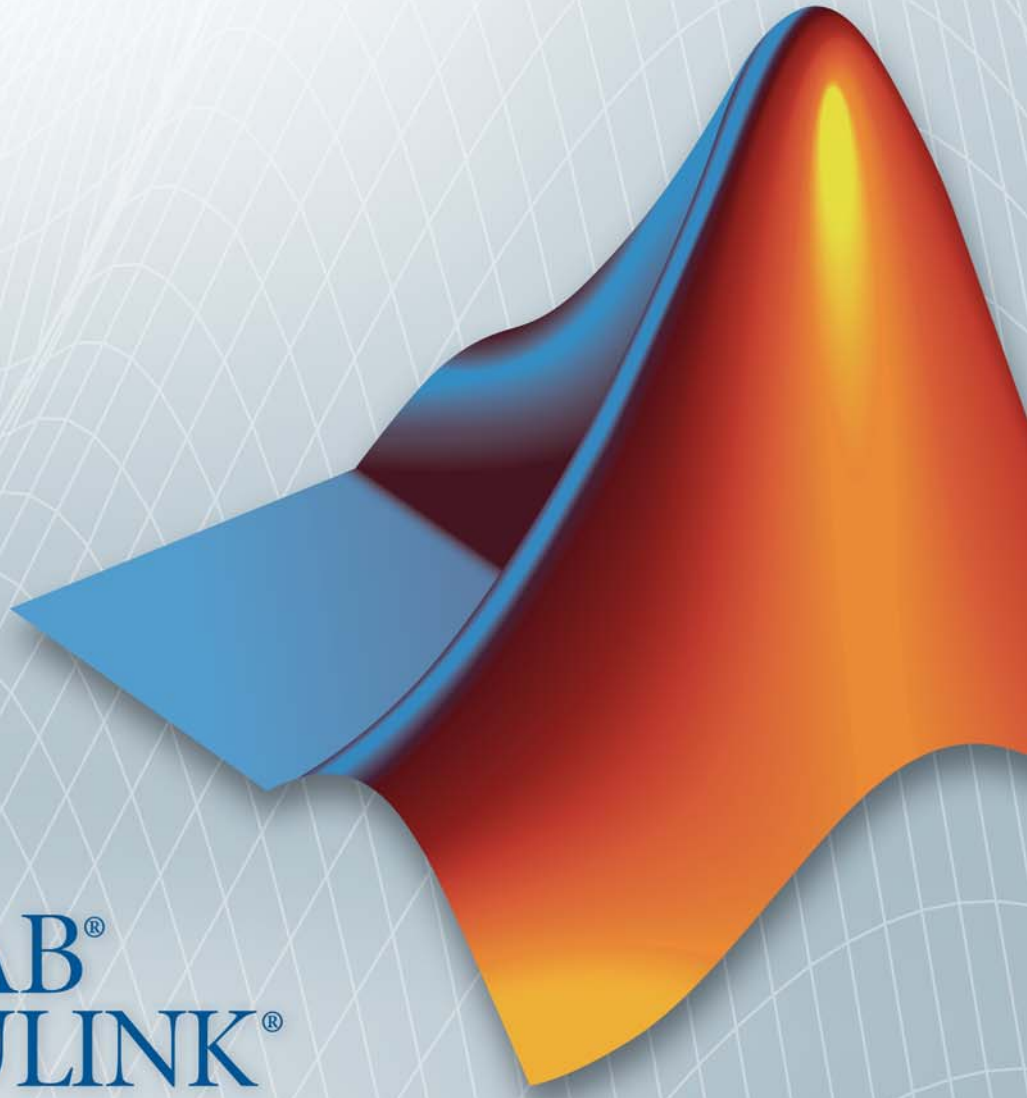
SimRF™ 3
User's Guide



MATLAB®
&SIMULINK®

MathWorks®

How to Contact MathWorks

`www.mathworks.com`	Web
`comp.soft-sys.matlab`	Newsgroup
`www.mathworks.com/contact_TS.html`	Technical Support

`suggest@mathworks.com`	Product enhancement suggestions
`bugs@mathworks.com`	Bug reports
`doc@mathworks.com`	Documentation error reports
`service@mathworks.com`	Order status, license renewals, passcodes
`info@mathworks.com`	Sales, pricing, and general information

508-647-7000 (Phone)

508-647-7001 (Fax)

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

For contact information about worldwide offices, see the MathWorks Web site.

SimRF™ User's Guide

© COPYRIGHT 2010–2011 by The 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` 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` for more information.

Revision History

September 2010	Online only	New for Version 3.0 (Release 2010b)
April 2011	Online only	New for Version 3.0.2 (Release 2011a)

Contents

1 Sensitivity

Modeling System Noise Figure 1-2
Create a Low-IF Receiver Model 1-2
Simulate a Thermal Noise Floor 1-6
Computing System Noise Figure 1-7

Designing a Receiver with an ADC 1-9
Overcome Quantization Error of an ADC 1-9
Measuring the Quantization Noise Floor 1-12
Improving Receiver-ADC Performance 1-13

2 Interference

Carrier to Interference Performance of a Weaver Receiver 2-2
Create a Model with RF Interference 2-2
Modeling RF and Blocker Sources 2-7
Simulating LO Phase Offset 2-7

Modeling LO Phase Noise 2-8
Create a Model with Phase Noise 2-8
Shaping the LO Noise Skirt 2-11

3 Intermodulation Distortion

Modeling a Direct Conversion Receiver 3-2
Create a Direct Conversion Receiver Model 3-2

Modeling IMD in System-Level Components **3-6**
Examining DC Impairments **3-7**

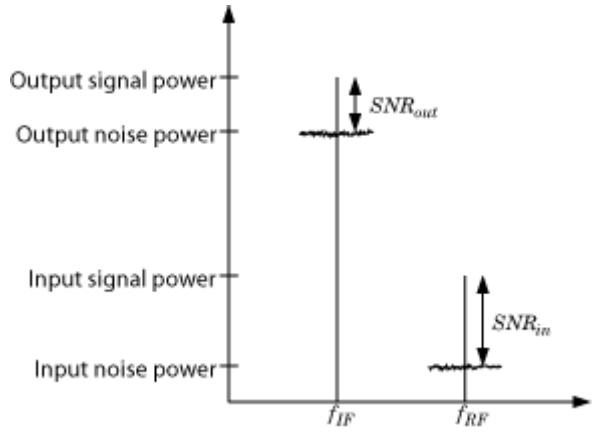
1

Sensitivity

- "Modeling System Noise Figure" on page 1-2
- "Designing a Receiver with an ADC" on page 1-9

Modeling System Noise Figure

RF receivers amplify signals and translate them to lower frequencies. The receiver itself introduces noise that degrades the received signal. The signal-to-noise ratio (SNR) at the receiver output ultimately determines the usability of the receiver.



