

RF Blockset

For Use with Simulink®

- Modeling
- Simulation
- Implementation

User's Guide

Version 1



How to Contact The MathWorks



www.mathworks.com
comp.soft-sys.matlab
www.mathworks.com/contact_TS.html

Web
Newsgroup
Technical Support



suggest@mathworks.com
bugs@mathworks.com
doc@mathworks.com
service@mathworks.com
info@mathworks.com

Product enhancement suggestions
Bug reports
Documentation error reports
Order status, license renewals, passcodes
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.

RF Blockset User's Guide

© COPYRIGHT 2004–2006 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, Simulink, Stateflow, Handle Graphics, Real-Time Workshop, and xPC TargetBox are registered trademarks, and SimBiology, SimEvents, and SimHydraulics are trademarks of The MathWorks, Inc.

Other product or brand names are trademarks or registered trademarks of their respective holders.

Patents

The MathWorks products are protected by one or more U.S. patents. Please see www.mathworks.com/patents for more information.

Revision History

June 2004 Online only
August 2004 Online only
March 2005 Online only
September 2005 Online only
March 2006 Online only
September 2006 Online only

New for Version 1.0 (Release 14)
Revised for Version 1.0.1 (Release 14+)
Revised for Version 1.1 (Release 14SP2)
Revised for Version 1.2 (Release 14SP3)
Revised for Version 1.3 (Release 2006a)
Revised for Version 1.3.1 (Release 2006b)

Getting Started

1

| | |
|--|-------------|
| What Is the RF Blockset? | 1-2 |
| Required and Related Products | 1-3 |
| Product Demos | 1-4 |
| RF Blockset Libraries | 1-6 |
| Opening RF Blockset Libraries | 1-6 |
| Physical Library | 1-7 |
| Mathematical Library | 1-9 |
| RF Blockset Workflow | 1-10 |
| Example — Modeling an LC Bandpass Filter | 1-12 |
| Selecting Blocks to Represent System Components | 1-12 |
| Building the Model | 1-13 |
| Specifying Model Parameters | 1-15 |
| Validating Filter Components and Running the Simulation | 1-22 |
| Analyzing the Simulation Results | 1-24 |

Modeling an RF System

2

| | |
|---|------------|
| Modeling RF Components | 2-2 |
| Adding RF Blocks to a Simulink Model | 2-2 |
| Connecting Model Blocks | 2-2 |
| Specifying or Importing Component Data | 2-6 |

| | |
|--|-------------|
| Specifying Parameter Values | 2-6 |
| Importing Data Files into RF Blocks | 2-7 |
| Example — Importing a Touchstone Data File into an RF Model | 2-8 |
| Importing Circuits from the MATLAB Workspace | 2-14 |
| Example — Importing a Bandstop Filter into an RF Model | 2-15 |
| Modeling Thermal Noise | 2-22 |
| Amplifier and Mixer Noise Specifications | 2-22 |
| Adding Noise to Your System | 2-23 |
| Plotting Noise | 2-24 |

Plotting Model Data

3

| | |
|---|-------------|
| Creating Plots | 3-2 |
| Validating Individual Blocks and Subsystems | 3-2 |
| Types of Plots | 3-3 |
| Plot Formats | 3-4 |
| How to Create a Plot | 3-9 |
| Example — Plotting Component Data on a Z Smith Chart | 3-15 |
| Updating Plots | 3-19 |
| Modifying Plots | 3-20 |
| Example — Creating and Modifying Subsystem Plots | 3-22 |
| Plotting the Network Parameters of a Subsystem | 3-22 |
| Adding Data to an Existing Plot | 3-24 |
| Changing Data on an Existing Plot | 3-26 |

Blocks — By Category

4

| | |
|---------------------------|------------|
| Mathematical | 4-2 |
| Physical | 4-3 |
| Ladder Filters | 4-3 |
| Transmission Lines | 4-3 |
| Black Box Elements | 4-4 |
| Amplifiers | 4-4 |
| Mixers | 4-5 |
| Input/Output Ports | 4-5 |

Blocks — Alphabetical List

5

RF Blockset Algorithms

A

| | |
|---|-------------|
| Simulating an RF Model | A-2 |
| Determining the Modeling Frequencies | A-3 |
| Mapping Network Parameters to Modeling Frequencies | A-5 |
| Modeling Noise in an RF System | A-7 |
| Calculating Noise Figure at Modeling Frequencies | A-8 |
| Calculating Output Noise Power | A-9 |
| Creating a Complex Baseband-Equivalent Model | A-10 |
| Baseband-Equivalent Modeling | A-10 |
| Simulation Efficiency of a Complex Baseband-Equivalent Model | A-13 |

Converting to and from Simulink Signals A-15

Examples

B

Examples B-2

Index

Getting Started

| | |
|--|---|
| What Is the RF Blockset? (p. 1-2) | Introduces the RF Blockset and describes its capabilities |
| Required and Related Products (p. 1-3) | Describes products you can use to extend the capabilities of the RF Blockset |
| Product Demos (p. 1-4) | Describes how to access the RF Blockset demos in the Help browser |
| RF Blockset Libraries (p. 1-6) | Summarizes the RF Blockset libraries |
| RF Blockset Workflow (p. 1-10) | Describes a typical workflow |
| Example — Modeling an LC Bandpass Filter (p. 1-12) | Describes how to build an RF model, modify the parameters, run the model in Simulink, and examine the results |

What Is the RF Blockset?

The RF Blockset extends Simulink® with a library of blocks for modeling RF systems that include RF filters, transmission lines, amplifiers, and mixers. For more information about creating and running Simulink models, see “Building a Model” in the Simulink User’s Guide.

You use blocks from the RF Blockset to represent the components of your RF system in a Simulink model. The blockset provides several types of component representations using network parameters (S, Y, Z, ABCD, H, and T format), mathematical descriptions, and physical properties.

In the Simulink model, you cascade the components to represent your RF architecture and run the simulation. During the simulation, all blocks are modeled using a time-domain, complex-baseband representation. This way of modeling results in fast simulation and enables compatibility with other Simulink blocks.

The RF Blockset lets you visualize the network parameters of the blocks using plots and Smith charts.

A validated Simulink model of an RF system can provide an executable specification for RF circuit design for wireless communication systems.

Required and Related Products

In addition to MATLAB and Simulink, you must have the following products installed to use the RF Blockset:

- RF Toolbox — Provides MATLAB functions for defining, simulating, and visualizing RF components.
- Signal Processing Toolbox — Provides MATLAB functions for filtering wireless communication signals.
- Signal Processing Blockset — Provides Simulink blocks for time-domain simulation of communication signals.

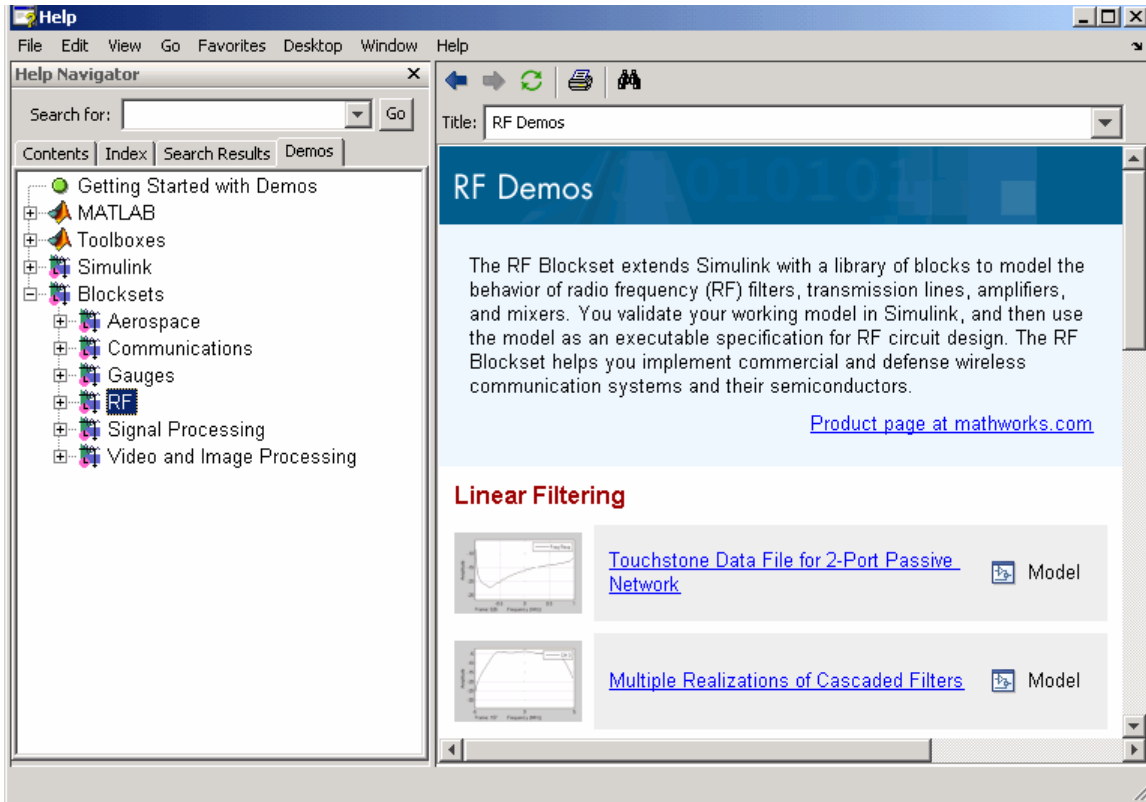
You can build sophisticated wireless communication system models by incorporating blocks from other blocksets, such as the Signal Processing Blockset and Communications Blockset.

The MathWorks provides several products that are especially relevant to the kinds of tasks you can perform with the RF Blockset. The following table summarizes the related products and describes how they complement the features of the RF Blockset.

| Product | Description |
|-------------------------|--|
| Communications Blockset | Simulink blocks for time-domain simulation of modulation and demodulation of a wireless communications signal. |
| Communications Toolbox | MATLAB functions for signal modulation and demodulation. |
| Filter Design Toolbox | MATLAB functions for filtering the modulated communication signal. |

Product Demos

You can find interactive RF Blockset demos in the MATLAB Help browser, as shown in the following figure.



This example shows you how to locate and open an RF Blockset demo:

- 1 Type **demos** at the MATLAB prompt to open the Help browser to the **Demos** tab.
- 2 Select **Blocksets > RF** in the **Demos** tab to see a list of demo categories.
- 3 Select a model, and click **Open this model** in the upper-right corner of the demo window to display the Simulink model for this demo.

- 4** In the model window, select **Simulation > Start** to run the demo simulation.

RF Blockset Libraries

The RF Blockset consists of the Physical and Mathematical libraries of components for modeling RF systems within the Simulink environment. An RF model can contain blocks from both the Physical and Mathematical libraries. It can also include Simulink blocks and blocks from other MathWorks blocksets, such as those described in “Required and Related Products” on page 1-3.

This section contains the following topics:

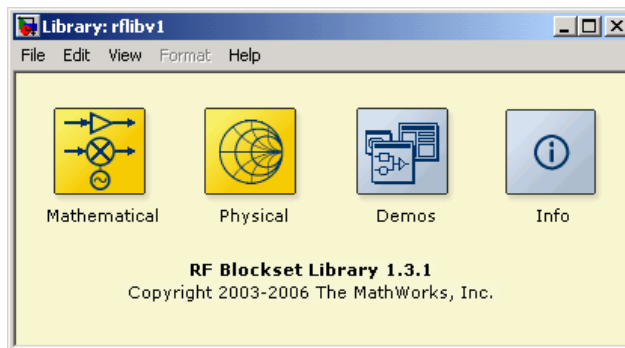
- “Opening RF Blockset Libraries” on page 1-6
- “Physical Library” on page 1-7
- “Mathematical Library” on page 1-9

Opening RF Blockset Libraries

To open the main library window, type the following at the MATLAB prompt:

```
rflib
```

The window is shown in the following figure. Each yellow icon in the window represents a library. Double-click an icon to open the corresponding library.



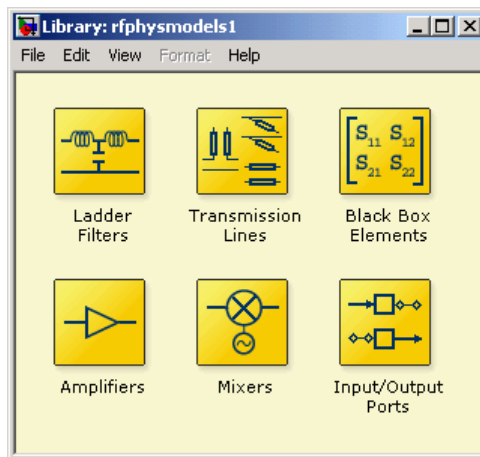
The Physical and Mathematical libraries are discussed in the following sections.

Note The blue icons take you to the MATLAB Help browser.

- Double-click the Demos icon to open the RF Blockset demos.
 - Double-click the Info icon to open the RF Blockset documentation.
-

Physical Library

Use blocks from the Physical library to model physical and electrical components by specifying physical properties or by importing measured data. The Physical library includes several sublibraries, as shown in the following figure.



The following table describes the sublibraries and how they can be used.

| Sublibrary | Description |
|------------|--|
| Amplifiers | RF amplifiers, specified using S-, Y-, or Z-parameters, noise figure, and IP3, or a data file containing these parameters. |

| Sublibrary | Description |
|--------------------|--|
| Ladder Filters | RF filters, specified using RLC parameters. The network parameters of the blocks are calculated from the topology of the filter and the RLC values. |
| Mixers | RF mixers that contain local oscillators, specified using S-, Y-, or Z-parameters and phase noise, or a data file containing these parameters. |
| Transmission Lines | RF filters, specified using physical dimensions and electrical characteristics. |
| Black Box | Passive RF components, specified using S-, Y-, or Z-parameters, or a data file containing these parameters. |
| Input/Output Ports | <p>Blocks for specifying simulation information that pertains to all blocks in a physical <i>subsystem</i>, such as center frequency and sample time.</p> <hr/> <p>Note A physical subsystem is a collection of one or more physical blocks bracketed by an Input Port block and an Output Port block that bridge the physical and mathematical parts of the model.</p> <hr/> |

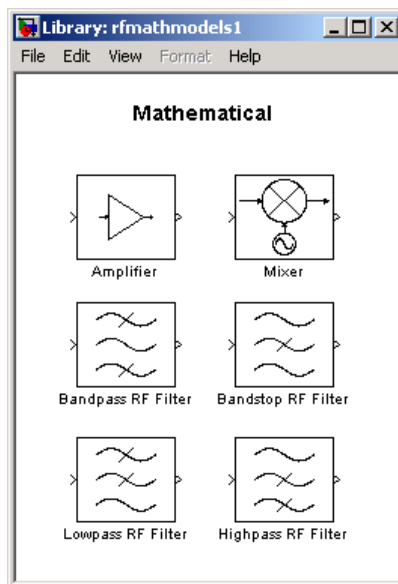
For more information on defining components, see “Specifying or Importing Component Data” on page 2-6.

Mathematical Library

The Mathematical library contains mathematical representations of the amplifier, mixer, and filter blocks. Use a block from the Mathematical library to model an RF component in terms of mathematical equations that describe how the block operates on an input signal.

Note Mathematical blocks assume perfect impedance matching and a nominal impedance of 1 ohm. In contrast, the physical blocks do not assume perfect matching—these blocks model the reflections that occur between blocks, and you can specify the source and load impedances using the Input Port and Output Port blocks.

The mathematical library is shown in the following figure.



RF Blockset Workflow

When you analyze an RF system using the RF Blockset, your workflow might include the following tasks:

- 1 Create a Simulink model of RF components.

For more information, see “Modeling RF Components” on page 2-2.

- 2 Define component data by

- Specifying network parameters, mathematical relationships, or physical properties
- Importing data from an industry-standard Touchstone file, a MathWorks AMP file, or the MATLAB workspace

The RF Blockset lets you access component data in Touchstone SnP, YnP, ZnP, and HnP formats. You can also import amplifier network parameters and power data from a MathWorks AMP file.

For more information, see “Specifying or Importing Component Data” on page 2-6.

- 3 Add thermal noise to the system.

For more information, see “Modeling Thermal Noise” on page 2-22.

- 4 Validate the behavior of individual blocks by plotting component data.

Note You can plot data for individual blocks from the RF Physical library that model physical components either before or after you run a simulation.

For more information, see “Creating Plots” on page 3-2.

- 5 Run the simulation.

For more information on how the RF Blockset performs time-domain simulation of an RF system, see “Simulating an RF Model” on page A-2.

- 6 Generate plots to gain insight into system behavior.

For more information, see “Creating Plots” on page 3-2.

The following plots and charts are available in the RF Blockset:

- Rectangular plots
- Polar plots
- Smith charts
- Composite plots
- Budget plots

Example – Modeling an LC Bandpass Filter

In this example, you model the signal attenuation caused by an RF filter by comparing the signals at the input and output of the filter.

The RF filter you use in this example is an LC bandpass filter with a bandwidth of 200 MHz, centered at 700 MHz. You use a three-tone input signal to stimulate a range of in-band and out-of-band frequencies of the filter. The input signal has the following tones:

- 700 MHz — Center of the filter
- 600 MHz — Lower edge of the filter passband
- 900 MHz — Outside the filter passband

You simulate the effects of the filter over a bandwidth of 500 MHz.

This example illustrates how to perform the following tasks:

- “Selecting Blocks to Represent System Components” on page 1-12
- “Building the Model” on page 1-13
- “Specifying Model Parameters” on page 1-15
- “Validating Filter Components and Running the Simulation” on page 1-22
- “Analyzing the Simulation Results” on page 1-24

Selecting Blocks to Represent System Components

In this part of the example, you select the blocks to represent the input signal, the RF filter, and the signal displays.

You model the RF filter using a physical *subsystem*, which is a collection of one or more physical blocks bracketed by an Input Port block and an Output Port block. The RF filter subsystem consists of an LC Bandpass Pi block, and the Input Port and Output Port blocks. The function of the Input Port and Output Port blocks is to bridge the physical and mathematical parts of the model.

The following table lists the blocks that represent the RF system components and a description of the role of each block.

| Block | Description |
|-----------------------|---|
| Sine Wave | Generates a three-channel signal. |
| Matrix Sum | Combines the three channel signal into a single three-tone source signal. |
| Input Port | Establishes parameters that are common to all blocks in the RF filter subsystem, including the source impedance of the subsystem that is used to convert Simulink signals to the RF Blockset physical modeling environment. |
| LC Bandpass Pi | Models the signal attenuation caused by the RF filter which, in this example, is the LC Bandpass Pi filter. |
| Output Port | Establishes parameters that are common to all blocks in the RF filter subsystem. These parameters include the load impedance of the subsystem, which is used to convert RF signals to Simulink signals. |
| Spectrum Scope | Displays signals at the input to and output of the filter. |

Building the Model

In this part of the example, you create a Simulink model, add blocks to the model, and connect the blocks.

1 Create a Simulink model.

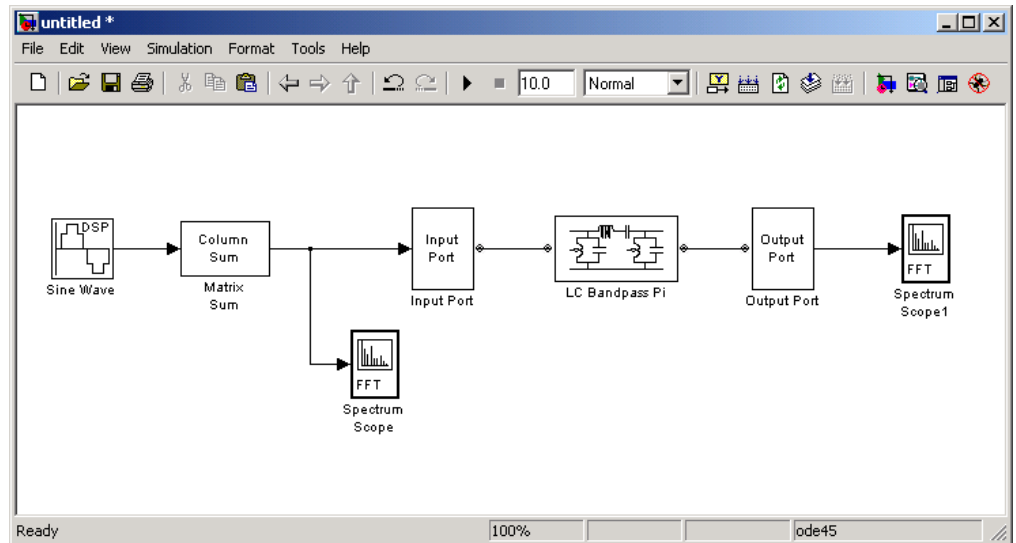
If you are new to Simulink, see the introductory Simulink example, “Building a Model”, for information on how to create a Simulink model.

2 Add to the model the blocks shown in the following table. The Library column of the table specifies the hierarchical path to each block.

| Block | Library Path | Quantity |
|-----------------------|---|-----------------|
| Sine Wave | Signal Processing Blockset > Signal Processing Sources | 1 |
| Matrix Sum | Signal Processing Blockset > Math Functions > Matrices and Linear Algebra > Matrix Operations | 1 |
| Spectrum Scope | Signal Processing Blockset > Signal Processing Sinks | 2 |
| Input Port | RF Blockset > Physical > Input/Output Ports | 1 |
| LC Bandpass Pi | RF Blockset > Physical > Ladder Filters | 1 |
| Output Port | RF Blockset > Physical > Input/Output Ports | 1 |

3 Connect the blocks as shown in the following figure.

For more information on connecting physical and mathematical blocks, see “Connecting Model Blocks” on page 2-2.



Now you are ready to specify block parameters.

Specifying Model Parameters

In this part of the example, you specify the following parameters to represent the behavior of the system components:

- “Input Signal Parameters” on page 1-15
- “Filter Subsystem Parameters” on page 1-18
- “Signal Display Parameters” on page 1-21

Input Signal Parameters

You generate the three-tone source signal using two blocks. You use the Sine Wave block to build a complex three-channel signal, where each channel

corresponds to a different frequency. Then, you use the Matrix Sum block to combine the channels into a single three-tone source signal.

The algorithm used by the RF Blockset requires you to shift the frequencies of the input signal. The RF Blockset simulates the filter subsystem using a complex-baseband modeling technique, which shifts the filter response so it is centered at zero. You must shift the frequencies of the signals outside the physical subsystem by the same amount.

For more information on complex-baseband modeling, see “Creating a Complex Baseband-Equivalent Model” on page A-10.

Note The RF Blockset requires that all signals in the RF model be complex to match the signals in the physical subsystem, so you create a complex input signal.

The center frequency of the LC bandpass filter is 700 MHz, so you use a three-tone source signal with tones that are 700 MHz below the actual tones, at -100 MHz, 0 MHz, and 200 MHz, respectively.

1 Set the Sine Wave block parameters as follows:

- **Amplitude** = $1e-6$
- **Frequency (Hz)** = $[-100 \ 0 \ 200]*1e6$
- **Output complexity** = Complex
- **Sample time** = $1/500e6$

- **Samples per frame = 128**

Source Block Parameters: Sine Wave

Sine Wave (mask) (link)

Output samples of a sinusoid. To generate more than one sinusoid simultaneously, enter a vector of values for the Amplitude, Frequency, and Phase offset parameters.

Main | Data Types

Amplitude: 1e-6

Frequency (Hz): [-100 0 200]*1e6

Phase offset (rad): 0

Sample mode: Discrete

Output complexity: Complex

Computation method: Trigonometric fcn

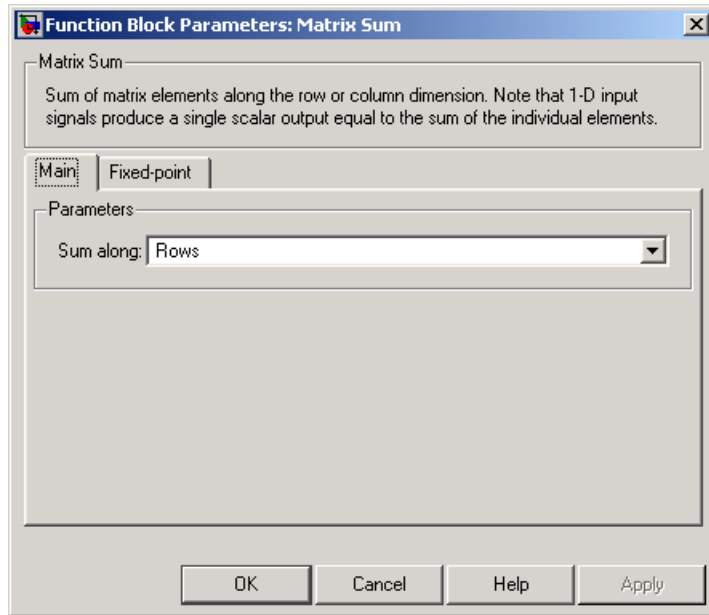
Sample time: 1/500e6

Samples per frame: 128

Resetting states when re-enabled: Restart at time zero

OK Cancel Help

- 2 Set the Matrix Sum block **Sum along** parameter to Rows.

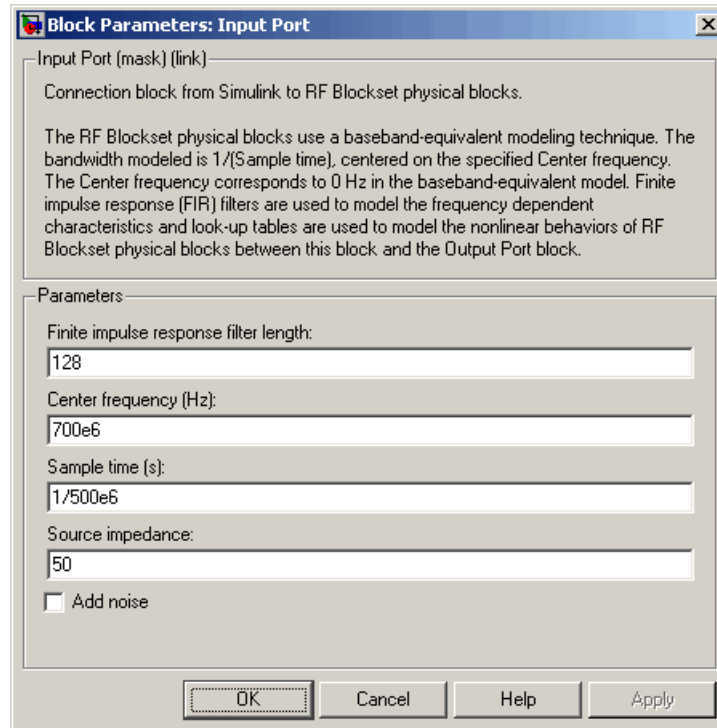


Filter Subsystem Parameters

In this part of the example, you configure the blocks that model the RF filter subsystem—the Input Port, LC Bandpass Pi, and Output Port blocks.

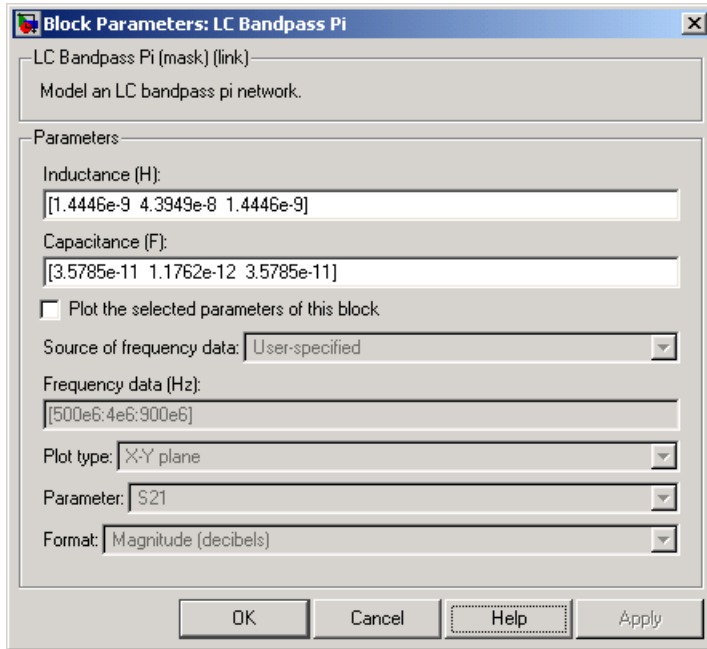
- 1 Set the Input Port block parameters as follows:
 - **Center frequency** = 700e6
 - **Sample time(s)** = 1/500e6

- Clear the **Add noise** check box so the RF Blockset will not include thermal noise in the simulation. To learn how to model thermal noise, see “Modeling Thermal Noise” on page 2-22.