The preceding figure illustrates the effect of the receiver on the signal. The receiver amplifies a low-power RF signal at the carrier f_{RF} with a high SNR and downconverts the signal to f_{IF}. The noise figure (NF) of the system determines the difference between the SNR at the output and the SNR at the input:

$$SNR_{out} = SNR_{in} - NF_{sys}$$

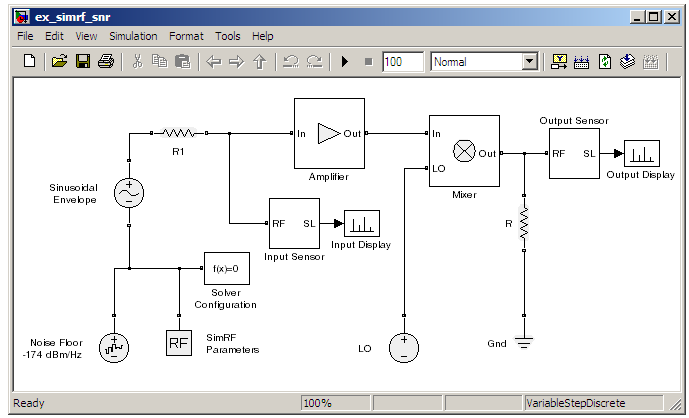
where the difference is calculated in decibels. Excessive noise figure in the system causes the noise to overwhelm the signal, making the signal unrecoverable.

Create a Low-IF Receiver Model

The model

```
ex_simrf_snr
```

simulates a simplified IF receiver architecture. A Sinusoid block and a Noise block model a two-tone input centered at f_{RF} and low-level thermal noise. The RF system amplifies the signal and mixes it with the local oscillator f_{LO} down to an intermediate frequency f_{IF}. A voltage sensor recovers the signal at the IF.



The amplifier contributes 40 dB of gain and a 15-dB noise figure, and the mixer contributes 0 dB of gain and a 20-dB noise figure, which are values characteristic of a relatively noisy, high-gain receiver. The two-tone input has a specified level of .1 µV. A 1-V level in the local oscillator ensures consistency with the formulation of the conversion gain of the mixer.

To run the model:

1. Open the model by clicking the link or by typing the model name at the Command Window prompt.
2. Select **Simulation > Start**.

Models that contain SimRF™ Amplifier, Mixer, or S-Parameter blocks generate files in the current MATLAB® folder at run time. However, you can configure the output location for these files by specifying a cache folder in the Simulink Preferences dialog box. To specify a cache folder, open the Simulink Preferences dialog box (**File > Preferences**) and specify a location on your

file system for the **Simulink cache folder** parameter. For more information about the Simulink® interface, see Simulink Preferences Window.

Set Up the SimRF Environment

The model simulates according to the following settings:

- In the SimRF Parameters Block Parameters dialog box, the **Carrier Frequencies** parameter specifies the carriers of the SimRF environment:
  - f_{RF}, the carrier of the desired signal, equal to 2 GHz.
  - f_{LO}, the frequency of the LO in the first mixing stage, equal to 1.9999 GHz.
  - f_{IF}, the intermediate frequency, equal to $f_{RF} - f_{LO}$, or 100 kHz.

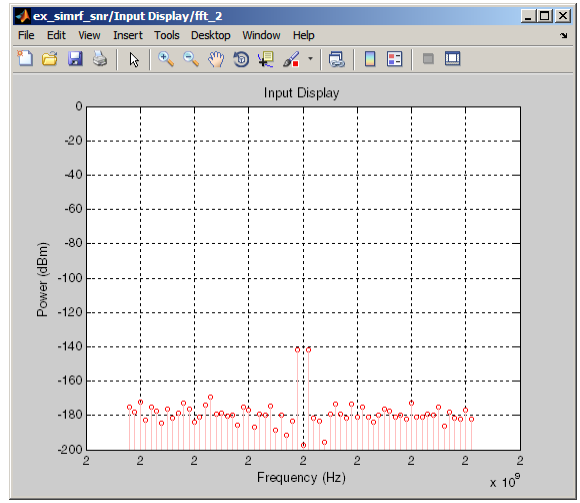
  This example uses the variable `car_env`, defined in the initialization function, to specify the carriers.
- In the Solver Configuration Block Parameters dialog box, the **Use local solver** box is selected. This setting causes the SimRF environment to simulate with a local solver with the following settings:
  - The **Solver type** is set to `Trapezoidal rule`.
  - The **Sample time** parameter is set to `1/64`.

  Since the model uses a local solver, the global solver settings do not affect the simulation within the SimRF environment. For more information on global and local solvers, see Choosing Simulink and Simscape™ Solvers.

View Simulation Output

The model uses subsystems with a MATLAB® Coder™ implementation of a fast Fourier transform (FFT) to generate two plots. The FFT uses 64 bins, so for a sampling frequency of 64 Hz, the bandwidth of each bin is 1 Hz. Subsequently, the power levels shown in the figures also represent the power spectral density (PSD) of the signals in dBm/Hz.

- The Input Display plot shows the power spectrum of the signal and noise at the input of the receiver.



The measured power of each tone is consistent with the expected power level of a .1-µV two-tone envelope:

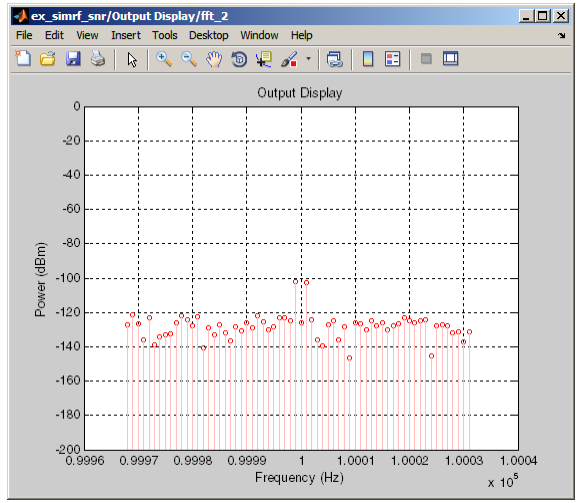
$$P_{in} = 10\log_{10}\left(\frac{V^2}{2R}\right) + 30$$

$$= 10\log_{10}\left(\frac{\left(\frac{1}{2}\cdot\frac{10^{-7}}{2}\right)^2}{2\cdot 50}\right) + 30 = -142 \text{ dBm}$$

A factor of 1/2 is due to voltage division across source and load resistors, and another factor of 1/2 is due to envelope scaling. See the demo Two-Tone Envelope Analysis Using Real Signals for more discussion on scaling envelope signals for power calculation.

The measured noise floor at -177 dBm/Hz is reduced by 3 dB from the specified -174 dBm/Hz noise floor. The difference is due to power transfer from the source to the input of the amplifier.

- The Output Display plot shows the power spectrum of the signal and noise at the output of the receiver.



The measured PSD of -102 dBm/Hz for each tone is consistent with the 40-dB combined gain of the amplifier and mixer. The noise PSD in the figure is shown to be approximately 50 dB higher at the output, due to the gain and noise figure of the system.

If you have DSP System Toolbox™ software installed, you can replace the MATLAB Coder subsystems with Vector Scope or Spectrum Scope blocks.

Simulate a Thermal Noise Floor

The model uses a block to model a thermal noise floor according to the equation

$$P_{noise} = 4k_B T R_s \Delta f$$

where:

- k_B is Boltzmann's constant, equal to 1.38065×10^{-23} J/K.
- T is the noise temperature, specified as 293 K in this example.
- R_s is the noise source impedance, specified as 50 Ω in this example to agree with the resistance value of the Resistor block labeled `R1`.
- Δf is the noise bandwidth.

To model the noise floor, the model uses a combination of settings in two different blocks.

- In the Noise block dialog box, the **Noise Power Spectral Density (Watts/Hz)** parameter is set to the variable `noise_level`, which is calculated in the initialization function to equal the value of the expression

  $P_{noise} / \Delta f = 4k_B T R_s$.
- In the SimRF Parameters block dialog box:
  - The **Simulate Noise** box is selected. When this box is cleared, the model simulates without noise.
  - The **Noise bandwidth type** parameter is set to `Absolute bandwidth`.
  - The **Noise bandwidth** is `1/sample_time`, where `sample_time` is the discrete sample time used by the local solver. This setting matches the **Sample time** parameter in the Solver Configuration dialog box. This value specifies Δf in the preceding expressions.

Computing System Noise Figure

To model RF noise from component noise figures:

1. Select **Simulate noise** in the SimRF Parameters block dialog box, if it is not already selected.
2. Specify a value for the **Noise figure (dB)** parameter of an Amplifier and Mixer blocks.

The noise figures are not strictly additive. The amplifier contributes more noise to the system than the mixer because it appears first in the cascade. To calculate the total noise figure of the RF system with n stages, use the Friis equation:

$$F_{sys} = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_1 G_2} + \ldots + \frac{F_n}{G_1 G_2 \ldots G_{n-1}}$$

where F_i and G_i are the noise factor and gain of the ith stage, and $NF_i = 10\log_{10}(F_i)$.

In this example, the noise figure of the amplifier is 10 dB, and the noise figure of the mixer is 15 dB, so the noise figure of the system is:

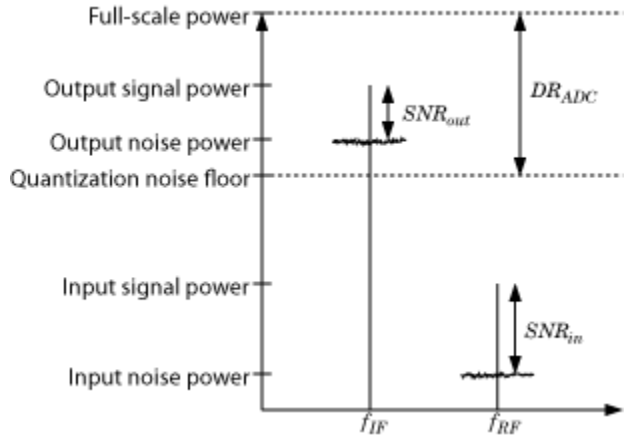
$$10\log_{10}\left(10^{10/10}+\frac{10^{15/10}-1}{10000}\right)=10.0\text{ dB}$$

The Friis equation shows that although the mixer has a higher noise figure, the amplifier contributes more noise to the system.

For more information on RF system noise figure, see the demo Impact of RF Receiver on Communcations System Performance.

Designing a Receiver with an ADC

Most RF receivers in modern communications or radar systems feed signals to an analog-to-digital converter (ADC). Due to their finite resolution, ADCs introduce quantization error into the system. The resolution of the ADC is determined by the number of bits and the full-scale (FS) range of the ADC.



The preceding figure illustrates an RF signal that falls within the dynamic range (DR) of an ADC. The input signal and noise at the carrier f_{RF} has high signal-to-noise ratio (SNR). The received signal at f_{IF} has reduced SNR due to system noise figure. However, if the quantization error is near or above the receiver noise, system performance degrades.

To ensure that the ADC contributes no more than 0.1 dB of noise to the signal at f_{IF}, the quantization noise floor must be 16 dB lower than the receiver noise. This condition can be met by:

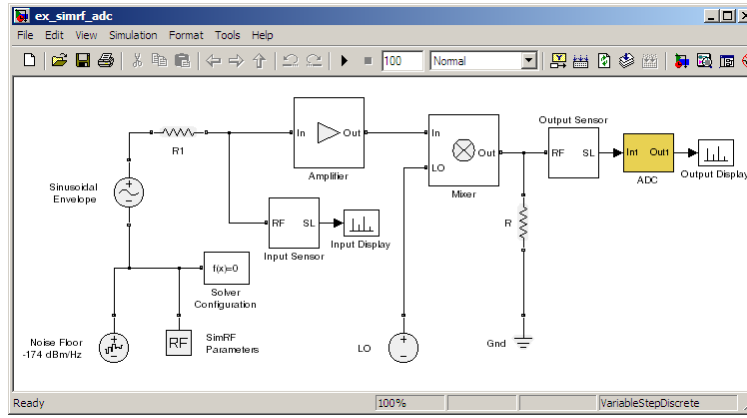
- Reducing the full-scale range or increasing the resolution of the ADC, which lowers the quantization noise floor.
- Increasing the gain of the RF receiver, which raises the receiver noise floor.