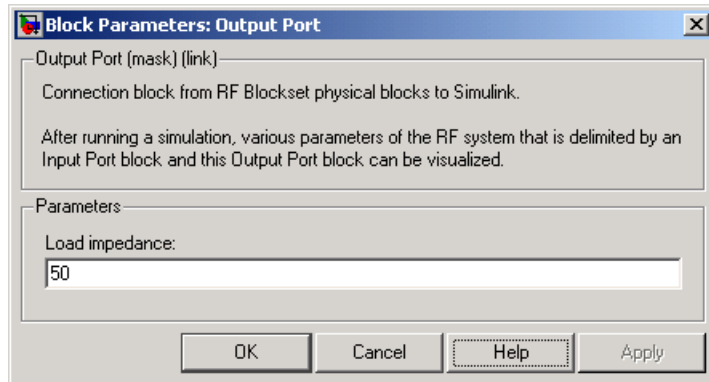


Note You must enter the **Sample time (s)** because the Input Port block does not inherit a sample time from the input signal. The specified sample time must match the sample time of the input signal. The **Sample time (s)** of 1/500e6 second used in this example is equivalent to a bandwidth of 500 MHz.

- 2 Accept default parameters for inductance and capacitance in the LC Bandpass Pi block. These parameters create a filter with the desired bandwidth of 200 MHz, centered at 700 MHz.



- 3 Accept the default parameters for the Output Port block to use a load impedance of 50 ohms.



Signal Display Parameters

In this part of the example, you specify the parameters of the Spectrum Scope block to display the source signal, and the parameters of the Spectrum Scope1 block to display the filtered signal.

For each scope, you set the range of the X- and Y-axes to make sure that the entire signal is visible.

By default, the scope displays appear stacked on top of each other on the screen when you run the simulation, so you can only see one of them. To ensure that both scopes are visible during the simulation, you specify a different position for each scope on the screen.

- 1 Set the Spectrum Scope block parameters as follows:
 - **Axis Properties** pane, **Frequency range** = $[-Fs/2 \dots Fs/2]$
 - **Axis Properties** pane, **Minimum Y-limit** = -160
 - **Axis Properties** pane, **Maximum Y-limit** = -100
 - **Display Properties** pane, **Scope position** = `get(0,'defaultfigureposition').* [.15 1 1 1]`

2 Set the Spectrum Scope1 block parameters as follows:

- **Axis Properties** pane, **Frequency range** = $[-Fs/2 \dots Fs/2]$
- **Axis Properties** pane, **Minimum Y-limit** = -160
- **Axis Properties** pane, **Maximum Y-limit** = -105
- **Display Properties** pane, **Scope position** = `get(0,'defaultfigureposition').*[1.85 1 1 1]`

Note If you do not specify the **Display Properties** of the scopes, you can click and drag the displays to arrange them on the screen after the simulation starts.

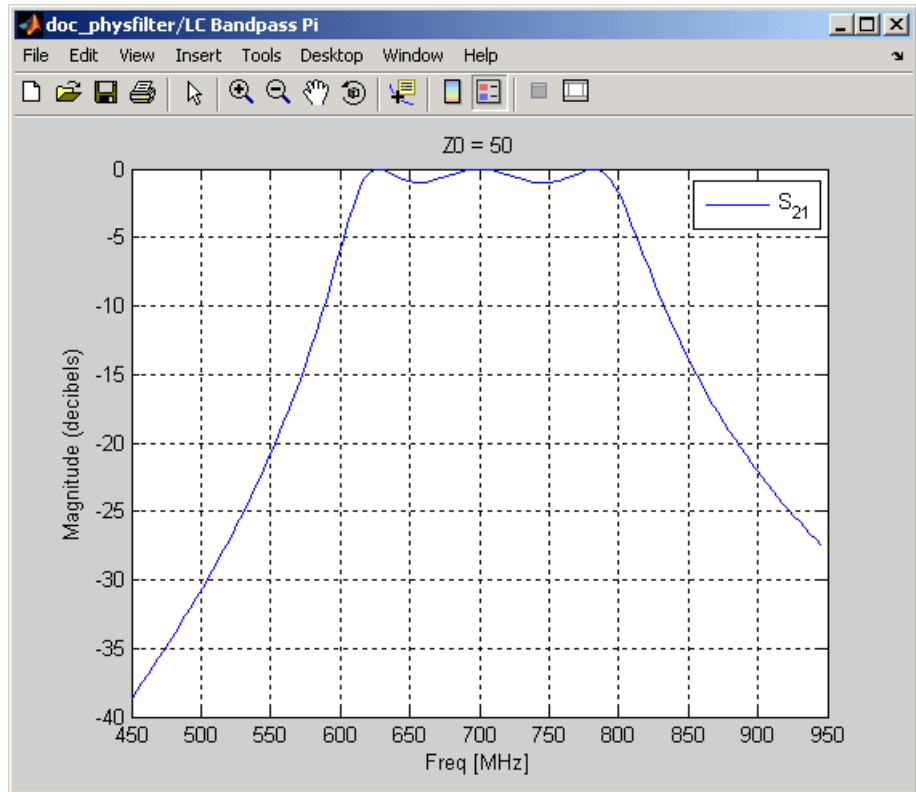
Validating Filter Components and Running the Simulation

In this part of the example, you validate the behavior of the LC Bandpass Pi filter block by plotting its network parameters and then run the simulation.

Note When you plot information about a physical block, the plot displays the actual frequency response of the block at the selected passband (i.e. the response at the unshifted frequencies), and *not* the response at the shifted frequencies. Recall that this shift was introduced in “Input Signal Parameters” on page 1-15.

1 Double-click the LC Bandpass Pi block to open the block dialog box.

- 2 Select the **Plot the selected parameters of this block** check box. Click **Apply** to plot the frequency response of the filter. This plots the magnitude of S_{21} as a function of frequency, which represents the gain of the filter.



Filter Gain

Note The physical blocks only model a band of frequencies around the center frequency of the physical subsystem. You must choose the sample time and center frequency such that all important frequency characteristics of your physical subsystem fall in this band of frequencies. The plot shows the frequency response of the filter for the portion of the RF spectrum that the physical blocks model. In this example, the physical blocks model a 500 MHz band that is centered at 700 MHz, defined by the Input Port block.

3 In the Simulink model window, select **Simulation > Start** to run the simulation.

Analyzing the Simulation Results

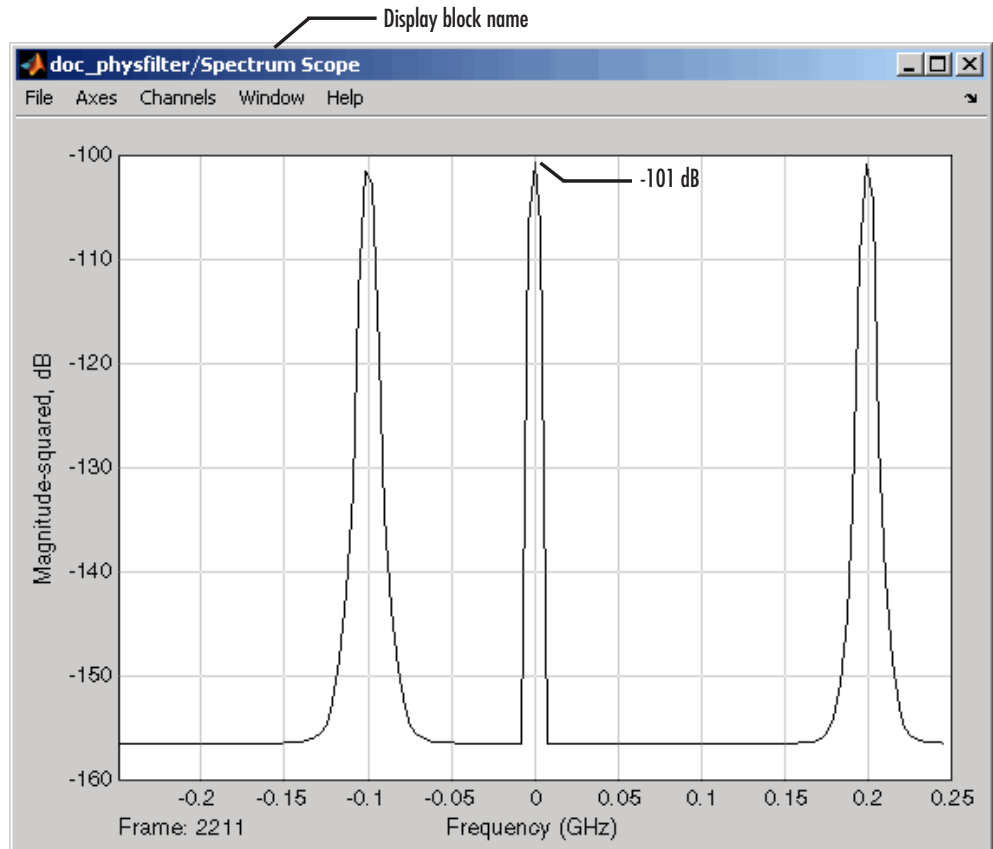
In this part of the example, you analyze the results of the simulation. This section contains the following topics:

- “Comparing the Input and Output Signals of the RF Filter” on page 1-24
- “Plotting Model Parameters of the Filter Subsystem” on page 1-27

Comparing the Input and Output Signals of the RF Filter

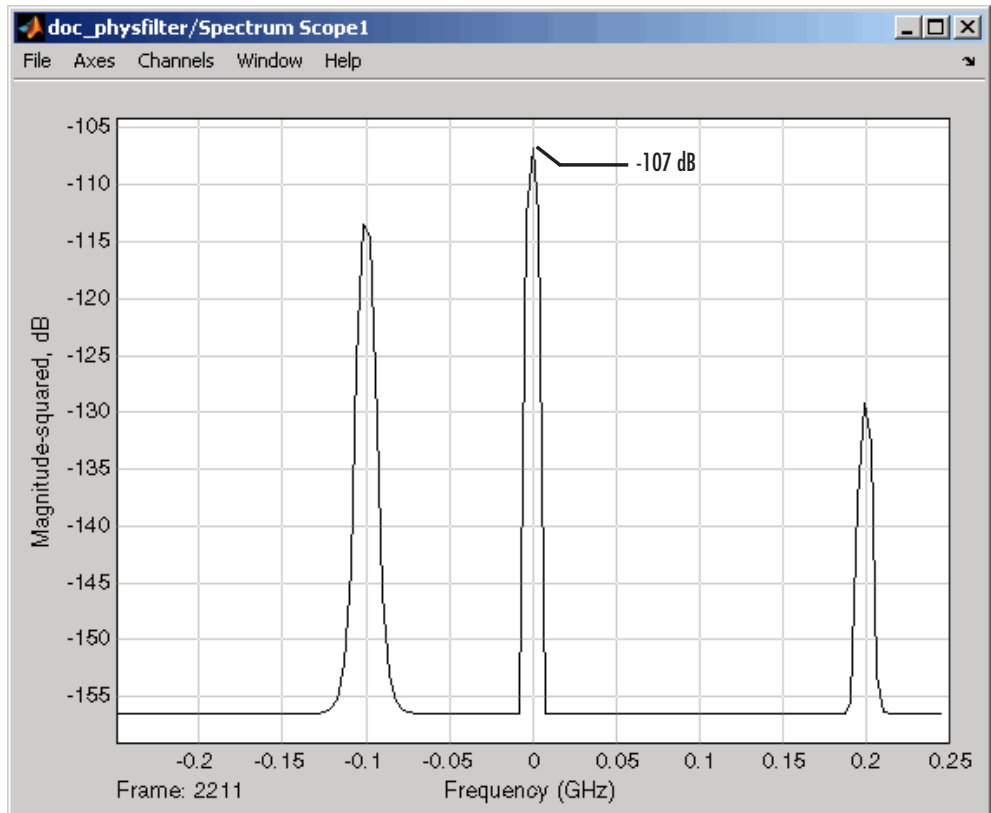
You can view the source signal and the filtered signal in the Spectrum Scope windows while the model is running. These windows appear automatically when you start the simulation.

The following plot shows the RF filter input signal you specified in the Sine Wave block.



Input to RF Filter

The next plot shows the filtered signal. Notice that the amplitude of the peaks is attenuated as a result of the RF filter.



Attenuated Output of RF Filter

Together, the Input Port and Output Port blocks introduce a 6-dB attenuation into the physical system at all frequencies. You can see this loss in the plots by comparing the input and output signals at the center frequency of the filter. The magnitude of the input signal at the center frequency approaches -101 dB in the Spectrum Scope window. The magnitude of the output signal at the center frequency approaches -107 dB in the Spectrum Scope1 window. However, as shown in the plot in the previous section, Filter Gain on page 1-23, the filter does not attenuate the signal at the center frequency. The 6 dB of loss is caused by the source and the load in the model. For more

information on why this loss occurs, see the note in “Converting to and from Simulink Signals” on page A-15.

Plotting Model Parameters of the Filter Subsystem

After you simulate an RF model, you can evaluate the behavior of the physical subsystem by plotting the network parameters of the Output Port block.

Note When you plot information about a physical subsystem, the plot displays the actual frequency response of the subsystem at the selected passband (i.e. the response at the unshifted frequencies), and *not* the response at the shifted frequencies.

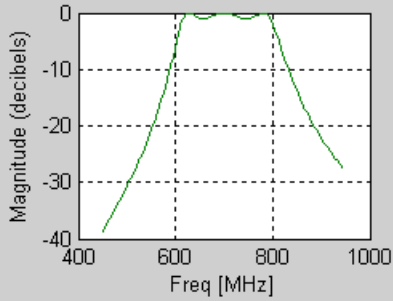
To understand the frequency response of the filter, examine the S-parameters as a function of frequency for the RF filter subsystem on a composite plot.

- 1 Open the dialog box of the Output Port block by double-clicking the block.
- 2 Select the **Plot the model parameters of this block** check box, and click **Apply**.

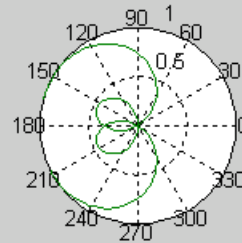
The composite plot, shown in the following figure, contains four separate plots in one figure. For the Output Port block, the composite plot shows the following as a function of frequency (counterclockwise from the upper-left plot):

- An X-Y plane plot of the magnitude of the filter gain, S_{21} , in decibels.
- An X-Y plane plot of the phase of the filter gain, S_{21} , in degrees.
- A Z Smith chart showing the real and imaginary parts of the filter reflection coefficient, S_{11} .
- A Polar plane plot showing the magnitude and phase of the filter reflection coefficient, S_{11} .

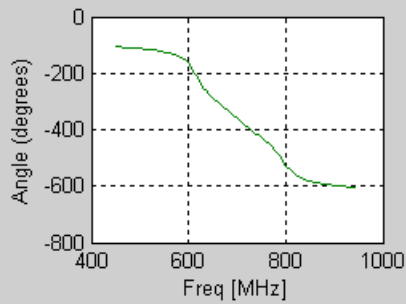
X-Y Plot,
Magnitude of S21



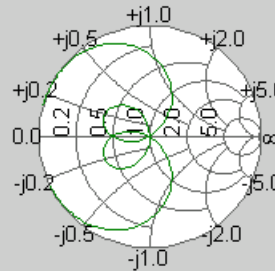
Polar Plot, S11



X-Y Plot,
Phase of S21



Z Smith Chart, S11



Modeling an RF System

| | |
|---|---|
| Modeling RF Components (p. 2-2) | Describes how to add and connect blocks in a Simulink model to represent RF components |
| Specifying or Importing Component Data (p. 2-6) | Describes how to define block behavior by entering parameter values and importing file and workspace data |
| Modeling Thermal Noise (p. 2-22) | Explains how to model thermal noise in a physical system |

Modeling RF Components

The RF Blockset lets you model an RF system using Simulink. This section describes how to build a system model that incorporates RF components and contains the following topics:

- “Adding RF Blocks to a Simulink Model” on page 2-2
- “Connecting Model Blocks” on page 2-2

Adding RF Blocks to a Simulink Model

You can include blocks from the RF Blockset Physical and Mathematical libraries in a Simulink model. For more information on the libraries and the available RF blocks, see “RF Blockset Libraries” on page 1-6.

To add RF blocks to a Simulink model:

- 1 Type `rf1ib` at the MATLAB prompt to open the RF Blockset library.
- 2 Navigate to the desired library or sublibrary.
- 3 Drag instances of RF Blockset blocks into the model window using the mouse.

Note You can also access RF Blockset blocks and other Simulink blocks from the Simulink Library Browser window. Open this window by typing `simulink` at the MATLAB prompt. Add blocks to the model by dragging them from this window and dropping them into the model window.

Connecting Model Blocks

You follow the same procedure for connecting RF Blockset blocks as for connecting Simulink blocks: you click a port and drag the mouse to draw a line to another port on a different block. For more information on connecting Simulink blocks, see “Connecting the Blocks” in the Simulink documentation.

You can only connect blocks that use the same type of signal. RF Blockset Physical blocks use different types of signals than Mathematical blocks, and

are represented graphically by a different port style. Therefore, you can freely connect pairs of Mathematical modeling blocks. You can also freely connect pairs of Physical modeling blocks. However, you cannot directly connect Physical blocks to Mathematical blocks. Instead, you must use the Input Port and Output Port blocks to bridge them.

For more information on the RF Blockset libraries, including how to open the libraries and a description of the available blocks, see “RF Blockset Libraries” on page 1-6.

This section contains the following topics:

- “Connecting Mathematical Blocks” on page 2-3
- “Connecting Physical Blocks” on page 2-4
- “Bridging Physical and Mathematical Blocks” on page 2-4

Connecting Mathematical Blocks

The RF Blockset Mathematical blocks use the same input and output ports as standard Simulink blocks. These ports show the direction of the signal at the port, as shown in the following diagram.

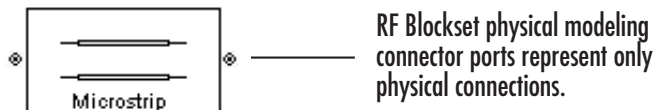


Similar to standard Simulink blocks, you draw lines between the ports of the Mathematical modeling blocks, called *signal lines*, to represent signals that are inputs to and outputs from the mathematical functions represented by the blocks. Therefore, you can connect Simulink, Signal Processing, and RF Blockset mathematical blocks by drawing signal lines between their ports.

You can connect a port to multiple ports by branching the signal line, or you can leave a port unconnected. For more information on connecting Simulink blocks, see “Connecting the Blocks” in the Simulink documentation.

Connecting Physical Blocks

The RF Blockset Physical blocks have specialized *connector ports*. These ports only represent physical connections; they do not imply signal direction.



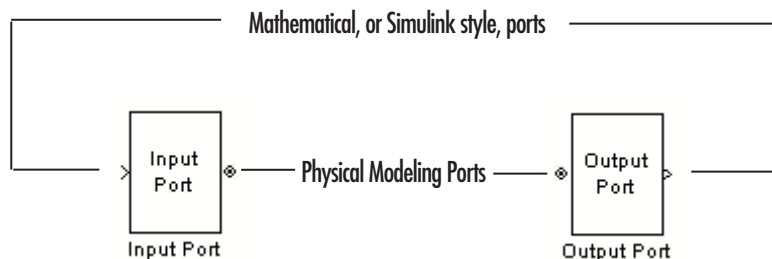
The lines you draw between the RF Blockset physical modeling blocks, called *connection lines*, represent physical connections among the block components. Connection lines appear as solid black when connected and as dashed red lines when either end is unconnected.

You can draw connection lines only between the connector ports of physical modeling blocks. You cannot branch these connection lines. You cannot leave connector ports unconnected.

Bridging Physical and Mathematical Blocks

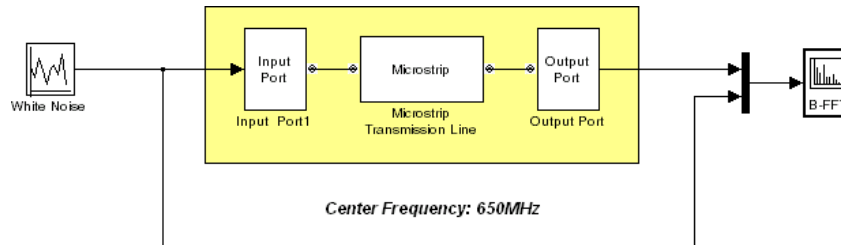
The RF Blockset provides the Input Port and Output Port blocks to connect the physical and mathematical parts of the model. These blocks convert mathematical signals to and from the physical modeling environment.

The Input Port and Output Port blocks have one of each kind of connector port: a standard Simulink style input port and a physical modeling port. These ports are shown in the following figure:



The Input Port and Output Port blocks must bound a physical subsystem to connect it to the mathematical part of a model.

For example, a simple RF model of a coaxial transmission line might resemble the following figure.



The Microstrip Transmission Line block uses an Input Port block to get its white noise input from a Random Source block, and an Output Port block to pass its output to a Spectrum Scope block. The Random Source and Spectrum Scope blocks are from Signal Processing Blockset library.

For information on how the RF Blockset converts mathematical signals to and from the physical modeling environment, see “Converting to and from Simulink Signals” on page A-15.

Specifying or Importing Component Data

You can specify RF Blockset blocks by entering the RF component parameter values. The RF Blockset also lets you import

- Industry-standard file formats — Touchstone S2P, Y2P, Z2P, and H2P formats specify the network parameters and noise information for measured and simulated data.

For more information on Touchstone files, see http://www.vhdl.org/pub/ibis/connector/touchstone_spec11.pdf.

- MathWorks amplifier (AMP) file format — Specifies amplifier network parameters, power data, noise data, and third-order intercept point

For more information about .amp files, see “AMP File Format” in the RF Toolbox documentation.

- MATLAB circuits — RF Toolbox circuit objects in the MATLAB workspace specify network parameters, noise data, and third-order intercept point information of circuits with different topologies.

For more information about RF circuit objects, see the RF Toolbox `rfckt` reference page.

This section contains the following topics:

- “Specifying Parameter Values” on page 2-6
- “Importing Data Files into RF Blocks” on page 2-7
- “Example — Importing a Touchstone Data File into an RF Model” on page 2-8
- “Importing Circuits from the MATLAB Workspace” on page 2-14
- “Example — Importing a Bandstop Filter into an RF Model” on page 2-15

Specifying Parameter Values

There are two ways to set block parameter values:

- Using the GUI — Enter information in the block dialog boxes, which open when you double-click a block in the Simulink window.

- Using commands — Use the Simulink `set_param` and `get_param` commands to set and get parameter values of the blocks, respectively. For more information on these commands, see the `set_param` and `get_param` reference pages.

Importing Data Files into RF Blocks

The RF Blockset lets you import industry-standard data files and MathWorks AMP files into specific blocks to simulate the behavior of measured components in Simulink.

This section contains the following topics:

- “Blocks Used to Import Data” on page 2-7
- “How to Import Data Files” on page 2-7

Blocks Used to Import Data

Three blocks in the Physical library accept data from a file. The following table lists the blocks and any corresponding data format that each supports.

| Block | Description | Supported Format(s) |
|-------------------------|---------------------------|---------------------|
| General Amplifier | Generic amplifier | Touchstone, AMP |
| General Mixer | Generic mixer | Touchstone, AMP |
| General Passive Network | Generic passive component | Touchstone |

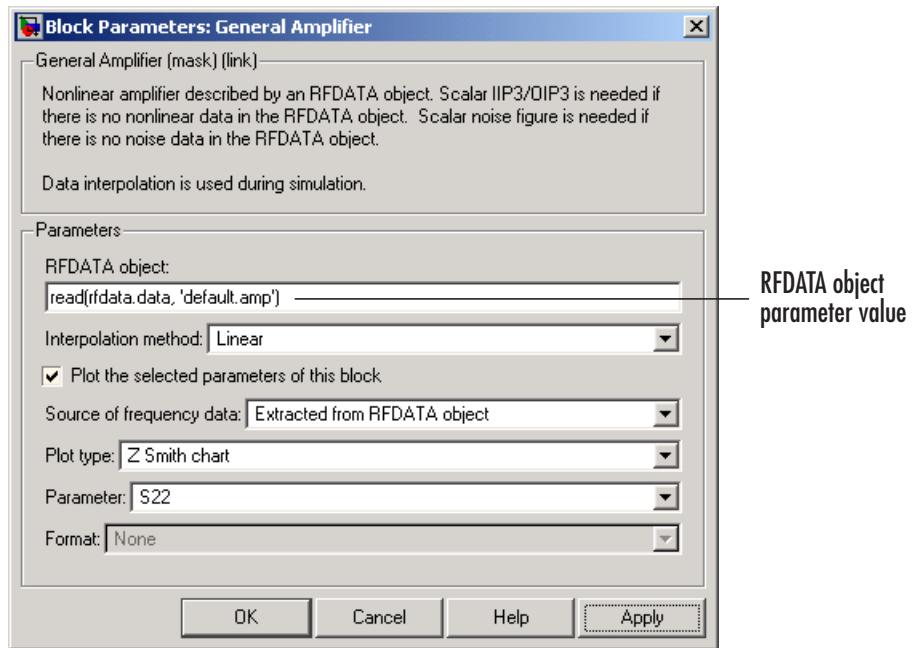
How to Import Data Files

To import a data file:

- 1 Choose the block that best represents your component from the list of blocks that accept file data shown in “Blocks Used to Import Data” on page 2-7.
- 2 Open the RF Blockset Physical library, and navigate to the sublibrary that contains the block.
- 3 Click and drag the block into your Simulink model.

- 4 In the block dialog box, enter a value of `read(rfdata.data, 'filename')` for the **RFCKT object** parameter, where *filename* is the name of your data file.

This value specifies that the RF Toolbox read function reads data from *filename* into an RF Toolbox data object. The RF Blockset uses this data object to simulate the data. For more information on RF Toolbox data objects, see the `rfdata` reference page.



This procedure is illustrated by example in the following section.

Example – Importing a Touchstone Data File into an RF Model

In this example, you simulate the frequency response of a passive component using data from a Touchstone file, `passive.s2p`.

You use a model from one of the RF Blockset demos to perform the following tasks:

- “Importing Data into a General Passive Network Block” on page 2-9
- “Validating the Passive Component” on page 2-11
- “Running the Simulation and Analyzing the Results” on page 2-13

Importing Data into a General Passive Network Block

In this part of the example, you inspect the passive.s2p file and import data into the RF model using the General Passive Network block.

- 1 Type the following at the MATLAB prompt to open the passive.s2p file:

```
edit passive.s2p
```

The following figure shows a portion of the .s2p file.

```

1 # Hz S DB R SO
2 ! S-Parameters data
3 ! FREQ dbS11 angS11 dbS21 angS21 dbS1
4 315074.664 -32.010394 81.245846 -0.028574
5 330906.814 -31.591401 81.414291 -0.030716
6 347534.511 -31.172839 81.595851 -0.034711
7 364997.732 -30.760176 81.829922 -0.029644
8 383338.459 -30.327674 82.009124 -0.029974
9 402600.788 -29.907735 82.225833 -0.035568
10 422831.028 -29.488195 82.366016 -0.036152
11 444077.814 -29.064399 82.418274 -0.036146

```

The option line

```
# Hz S DB R 50
```

specifies the following information about the contents of the data file:

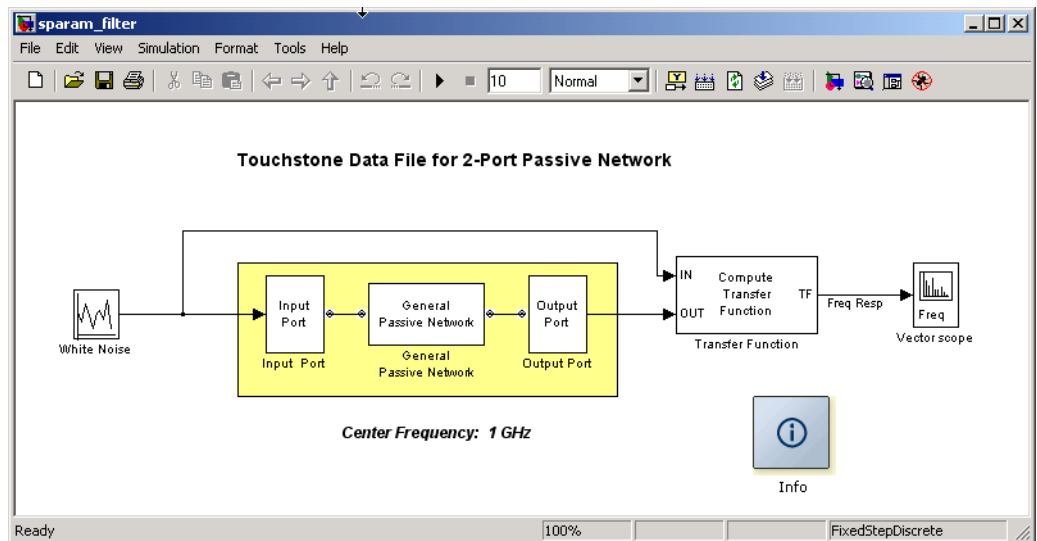
- Hz — Frequency units.
- S — Network parameters are S-parameters.
- DB — Network parameters are specified as magnitude in dB and phase in degrees.
- R 50 — Reference impedance is 50 ohms.