Overcome Quantization Error of an ADC

The model

```
ex_simrf_adc
```

simulates a low-IF receiver with an ADC. This model is based on the model `ex_simrf_snr` described in the section "Create a Low-IF Receiver Model" on page 1-2. At the output of the RF system, the ADC subsystem models an ADC with an FS range of `sqrt(100e-3)` V and a resolution of 16 bits.



The power a voltage signal at the full-scale range of the ADC is

$$10\log_{10}\sqrt{100^{-3}} + 30 = 0 \text{ dBm}$$

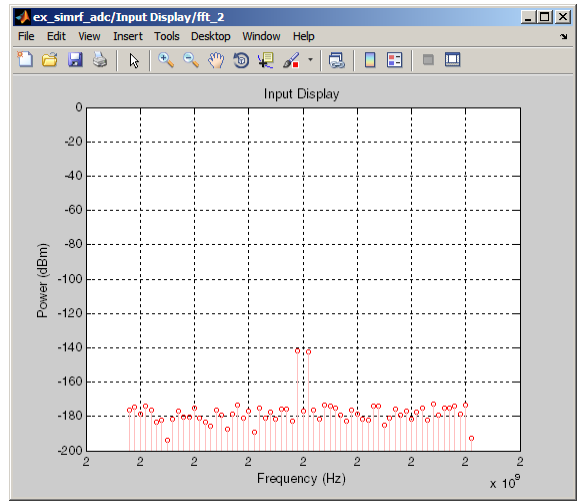
To run the model:

1. Open the model by clicking the link or by typing the model name at the Command Window prompt.
2. Select **Simulation > Start**.

View Simulation Output

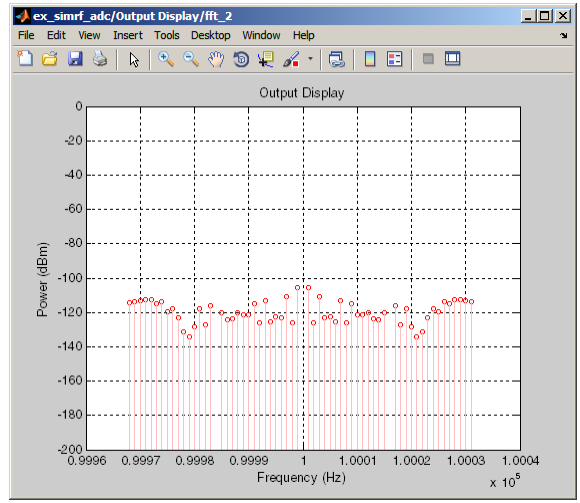
The model uses subsystems with a MATLAB Coder implementation of a fast Fourier transform (FFT) to generate two plots. The FFT uses 64 bins, so for a sampling frequency of 64 Hz, the bandwidth of each bin is 1 Hz. Subsequently, the power levels shown in the figures also represent the power spectral density (PSD) of the signals in dBm/Hz.

- The Input Display plot shows the power spectrum of the two-tone signal and noise at the input of the receiver-ADC system.



  The measured power of each tone of -142 dBm is consistent with the expected power level of a .1-µV signal. The power level of the noise is consistent with a -174 dBm/Hz noise floor.

- The Output Display plot shows power spectrum of the output signal.



The quantization error exceeds the receiver noise.

If you have DSP System Toolbox software installed, you can replace the MATLAB Coder subsystems with Vector Scope or Spectrum Scope blocks.

Measuring the Quantization Noise Floor

To calculate the quantization noise floor (QNF) of the ADC, subtract the dynamic range from the full-scale power, which is 0 dBm. To calculate the dynamic range PSD for the ADC, use the equation:

$$DR_{ADC} = 6.02 \cdot N_{bits} + 10\log_{10}(\Delta f) + 1.76 = 116.1 \text{ dBm/Hz}$$

where

- N_{bits} is the resolution. The ADC in this example uses 16 bits.
- Δf is the bandwidth of the FFT, which is 64 in this example. Oversampling in an ADC yields lower quantization noise.
- The value 1.76 is a correction factor for a pure sinusoidal input.

Therefore, the quantization noise floor is -116 dBm/Hz, in agreement with the measured output levels.

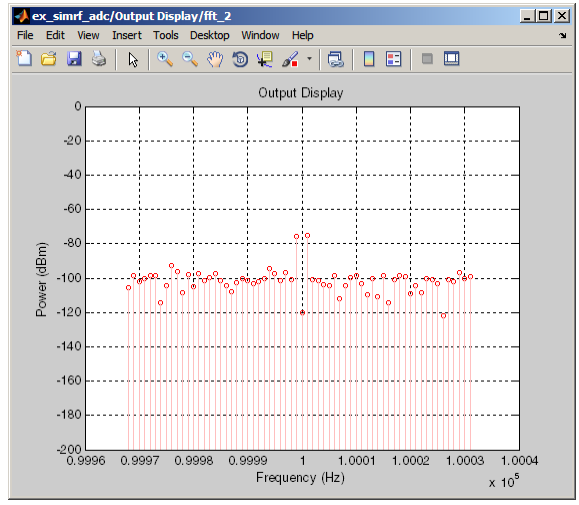
Improving Receiver-ADC Performance

Increasing the gain in the mixer raises the receiver noise without increasing the noise figure. Calculate the mixer gain required to achieve a 16-dB margin between the quantization noise floor and the receiver noise:

$$\begin{aligned} G_{mixer} &= (QNF_{ADC} + 16) - (-174 + G_{sys} + NF_{sys}) \\ &= (-116.1 + 16) - (-174 + 40 + 10.0) \\ &= -100.1 + 124.0 \\ &= 23.9 \text{ dB} \end{aligned}$$

To simulate a receiver that clears the quantization noise floor:

1. Set the **Available power gain** parameter of the mixer to `23.9`.
2. Select **Simulation > Start**.



The figure shows that the receiver noise is 16 dB above the quantization noise floor. Therefore, the SNR at the output is dominated by the receiver noise.

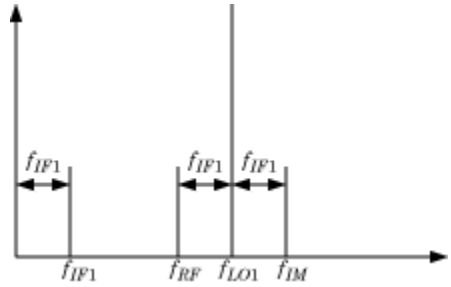
2

Interference

- “Carrier to Interference Performance of a Weaver Receiver” on page 2-2
- “Modeling LO Phase Noise” on page 2-8

Carrier to Interference Performance of a Weaver Receiver

A classic superhetorodyne architecture filters images prior to frequency conversion. In contrast, image-reject receivers remove the images at the output without filtering but are sensitive to phase offsets.



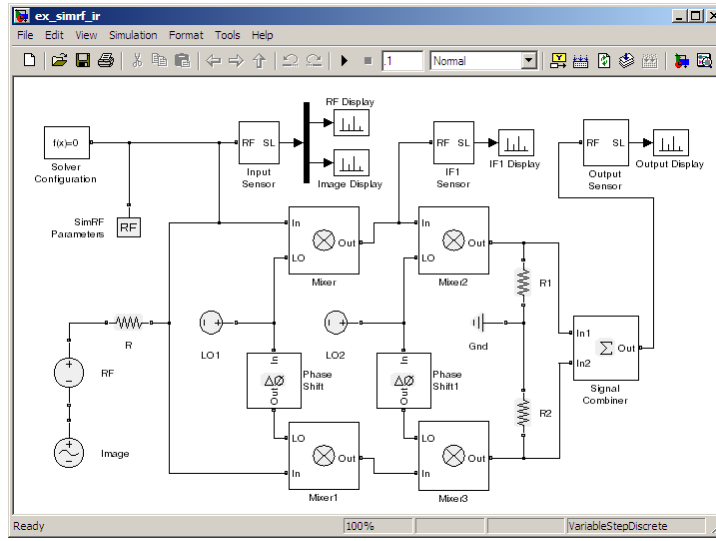
The preceding figure illustrates two input signals at the carriers f_{RF} and f_{IM} that both differ from the LO frequency, f_{LO1}, by an amount f_{IF1}. Mixing translates both input signals down to f_{IF1}. Perfect image rejection in the final stage of the receiver removes the image signal from the output entirely.

Create a Model with RF Interference

The model

```
ex_simrf_ir
```

simulates image rejection in a Weaver architecture. The receiver downconverts the signals at f_{RF} and f_{IM} to f_{IF1} and f_{IF2} in two sequential stages.



To run the model:

1. Open the model by clicking the link or by typing the model name at the Command Window prompt.
2. Select **Simulation > Start**.

Models that contain SimRF Amplifier, Mixer, or S-Parameter blocks generate files in the current MATLAB folder at run time. However, you can configure the output location for these files by specifying a cache folder in the Simulink Preferences dialog box. To specify a cache folder, open the Simulink Preferences dialog box (**File > Preferences**) and specify a location on your file system for the **Simulink cache folder** parameter. For more information about the Simulink interface, see Simulink Preferences Window.

Set Up the SimRF Environment

The model runs according to the following settings:

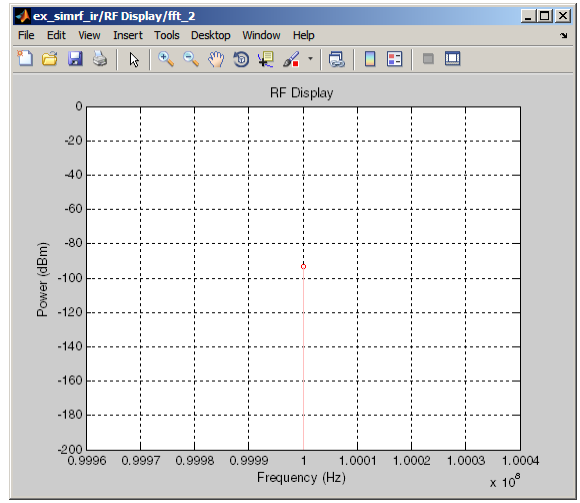
- In the SimRF Parameters Block Parameters dialog box, the **Carrier Frequencies** parameter specifies the carriers of the SimRF environment:
  - f_{RF}, the RF carrier.
  - f_{IM}, the image carrier.
  - f_{LO1}, the frequency of the LO in the first mixing stage.
  - f_{LO2}, the frequency of the LO in the second mixing stage.
  - f_{IF1}, the intermediate frequency of the signal after the first mixing stage, equal to $f_{LO1} - f_{RF}$ and $f_{IM} - f_{LO1}$.
  - f_{IF2}, the intermediate frequency of the signal after the second mixing stage, equal to $f_{LO2} - f_{IM}$.
- In the Solver Configuration Block Parameters dialog box, the **Use local solver** box is selected. This setting causes the SimRF environment to simulate with a local solver with the following settings:
  - **Solver type** is `Trapezoidal rule`.
  - **Sample time** is below the Nyquist frequency of the modulation.

  Since the model uses a local solver, the global solver settings do not affect the simulation within the SimRF environment. For more information on global and local solvers, see Choosing Simulink and Simscape Solvers.

View Simulation Output

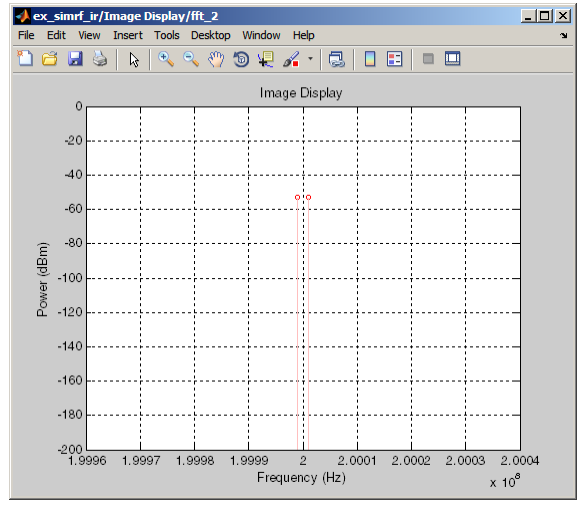
The model uses subsystems with a MATLAB Coder implementation of a fast Fourier transform (FFT) to generate four plots.

- The RF Display plot shows the power spectrum of the signal recovered from the carrier f_{RF}, specified as `carriers.RF` in the **Carrier Frequencies** parameter of the preceding SimRF Outport block.