For more information about the Touchstone specification, including the option line, see http://www.vhdl.org/pub/ibis/connector/touchstone_spec11.pdf.

2 At the MATLAB prompt, type

```
sparam_filter
```

This opens the RF Blockset demo called “Touchstone Data File for 2-Port Passive Networks.”



- 3 Double-click the General Passive Network block to display its parameters.

The **RFDATA object** parameter specifies the data file to import.

The **RFDATA object** parameter is set to `read(rfdata.data, 'passive.s2p')`. This setting reads the data from the file `passive.s2p` into an `rfdata.data` object. The block uses this data with the other block parameters during simulation.

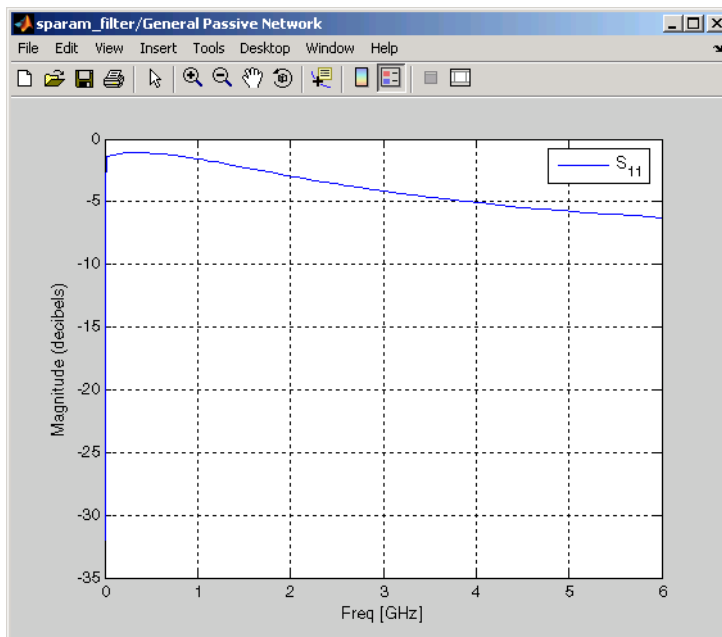
Note When the imported file contains data that is measured at frequencies other than the modeling frequencies, use the **Interpolation method** parameter to specify how the block determines the data values at the modeling frequencies. For more information, see “Determining the Modeling Frequencies” on page A-3 and “Mapping Network Parameters to Modeling Frequencies” on page A-5.

Validating the Passive Component

In this part of the example, you plot the network parameters of the General Passive Network block to validate the data you imported in “Importing Data into a General Passive Network Block” on page 2-9.

- 1 Open the General Passive Network block dialog box, and perform the following actions:
 - a Select the **Plot the selected parameters of this block** check box.
 - b Set the **Parameter** parameter to S11.
 - c Click **Apply**.

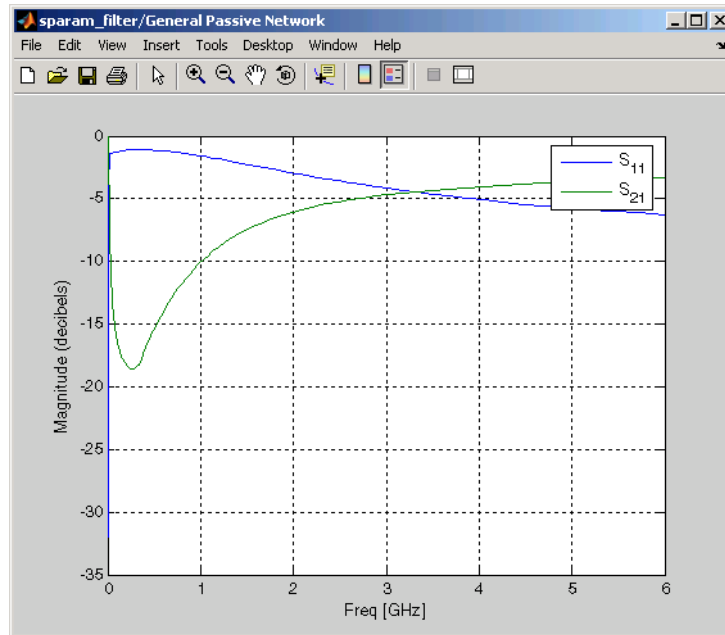
These dialog box selections create a plot of S11 as a function of frequency.



S11 versus Frequency for the Imported Data

- 2 Open the General Passive Network block dialog box, set the **Parameter** parameter to S21, and click **Apply**.

The block adds the S21 data to the plot.

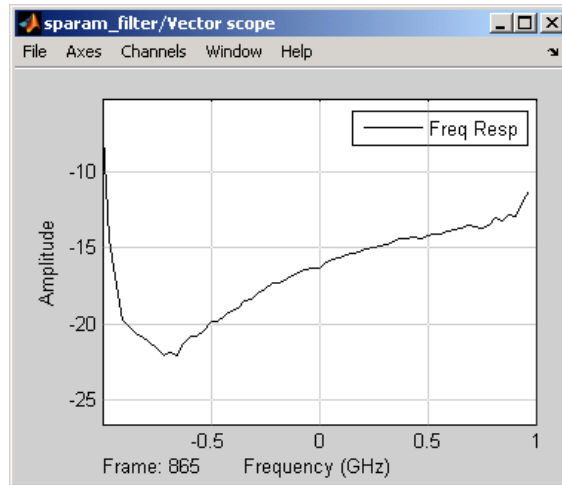


S11 and S21 versus Frequency for the Imported Data

Running the Simulation and Analyzing the Results

In this part of the example, you run the simulation and examine the frequency response of the passive component.

Start the simulation by selecting **Simulation > Start** in the Simulink model window. This action opens the Vector scope plot, which displays the amplitude of the transfer function of the system in decibels as a function of frequency, as shown in the following figure:



Transfer Function of General Passive Network Subsystem

Note The transfer function differs from S21. The differences arise from the following factors:

- The 6 dB of loss introduced by the source and the load in the `sparam_filter` model. This loss is described in the note in “Converting to and from Simulink Signals” on page A-15.
 - The numerical error introduced by the calculation of the transfer function.
-

Importing Circuits from the MATLAB Workspace

You can only connect the RF Blockset Physical blocks in cascade. However, the RF Blockset works with the RF Toolbox to let you include additional circuit topologies in an RF model. To model circuit topologies that contain other types of connections, you must define a circuit in the MATLAB workspace and import it into an RF model.

To import a circuit from the MATLAB workspace:

- 1 Define the circuit object in the MATLAB workspace using the RF Toolbox functions.

For more information about RF circuit objects, see the RF Toolbox `rfckt` reference page.

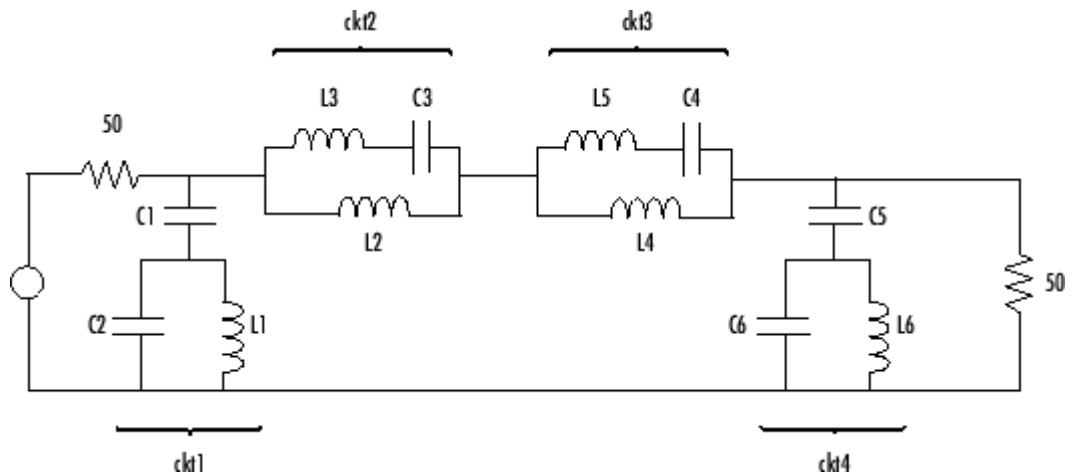
- 2 Add a General Circuit Element block to your RF model from the Black Box Elements sublibrary of the Physical library. For information on how to open this library, see “Opening RF Blockset Libraries” on page 1-6.
- 3 Enter the circuit object name in the **RFCKT object** parameter in the General Circuit Element block dialog box.

This procedure is illustrated by example in the following section.

Example – Importing a Bandstop Filter into an RF Model

In this example, you simulate the frequency response of a filter that you model using circuit objects from the MATLAB workspace.

The filter in this example is the 50-ohm bandstop filter shown in the following figure.



Bandstop Filter Diagram

You represent the filter using four circuit objects that correspond to the four parts of the filter, `ckt1`, `ckt2`, `ckt3`, and `ckt4` in the diagram. You use an input signal with random, complex input values that have a Gaussian distribution to stimulate the filter. The scope block displays the output signal.

This example illustrates how to perform the following tasks:

- “Creating Circuit Objects in the MATLAB Workspace” on page 2-16
- “Building the Model” on page 2-17
- “Specifying and Importing Component Data” on page 2-18
- “Running the Simulation and Plotting the Results” on page 2-20

Creating Circuit Objects in the MATLAB Workspace

In this part of the example, you define MATLAB variables to represent the physical properties of the filter shown in the previous figure, Bandstop Filter Diagram on page 2-15, and use RF Toolbox functions to create RF circuit objects that model the filter components.

- 1 Type the following at the MATLAB prompt to define the filter’s resistance, capacitance, and inductance values in the MATLAB workspace:

```
C1 = 1.734e-12;  
C2 = 4.394e-12;  
C3 = 7.079e-12;  
C4 = 7.532e-12;  
C5 = 1.734e-12;  
C6 = 4.394e-12;  
L1 = 25.70e-9;  
L2 = 3.760e-9;  
L3 = 17.97e-9;  
L4 = 3.775e-9;  
L5 = 17.63e-9;  
L6 = 25.70e-9;
```

- 2** Type the following at the MATLAB prompt to create RF circuit objects that model the components labeled ckt1, ckt2, ckt3, and ckt4 in the circuit diagram:

```

ckt1 =
rfckt.series('Ckts',{rfckt.shuntrlc('C',C1),rfckt.shuntrlc('L'
,L1,'C',C2)});
ckt2 =
rfckt.parallel('Ckts',{rfckt.seriesrlc('L',L2),rfckt.seriesrlc
('L',L3,'C',C3)});
ckt3 =
rfckt.parallel('Ckts',{rfckt.seriesrlc('L',L4),rfckt.seriesrlc
('L',L5,'C',C4)});
ckt4 =
rfckt.series('Ckts',{rfckt.shuntrlc('C',C5),rfckt.shuntrlc('L'
,L6,'C',C6)});

```

For more information about the RF Toolbox functions used in this example, see the `rfckt.series`, `rfckt.parallel`, `rfckt.shuntrlc`, and `rfckt.seriesrlc` function reference pages in the RF Toolbox documentation.

Building the Model

In this portion of the example, you create a Simulink model. For more information about adding and connecting components, see “Modeling RF Components” on page 2-2.

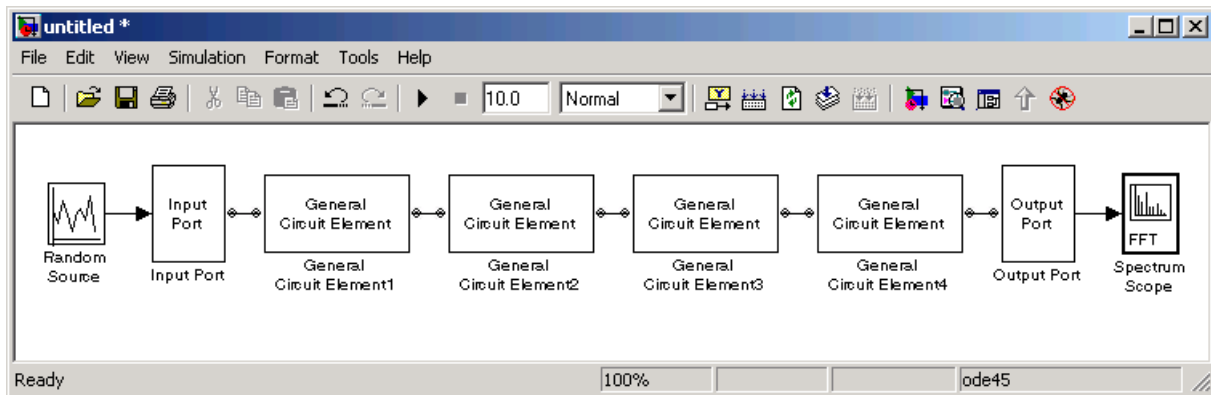
- 1** Create a new Simulink model.
- 2** Add to the model the blocks shown in the following table. The Library column of the table specifies the hierarchical path to each block.

| Block | Library | Quantity |
|---------------|--|----------|
| Random Source | Signal Processing Blockset > Signal Processing Sources | 1 |

| Block | Library | Quantity |
|-------------------------|--|----------|
| Input Port | RF Blockset > Physical > Input/Output Ports | 1 |
| General Circuit Element | RF Blockset > Physical > Black Box Elements | 4 |
| Output Port | RF Blockset > Physical > Input/Output Ports | 1 |
| Spectrum Scope | Signal Processing Blockset > Signal Processing Sinks | 1 |

3 Connect the blocks as shown in the following figure.

Change the names of your General Circuit Element blocks to match those in the figure by double-clicking the text below the block and typing a new name.



Specifying and Importing Component Data

In this portion of the example, you specify block parameters. To open the parameter dialog box for each block, double-click the block.

1 Set the Random Source block parameters as follows:

- **Source type** = Gaussian
- **Sample time** = $1 / (100e6)$
- **Samples per frame** = 256
- **Complexity** = Complex

Selecting these settings creates an input signal with random, complex input values that have a Gaussian distribution.

2 Set the Input Port block parameters as follows:

- **Finite impulse response filter length** = 256
- **Center frequency (Hz)** = $400e6$
- **Sample time** = $1 / (100e6)$
- **Source impedance** = 50
- Clear the **Add noise** check box.

Selecting these settings defines the physical characteristics and modeling bandwidth of the filter.

3 Set the parameters of the General Circuit Element blocks as follows:

- In the General Circuit Element1 block dialog box, set the **RFCKT object** parameter to ckt1.
- In the General Circuit Element2 block dialog box, set the **RFCKT object** parameter to ckt2.
- In the General Circuit Element3 block dialog box, set the **RFCKT object** parameter to ckt3.
- In the General Circuit Element4 block dialog box, set the **RFCKT object** parameter to ckt4.

Selecting these settings imports the circuit objects that model the filter components into the Simulink model.

4 Type 50 in the **Load impedance** field of the Output Port block to represent an impedance of 50 ohms.

5 Set the Spectrum Scope block parameters as follows:

- On the **Scope Properties** pane, set the **Number of spectral averages** parameter to 100.

This parameter establishes the number of spectra that the scope averages to produce the displayed signal. You use a value of 100 because the input signal is random and you want to display the average filter response over a large number of input values.

- On the **Axis Properties** pane, set the **Frequency range** parameter to $[-F_s/2 \dots F_s/2]$, the **Minimum Y-limit** parameter to -35, and the **Maximum Y-limit** parameter to 3.6.

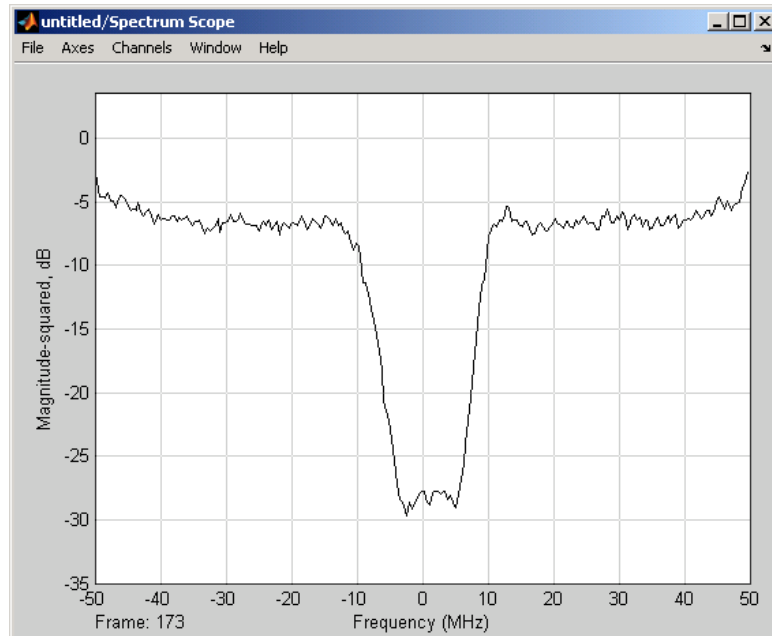
These values set the range of X- and Y-values on the display such that the entire signal is visible when you run the simulation.

Running the Simulation and Plotting the Results

In this part of the example, you run the simulation and examine the frequency response of the filter.

Select **Simulation > Start** in the Simulink model window to start the simulation.

The Spectrum Scope window appears automatically and displays the following plot, which shows the frequency response of the filter.



Frequency Response of Bandstop Filter

References

Geffe, P.R., "Novel designs for elliptic bandstop filters," *RF Design*, February 1999.

Modeling Thermal Noise

The RF Blockset lets you include the thermal noise generated by any physical block in your RF model. You only need to specify noise information for the physical amplifier and mixer blocks that generate noise other than resistor noise. For the other blocks, the RF Blockset calculates the noise automatically based on the resistor values.

This section contains the following topics:

- “Amplifier and Mixer Noise Specifications” on page 2-22
- “Adding Noise to Your System” on page 2-23
- “Plotting Noise” on page 2-24

Amplifier and Mixer Noise Specifications

You specify noise for the physical amplifier and mixer blocks as spot noise data or as noise figure values. The following table summarizes the noise specification options for each type of physical amplifier and mixer block.

| Block | Noise Specification |
|------------------------|--|
| General Amplifier | Spot noise data (using a Touchstone or AMP data file) or noise figure values |
| S-Parameters Amplifier | Noise figure values |
| Y-Parameters Amplifier | Noise figure values |
| Z-Parameters Amplifier | Noise figure values |
| General Mixer | Spot noise data (using a Touchstone or AMP data file) or noise figure values |
| S-Parameters Mixer | Noise figure values |
| Y-Parameters Mixer | Noise figure values |
| Z-Parameters Mixer | Noise figure values |

Adding Noise to Your System

To simulate the thermal noise of a physical subsystem, you perform the following tasks:

- “Specifying or Importing Noise Data” on page 2-23
- “Adding Noise to the Simulation” on page 2-24

Specifying or Importing Noise Data

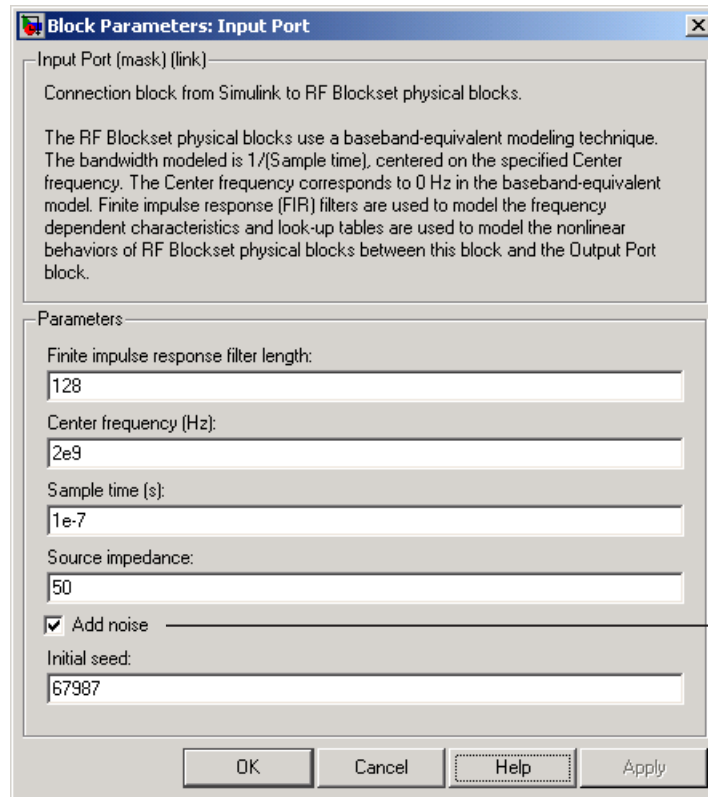
The method you use to add noise data to a block depends on whether you are specifying noise figure or importing spot-noise data. The following table provides instructions for adding noise data.

| Noise Specification | Instructions |
|---------------------|--|
| Noise figure | Enter the noise figure value in the Noise figure (dB) parameter in the block dialog box. |
| Spot noise data | Import file data that includes noise information into the RFCKT object parameter of the General Amplifier or General Mixer block. |

Note If you import file data with no noise information into a General Amplifier or General Mixer block, the RF Blockset adds the **Noise figure (dB)** parameter to the block dialog box . This parameter lets you specify noise manually.

Adding Noise to the Simulation

To include noise in the simulation, you must select the **Add noise** check box on the Input Port block dialog box. This check box is selected by default.



Select this check box to take the noise data in the physical blocks into account. This check box is selected by default.

For information on how the RF Blockset simulates thermal noise, see “Modeling Noise in an RF System” on page A-7.

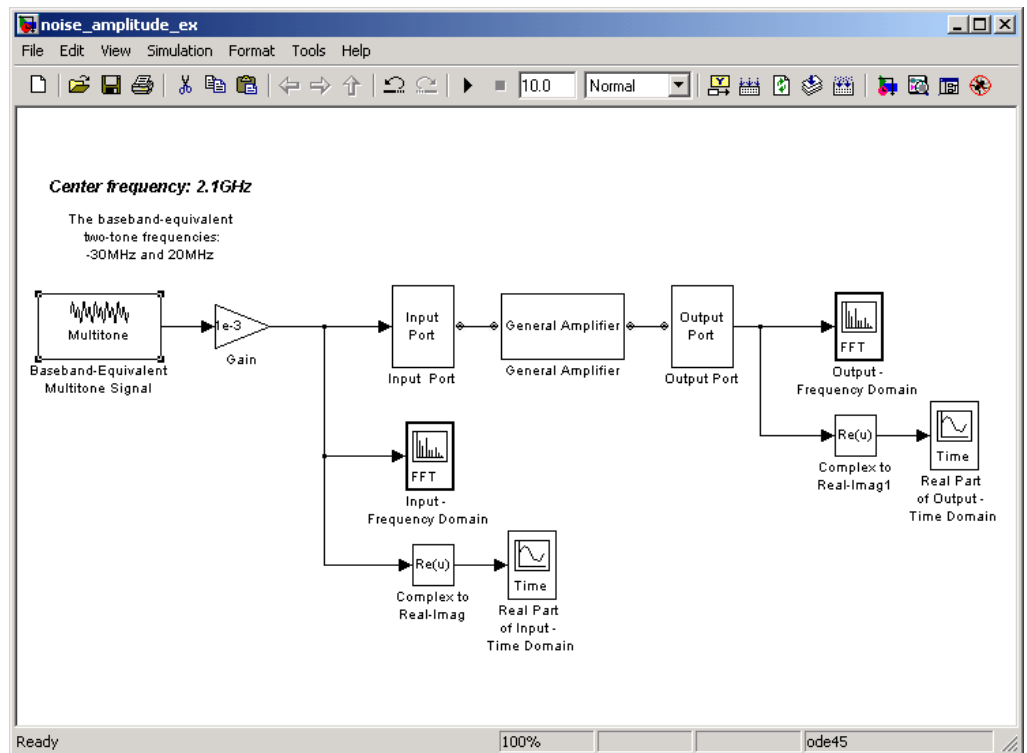
Plotting Noise

The RF Blockset models communications systems. The noise in these systems has a very small amplitude, typically from $1e-6$ to $1e-12$ Watts. In contrast, the default signal power of a Communications Blockset modulator block is 1 Watt at a nominal 1 ohm. Therefore, the signal-to-noise ratio in an RF system

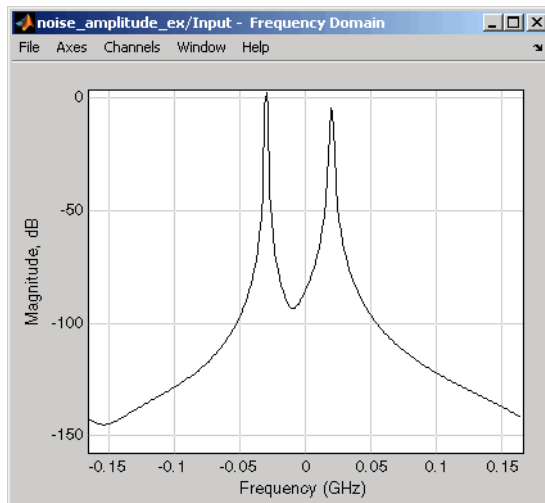
simulation is large, making it difficult to view the noise the RF Blockset adds to your signal.

To display the noise on a plot, you might need to attenuate the signal amplitude to a value within a couple orders of magnitude of the noise.

For example, suppose you have the following model that contains a multitone test signal source.

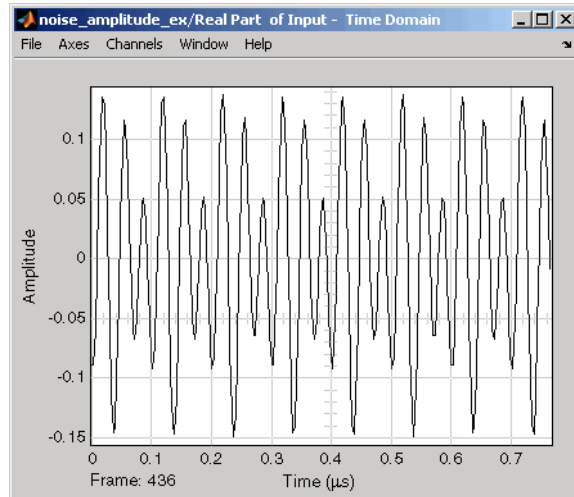


When you simulate this model, Simulink brings up several windows showing the input and output for the physical subsystem. The Input - Frequency Domain window shown in the following figure displays the input signal in the frequency domain.



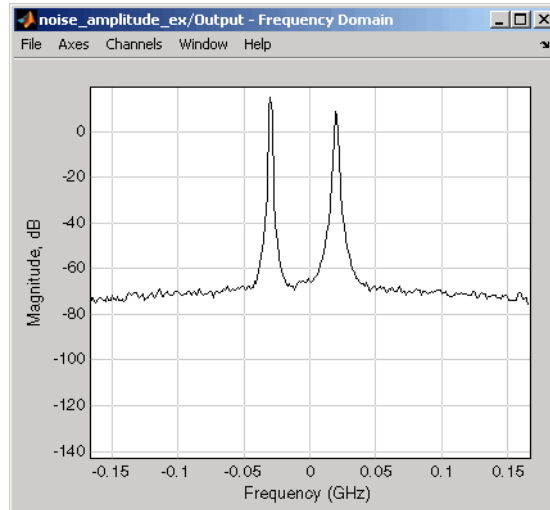
Input Signal Spectrum

The Real Part of Input - Time Domain window displays the real part of the complex-valued input signal in the time domain.



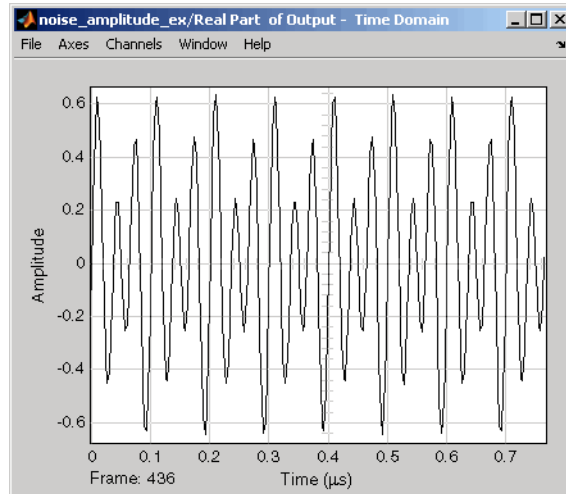
Real Part of Input Signal

In the model, the physical subsystem adds noise to the input signal. The Output - Frequency Domain window shows the noisy output signal in the frequency domain.