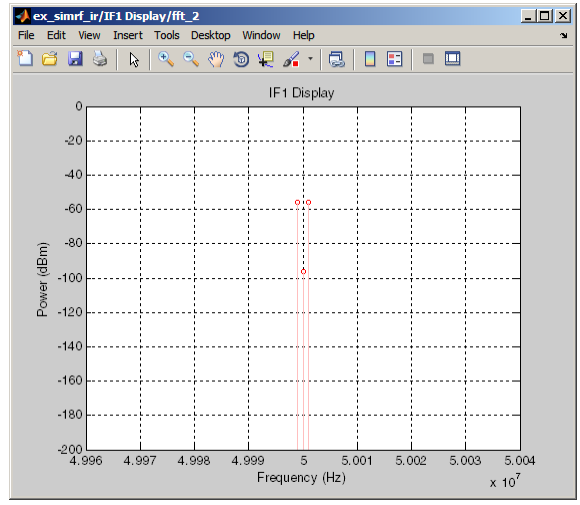
The modulation of the RF carrier is a constant envelope generated by a Continuous Wave block which generates a single peak centered at the carrier.

- The Image Display plot shows the power spectrum of the image. The signal is recovered from the carrier f_{IM}, specified as `carriers.IM` in the **Carrier Frequencies** parameter of the preceding SimRF Outport block.



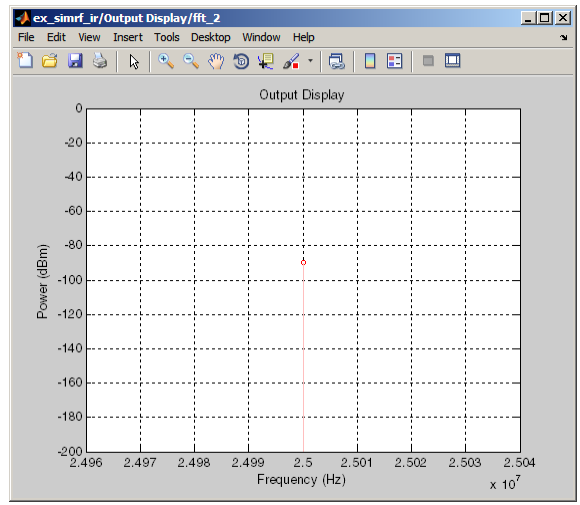
The Sinusoid source generates a two-tone signal centered at f_{IM}.

- The IF1 Display plot shows a power spectrum centered at the first intermediate frequency, measured between the first and second stages. The sensor outputs the modulation from the carrier f_{IF1}, specified as `carriers.IF2` in the **Carrier Frequencies** parameter of the preceding SimRF Outport block.



Both the RF and image appear on the carrier. The power level of the image 40 dB higher than the RF.

- The Output Display plot shows the effects of the RF system. The sensor outputs the modulation from the carrier f_{IF2}, specified as `carriers.IF1` in the **Carrier Frequencies** parameter of the preceding SimRF Outport block.



As expected, the system removes the image and amplifies the RF by 6 dB.

If you have DSP System Toolbox software installed, you can replace the MATLAB Coder subsystems with Vector Scope or Spectrum Scope blocks.

Modeling RF and Blocker Sources

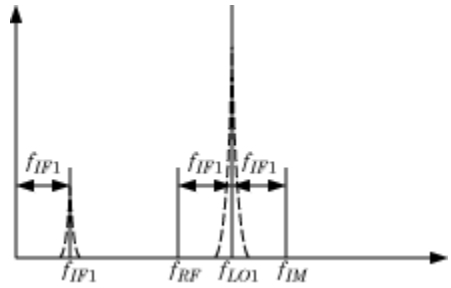
To model more robust input signals, you can use a SimRF Inport block to specify a circuit envelope generated using blocks from other Simulink libraries. For example, the demo Impact of RF Receiver on Communications System Performance uses Communications System Toolbox™ blocks to model a QPSK-modulated waveform of random bits with SimRF Inport that brings the signal into the SimRF environment.

Simulating LO Phase Offset

The phase shifters have specified **Phase shift** parameters of `90`. Deviation from this value results in a *phase offset* and causes imperfect image rejection. The demo Simulating Image Rejection Ratio Measurements analyzes the IRR of the Weaver and Hartley architectures several times, calculating the image rejection ratio (IRR) for several different phase offsets.

Modeling LO Phase Noise

A mixer transfers local oscillator (LO) phase noise directly to its output.



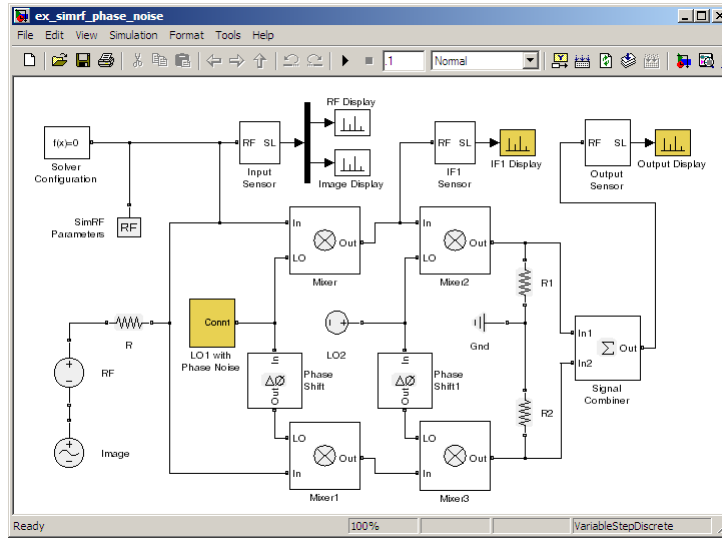
The preceding figure shows the transfer of phase noise from f_{LO1} to f_{IF1}.

Create a Model with Phase Noise

The model

```
ex_simrf_phase_noise
```

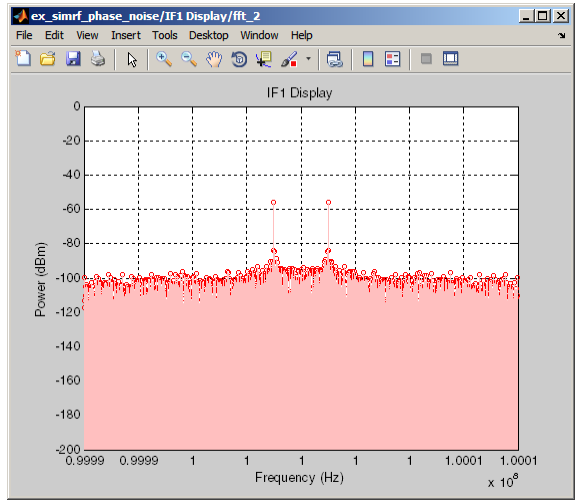
introduces phase noise into the model from the section "Create a Model with RF Interference" on page 2-2. The first mixing stage downconverts the RF and image to f_{IF}



View Simulation Output

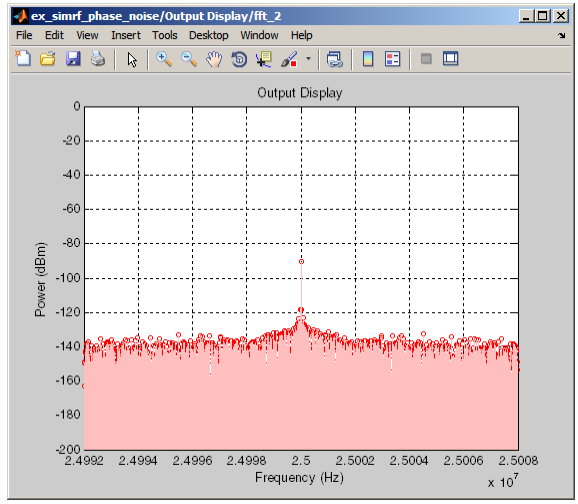
The model uses subsystems with a MATLAB Coder implementation of a fast Fourier transform (FFT) to generate four plots.

- The IF1 Display plot shows a power spectrum centered at the first intermediate frequency, measured between the first and second stages.



The figure shows that the LO phase noise has been transferred to the image. The RF signal on the carrier f_{IF1} is not visible in the figure because its power level is below the phase noise power of the downconverted image signal. The two visible peaks are at the same power as the IF shown in the previous section, “Create a Model with RF Interference” on page 2-2.

- The Output Display plot shows the downconverted RF with the images removed.

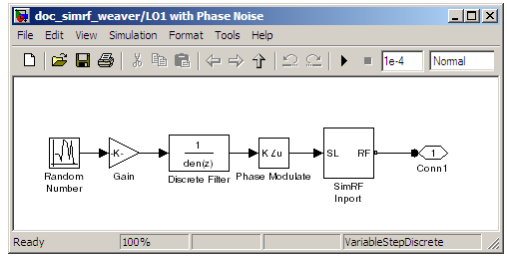


The LO phase noise has been transferred to the receiver output. The peak signal power is the same as in the previous section, “Create a Model with RF Interference” on page 2-2.

If you have DSP System Toolbox software installed, you can replace the MATLAB Coder subsystems with Vector Scope or Spectrum Scope blocks.

Shaping the LO Noise Skirt

To simulate phase noise, the model phase modulates pink noise generated in the LO with Phase Noise subsystem:



The subsystem contains the following blocks:

- A Random Number block outputs a Gaussian random number at discrete time steps to generate white noise.
- A Gain block scales the signal by a factor of

  $$\sqrt{2\pi f_{LO1} 10^{P_{rel}/10}}$$

  where f_{LO1} is the LO frequency and P_{rel} is the relative noise power density in dBc/Hz.
- A Discrete Filter block filters the uniform white noise to generate $1/f$ noise.
- A Magnitude-Angle to Complex block phase modulates an input signal φ. The output is of the form $\exp(j\varphi)$.
- A SimRF Inport block models a controlled voltage source in the SimRF environment, modulating the carrier f_{LO} with the input signal $\exp(j\varphi)$. The circuit-envelope equivalent signal is $\exp[j(2\pi f_{LO1} t + \varphi(t))]$.

If you have Communications System Toolbox software installed, use the Phase Noise block to add phase noise to a given input signal.

3

Intermodulation Distortion

Modeling a Direct Conversion Receiver

In this section...

"Create a Direct Conversion Receiver Model" on page 3-2
"Modeling IMD in System-Level Components" on page 3-6
"Examining DC Impairments" on page 3-7

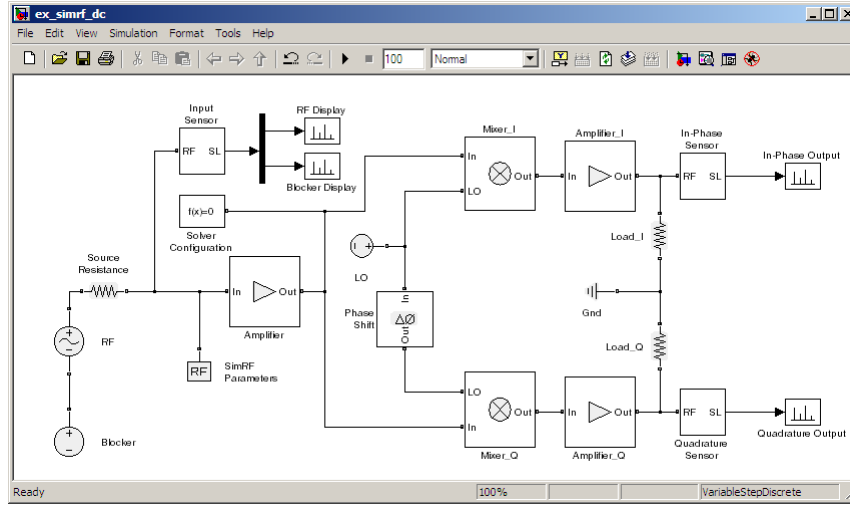
Direct-conversion receivers are sensitive to second-order intermodulation products because they transfer the RF signal directly to baseband.

Create a Direct Conversion Receiver Model

The model

```
ex_simrf_dc
```

models a direct-conversion receiver within the SimRF environment. The RF system consists of a low-noise amplification (LNA) stage, a direct-conversion stage, and a final amplification stage. The receiver specifications are similar to the specifications used in the Design and Simulation of a Direct Conversion Receiver demo, which elaborates on the impairments shown in this example.



To run the model:

1. Open the model by clicking the link or by typing the model name at the Command Window prompt.
2. Select **Simulation > Start**.

Models that contain SimRF Amplifier, Mixer, or S-Parameter blocks generate files in the current MATLAB folder at run time. However, you can configure the output location for these files by specifying a cache folder in the Simulink Preferences dialog box. To specify a cache folder, open the Simulink Preferences dialog box (**File > Preferences**) and specify a location on your file system for the **Simulink cache folder** parameter. For more information about the Simulink interface, see Simulink Preferences Window.