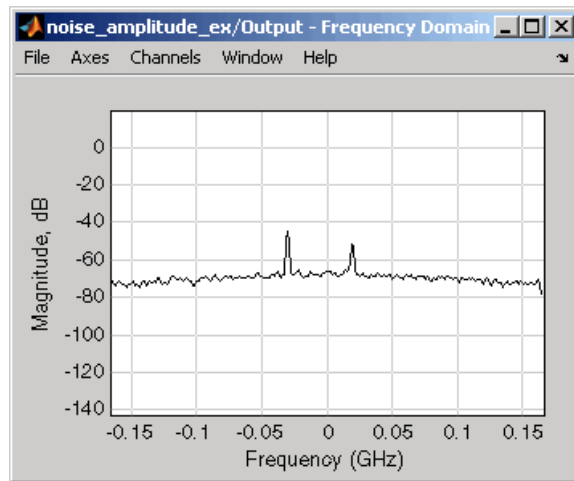
Output Signal Spectrum

The amplitude of the signal is large compared to the amplitude of the noise, so the noise is not visible in the Real Part of Output - Time Domain window that shows the real part of the time-domain output signal. Therefore, you must attenuate the amplitude of the input signal to display the noise of the time-domain output signal.



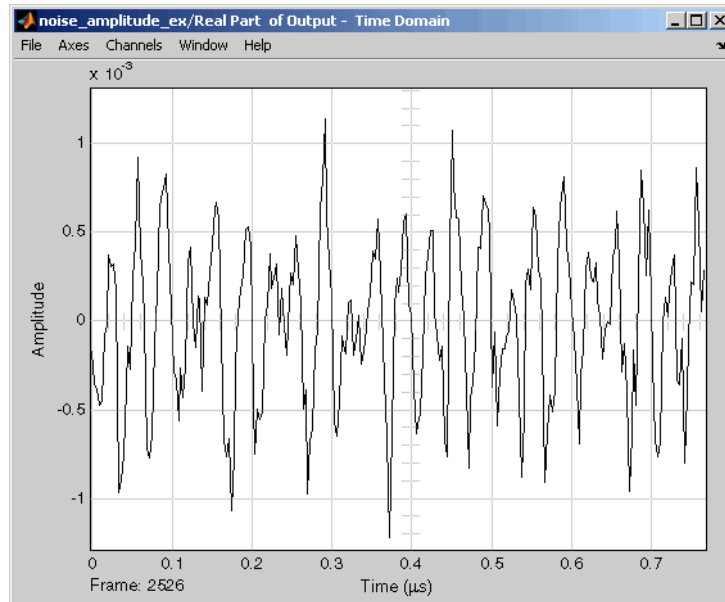
Real Part of Output Signal

Attenuate the amplitude of the input signal by setting the **Gain** parameter to $1e-3$. This is equivalent to attenuating the input signal by 60 dB. When you run the model again, the two signal peaks are not as pronounced in the Output - Frequency Domain window.



Output Signal Spectrum for Attenuated Input

You can now view the noise the RF Blockset adds to your signal in the Real Part of Output - Time Domain window.



Real Part of Output Signal Showing Noise

Plotting Model Data

| | |
|--|--|
| Creating Plots (p. 3-2) | Describes the available plots and explains how to plot data for components and subsystems |
| Updating Plots (p. 3-19) | Describes how to update existing plots after changing model parameters |
| Modifying Plots (p. 3-20) | Describes how to add data to a plot and how to change the plot type, format, and frequency range |
| Example — Creating and Modifying Subsystem Plots (p. 3-22) | Shows how to plot RF subsystem data, add data to the plot, and change the plot type |

Creating Plots

The RF Blockset lets you validate the behavior of individual RF components and physical subsystems in your model by plotting the following data:

- S-parameters
- Noise figure
- Output third-order intercept point
- Power data
- Phase noise
- Voltage standing-wave ratio

The following topics describe the RF Blockset plotting options, explain the procedure for creating a plot, and illustrate this procedure with an example:

- “Validating Individual Blocks and Subsystems” on page 3-2
- “Types of Plots” on page 3-3
- “Plot Formats” on page 3-4
- “How to Create a Plot” on page 3-9
- “Example — Plotting Component Data on a Z Smith Chart” on page 3-15

Validating Individual Blocks and Subsystems

You can plot model data for an individual physical block or for a physical subsystem. A *subsystem* is a collection of one or more physical blocks bracketed by an Input Port block and an Output Port block. To gain insight into the behavior of specific subsystems, plot the data of the corresponding Output Port block after you run a simulation.

To validate the behavior of individual RF components in the model, plot the data of the corresponding physical blocks. You can plot data for individual blocks from each of these components either before or after you run a simulation.

You create a plot by selecting options in the block dialog box, as shown in “Example — Creating and Modifying Subsystem Plots” on page 3-22. To

learn about the available plots, see “Types of Plots” on page 3-3. For more information about creating plots, see “How to Create a Plot” on page 3-9.

Types of Plots

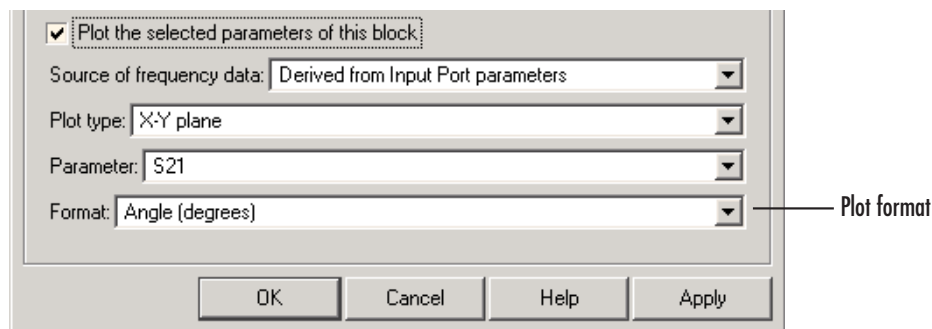
The RF Blockset provides a variety of plots for analyzing the behavior of RF components and subsystems. The following table summarizes the available plots and charts and describes each one.

| Plot Type | Plot Contents |
|------------------------------|---|
| X-Y Plane (Rectangular) Plot | Parameters as a function of frequency, such as <ul style="list-style-type: none"> • S-parameters • Noise figure (NF) • Voltage standing-wave ratio (VSWR) • Output third-order intercept point (OIP3) |
| Link Budget Plot (3-D) | Parameters as a function of frequency for each component in a physical subsystem, where the curve for a given component represents the cumulative contribution of each RF component up to and including the parameter value of that component. For more information, see “Link Budget” on page 3-7. |
| Polar Plane Plot | Magnitude and phase of S-parameters as a function of frequency. |
| Smith Chart | Real and imaginary parts of S-parameters as a function of frequency, used for analyzing the reflections caused by impedance mismatch. |
| Composite Plot | Multiple plots and charts in one figure. |

To learn how to create these plots, see “How to Create a Plot” on page 3-9.

Plot Formats

When you create a plot from a block dialog box, you must specify the plot **Format**.



This plot option defines how the RF Blockset displays the data on the plot.

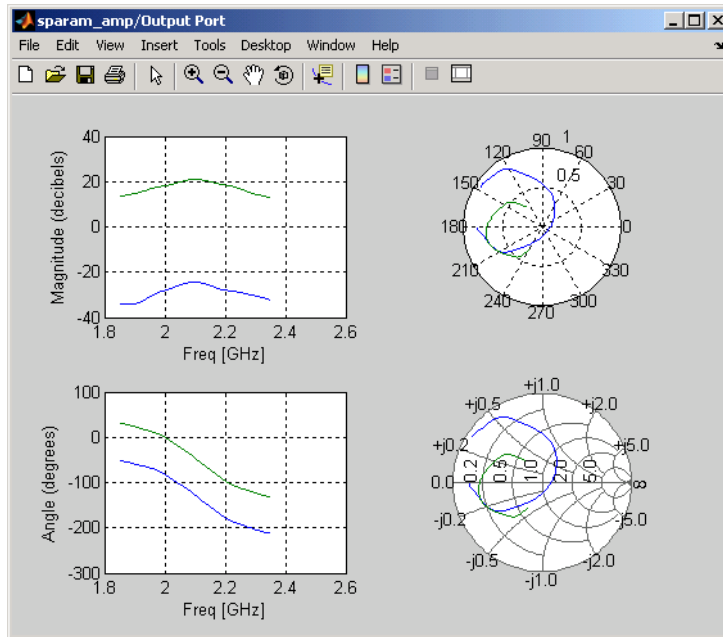
The available formats vary with the data you select to plot. The data you can plot depends on the plot type you select. The plot format determines whether the RF Blockset converts the data to a new set of units, or performs a calculation on the data. For example, setting the format to Real tells the RF Blockset to compute and plot the real part of the selected parameter.

The following topics describe the available parameters and formats for each plot type:

- “Composite Data” on page 3-5
- “X-Y Plane” on page 3-7
- “Link Budget” on page 3-7
- “Polar Plane Plots and Smith Charts” on page 3-9

Composite Data

The composite data plot automatically generates four separate plots in one figure window, showing the frequency dependence of several parameters. The following figure shows an example of such a plot.



Example – Composite Data Plot

Note For composite data plots, you do not need to specify the parameters or the plot format—both are set automatically.

The combination of plots differs based on the type of block and the specified block data. The following table describes the contents of the composite data plot for each specification. The Plot Contents column lists the types of plots as they appear on the composite plot, counterclockwise and starting in the upper-left corner.

| Block | Specified Data | Plot Contents |
|------------------------------------|--|--|
| General Amplifier or General Mixer | Network parameters OR Network parameters and noise | X-Y plot, magnitude of S12 and S21 in decibels X-Y plot, phase of S12 and S21 in degrees Z Smith chart, real and imaginary parts of S11 and S22 Polar plot, magnitude and phase of S11 and S22 |
| | Network parameters and power OR Network parameters, noise, and power | X-Y plot, magnitude of S12 and S21 in decibels X-Y plot, output power (P_{out}) in dBm (decibels referenced to one milliwatt) Z Smith chart, real and imaginary parts of S11 and S22 Polar plot, magnitude and phase of S11 and S22 |
| Other Physical block | Network parameters Note Only the General Amplifier and General Mixer blocks accept power and noise data. | X-Y plot, magnitude of S12 and S21 in decibels X-Y plot, phase of S12 and S21 in degrees Z Smith chart, real and imaginary parts of S11 and S22 Polar plot, magnitude and phase of S11 and S22 |

X-Y Plane

You can plot any parameters that are relevant to your block on an X-Y plane plot. The following table summarizes the parameters and formats for this type of plot.

| Parameter | Format |
|--|--|
| S11, S12, S21, S22 | Magnitude (decibels) Magnitude (linear) Angle (degrees) Real Imaginary |
| VSWRIn, VSWROut | Magnitude (decibels) Magnitude (linear) |
| OIP3 (Output Port block only) | dBm W mW |
| NF (Output Port block only) | Magnitude (decibels) Magnitude (linear) |
| Pout (General Amplifier block with power data only) | dBm dBW W mW |
| Phase (General Amplifier block with power data only) | Angle (degrees) Angle (radians) |
| AM/AM (General Amplifier block with power data only) | Magnitude (decibels) Magnitude (linear) |
| AM/PM (General Amplifier block with power data only) | Angle (degrees) Angle (radians) |
| PhaseNoise (Mixer blocks only) | dBc/Hz |

Link Budget

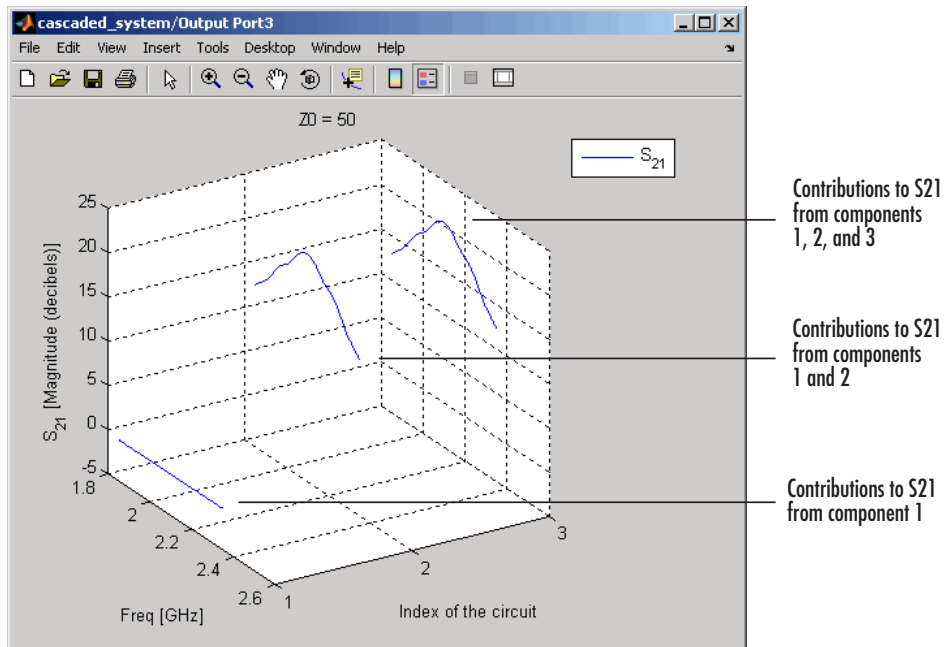
You use the Link budget plot to understand the individual contribution of each block to a plotted parameter value in a physical subsystem with multiple components between the Input Port and the Output Port blocks.

The link budget plot is a three-dimensional plot that shows one or more curves of parameter values as a function of frequency, ordered by the subsystem circuit index.

The following figure shows how the circuit index is assigned to a component in a physical subsystem based on its sequential position in the subsystem.



A curve on the link budget plot for each circuit index represents the contributions to the parameter value of the RF components up to that index. The following figure shows an example of a link budget plot.



Example – Link Budget Plot

The following table summarizes the parameters and formats that are available for a link budget plot.

| Parameter | Format |
|--------------------|--|
| S11, S12, S21, S22 | Magnitude (decibels) Magnitude (linear) Angle (degrees) Real Imaginary |
| VSWRIn, VSWROut | Magnitude (decibels) Magnitude (linear) |
| OIP3 | dBm W mW |
| NF | Magnitude (decibels) Magnitude (linear) |

Polar Plane Plots and Smith Charts

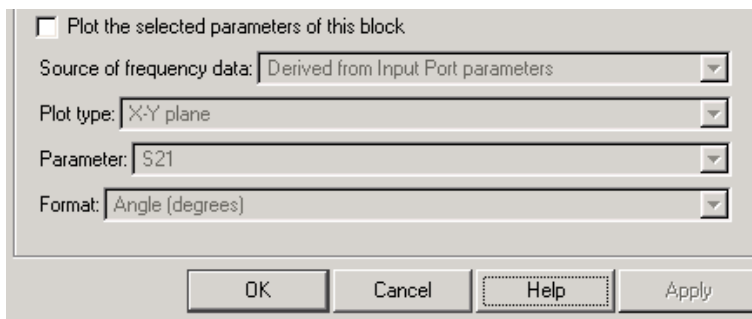
You can use the RF Blockset to generate Polar plots and Smith charts. When you select these plot types, you do not need to specify the plot format—it is set automatically.

The following table describes the Polar plot and Smith chart options, as well as the available parameters.

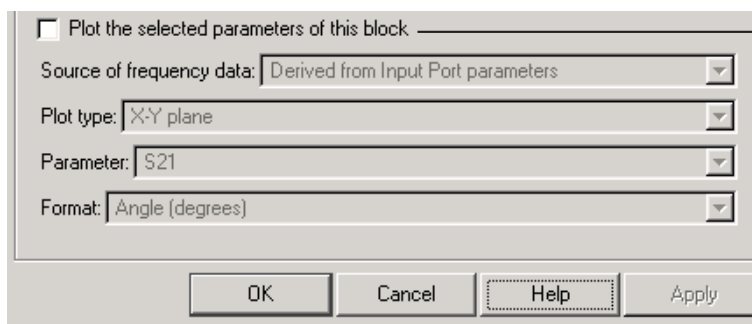
| Plot Type | Parameter |
|----------------|--------------------|
| Polar plane | S11, S12, S21, S22 |
| Z Smith chart | S11, S22 |
| Y Smith chart | S11, S22 |
| ZY Smith chart | S11, S22 |

How to Create a Plot

- 1 Double-click the block to open the block dialog box. The following figure shows the area of the dialog box where you specify plotting options.

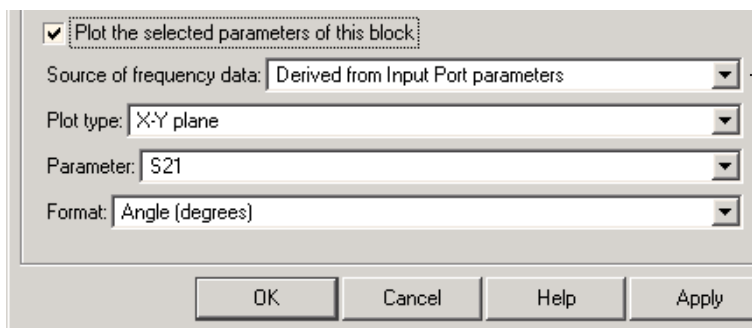


- 2** Select the **Plot the selected parameters of this block** check box to display the plot options.




Select this check box to display the plot options

- 3** Select the **Source of frequency data**.



Select the source of frequencies at which to plot block data

This is the source of the frequency values at which to plot block data. The following table summarizes the available types of sources for the various types of blocks.

| Source of frequency data | Description | Blocks |
|---|--|---------------------|
| Derived from Input Port parameters (Available after running a simulation or clicking the Update Diagram button ) | Modeling frequencies derived from the Input Port block parameters. For information on how the RF Blockset computes the modeling frequencies, see “Determining the Modeling Frequencies” on page A-3. | All physical blocks |
| User-specified | <p>Vector of frequencies that you enter.</p> <p>When you select User-specified in the Source of frequency data list, the Frequency range (Hz) field is displayed. Enter a vector specifying the range of frequencies you want to plot.</p> <p>For example, to plot block data from 0.3 MHz to 5 GHz by 0.1 MHz, enter [0.3e6:0.1e6:5e9].</p> | All physical blocks |

| Source of frequency data | Description | Blocks |
|---------------------------------|---|---|
| | <p>Note When you select PhaseNoise in the Parameter list and User-specified in the Source of frequency data list, the Frequency range (Hz) field is not displayed. You use the Phase noise frequency offset (Hz) block parameter to specify the frequency values at which to plot block data.</p> | |
| Same as the Frequency parameter | Frequency values specified in the Frequency block parameter. | S-Parameters Passive Network, Y-Parameters Passive Network, Z-Parameters Passive Network, S-Parameters Amplifier, Y-Parameters Amplifier, Z-Parameters Amplifier, S-Parameters Mixer, Y-Parameters Mixer, |

| Source of frequency data | Description | Blocks |
|------------------------------|--|---|
| | | and Z-Parameters Mixer |
| Extracted from RFDATA object | Frequency values imported into the RFDATA object block parameter. For information about rfddata objects, see the RF Toolbox rfddata reference page. | General Passive Network, General Amplifier, and General Mixer |

4 Select the **Plot type**.

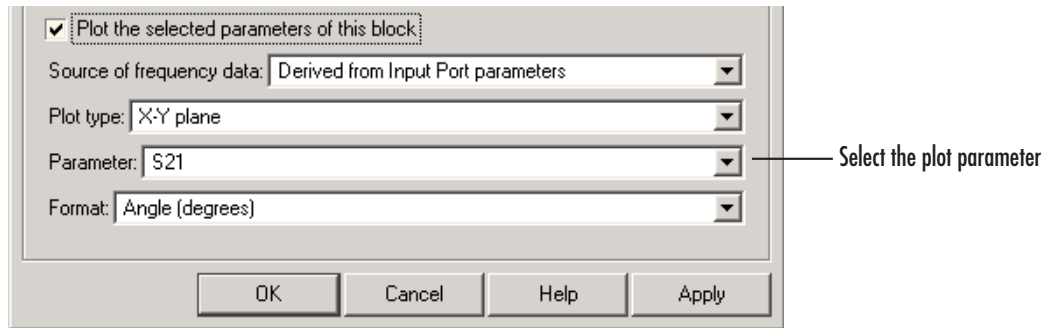
The screenshot shows a dialog box with the following settings:

- Plot the selected parameters of this block
- Source of frequency data: Derived from Input Port parameters
- Plot type: X-Y plane (highlighted with a callout "Select the plot type")
- Parameter: S21
- Format: Angle (degrees)

Buttons at the bottom: OK, Cancel, Help, Apply

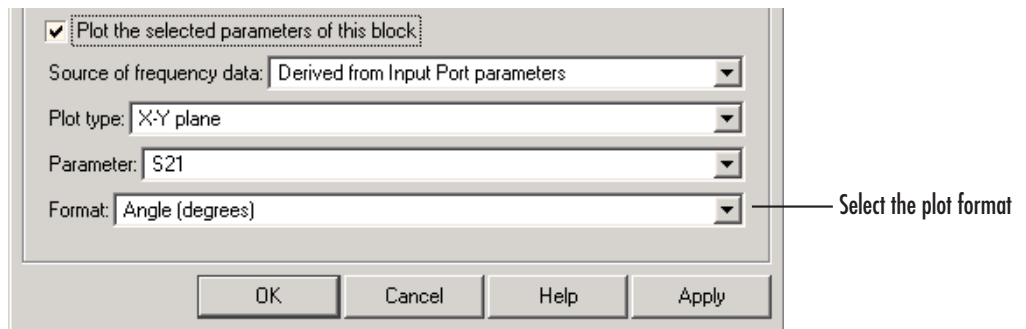
This is the type of plot. For a description of the options, see “Types of Plots” on page 3-3.

5 Select the **Parameter**.



This is the data to be plotted. The available choices vary with the type of plot. For a description of the options for a particular plot type, see the topic on that plot type in “Plot Formats” on page 3-4.

6 Choose the **Format**.



This is the format for plotting the selected parameter. The available choices vary based on the selected parameter. For a description of the options for a particular plot type, see the topic on that plot type in “Plot Formats” on page 3-4.

7 Click **Apply**.

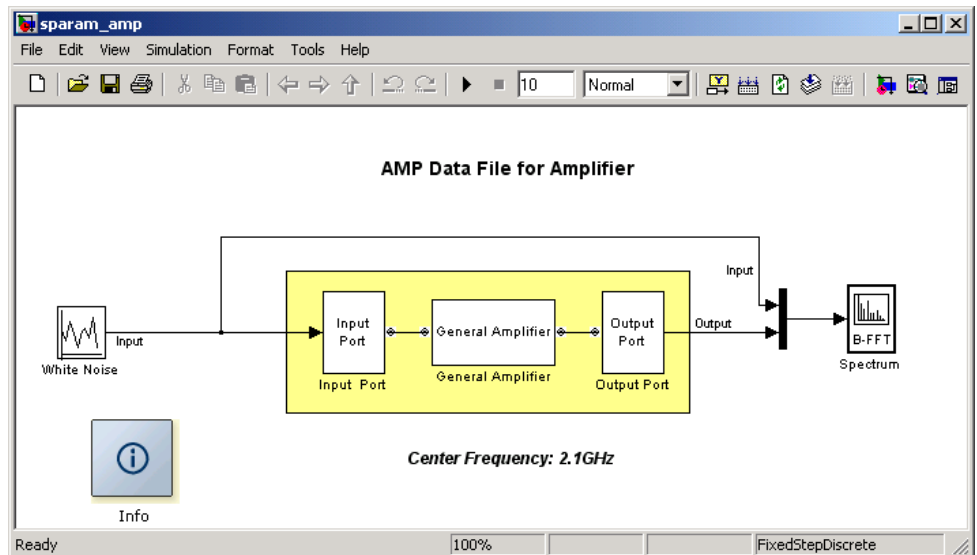
Note By default, the RF Blockset does not add a legend to some plots. To display the plot legend, type `legend show` at the MATLAB prompt.

Example – Plotting Component Data on a Z Smith Chart

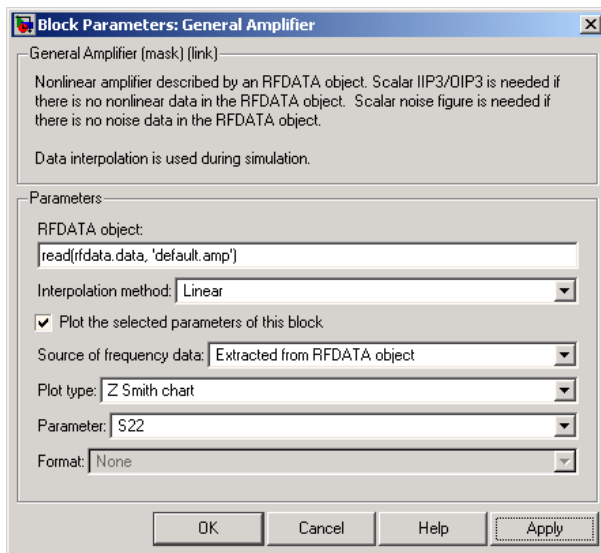
In this example, you simulate the frequency response of an amplifier using data from the `default.amp` AMP file.

Using a model from one of the RF Blockset demos, you import the data file into a General Amplifier block and validate the amplifier by plotting the S-parameters of the block on a Z Smith Chart.

- 1 Type `sparam_amp` at the MATLAB prompt to open the RF Blockset demo called “AMP Data File for Amplifier”.



2 Double-click the General Amplifier block to display its parameters.



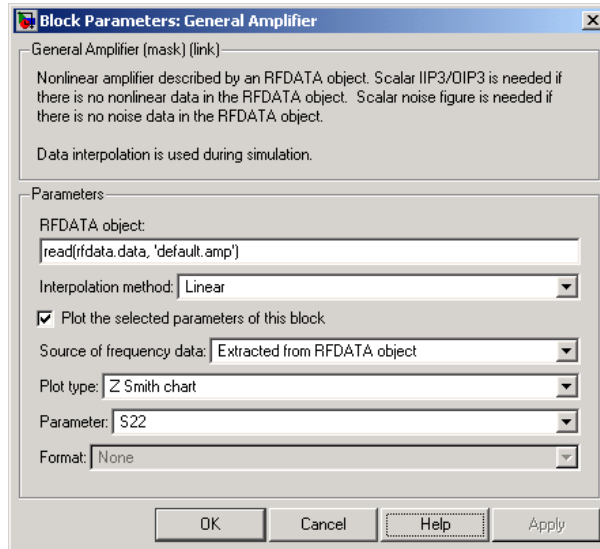
Notice that the **RFDATA object** parameter is set to `read(rfdata.data, 'default.amp')`. This value uses the RF Toolbox `read` function to import data from the file `default.amp` into an `rfdata.data` object. The block uses this data, along with the other block parameters, in simulation.

Note The General Amplifier block models the nonlinear amplifier described by an `rfdata.data` object. See the RF Toolbox `rfdata` reference page for information about `rfdata` objects.

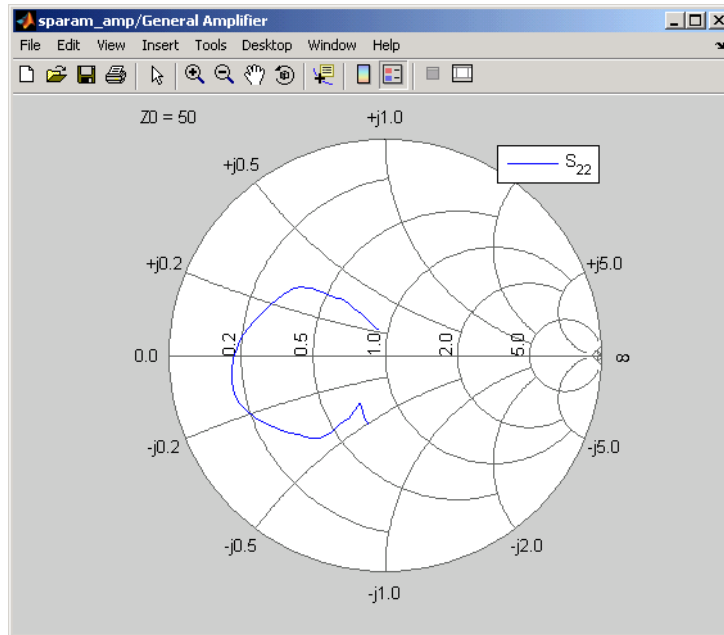
See “AMP File Format” in the RF Toolbox documentation for information about `.amp` files.

3 Set the General Amplifier block parameters as follows:

- Select the **Plot the selected parameters of this block** check box.
- In the **Plot type** list, select Z Smith chart.
- In the **Parameter** list, select S22.

**4** Click **Apply**.

This action creates a Z Smith® chart of the S22 parameters using the frequency data from the default.amp file.



General Amplifier Frequency Response

Note To display data tips for a plotted line, select **Tools > Data Cursor**. Click the data cursor on the plotted line to see the frequency and the parameter value at that point. See “Data Cursor — Displaying Data Values Interactively” in the MATLAB documentation for more information.

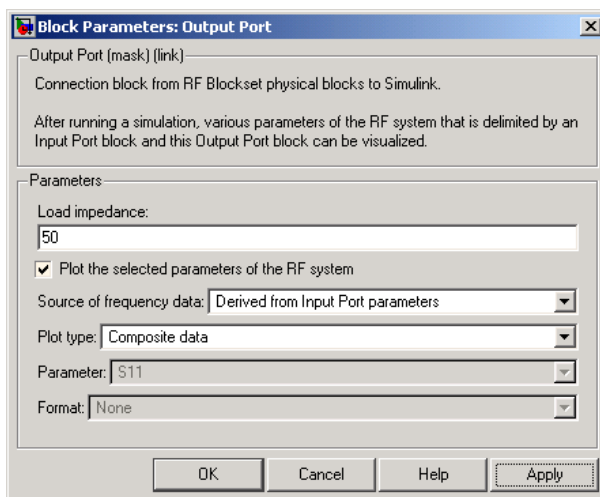
Updating Plots

When you run a simulation, the RF Blockset continues to display any open plots, but does not update the plots to reflect new simulation results. You must update the subsystem plots after the simulation to display the behavior of the revised subsystem.

Note You do not need to update the plots that represent individual RF components, because the RF Blockset redraws these plots when you make changes to the block parameters.

To update an existing subsystem plot:

- 1 Double-click the Output Port block to open the block dialog box.



- 2 Clear and select the **Plot the selected parameters of this block** check box.
- 3 Click **Apply**.

Modifying Plots

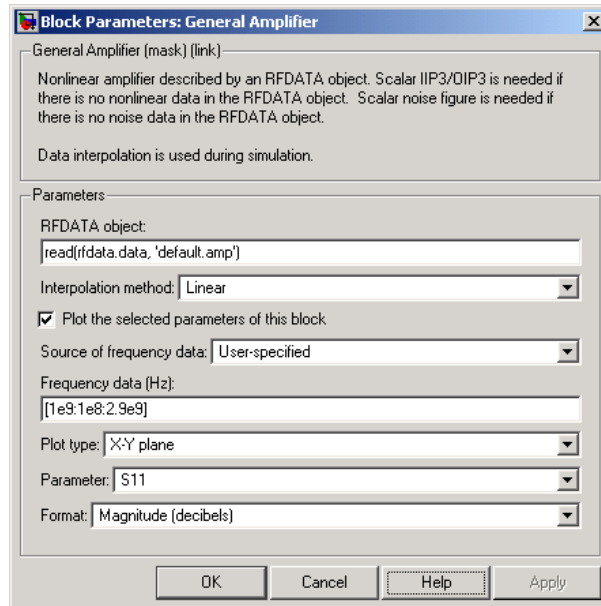
You can modify an existing plot by changing the plot options. The outcome depends on the parameter you change.

The following table summarizes the results of changing the plot options.

| Block Parameter | Plot Change |
|--|--|
| Source of frequency data OR Frequency range | Redraws plot using the new frequency data. |
| Plot type | Redraws figure in the new plot type, with the following considerations: <ul style="list-style-type: none"> • If the current plot options are valid for the new plot type, they retain their values. Otherwise, they revert to their default values. • If you change the Plot type to or from <code>Composite data</code>, the RF Blockset draws the new plot in a new figure and does not update the existing figure. |
| Parameter | If the new parameter has the same independent variable as the one on the plot, the RF Blockset adds the new parameter to the existing plot. Otherwise, it redraws the plot for the new parameter and independent variable. |
| Format | Redraws plot using the new format. |

To modify a plot:

- 1 Double-click the block to open the block dialog box.



Example Block Dialog Box Showing Plot Parameters

- 2 Change the plot options.
- 3 Click **Apply**.

Example – Creating and Modifying Subsystem Plots

In this example, you perform the following tasks:

- “Plotting the Network Parameters of a Subsystem” on page 3-22
- “Adding Data to an Existing Plot” on page 3-24
- “Changing Data on an Existing Plot” on page 3-26

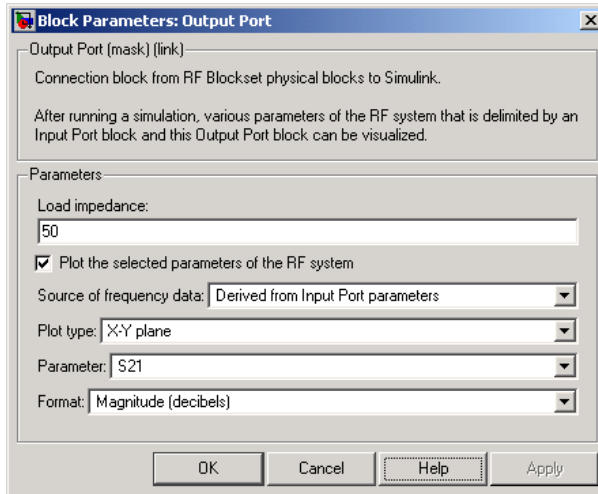
Plotting the Network Parameters of a Subsystem

In this part of the example, you open and run an RF Blockset demo that uses file data to specify an amplifier in a physical subsystem. Then, you plot the network parameters of the physical subsystem, which consists of the General Amplifier, the Input Port, and the Output Port blocks.

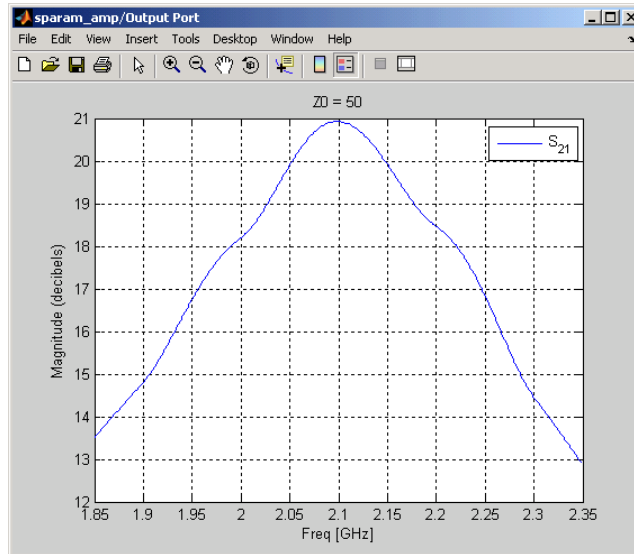
- 1 Type `sparam_amp` at the MATLAB prompt to open the RF Blockset demo called “AMP Data File for Amplifier”.
- 2 In the Simulink model window, select **Simulation > Start** to run the simulation.
- 3 Double-click the Output Port block to open the block dialog box.

4 Set the Output Port block parameters as follows:

- Select the **Plot the selected parameters of this block** check box.
- In the **Plot type** list, select X-Y plane.
- In the **Parameter** list, select S21.

**5** Click **Apply**.

This action plots the magnitude of S21 (in decibels) as a function of frequency on an X-Y plot.

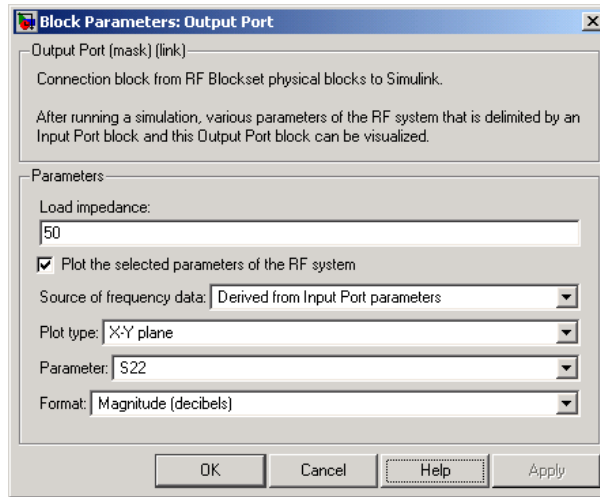


S21 versus Frequency for a Physical Subsystem

Adding Data to an Existing Plot

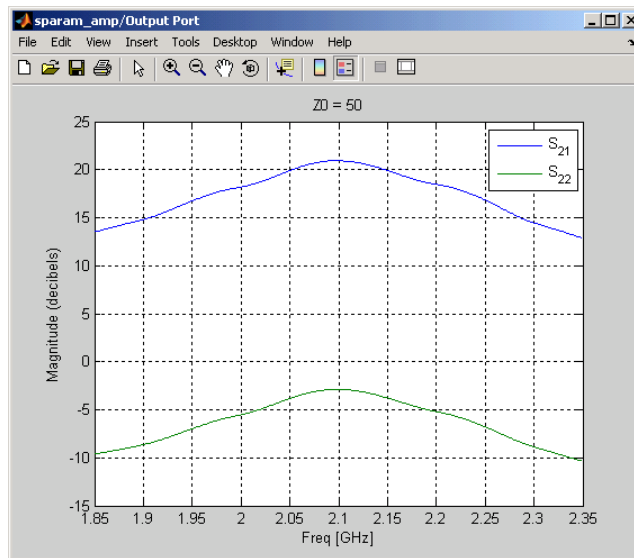
In this part of the example, you add data to the plot you created in “Plotting the Network Parameters of a Subsystem” on page 3-22.

- 1 Double-click the Output Port block to open the block dialog box.
- 2 Change the value of **Parameter** to S22.



3 Click **Apply**.

This action adds S22 to the plot.



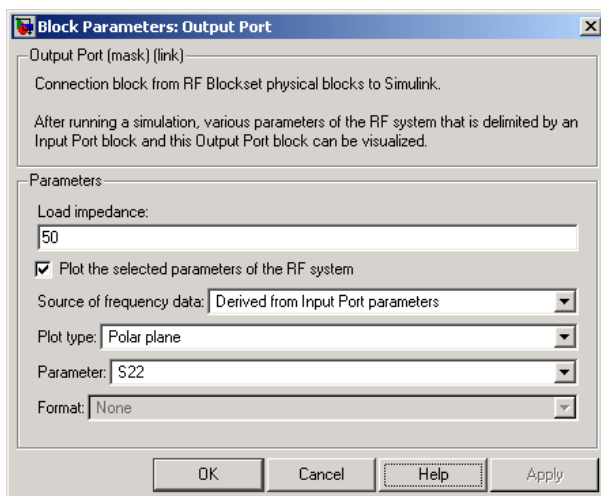
S21 and S22 versus Frequency for a Physical Subsystem

Changing Data on an Existing Plot

In this part of the example, you change the data on the plot you created in the previous steps of the example by modifying the **Plot type**.

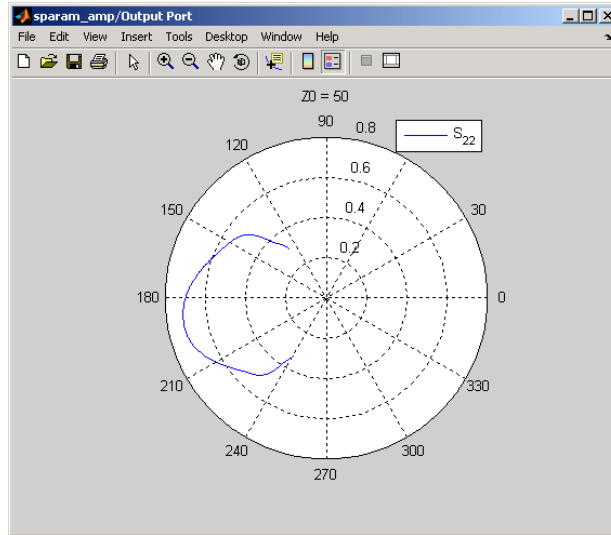
- 1 Double-click the Output Port block to open the block dialog box.
- 2 Change the value of **Plot type** to **Polar plane**, as shown in the following figure.

Notice that the value of **Parameter** remains as **S22**, the last parameter selected for the previous plot.



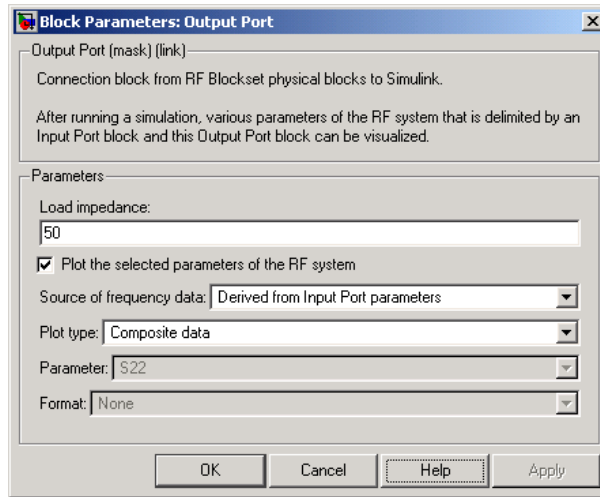
- 3 Click **Apply**.

This action creates a Polar plane plot of S_{22} as a function of frequency.

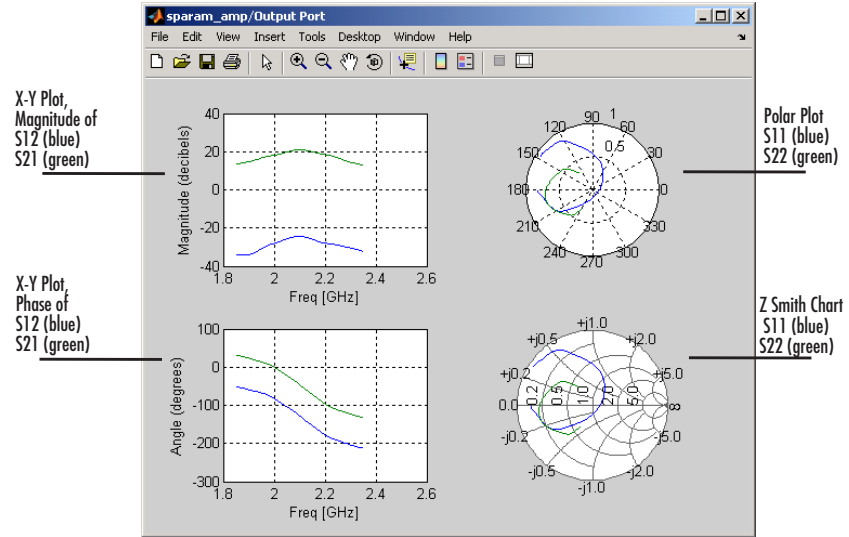


S_{22} versus Frequency for a Physical Subsystem

- 4 In the Output Port block dialog box, change the **Plot type** to Composite data to generate four plots in one figure. The parameters for the plots are defined by the block, so the **Parameter** field becomes invisible.



The RF Blockset creates the plot in a new figure.



Composite Plot for a Physical Subsystem

Blocks — By Category

Mathematical (p. 4-2)

Model RF components in terms of mathematical equations.

Physical (p. 4-3)

Model RF components in terms of physical properties or measured data

Mathematical

| | |
|--------------------|--|
| Amplifier | Complex baseband model of amplifier with noise |
| Bandpass RF Filter | Standard bandpass RF filters in baseband-equivalent complex form |
| Bandstop RF Filter | Standard bandstop RF filters in baseband-equivalent complex form |
| Highpass RF Filter | Standard highpass RF filters in baseband-equivalent complex form |
| Lowpass RF Filter | Standard lowpass RF filters in baseband-equivalent complex form |
| Mixer | Complex baseband model of mixer with phase noise |

Physical

| | |
|-----------------------------|---------------------------|
| Ladder Filters (p. 4-3) | Ladder filter blocks |
| Transmission Lines (p. 4-3) | Transmission line blocks |
| Black Box Elements (p. 4-4) | Black box elements blocks |
| Amplifiers (p. 4-4) | Amplifier blocks |
| Mixers (p. 4-5) | Mixer blocks |
| Input/Output Ports (p. 4-5) | Connector blocks |

Ladder Filters

| | |
|-----------------|-------------------------------|
| LC Bandpass Pi | Model LC bandpass pi network |
| LC Bandpass Tee | Model LC bandpass tee network |
| LC Bandstop Pi | Model LC bandstop pi network |
| LC Bandstop Tee | Model LC bandstop tee network |
| LC Highpass Pi | Model LC highpass pi network |
| LC Highpass Tee | Model LC highpass tee network |
| LC Lowpass Pi | Model LC lowpass pi network |
| LC Lowpass Tee | Model LC lowpass tee network |
| Series RLC | Model series RLC network |
| Shunt RLC | Model shunt RLC network |

Transmission Lines

| | |
|--------------------------------------|--|
| Coaxial Transmission Line | Model coaxial transmission line |
| Coplanar Waveguide Transmission Line | Model coplanar waveguide transmission line |
| Microstrip Transmission Line | Model microstrip transmission line |

| | |
|----------------------------------|--|
| Parallel-Plate Transmission Line | Model parallel-plate transmission line |
| RLCG Transmission Line | Model RLCG transmission line |
| Transmission Line | Model general transmission line |
| Two-Wire Transmission Line | Model two-wire transmission line |

Black Box Elements

| | |
|------------------------------|---|
| General Circuit Element | Model two-port network described by <code>rfckt</code> object |
| General Passive Network | Model two-port passive network described by <code>rfddata</code> object |
| S-Parameters Passive Network | Model passive network using its S-parameters |
| Y-Parameters Passive Network | Model passive network using its Y-parameters |
| Z-Parameters Passive Network | Model passive network using its Z-parameters |

Amplifiers

| | |
|------------------------|--|
| General Amplifier | Model nonlinear amplifier described by <code>rfddata</code> object |
| S-Parameters Amplifier | Model nonlinear amplifier using its S-parameters |
| Y-Parameters Amplifier | Model nonlinear amplifier using its Y-parameters |
| Z-Parameters Amplifier | Model nonlinear amplifier using its Z-parameters |

Mixers

| | |
|--------------------|---|
| General Mixer | Model mixer described by rfddata object |
| S-Parameters Mixer | Model mixer using its S-parameters |
| Y-Parameters Mixer | Model mixer using its Y-parameters |
| Z-Parameters Mixer | Model mixer using its Z-parameters |

Input/Output Ports

| | |
|-------------|--|
| Input Port | Connection block from Simulink environment to RF physical blocks |
| Output Port | Connection block from RF physical blocks to Simulink environment |

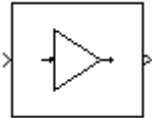
Blocks — Alphabetical List

Amplifier

Purpose Complex baseband model of amplifier with noise

Library Mathematical

Description The Amplifier block generates a complex baseband model of an amplifier with thermal noise. It provides six methods for modeling nonlinearity and three ways to specify noise.



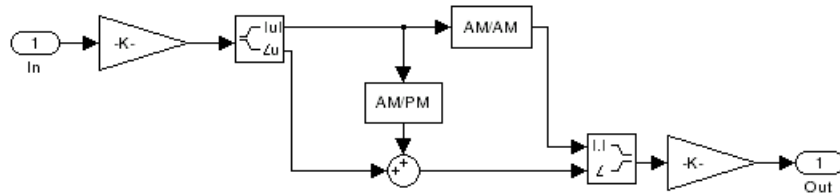
Note This block assumes a nominal impedance of 1 ohm.

Modeling Nonlinearity

Specify the method of your choice using this parameter in the block dialog box. The options for the **Method** parameter are

- Linear
- Cubic polynomial
- Hyperbolic tangent
- Saleh model
- Ghorbani model
- Rapp model

The linear method is implemented by a Gain block. The other nonlinear methods are implemented by subsystems underneath the block's mask. Each subsystem has the same basic structure, as shown in the figure below.



Application of Nonlinearity

All five subsystems for the nonlinear methods apply a memoryless nonlinearity to the complex baseband input signal. Each one

- 1 Multiplies the signal by a gain factor.
- 2 Splits the complex signal into its magnitude and angle components.
- 3 Applies an AM/AM conversion to the magnitude of the signal, according to the selected interpolation method, to produce the magnitude of the output signal.
- 4 Applies an AM/PM conversion to the phase of the signal, according to the selected interpolation method, and adds the result to the angle of the signal to produce the angle of the output signal.
- 5 Combines the new magnitude and angle components into a complex signal and multiplies the result by a gain factor, which is controlled by the **Linear gain** parameter.

AM/AM and AM/PM Conversions

The subsystems for the nonlinear methods implement the AM/AM and AM/PM conversions differently, according to the interpolation method you specify. To see exactly how the Amplifier block implements the conversions for a specific method, you can view the AM/AM and AM/PM subsystems that implement these conversions as follows:

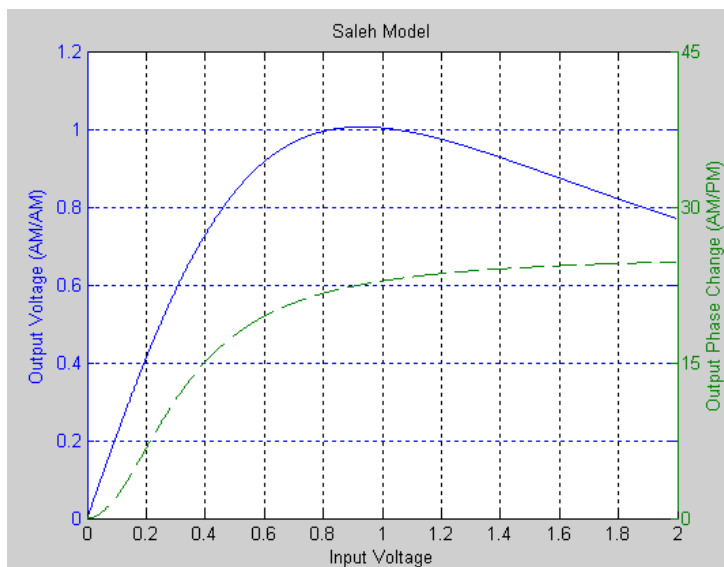
- 1 Right-click the Amplifier block.

Amplifier

- 2 Select **Look under mask** in the pop-up menu. This displays the block's configuration underneath the mask. The block contains five subsystems corresponding to the five nonlinearity methods.
- 3 Double-click the subsystem for the method in which you are interested. A subsystem displays similar to the one shown in the preceding figure.
- 4 Double-click one of the subsystems labeled AM/AM or AM/PM to view how the block implements the conversions.

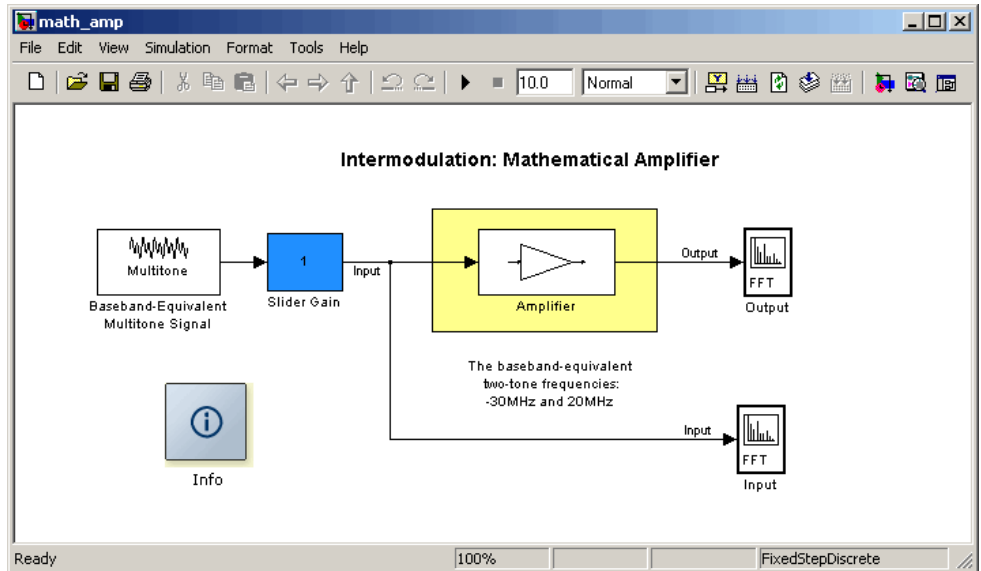
The following figure shows, for the Saleh method, plots of

- Output voltage against input voltage for the AM/AM conversion
- Output phase against input voltage for the AM/PM conversion



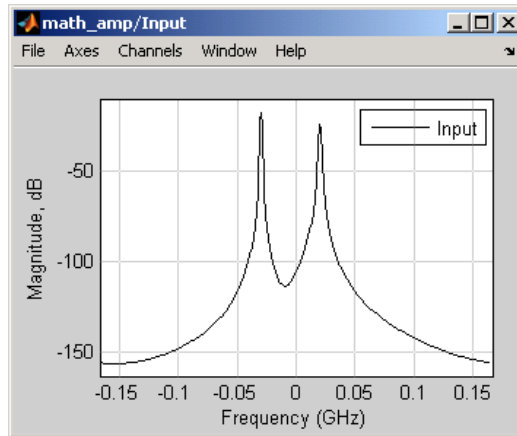
Effects of the Amplifier Block

You can see the effect of the Amplifier block in the demo Intermodulation: Mathematical Amplifier.

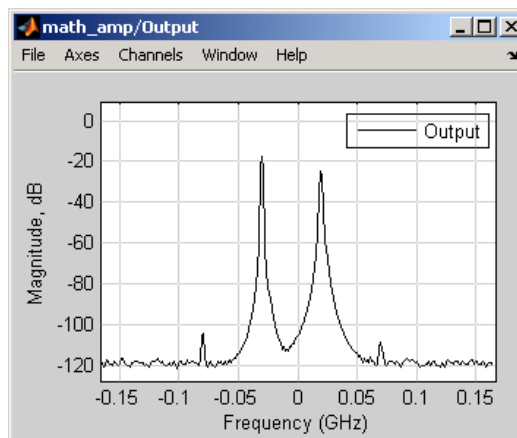


This demo uses a baseband-equivalent multitone signal as input to the Amplifier block. A Simulink Slider Gain block enables you to vary the gain from 1 to 10. The following figure shows the input signal with gain set to the default 1.

Amplifier



The next figure shows the same signal after it passes through the Amplifier block, with the **Method** parameter set to Hyperbolic tangent. The demo uses the default Amplifier block **IIP3 (dBm)** value of 30. It uses no AM/PM conversion. The demo specifies thermal noise as Noise figure, for which it uses the default 3.01 dB.



Parameters for Nonlinear Methods

The following sections discuss parameters specific to the following models:

- “Cubic Polynomial Model” on page 5-7
- “Hyperbolic Tangent Model” on page 5-8
- “Saleh Model” on page 5-8
- “Ghorbani Model” on page 5-9
- “Rapp Model” on page 5-10

Note The Amplifier block also enables you to model a linear amplifier.

Cubic Polynomial Model

The third-order input intercept point **IIP3 (dBm)** parameter is used to compute a scaling factor, which is then applied to the input signal. The scaling factor is equal to 3 divided by the **IIP3** parameter value, converted from decibels to linear units.

The scaled input is then limited to a maximum value of 1 and the amplitude gain is applied according to the following function

$$F_{AM/AM}(u) = u - \frac{u}{3}$$

where u is the magnitude of the scaled signal.

The **AM/PM conversion (degrees per dB)** parameter specifies the linear phase change. This is applied within the power limits specified by the **Lower input power limit for AM/PM conversion (dBm)** parameter and the **Upper input power limit for AM/PM conversion (dBm)** parameter. Outside those limits, the phase change is constant at the values corresponding to the lower and upper input power limits, which are zero and

(AM/PM conversion) · (upper input power limit – lower input power limit),
respectively.

The **Linear gain (dB)** parameter scales the output signal.

Hyperbolic Tangent Model

The third-order input intercept point **IIP3 (dBm)** parameter is used to compute a scaling factor, which is then applied to the input signal. The scaling factor is equal to 3 divided by the **IIP3** parameter value, converted from decibels to linear units.