Set Up the SimRF Environment

The model runs according to the following environment settings:

- In the SimRF Parameters Block Parameters dialog box, the **Carrier Frequencies** parameter specifies the carriers in the SimRF environment:
  - $f_{RF} = f_{LO}$, the carrier of the RF and the local oscillator.
  - f_{BL}, the blocker carrier

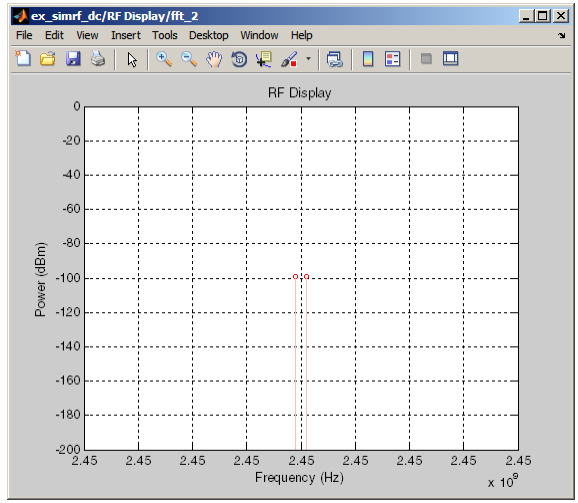
  The SimRF environment always simulates the 0 Hz carrier, regardless of whether the SimRF Parameters block specifies it.
- In the Solver Configuration Block Parameters dialog box, the **Use local solver** box is selected. This setting causes the SimRF environment to simulate with a local solver with the following settings:
  - **Solver type** is `Trapezoidal rule`.
  - **Sample time** is `sample_time`, defined as `1.25e-4` in the model initialization function.

  Since the model uses a local solver, the global solver settings do not affect the simulation within the SimRF environment. For more information on global and local solvers, see Choosing Simulink and Simscape Solvers.

View Simulation Output

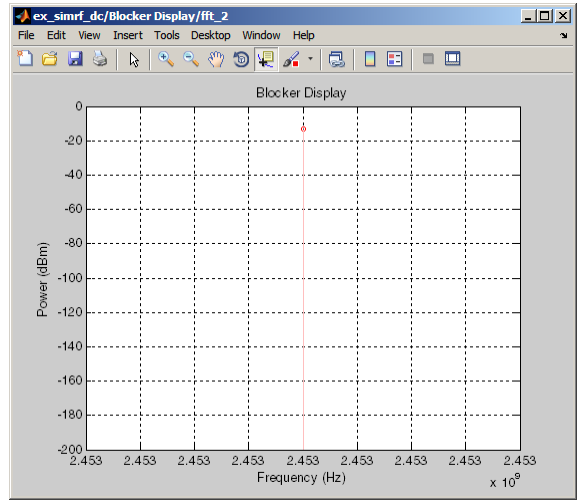
The model uses subsystems with a MATLAB Coder implementation of a fast Fourier transform (FFT) to generate four plots:

- The RF Display plot shows the power level of the RF signal.



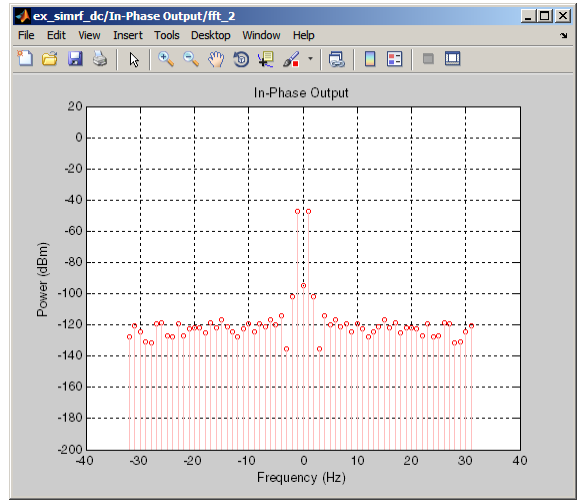
The power level of the RF is about 100 dBm.

- The Blocker Display plot shows the power spectrum centered at the carrier f_{BL}..



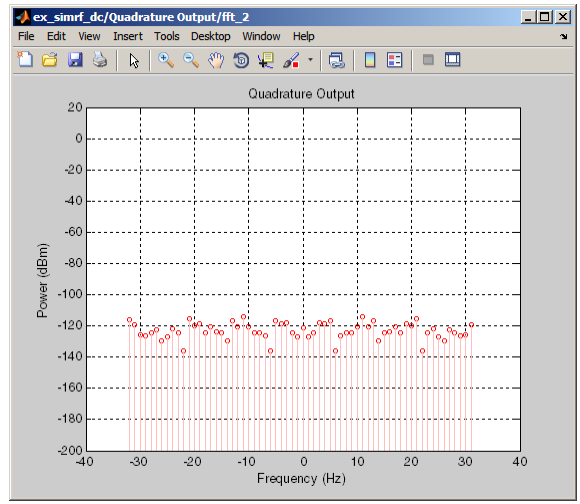
The power level of the blocker is about 90 dB higher than the signal power of the RF..

- The In-Phase Output plot shows the power spectrum of the in-phase signal at baseband.



In the figure, DC power is a direct result of the blocker and the IP2 in the mixers.

- The Quadrature Output plot shows the power spectrum of the quadrature signal at baseband.



The quadrature output only contains noise because the input signal and blocker have no quadrature components.

If you have DSP System Toolbox software installed, you can replace the MATLAB Coder subsystems with Vector Scope or Spectrum Scope blocks.

Modeling IMD in System-Level Components

The **IP2** and **IP3** parameters specify the second- and third-order intercept points of Amplifier and Mixer blocks:

- The amplifiers have infinite **IP2** and **IP3**, so the amplifiers are linear.
- **IP2** of the mixer is `-10` dB

Amplifier and Mixer components have specified gains and noise figures:

- The gain and noise figure in the LNA stage are 25 dB and 6 dB, respectively.
- The gain and noise figure in the mixing stage are 10 dB and 10 dB. The **Input impedance (ohms)** parameters of the two mixers are both

100, which sum in parallel to a resistance of 50 Ω to match the output impedance of the LNA.

- The gain and noise figure in the final amplification stage are 20 dB and 15 dB, respectively.

To calculate RF system noise figure, use the Friis equation:

$$F_{sys} = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_1 G_2} + \ldots + \frac{F_n}{G_1 G_2 \ldots G_{n-1}}$$

where F_i and G_i are the noise factor and gain of the ith stage. For more information on RF system noise figure, see the demo Impact of RF Receiver on Communcations System Performance.

Examining DC Impairments

In addition to intermodulation distortion from IP2, direct-conversion receivers are subject to additional DC impairments. For example, coupling between mixer input and local oscillator (LO) ports causes self-mixing of the LO. For more information, see the demo Executable Specification of a Direct Conversion Receiver.