The amplitude gain is then applied to the scaled input according to the following function

$$F_{AM/AM}(u) = \tanh(u)$$

where u is the magnitude of the scaled signal.

The **AM/PM conversion (degrees per dB)** parameter specifies the linear phase change. This is applied within the power limits specified by the **Lower input power limit for AM/PM conversion (dBm)** parameter and the **Upper input power limit for AM/PM conversion (dBm)** parameter. Outside those limits, the phase change is constant at the values corresponding to the lower and upper input power limits, which are zero and (AM/PM conversion) · (upper input power limit – lower input power limit), respectively.

The **Linear gain (dB)** parameter scales the output signal.

Saleh Model

The **Input scaling (dB)** parameter scales the input signal before the nonlinearity is applied. The block multiplies the input signal by the parameter value, converted from decibels to linear units. If you set the parameter to be the inverse of the input signal amplitude, the scaled signal has amplitude normalized to 1.

The AM/AM parameters, alpha and beta, are used to compute the amplitude gain for an input signal using the following function

$$F_{AM/AM}(u) = \frac{\alpha u}{1 + \beta u^2}$$

where u is the magnitude of the scaled signal.

The AM/PM parameters, α and β , are used to compute the phase change for an input signal using the following function

$$F_{AM/PM}(u) = \frac{\alpha u^2}{1 + \beta u^2}$$

where u is the magnitude of the input signal. Note that the AM/AM and AM/PM parameters, although similarly named α and β , are distinct.

The **Output scaling (dB)** parameter scales the output signal similarly.

Ghorbani Model

The **Input scaling (dB)** parameter scales the input signal before the nonlinearity is applied. The block multiplies the input signal by the parameter value, converted from decibels to linear units. If you set the parameter to be the inverse of the input signal amplitude, the scaled signal has amplitude normalized to 1.

The AM/AM parameters, $[x_1 x_2 x_3 x_4]$, are used to compute the amplitude gain for an input signal using the following function

$$F_{AM/AM}(u) = \frac{x_1 u^{x_2}}{1 + x_3 u^{x_2}} + x_4 u$$

where u is the magnitude of the scaled signal.

The AM/PM parameters, $[y_1 y_2 y_3 y_4]$, are used to compute the phase change for an input signal using the following function

$$F_{AM/PM}(u) = \frac{y_1 u^{y_2}}{1 + y_3 u^{y_2}} + y_4 u$$

where u is the magnitude of the scaled signal.

The **Output scaling (dB)** parameter scales the output signal similarly.

Rapp Model

The **Smoothness factor** and **Output saturation level** parameters are used to compute the amplitude gain for an input signal by the following function

$$F_{AM/AM}(u) = \frac{u}{\left(1 + \left(\frac{u}{O_{\text{sat}}}\right)^{2dS}\right)^{1/2S}}$$

where u is the magnitude of the scaled signal, S is the **Smoothness factor** and O_{sat} is the **Output saturation level**.

The Rapp model does not apply a phase change to the input signal.

The **Output saturation level** parameter limits the output signal level. The **Smoothness factor** parameter controls the transition for the amplitude gain as the input amplitude approaches saturation. The smaller the smoothness factor, the smoother the curve.

Thermal Noise Simulation

You can specify the amount of thermal noise in three ways, according to the **Specification method** parameter you select.

- Noise temperature — Specifies the noise in kelvin.
- Noise factor — Specifies the noise by the following equation:

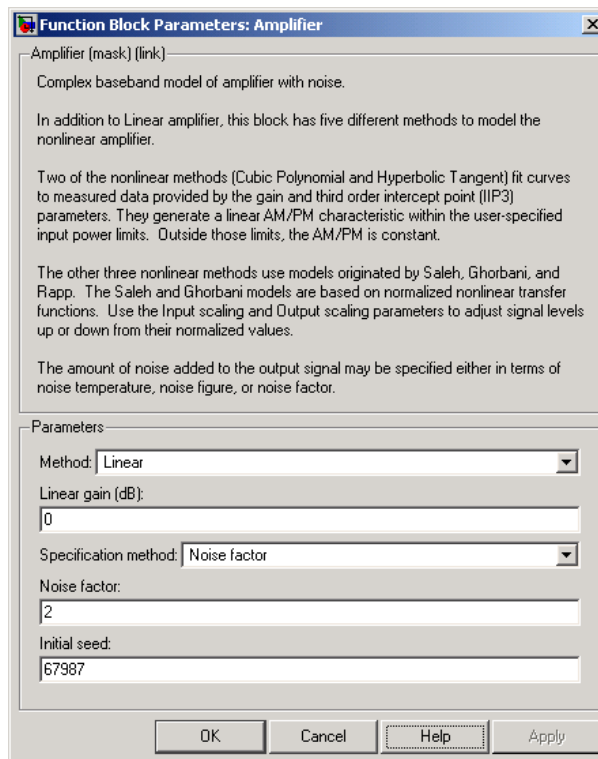
$$\text{Noise factor} = 1 + \frac{\text{Noise temperature}}{290}$$

- Noise figure — Specifies the noise in decibels relative to a noise temperature of 290 kelvin. In terms of noise factor,

$$\text{Noise figure} = 10\log(\text{Noise factor})$$

Note Some RF blocks require the sample time to perform baseband modeling calculations. To ensure the accuracy of these calculations, the Input Port block, as well as the mathematical RF blocks, compare the input sample time to the sample time you provide in the mask. If they do not match, or if the input sample time is missing because the blocks are not connected, an error message appears.

Dialog Box



The parameters displayed in the dialog box vary for different methods of modeling nonlinearity. Only some of these parameters are visible in the dialog box at any one time.

You can change tunable parameters while the model is running.

Method

Method used to model the nonlinearity. The choices are Linear, Cubic polynomial, Hyperbolic tangent, Saleh model, Ghorbani model, Rapp model. Tunable.

Linear gain (dB)

Scalar specifying the linear gain for the output function. This field becomes visible if you select Linear, Cubic polynomial, Hyperbolic tangent, or Rapp model as the **Method** parameter. Tunable.

IIP3 (dBm)

Input power intercept point as a scalar value. This field becomes visible if you select Cubic polynomial or Hyperbolic tangent as the **Method** parameter. For both of these methods, the nominal impedance is 1 ohm. Tunable.

AM/PM conversion (degrees per dB)

Scalar specifying the AM/PM conversion in degrees per decibel. This field becomes visible if you select Cubic polynomial or Hyperbolic tangent as the **Method** parameter. Tunable.

Lower input power limit for AM/PM conversion (dBm)

Scalar specifying the minimum input power for which AM/PM conversion scales linearly with input power value. Below this value, the phase shift resulting from AM/PM conversion is zero. This field becomes visible if you select Cubic polynomial or Hyperbolic tangent as the **Method** parameter. Tunable.

Upper input power limit for AM/PM conversion (dBm)

Scalar specifying the maximum input power for which AM/PM conversion scales linearly with input power value. Above this value, the phase shift resulting from AM/PM conversion is constant. The value of this maximum shift is given by:

$$(\text{AM/PM conversion}) \cdot (\text{upper input power limit} - \text{lower input power limit})$$

This field becomes visible if you select Cubic polynomial or Hyperbolic tangent as the **Method** parameter. Tunable.

Input scaling (dB)

Number that scales the input signal level. This field becomes visible if you select Saleh model or Ghorbani model as the **Method** parameter. Tunable.

Output scaling (dB)

Number that scales the output signal level. This field becomes visible if you select Saleh model or Ghorbani model as the **Method** parameter. Tunable.

AM/AM parameters [alpha beta]

Vector specifying the AM/AM parameters. This field becomes visible if you select Saleh model as the **Method** parameter. Tunable.

AM/PM parameters [alpha beta]

Vector specifying the AM/PM parameters. This field becomes visible if you select Saleh model as the **Method** parameter. Tunable.

AM/AM parameters [x1 x2 x3 x4]

Vector specifying the AM/AM parameters. This field becomes visible if you select Ghorbani model as the **Method** parameter. Tunable.

AM/PM parameters [y1 y2 y3 y4]

Vector specifying the AM/PM parameters. This field becomes visible if you select Ghorbani model as the **Method** parameter. Tunable.

Smoothness factor

Scalar specifying the smoothness factor. This field becomes visible if you select Rapp model as the **Method** parameter. Tunable.

Output saturation level

Scalar specifying the output saturation level. This field becomes visible if you select Rapp model as the **Method** parameter. Tunable.

Specification method

The method by which you specify the amount of noise. The choices are Noise temperature, Noise figure, and Noise factor. Tunable.

Noise temperature (K)

Scalar specifying the amount of noise. This field becomes visible if you select Noise temperature as the **Specification method** parameter. Tunable.

Noise figure (dB)

Scalar specifying the amount of noise relative to a noise temperature of 290 kelvin. A Noise figure setting of 0 dB indicates a noiseless system. This field becomes visible if you select Noise figure as the **Specification method** parameter. Tunable.

Noise factor

Scalar specifying the amount of noise relative to a noise temperature of 290 kelvin. This field becomes visible if you select Noise factor as the **Specification method** parameter. Tunable.

Initial seed

Nonnegative integer specifying the initial seed for the random number generator the block uses to generate noise.

References

- [1] Ghorbani, A. and M. Sheikhan, "The Effect of Solid State Power Amplifiers (SSPAs) Nonlinearities on MPSK and M-QAM Signal Transmission," *Sixth Int'l Conference on Digital Processing of Signals in Comm.*, 1991, pp. 193-197.
- [2] Rapp, C., "Effects of HPA-Nonlinearity on a 4-DPSK/OFDM-Signal for a Digital Sound Broadcasting System," in *Proceedings of the Second European Conference on Satellite Communications*, Liege, Belgium, Oct. 22-24, 1991, pp. 179-184.

[3] Saleh, A.A.M., "Frequency-independent and frequency-dependent nonlinear models of TWT amplifiers," *IEEE Trans. Communications*, vol. COM-29, pp.1715-1720, November 1981.

See Also

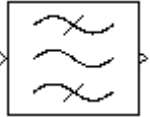
Bandpass RF Filter, Bandstop RF Filter, Highpass RF Filter, Lowpass RF Filter, Mixer

Bandpass RF Filter

Purpose Standard bandpass RF filters in baseband-equivalent complex form

Library Mathematical

Description The Bandpass RF Filter block lets you design standard analog bandpass filters, implemented in baseband-equivalent complex form. The following table describes the available design methods.



| Design Method | Description |
|---------------|--|
| Butterworth | The magnitude response of a Butterworth filter is maximally flat in the passband and monotonic overall. |
| Chebyshev I | The magnitude response of a Chebyshev I filter is equiripple in the passband and monotonic in the stopband. |
| Chebyshev II | The magnitude response of a Chebyshev II filter is monotonic in the passband and equiripple in the stopband. |
| Elliptic | The magnitude response of an elliptic filter is equiripple in both the passband and the stopband. |
| Bessel | The delay of a Bessel filter is maximally flat in the passband. |

The block input must be a discrete-time complex signal.

Note This block assumes a nominal impedance of 1 ohm.

Select the design of the filter from the **Design method** list in the dialog box. For each design method, the block enables you to specify the filter design parameters shown in the following table.

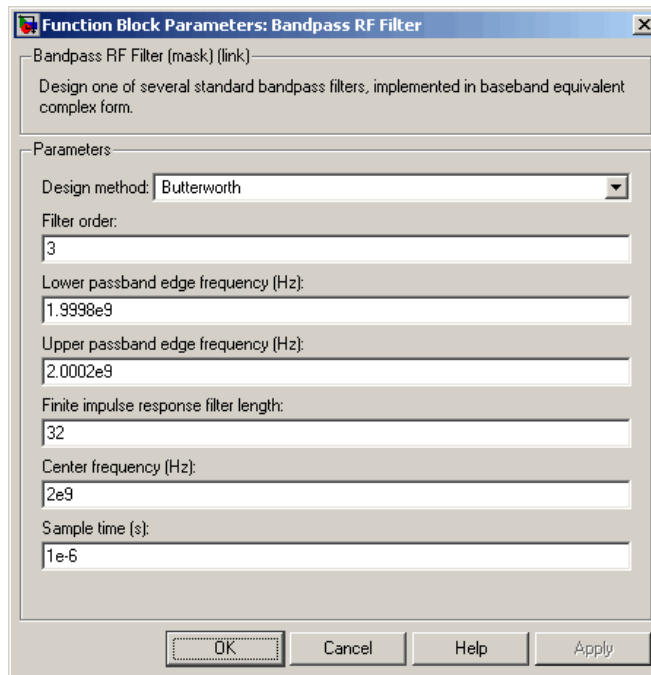
| Design Method | Filter Design Parameters |
|---------------|--|
| Butterworth | Order, lower passband edge frequency, upper passband edge frequency |
| Chebyshev I | Order, lower passband edge frequency, upper passband edge frequency, passband ripple |
| Chebyshev II | Order, lower stopband edge frequency, upper stopband edge frequency, stopband attenuation |
| Elliptic | Order, lower passband edge frequency, upper passband edge frequency, passband ripple, stopband attenuation |
| Bessel | Order, lower passband edge frequency, upper passband edge frequency |

The Bandpass RF Filter block designs the filters using the Signal Processing Toolbox filter design functions `buttap`, `cheb1ap`, `cheb2ap`, `ellipap`, and `besselap`.

Note Some RF blocks require the sample time to perform baseband modeling calculations. To ensure the accuracy of these calculations, the Input Port block, as well as the mathematical RF blocks, compare the input sample time to the sample time you provide in the mask. If they do not match, or if the input sample time is missing because the blocks are not connected, an error message appears.

Bandpass RF Filter

Dialog Box



The parameters displayed in the dialog box vary for different design methods. Only some of these parameters are visible in the dialog box at any one time.

You can change tunable parameters while the model is running.

Design method

Filter design method. The design method can be Butterworth, Chebyshev I, Chebyshev II, Elliptic, or Bessel. Tunable.

Filter order

Order of the lowpass analog prototype filter that forms the basis for the bandpass filter design. The order of the final filter is twice this value.

Lower passband edge frequency (Hz)

Lower passband edge frequency for Butterworth, Chebyshev I, elliptic, and Bessel designs. Tunable.

Upper passband edge frequency (Hz)

Upper passband edge frequency for Butterworth, Chebyshev I, elliptic, and Bessel designs. Tunable.

Lower stopband edge frequency (Hz)

Lower stopband edge frequency for Chebyshev II designs. Tunable.

Upper stopband edge frequency (Hz)

Upper stopband edge frequency for Chebyshev II designs. Tunable.

Passband ripple in dB

Passband ripple for Chebyshev I and elliptic designs. Tunable.

Stopband attenuation in dB

Stopband attenuation for Chebyshev II and elliptic designs. Tunable.

Finite impulse response filter length

Desired length of the baseband-equivalent impulse response for the filter.

Center frequency (Hz)

Center of the modeling frequencies.

Sample time (s)

Time interval between consecutive samples of the input signal.

See Also

Amplifier, Bandstop RF Filter, Highpass RF Filter, Lowpass RF Filter, Mixer

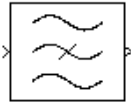
buttap, cheb1ap, cheb2ap, ellipap, besslap (Signal Processing Toolbox)

Bandstop RF Filter

Purpose Standard bandstop RF filters in baseband-equivalent complex form

Library Mathematical

Description The Bandstop RF Filter block lets you design standard analog bandstop filters, implemented in baseband-equivalent complex form. The following table describes the available design methods.



| Design Method | Description |
|---------------|--|
| Butterworth | The magnitude response of a Butterworth filter is maximally flat in the passband and monotonic overall. |
| Chebyshev I | The magnitude response of a Chebyshev I filter is equiripple in the passband and monotonic in the stopband. |
| Chebyshev II | The magnitude response of a Chebyshev II filter is monotonic in the passband and equiripple in the stopband. |
| Elliptic | The magnitude response of an elliptic filter is equiripple in both the passband and the stopband. |
| Bessel | The delay of a Bessel filter is maximally flat in the passband. |

The block input must be a discrete-time complex signal.

Note This block assumes a nominal impedance of 1 ohm.

Select the design of the filter from the **Design method** list in the dialog box. For each design method, the block enables you to specify the filter design parameters shown in the following table.

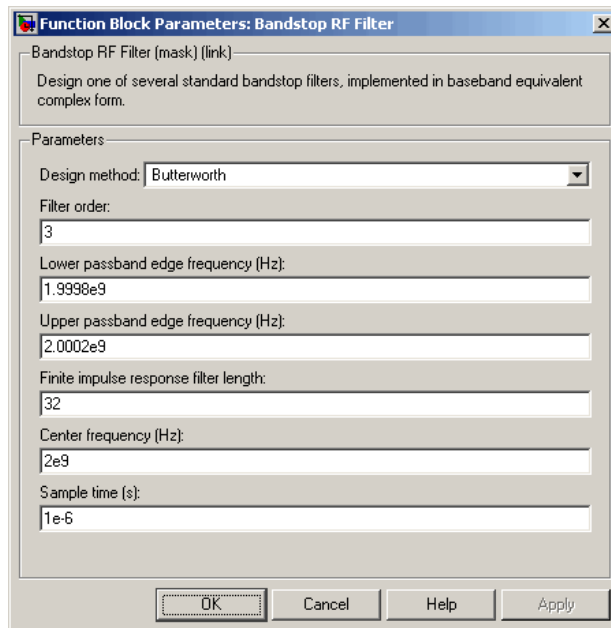
| Design Method | Filter Design Parameters |
|---------------|--|
| Butterworth | Order, lower passband edge frequency, upper passband edge frequency |
| Chebyshev I | Order, lower passband edge frequency, upper passband edge frequency, passband ripple |
| Chebyshev II | Order, lower stopband edge frequency, upper stopband edge frequency, stopband attenuation |
| Elliptic | Order, lower passband edge frequency, upper passband edge frequency, passband ripple, stopband attenuation |
| Bessel | Order, lower passband edge frequency, upper passband edge frequency |

The Bandstop RF Filter block designs the filters using the Signal Processing Toolbox filter design functions `buttap`, `cheb1ap`, `cheb2ap`, `ellipap`, and `besselap`.

Note Some RF blocks require the sample time to perform baseband modeling calculations. To ensure the accuracy of these calculations, the Input Port block, as well as the mathematical RF blocks, compare the input sample time to the sample time you provide in the mask. If they do not match, or if the input sample time is missing because the blocks are not connected, an error message appears.

Bandstop RF Filter

Dialog Box



The parameters displayed in the dialog box vary for different design methods. Only some of these parameters are visible in the dialog box at any one time.

You can change tunable parameters while the model is running.

Design method

Filter design method. The design method can be Butterworth, Chebyshev I, Chebyshev II, Elliptic, or Bessel. Tunable.

Filter order

Order of the lowpass analog prototype filter that forms the basis for the bandstop filter design. The order of the final filter is twice this value.

Lower passband edge frequency (Hz)

Lower passband edge frequency for Butterworth, Chebyshev I, elliptic, and Bessel designs. Tunable.

Upper passband edge frequency (Hz)

Upper passband edge frequency for Butterworth, Chebyshev I, elliptic, and Bessel designs. Tunable.

Lower stopband edge frequency (Hz)

Lower stopband edge frequency for Chebyshev II designs. Tunable.

Upper stopband edge frequency (Hz)

Upper stopband edge frequency for Chebyshev II designs. Tunable.

Passband ripple in dB

Passband ripple for Chebyshev I and elliptic designs. Tunable.

Stopband attenuation in dB

Stopband attenuation for Chebyshev II and elliptic designs. Tunable.

Finite impulse response filter length

Desired length of the baseband-equivalent impulse response for the filter.

Center frequency (Hz)

Center of the modeling frequencies.

Sample time (s)

Time interval between consecutive samples of the input signal.

See Also

Amplifier, Bandpass RF Filter, Highpass RF Filter, Lowpass RF Filter, Mixer

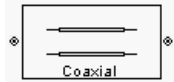
buttap, cheb1ap, cheb2ap, ellipap, besslap (Signal Processing Toolbox)

Coaxial Transmission Line

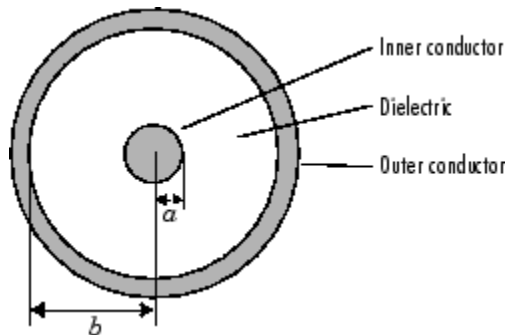
Purpose Model coaxial transmission line

Library Transmission Lines sublibrary of the Physical library

Description



The Coaxial Transmission Line block models the coaxial transmission line described in the block dialog box in terms of its frequency-dependent S-parameters. A coplanar waveguide transmission line is shown here in cross-section. Its physical characteristics include the radius of the inner conductor a and the radius of the outer conductor b .



The block lets you model the transmission line as a stub or as a stubless line.

Stubless Transmission Line

If you model a coaxial transmission line as a stubless line, the Coaxial Transmission Line block calculates the frequency-dependent S-parameters using the physical length of the transmission line, D , and the complex propagation constant, k .

$$S_{11} = 0$$

$$S_{12} = e^{-kD}$$

$$S_{21} = e^{-kD}$$

$$S_{22} = 0$$

k is a vector whose elements correspond to the elements of f , a vector of modeling frequencies. It can be expressed in terms of the resistance (R), inductance (L), conductance (G), and capacitance (C) per unit length (meters) as

$$k = k_r + jk_i = \sqrt{(R + j2\pi fL)(G + j2\pi fC)}$$

where

$$R = \frac{1}{2\pi\sigma_{\text{cond}}\delta} \left(\frac{1}{a} + \frac{1}{b} \right)$$

$$L = \frac{\mu}{2\pi} \ln(b/a)$$

$$G = \frac{2\pi\sigma_{\text{diel}}}{\ln(b/a)}$$

$$C = \frac{2\pi\epsilon}{\ln(b/a)}$$

In these equations, σ_{cond} is the conductivity in the conductor and σ_{diel} is the conductivity in the dielectric. μ is the permeability of the dielectric, ϵ is its permittivity, and skin depth δ is calculated as $1/\sqrt{\pi f \mu \sigma_{\text{cond}}}$. f is a vector of modeling frequencies determined by the Output Port block.

The Coaxial Transmission Line block normalizes the resulting S-parameters to a reference impedance of 50 ohms.

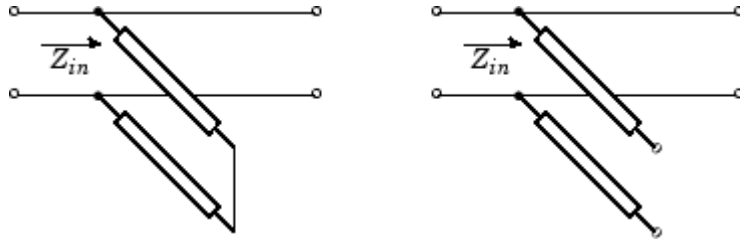
Shunt and Series Stubs

If you model the transmission line as a shunt or series stub, the Coaxial Transmission Line block first calculates the ABCD-parameters at each frequency contained in the modeling frequencies vector. It then uses the `abcd2s` function to convert the ABCD-parameters to S-parameters.

Shunt ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to Shunt, the two-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.

Coaxial Transmission Line



Z_{in} is the input impedance of the shunt circuit. The ABCD-parameters for the shunt stub are calculated as

$$A = 1$$

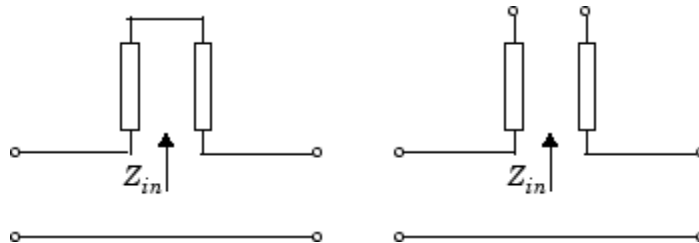
$$B = 0$$

$$C = 1/Z_{in}$$

$$D = 1$$

Series ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to Series, the two-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



Z_{in} is the input impedance of the series circuit. The ABCD-parameters for the series stub are calculated as

$$\begin{aligned}A &= 1 \\B &= Z_{in} \\C &= 0 \\D &= 1\end{aligned}$$

Dialog Box

Block Parameters: Coaxial Transmission Line

Coaxial Transmission Line (mask) (link)
Model a coaxial transmission line.

Parameters

Outer radius (m):
2.57e-3

Inner radius (m):
0.725e-3

Relative permeability constant:
1

Relative permittivity constant:
2.3

Conductivity in conductor (S/m):
inf

Conductivity in dielectric (S/m):
0

Transmission line length (m):
0.01

Stub mode: Not a stub

Plot the selected parameters of this block

Source of frequency data: User-specified

Frequency data (Hz):
[1e9; 1.0e6; 3e9]

Plot type: X-Y plane

Parameter: S21

Format: Angle (degrees)

OK Cancel Help Apply

Outer radius (m)

Radius of the outer conductor of the coaxial transmission line.

Coaxial Transmission Line

Inner radius (m)

Radius of the inner conductor of the coaxial transmission line.

Relative permeability constant

Relative permeability of the dielectric expressed as the ratio of the permeability of the dielectric to permeability in free space μ_0 .

Relative permittivity constant

Relative permittivity of the dielectric expressed as the ratio of the permittivity of the dielectric to permittivity in free space ϵ_0 .

Conductivity in conductor (S/m)

Conductivity of the conductor in siemens per meter.

Conductivity in dielectric (S/m)

Conductivity of the dielectric in siemens per meter.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

Type of stub. Choices are Not a stub, Shunt, or Series.

Termination of stub

Stub termination for stub modes Shunt and Series. Choices are Open or Short. This parameter becomes visible only when **Stub mode** is set to Shunt or Series.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

References

[1] Ludwig, Reinhold and Pavel Bretchko, *RF Circuit Design: Theory and Applications*, Prentice-Hall, 2000.

See Also

Coplanar Waveguide Transmission Line, General Passive Network, Transmission Line, Microstrip Transmission Line, Parallel-Plate Transmission Line, Two-Wire Transmission Line

Coplanar Waveguide Transmission Line

Purpose

Model coplanar waveguide transmission line

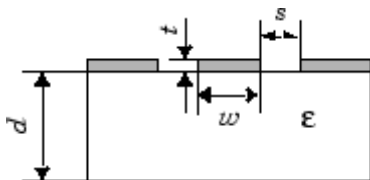
Library

Transmission Lines sublibrary of the Physical library

Description



The Coplanar Waveguide Transmission Line block models the coplanar waveguide transmission line described in the block dialog box in terms of its frequency-dependent S-parameters. A coplanar waveguide transmission line is shown here in cross-section. Its physical characteristics include the conductor width (w), the conductor thickness (t), the slot width (s), the substrate height (d), and the relative permittivity constant (ϵ).



The block lets you model the transmission line as a stub or as a stubless line.

Stubless Transmission Line

If you model a coplanar waveguide transmission line as a stubless line, the Coplanar Waveguide Transmission Line block calculates the frequency-dependent S-parameters using the physical length of the transmission line, D , and the complex propagation constant, k .

$$S_{11} = 0$$

$$S_{12} = e^{-kD}$$

$$S_{21} = e^{-kD}$$

$$S_{22} = 0$$

$k = \alpha_a + i\beta$, where α_a is the attenuation coefficient and β is the wave number. The attenuation coefficient α_a is related to the loss, α , by

Coplanar Waveguide Transmission Line

$$\alpha_a = -\ln 10 \frac{\alpha}{20}$$

where α is the reduction in signal strength, in dB, per unit length. α combines both conductor loss and dielectric loss and is derived from the physical parameters specified in the Coplanar Waveguide Transmission Line block dialog box.

The wave number β is related to the phase velocity, V_p , by

$$\beta = \frac{2\pi f}{V_p}$$

$V_p = c / \sqrt{\epsilon_{\text{eff}}}$, where ϵ_{eff} is the frequency-dependent effective dielectric constant. f is a vector of modeling frequencies determined by the Output Port block. The phase velocity V_p is also known as the wave propagation velocity.

The Coplanar Waveguide Transmission Line block normalizes the resulting S-parameters to a reference impedance of 50 ohms.

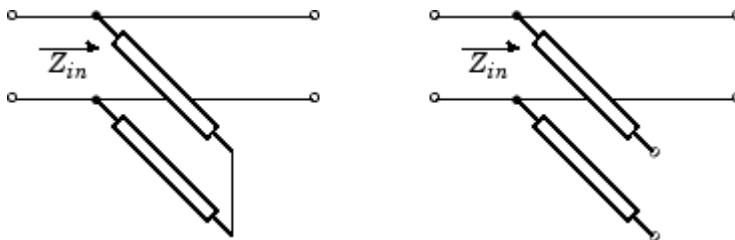
Shunt and Series Stubs

If you model the transmission line as a shunt or series stub, the Coplanar Waveguide Transmission Line block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. It then uses the `abcd2s` function to convert the ABCD-parameters to S-parameters.

Shunt ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to Shunt, the two-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.

Coplanar Waveguide Transmission Line



Z_{in} is the input impedance of the shunt circuit. The ABCD-parameters for the shunt stub are calculated as

$$A = 1$$

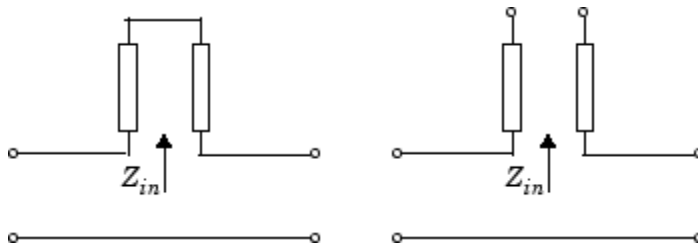
$$B = 0$$

$$C = 1/Z_{in}$$

$$D = 1$$

Series ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to Series, the two-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



Z_{in} is the input impedance of the series circuit. The ABCD-parameters for the series stub are calculated as

Coplanar Waveguide Transmission Line

$$\begin{aligned} A &= 1 \\ B &= Z_{in} \\ C &= 0 \\ D &= 1 \end{aligned}$$

Dialog Box

Block Parameters: Coplanar Waveguide Transmission Line

Coplanar Waveguide Transmission Line (mask) (link)
Model a coplanar waveguide transmission line.

Parameters:

Conductor width (m): 0.6e-3

Slot width (m): 0.2e-3

Substrate height (m): 0.635e-3

Strip thickness (m): 0.005e-3

Relative permittivity constant: 9.8

Conductivity in conductor (S/m): inf

Loss tangent in dielectric: 0

Transmission line length (m): 0.01

Stub mode: Not a stub

Plot the selected parameters of this block

Source of frequency data: User-specified

Frequency data (Hz): [1e9; 1.0e6; 3e9]

Plot type: X-Y plane

Parameter: S21

Format: Angle (degrees)

OK Cancel Help Apply

Coplanar Waveguide Transmission Line

Conductor width (m)

Physical width of the conductor.

Slot width (m)

Physical width of the slot.

Substrate height (m)

Thickness of the dielectric on which the conductor resides.

Strip thickness (m)

Physical thickness of the conductor.

Relative permittivity constant

Relative permittivity of the dielectric expressed as the ratio of the permittivity of the dielectric to permittivity in free space ϵ_0 .

Conductivity in conductor (S/m)

Conductivity of the conductor in siemens per meter.

Loss tangent in dielectric

Loss angle tangent of the dielectric.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

Type of stub. Choices are Not a stub, Shunt, or Series.

Termination of stub

Stub termination for stub modes Shunt and Series. Choices are Open or Short. This parameter becomes visible only when **Stub mode** is set to Shunt or Series.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

References

[1] Gupta, K. C., Ramesh Garg, Inder Bahl, and Prakash Bhartia, *Microstrip Lines and Slotlines*, 2nd Edition, Artech House, Inc., Norwood, MA, 1996.

Coplanar Waveguide Transmission Line

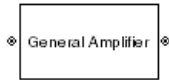
See Also

Coaxial Transmission Line, General Passive Network, Transmission Line, Microstrip Transmission Line, Parallel-Plate Transmission Line, Two-Wire Transmission Line

Purpose Model nonlinear amplifier described by `rfddata` object

Library Amplifiers sublibrary of the Physical library

Description The General Amplifier block models the nonlinear amplifier described by an RF Toolbox data (`rfddata.data`) object.



Network Parameters

If network parameter data and their corresponding frequencies exist as S-parameters in the `rfddata.data` object, the General Amplifier block interpolates the S-parameters to determine their values at the modeling frequencies. If the block contains network Y- or Z-parameters, the block first converts them to S-parameters. See Appendix A, “RF Blockset Algorithms” for more details.

Nonlinearity

If power data exists in the `rfddata.data` object, the block extracts the AMAM/AMPM nonlinearities from the power data.

If the `rfddata.data` object contains no power data, then you can enter either the OIP3 or IIP3 as a scalar value for nonlinearity in the General Amplifier block dialog box.

Active Noise

If active spot noise data exists in the `rfddata.data` object, the block uses the data to calculate the noise figure. The block first interpolates the noise data for the modeling frequencies, using the specified **Interpolation method**. It then calculates the noise figure using the resulting values.

If the `rfddata.data` object contains no noise data, then you can enter a value for the noise figure in the General Amplifier block dialog box.

Data Consistency

If you create the `rfddata.data` object by reading data from a MathWorks AMP file that contains both network parameter data and power data, the RF Blockset checks the data for consistency and reconciles it as necessary.

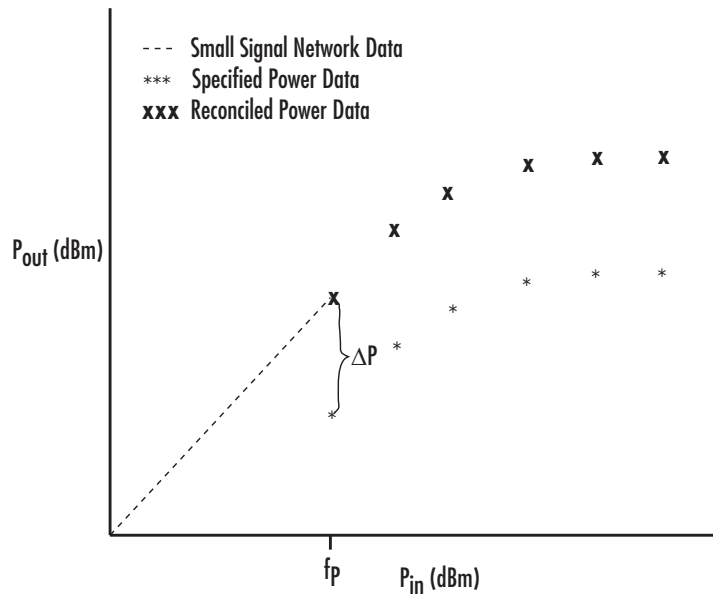
General Amplifier

The RF Blockset compares the small-signal amplifier gain defined by the network parameters, S_{21} , and by the power data, $P_{out}-P_{in}$. The discrepancy between the two is computed in dBm using the following equation:

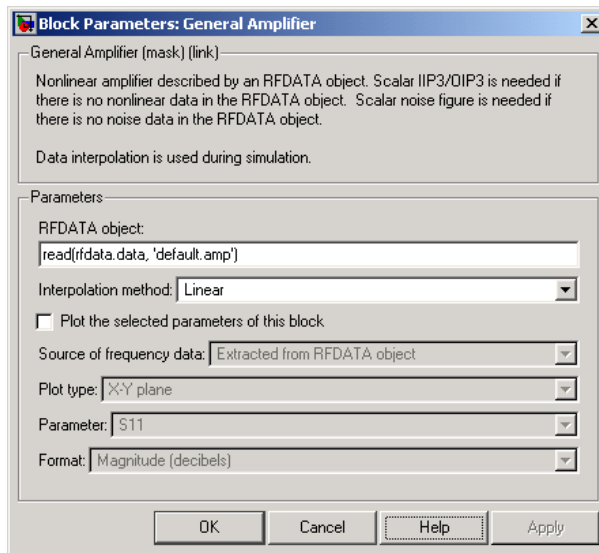
$$\Delta P = S_{21}(f_P) - P_{out}(f_P) + P_{in}(f_P) \quad (dBm)$$

where f_P is the lowest frequency for which power data is specified.

If ΔP is more than 0.4 dB, a warning appears and the RF Blockset adds ΔP to the output power values at each specified input power value to resolve the discrepancy for simulation. This discrepancy is shown in the following graph.



Dialog Box



RFDATA object

An RF Toolbox data (`rfddata.data`) object. You can specify the object as (1) the handle of a data object previously created using the RF Toolbox, (2) an RF Toolbox command such as `rfddata.data`, which creates a default data object, or (3) a MATLAB expression that generates such an object. See the RF Toolbox documentation for more information about data objects.

Interpolation method

For network data, the method used to interpolate the parameters contained in the `rfddata.data` object. Interpolation can be Cubic, Linear (default), or Spline.

IP3 type

Type of third-order intercept point. The value can be IIP3 (input intercept point) or OIP3 (output intercept point). This parameter becomes visible only if the `rfddata.data` object contains no power data.

General Amplifier

IIP3 (dBm)

Input power intercept point as a scalar value. This field becomes visible if you select IIP3 as the IP3 type. This parameter becomes visible only if the `rfdata.data` object contains no power data.

OIP3 (dBm)

Output power intercept point as a scalar value. This field becomes visible if you select OIP3 as the IP3 type. This parameter becomes visible only if the `rfdata.data` object contains no power data.

Noise figure (dB)

Scalar ratio of the available signal-to-noise power ratio at the input to the available signal-to-noise power ratio at the output, $(S_i/N_i)/(S_o/N_o)$. This parameter becomes visible only if the `rfdata.data` object contains no noise data.

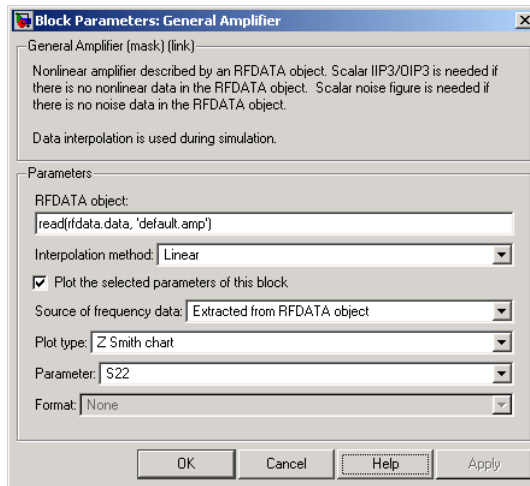
Note For information about plotting the amplifier parameters, see Chapter 3, “Plotting Model Data”. Use `rftool` or the RF Toolbox plotting functions to plot other data.

Examples

Creating a General Amplifier Block from File Data

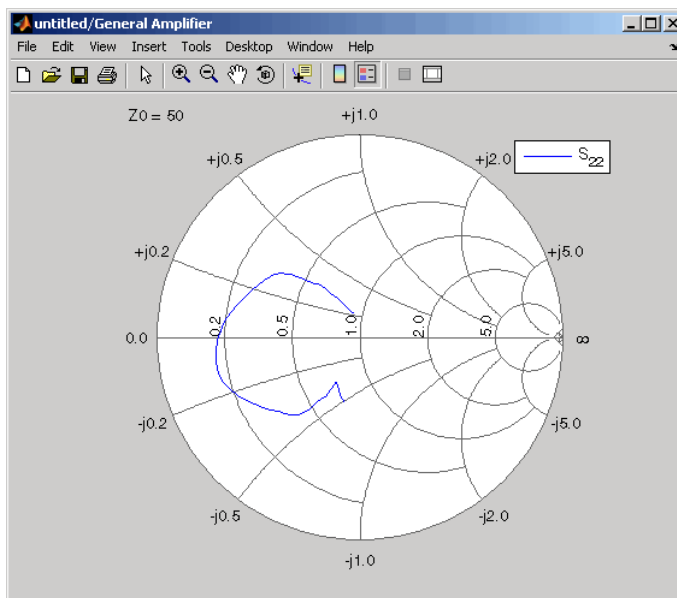
This example uses the RF Toolbox `read` function to create an `rfdata.data` object that describes the nonlinear amplifier in the file `default.amp`. The file, which is read into an RF Toolbox data (`rfdata.data`) object, contains S-parameters for frequencies from 1.0 to 2.9 GHz at intervals of 0.01 GHz, power data at frequency 2.1 GHz, and active noise parameters. The General Amplifier block uses linear interpolation to model the network described in the object.

Note See “AMP File Format” in the RF Toolbox documentation for information about `.amp` files.



The plot parameters in the dialog box request a Z Smith chart of the S22 parameters using the frequencies taken from the **RFDATA object** parameter.

General Amplifier



See Also

Output Port, S-Parameters Amplifier, Y-Parameters Amplifier, Z-Parameters Amplifier

`rfdata`, `rfdata.data` (RF Toolbox)

`interp1` (MATLAB)

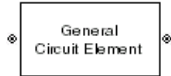
Purpose

Model two-port network described by `rfckt` object

Library

Black Box Elements sublibrary of the Physical library

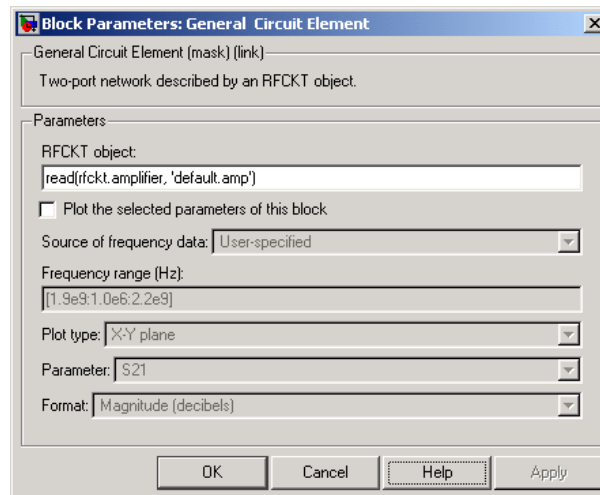
Description



The General Circuit Element block models the two-port network described by an RF Toolbox circuit (`rfckt`) object.

The block uses the `rfckt/analyze` method to calculate the network parameters at the modeling frequencies.

Dialog Box



RFCKT object

An RF Toolbox circuit (`rfckt`) object. You can specify the object as (1) the handle of a circuit object previously created using the RF Toolbox, (2) an RF Toolbox command such as `rfckt.txline`, `rfckt.coaxial`, or `rfckt.cascade` that creates a default circuit object of the specified type, or (3) a MATLAB expression that generates such an object. See the RF Toolbox documentation for more information about circuit objects.

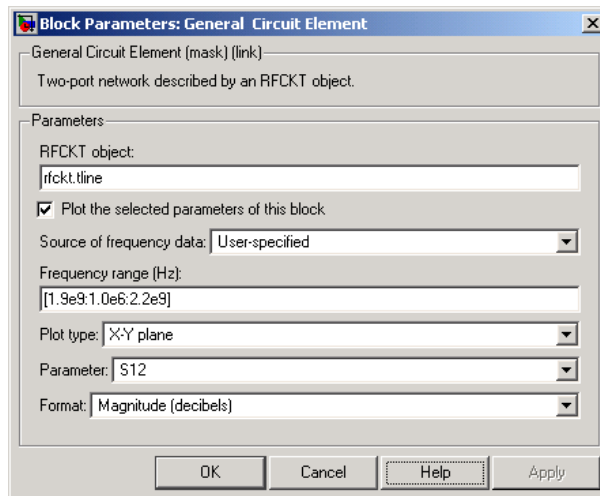
General Circuit Element

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

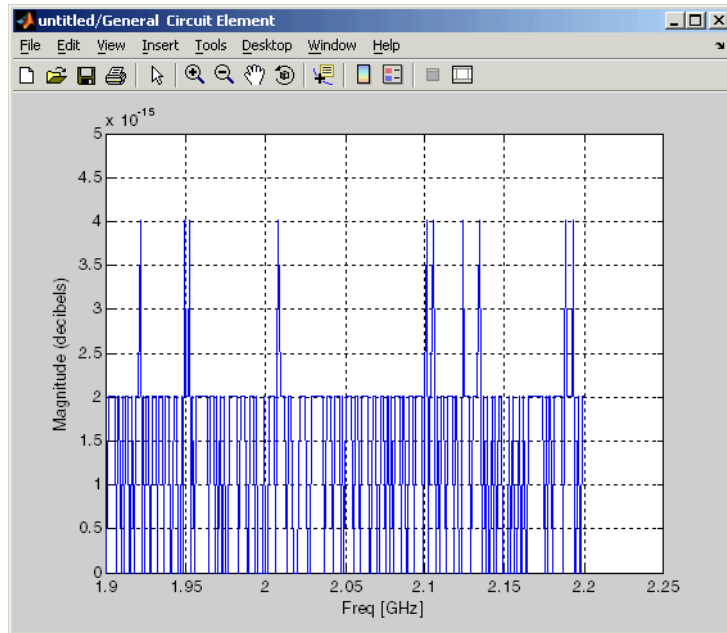
Examples

Creating a General Circuit Element from an RF Toolbox Object

This example uses the `rfckt.txline` object, which describes a transmission line.



The plot parameters in the dialog box request an X-Y plane plot of the S12 parameters in the frequency range 1.9 to 2.2 GHz.



See Also

General Passive Network, S-Parameters Passive Network, Y-Parameters
Passive Network, Z-Parameters Passive Network

rfckt (RF Toolbox)

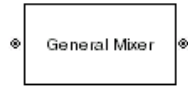
interp1 (MATLAB)

General Mixer

Purpose Model mixer described by `rfdata` object

Library Mixer sublibrary of the Physical library

Description The General Mixer block models the mixer described by an RF Toolbox data (`rfdata.data`) object.



Network Parameters

The network parameter values all refer to the mixer input frequency. If network parameter data and their corresponding frequencies exist as S-parameters in the `rfdata.data` object, the General Mixer block interpolates the S-parameters to determine their values at the modeling frequencies. If the block contains network Y- or Z-parameters, the block first converts them to S-parameters. See “Mapping Network Parameters to Modeling Frequencies” on page A-5 for more details.

The RF Blockset computes the normalized reflected amplitude at the mixer input (E_{r1}) and at the mixer output (E_{r2}) from the interpolated S-parameters as

$$\begin{aligned} E_{r1}(f_{in}) &= S_{11}(f_{in})E_{i1}(f_{in}) + S_{12}(f_{in})E_{i2}(f_{out}) \\ E_{r2}(f_{out}) &= S_{21}(f_{in})E_{i1}(f_{in}) + S_{22}(f_{in})E_{i2}(f_{out}) \end{aligned}$$

where

- f_{in} and f_{out} are the mixer input and output frequencies, respectively.
- E_{i1} and E_{i2} are the normalized incident amplitudes at the mixer input and output, respectively.

The interpolated S21 parameter values describe the conversion gain as a function of frequency, referred to the mixer input frequency.

Active Noise

If active spot noise data exists in the `rfdata.data` object, the block uses the data to calculate the noise figure. It first interpolates the noise

data for the modeling frequencies, using the specified **Interpolation method**. It then calculates the noise figure using the resulting values.

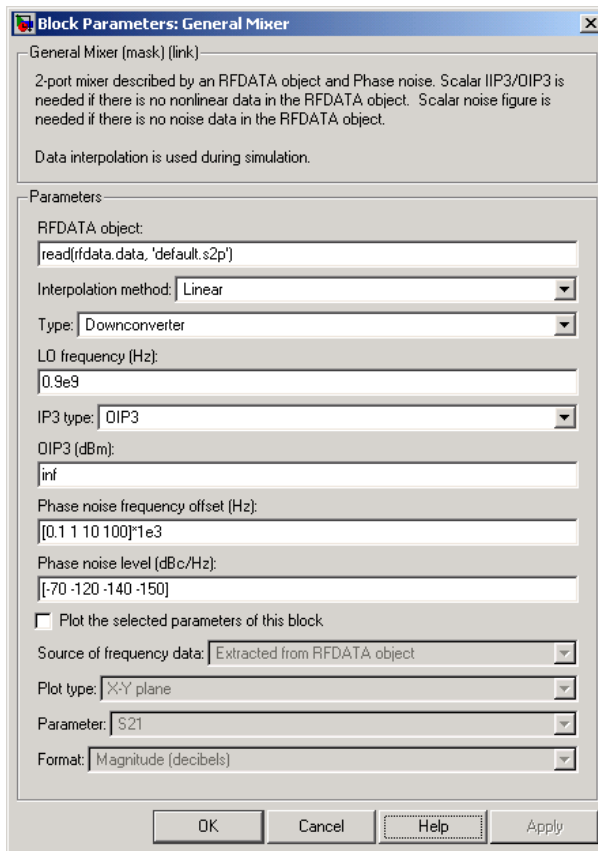
If the `rfdata.data` object contains no noise data, the General Mixer block dialog lets you enter a value for the noise figure.

Phase Noise

The General Mixer block applies phase noise to a complex baseband signal. The block first applies generates additive white Gaussian noise (AWGN) and filters it with a digital filter. It then adds the resulting noise to the angle component of the input signal.

General Mixer

Dialog Box



RFDATA object

An RF Toolbox data object (`rfdata.data`) that describes a mixer. You can specify the object as (1) the handle of a data object previously created using the RF Toolbox, (2) an RF Toolbox command such as `rfdata.data` that creates a default data object, or (3) a MATLAB expression that generates such an object. See the RF Toolbox documentation for more information about data objects.

Interpolation method

For network data, the method used to interpolate the parameters contained in the `rfddata.data` object. Interpolation can be Cubic, Linear (default), or Spline.

Type

Type of mixer. Choices are Downconverter (default) and Upconverter.

LO frequency (Hz)

Local oscillator frequency. If you choose Downconverter, the RF Blockset computes the mixer output frequency, f_{out} , from the mixer input frequency, f_{in} , and the local oscillator frequency, f_{lo} , as $f_{out} = f_{in} - f_{lo}$. If you choose Upconverter, $f_{out} = f_{in} + f_{lo}$.

Note The mixer output frequency must be positive. This means that if you choose a downconverting mixer, f_{in} must be greater than f_{lo} . Otherwise, an error appears.

IP3 type

Type of third-order intercept point. The value can be IIP3 (input intercept point) or OIP3 (output intercept point). This parameter becomes visible only if the `rfddata.data` object contains no power data.

IIP3 (dBm)

Input power intercept point as a scalar value. This field becomes visible if you select IIP3 as the IP3 type. This parameter becomes visible if the `rfddata.data` object contains no power data and you select IIP3 as the IP3 type.

OIP3 (dBm)

Output power intercept point as a scalar value. This field becomes visible if you select OIP3 as the IP3 type. This parameter becomes

General Mixer

visible if the `rfdata.data` object contains no power data and you select OIP3 as the IP3 type.

Noise figure (dB)

Scalar ratio of the available signal-to-noise power ratio at the input to the available signal-to-noise power ratio at the output, $(S_i/N_i)/(S_o/N_o)$. This parameter becomes visible only if the `rfdata.data` object contains no noise data.

Phase noise frequency offset (Hz)

Vector specifying the frequency offset.

Phase noise level (dBc/Hz)

Vector specifying the phase noise level.

Note For information about plotting the mixer parameters, see Chapter 3, “Plotting Model Data”. Use RF Tool or the RF Toolbox plotting functions to plot other data.

See Also

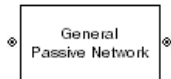
Output Port, S-Parameters Mixer, Y-Parameters Mixer, Z-Parameters Mixer

`rfdata`, `rfdata.data` (RF Toolbox)

Purpose Model two-port passive network described by `rfddata` object

Library Black Box Elements sublibrary of the Physical library

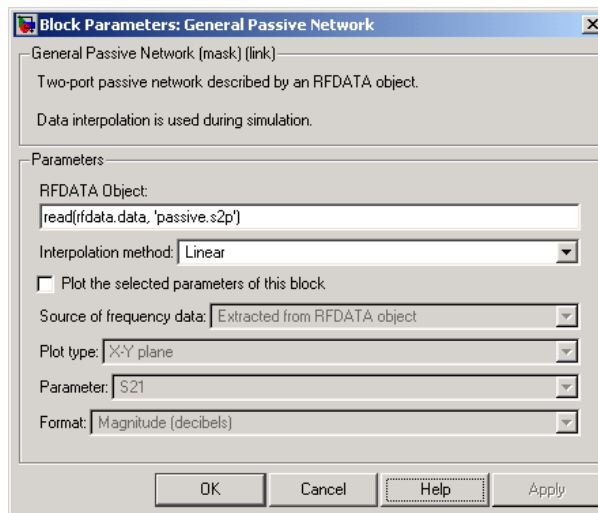
Description



The General Passive Network block models the two-port passive network described by an RF Toolbox data (`rfddata.data`) object.

If network parameter data and their corresponding frequencies exist as S-parameters in the `rfddata.data` object, the General Passive Network block interpolates the S-parameters to determine their values at the modeling frequencies. If the block contains network Y- or Z-parameters, the block first converts them to S-parameters. See “Mapping Network Parameters to Modeling Frequencies” on page A-5 for more details.

Dialog Box



RFDATA object

An RF Toolbox data (`rfddata.data`) object. You can specify the object as (1) the handle of a data object previously created using the RF Toolbox, (2) an RF Toolbox command such as `rfddata.data` that creates a default data object, or (3) a MATLAB expression

General Passive Network

that generates such an object. See the RF Toolbox documentation for more information about data objects.

Interpolation method

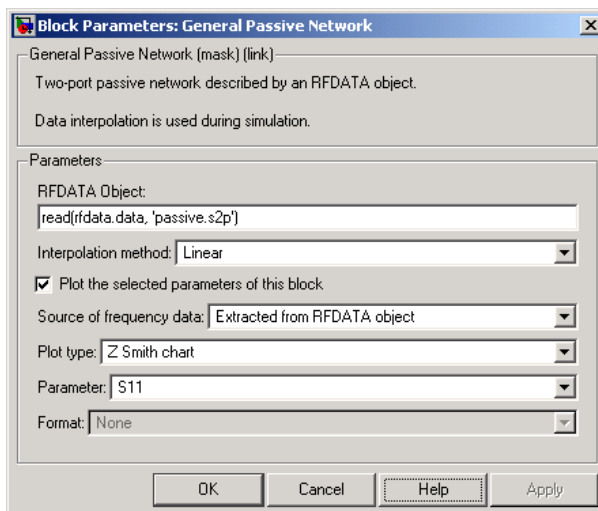
Method used to interpolate the parameters contained in the `rfddata` object. Interpolation can be Linear (default), Spline, or Cubic.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

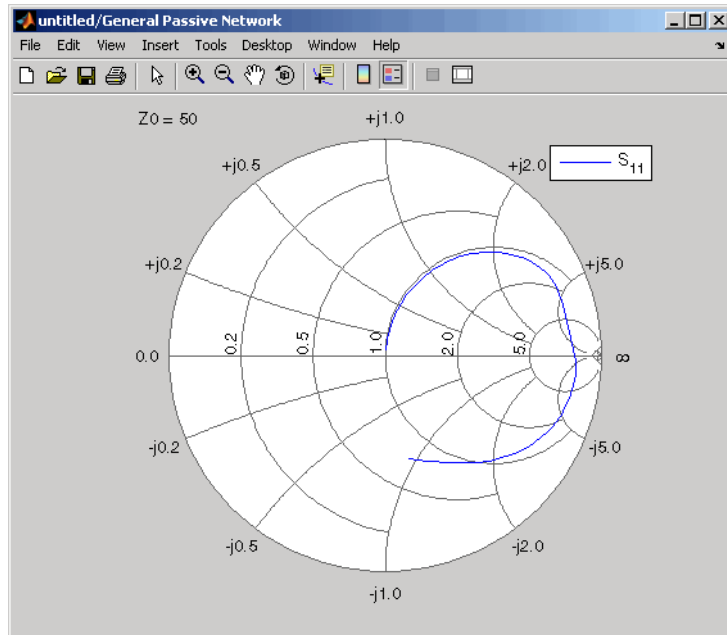
Examples

Creating a General Passive Network Block from File Data

This example uses the RF Toolbox `read` function to create an `rfddata.data` object that describes the two-port passive network in the file `passive.s2p`. The file contains S-parameters for frequencies from about 0.315 MHz to 6.0 GHz. The General Passive Network block uses linear interpolation to model the network described in the object.



The plot parameters in the dialog box request a Z Smith chart of the S_{11} parameters using the frequencies taken from the **RFDATA** object parameter.



See Also

General Circuit Element, Output Port, S-Parameters Passive Network, Y-Parameters Passive Network, Z-Parameters Passive Network

rfddata, rfddata.data (RF Toolbox)

interp1 (MATLAB)

Highpass RF Filter

Purpose Standard highpass RF filters in baseband-equivalent complex form

Library Mathematical

Description The Highpass RF Filter block lets you design standard analog highpass filters, implemented in baseband-equivalent complex form. The following table describes the available design methods.



| Design Method | Description |
|---------------|--|
| Butterworth | The magnitude response of a Butterworth filter is maximally flat in the passband and monotonic overall. |
| Chebyshev I | The magnitude response of a Chebyshev I filter is equiripple in the passband and monotonic in the stopband. |
| Chebyshev II | The magnitude response of a Chebyshev II filter is monotonic in the passband and equiripple in the stopband. |
| Elliptic | The magnitude response of an elliptic filter is equiripple in both the passband and the stopband. |
| Bessel | The delay of a Bessel filter is maximally flat in the passband. |

The block input must be a discrete-time complex signal.

Note This block assumes a nominal impedance of 1 ohm.

Select the design of the filter from the **Design method** list in the dialog box. For each design method, the block lets you specify the filter design parameters shown in the following table.

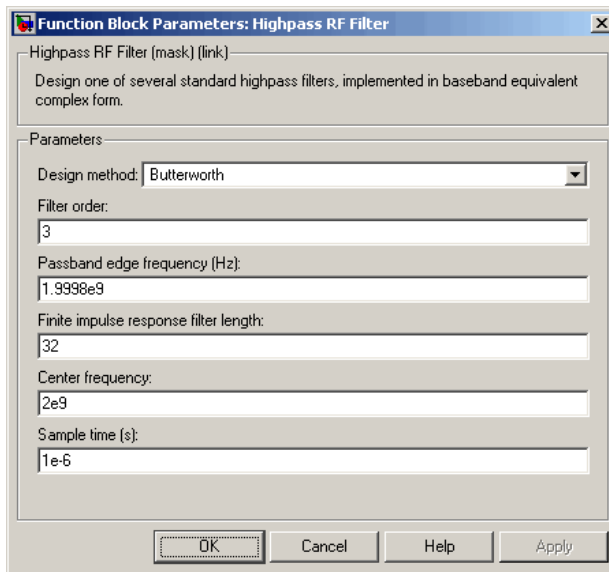
| Design Method | Filter Design Parameters |
|---------------|---|
| Butterworth | Order, passband edge frequency |
| Chebyshev I | Order, passband edge frequency, passband ripple |
| Chebyshev II | Order, stopband edge frequency, stopband attenuation |
| Elliptic | Order, passband edge frequency, passband ripple, stopband attenuation |
| Bessel | Order, passband edge frequency |

The Highpass RF Filter block designs the filters using the Signal Processing Toolbox filter design functions `buttap`, `cheb1ap`, `cheb2ap`, `ellipap`, and `besselap`.

Note Some RF blocks require the sample time to perform baseband modeling calculations. To ensure the accuracy of these calculations, the Input Port block, as well as the mathematical RF blocks, compare the input sample time to the sample time you provide in the mask. If they do not match, or if the input sample time is missing because the blocks are not connected, an error message appears.

Highpass RF Filter

Dialog Box



The parameters displayed in the dialog box vary for different design methods. Only some of these parameters are visible in the dialog box at any one time.

You can change tunable parameters while the model is running.

Design method

Filter design method. The design method can be Butterworth, Chebyshev I, Chebyshev II, Elliptic, or Bessel. Tunable.

Filter order

Order of the filter.

Passband edge frequency (Hz)

Passband edge frequency for Butterworth, Chebyshev I, elliptic, and Bessel designs. Tunable.

Stopband edge frequency (Hz)

Stopband edge frequency for Chebyshev II designs. Tunable.

Passband ripple in dB

Passband ripple for Chebyshev I and elliptic designs. Tunable.

Stopband attenuation in dB

Stopband attenuation for Chebyshev II and elliptic designs. Tunable.

Finite impulse response filter length

Desired length of the baseband-equivalent impulse response for the filter.

Center frequency (Hz)

Center of the modeling frequencies.

Sample time

Time interval between consecutive samples of the input signal.

See Also

Amplifier, Bandpass RF Filter, Bandstop RF Filter, Lowpass RF Filter, Mixer

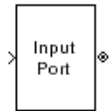
buttap, cheb1ap, cheb2ap, ellipap, besslap (Signal Processing Toolbox)

Input Port

Purpose Connection block from Simulink environment to RF physical blocks

Library Input/Output Ports sublibrary of the Physical library

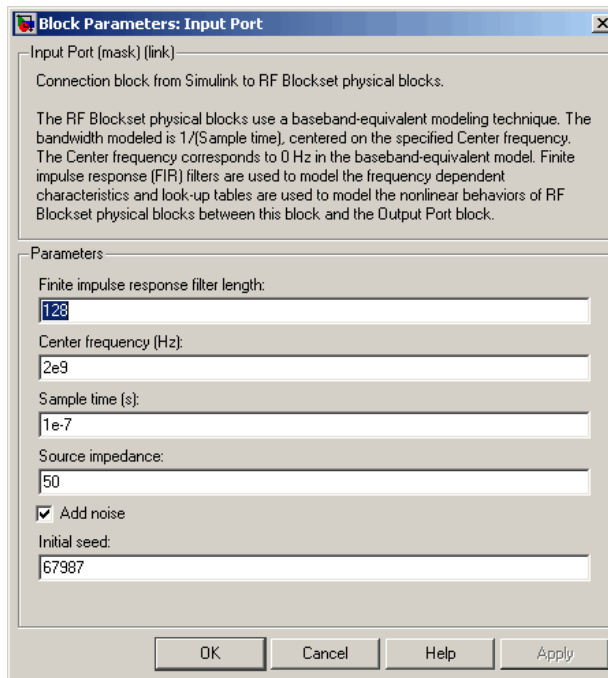
Description



The Input Port block serves as a connecting port from the Simulink, or mathematical, part of the model to an RF physical part of the model. The Input Port block lets you provide the parameter data needed to calculate the modeling frequencies and the baseband-equivalent impulse response for the physical subsystem.

For more information about how the Input Port block converts the mathematical Simulink signals to RF Blockset physical modeling environment signals, see “Converting to and from Simulink Signals” on page A-15. For more information about connecting mathematical and physical parts of a model, see Chapter 2, “Modeling an RF System”.

Note Some RF blocks require the sample time to perform baseband modeling calculations. To ensure the accuracy of these calculations, the Input Port block, as well as the mathematical RF blocks, compare the input sample time to the sample time you provide in the Input Port mask. If they do not match, or if the input sample time is missing because the blocks are not connected, an error message appears.

**Dialog
Box****Finite impulse response filter length**

Desired length of the baseband-equivalent impulse response for the physical model. The longer the FIR filter in the time-domain, the finer the frequency resolution in the frequency domain.

The frequency resolution is approximately equal to $1/(\text{finite impulse response filter length} \times \text{sample time})$. For a graphical representation of this parameter, see “Baseband-Equivalent Modeling” on page A-10.

Center frequency (Hz)

Center of the modeling frequencies. See the Output Port block reference page for information about calculating the modeling frequencies.

Input Port

Sample time (s)

Time interval between consecutive samples of the input signal.

Source impedance

Source impedance of the RF network described in the physical model to which it connects.

Add noise

If you select this parameter, noise data in the RF physical blocks that are bracketed by the Input Port block and Output Port block is taken into consideration. If you do not select this block, noise data is ignored.

Initial seed

Nonnegative integer specifying the initial seed for the random number generator the block uses to generate noise. This parameter becomes visible if you select the **Add noise** parameter.

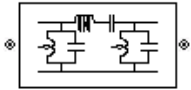
See Also

Output Port

Purpose Model LC bandpass pi network

Library Ladder Filters sublibrary of the Physical library

Description



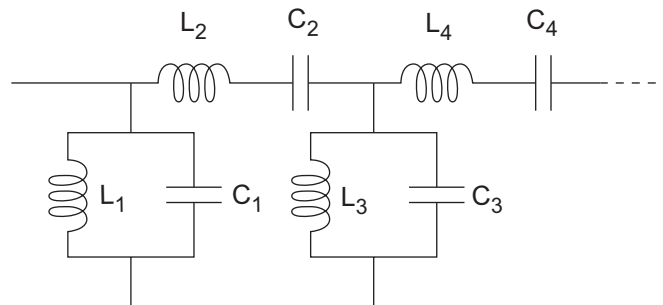
The LC Bandpass Pi block models the LC bandpass pi network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor pair in the network, the block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. For each series pair, $A = 1$, $B = Z$, $C = 0$, and $D = 1$, where Z is the impedance of the series pair. For each shunt pair, $A = 1$, $B = 0$, $C = Y$, and $D = 1$, where Y is the admittance of the shunt pair.

The LC Bandpass Pi block then cascades the ABCD-parameters for each series and shunt pair at each of the modeling frequencies, and converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

See the Output Port block for information about determining the modeling frequencies.

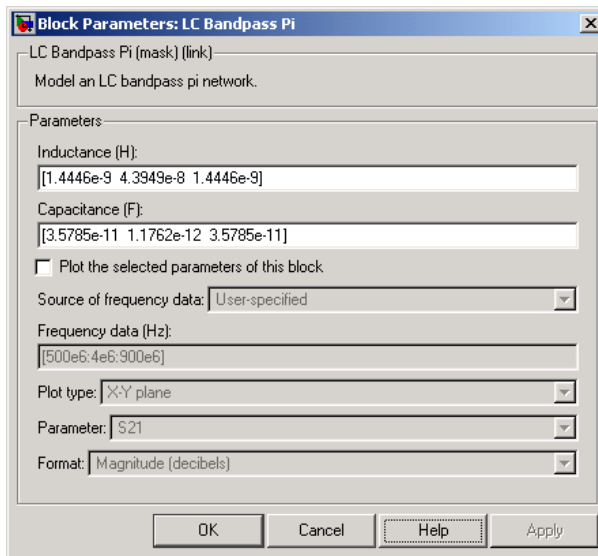
The LC bandpass pi network object is a two-port network as shown in the circuit diagram below.



$[L_1, L_2, L_3, L_4, \dots]$ is the value of the 'L' property, and $[C_1, C_2, C_3, C_4, \dots]$ is the value of the 'C' property.

LC Bandpass Pi

Dialog Box



Inductance (H)

Vector containing the inductances, in order from source to load, of all inductors in the network. The inductance vector must contain at least three elements. All values must be strictly positive.

Capacitance (F)

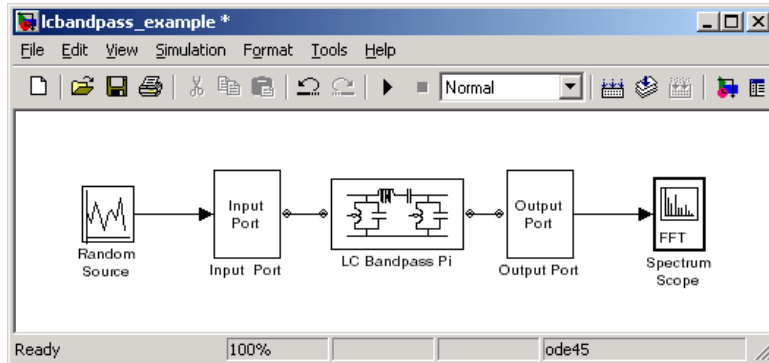
Vector containing the capacitances, in order from source to load, of all capacitors in the network. Its length must be equal to the length of the vector you provide in the **Inductance** parameter. All values must be strictly positive.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

Examples

Using a Ladder Filter Block to Filter Gaussian Noise

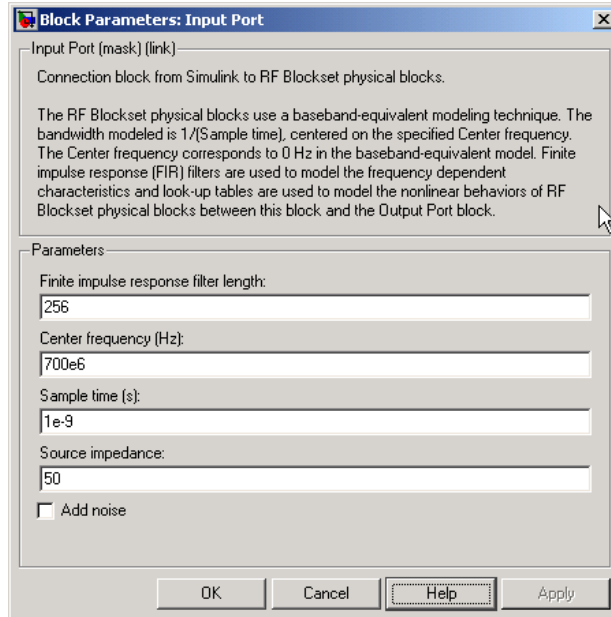
This example provides complex random noise in Gaussian form as input to an LC Bandpass Pi block. A Spectrum Scope block (Signal Processing Blockset) plots the filtered output.



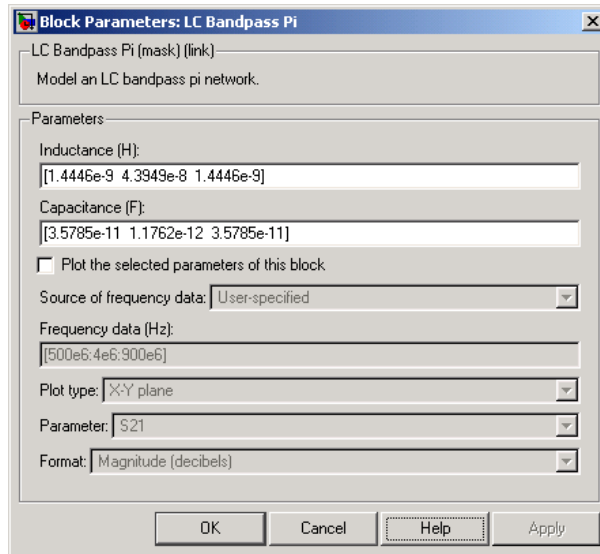
The Random Source block (Signal Processing Blockset) produces frame-based output at 512 samples per frame. Its **Sample time** parameter is set to 1.0e-9. This sample time must match the sample time for the physical part of the model, which you provide in the Input Port block diagram.

LC Bandpass Pi

The Input Port block specifies **Finite impulse response filter length** as 256, **Center frequency** as 700.0e6 Hz, **Sample time** as 1.0e-9, and **Source impedance** as 50 ohms.



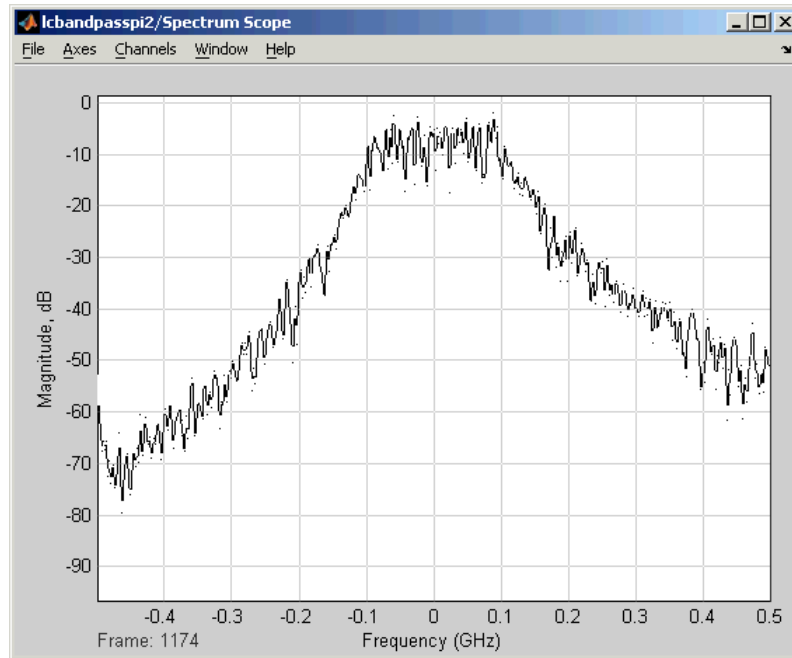
The LC Bandpass Pi block provides the inductances for three inductors, in order from source to load, [1.4446e-9, 4.3949e-8, 1.4446e-9]. Similarly, it provides the capacitances for three capacitors [3.5785e-11, 1.1762e-12, 3.5785e-11].



The following plot shows a sample of the baseband-equivalent RF signal generated by this LC Bandpass Pi block. Zero (0) on the frequency axis corresponds to the center frequency specified in the Input Port block. The bandwidth of the frequency spectrum is $1/\text{sample time}$. You specify the **Sample time** parameter in the Input Port block.

LC Bandpass Pi

The Axis Properties of the Spectrum Scope block have been adjusted to show the frequencies above and below the carrier. The **Minimum Y-limit** parameter is -90, and **Maximum Y-limit** is 0.



References

- [1] Ludwig, Reinhold and Pavel Bretchko, *RF Circuit Design: Theory and Applications*, Prentice-Hall, 2000.
- [2] Zverev, Anatol I., *Handbook of Filter Synthesis*, John Wiley & Sons, 1967.

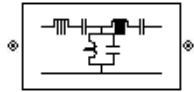
See Also

General Passive Network, LC Bandpass Tee, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Pi, LC Highpass Tee, LC Lowpass Pi, LC Lowpass Tee, Series RLC, Shunt RLC

Purpose Model LC bandpass tee network

Library Ladder Filters sublibrary of the Physical library

Description



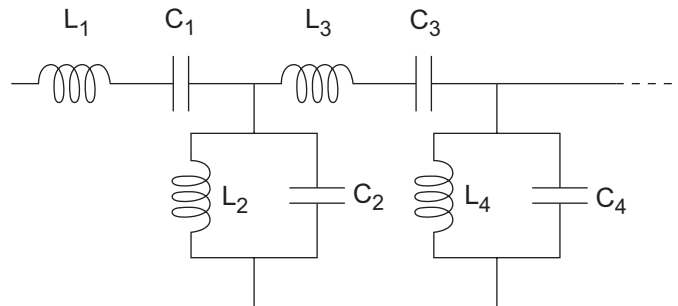
The LC Bandpass Tee block models the LC bandpass tee network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor pair in the network, the block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. For each series pair, $A = 1$, $B = Z$, $C = 0$, and $D = 1$, where Z is the impedance of the series pair. For each shunt pair, $A = 1$, $B = 0$, $C = Y$, and $D = 1$, where Y is the admittance of the shunt pair.

The LC Bandpass Tee block then cascades the ABCD-parameters for each series and shunt pair at each of the modeling frequencies, and converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

See the Output Port block reference page for information about determining the modeling frequencies.

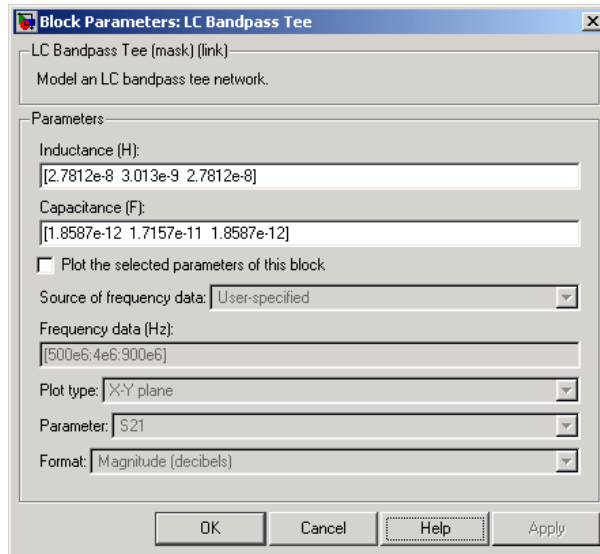
The LC bandpass tee network object is a two-port network as shown in the circuit diagram below.



$[L_1, L_2, L_3, L_4, \dots]$ is the value of the 'L' property, and $[C_1, C_2, C_3, C_4, \dots]$ is the value of the 'C' property.

LC Bandpass Tee

Dialog Box



Inductance (H)

Vector containing the inductances, in order from source to load, of all inductors in the network. The inductance vector must contain at least three elements. All values must be strictly positive.

Capacitance (F)

Vector containing the capacitances, in order from source to load, of all capacitors in the network. Its length must be equal to the length of the vector you provide in the **Inductance** parameter. All values must be strictly positive.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

Examples

See the LC Bandpass Pi block for an example of an LC filter.

References

[1] Ludwig, Reinhold and Pavel Bretchko, *RF Circuit Design: Theory and Applications*, Prentice-Hall, 2000.

[2] Zverev, Anatol I., *Handbook of Filter Synthesis*, John Wiley & Sons, 1967.

See Also

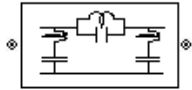
General Passive Network, LC Bandpass Pi, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Pi, LC Highpass Tee, LC Lowpass Pi, LC Lowpass Tee, Series RLC, Shunt RLC

LC Bandstop Pi

Purpose Model LC bandstop pi network

Library Ladder Filters sublibrary of the Physical library

Description



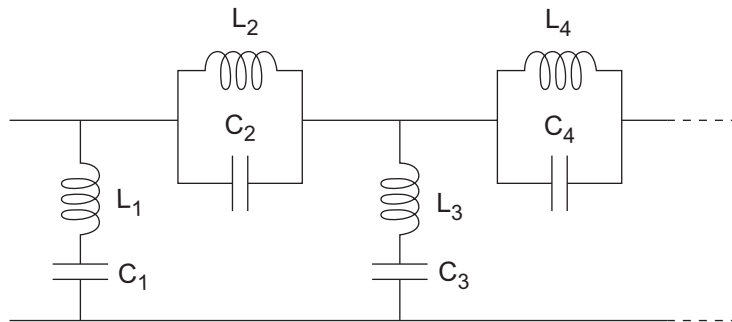
The LC Bandstop Pi block models the LC bandstop pi network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor pair in the network, the block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. For each series pair, $A = 1$, $B = Z$, $C = 0$, and $D = 1$, where Z is the impedance of the series pair. For each shunt pair, $A = 1$, $B = 0$, $C = Y$, and $D = 1$, where Y is the admittance of the shunt pair.

The LC Bandstop Pi block then cascades the ABCD-parameters for each series and shunt pair at each of the modeling frequencies, and converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

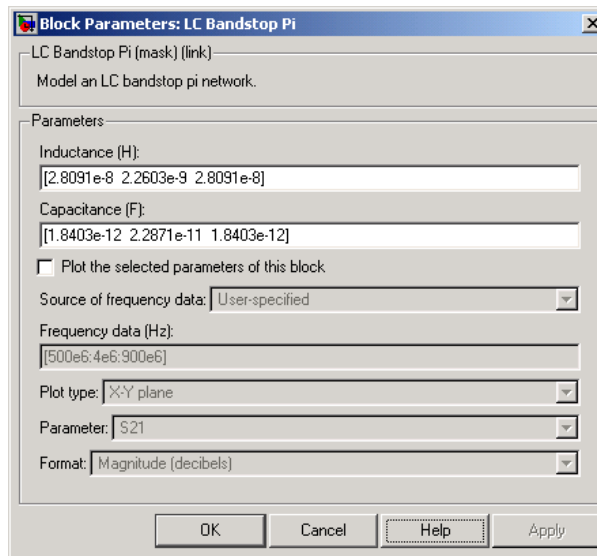
See the Output Port block for information about determining the modeling frequencies.

The LC bandstop pi network object is a two-port network as shown in the circuit diagram below.



$[L_1, L_2, L_3, L_4, \dots]$ is the value of the 'L' property, and $[C_1, C_2, C_3, C_4, \dots]$ is the value of the 'C' property.

Dialog Box



Inductance (H)

Vector containing the inductances, in order from source to load, of all inductors in the network. The inductance vector must contain at least three elements. All values must be strictly positive.

Capacitance (F)

Vector containing the capacitances, in order from source to load, of all capacitors in the network. Its length must be equal to the length of the vector you provide in the **Inductance** parameter. All values must be strictly positive.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

Examples

See the LC Bandpass Pi block for an example of an LC filter.

LC Bandstop Pi

References

[1] Ludwig, Reinhold and Pavel Bretchko, *RF Circuit Design: Theory and Applications*, Prentice-Hall, 2000.

[2] Zverev, Anatol I., *Handbook of Filter Synthesis*, John Wiley & Sons, 1967.

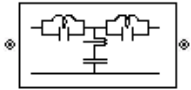
See Also

General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Tee, LC Highpass Pi, LC Highpass Tee, LC Lowpass Pi, LC Lowpass Tee, Series RLC, Shunt RLC

Purpose Model LC bandstop tee network

Library Ladder Filters sublibrary of the Physical library

Description



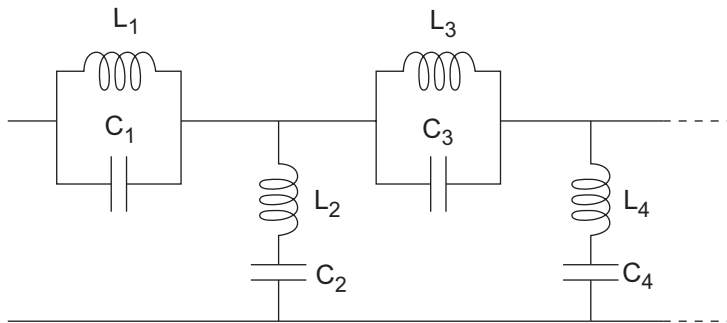
The LC Bandstop Tee block models the LC bandstop tee network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor pair in the network, the block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. For each series pair, $A = 1$, $B = Z$, $C = 0$, and $D = 1$, where Z is the impedance of the series pair. For each shunt pair, $A = 1$, $B = 0$, $C = Y$, and $D = 1$, where Y is the admittance of the shunt pair.

The LC Bandstop Tee block then cascades the ABCD-parameters for each series and shunt pair at each of the modeling frequencies, and converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

See the Output Port block for information about determining the modeling frequencies.

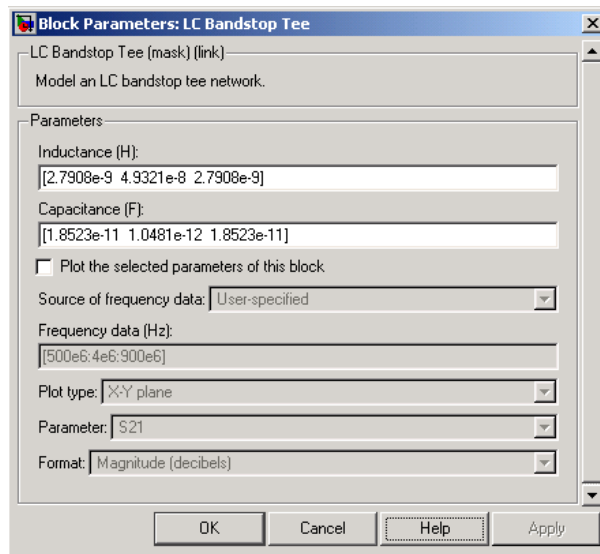
The LC bandstop tee network object is a two-port network as shown in the circuit diagram below.



$[L_1, L_2, L_3, L_4, \dots]$ is the value of the 'L' property, and $[C_1, C_2, C_3, C_4, \dots]$ is the value of the 'C' property.

LC Bandstop Tee

Dialog Box



Inductance (H)

Vector containing the inductances, in order from source to load, of all inductors in the network. The inductance vector must contain at least three elements. All values must be strictly positive.

Capacitance (F)

Vector containing the capacitances, in order from source to load, of all capacitors in the network. Its length must be equal to the length of the vector you provide in the **Inductance** parameter. All values must be strictly positive.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

Examples

See the LC Bandpass Pi block for an example of an LC filter.

References

[1] Ludwig, Reinhold and Pavel Bretchko, *RF Circuit Design: Theory and Applications*, Prentice-Hall, 2000.

[2] Zverev, Anatol I., *Handbook of Filter Synthesis*, John Wiley & Sons, 1967.

See Also

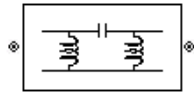
General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Pi, LC Highpass Pi, LC Highpass Tee, LC Lowpass Pi, LC Lowpass Tee, Series RLC, Shunt RLC

LC Highpass Pi

Purpose Model LC highpass pi network

Library Ladder Filters sublibrary of the Physical library

Description



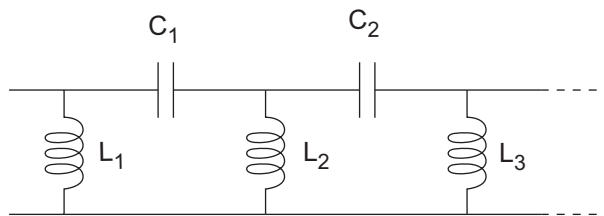
The LC Highpass Pi block models the LC highpass pi network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor in the network, the block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. For each series circuit, $A = 1$, $B = Z$, $C = 0$, and $D = 1$, where Z is the impedance of the series circuit. For each shunt, $A = 1$, $B = 0$, $C = Y$, and $D = 1$, where Y is the admittance of the shunt circuit.

The LC Highpass Pi block then cascades the ABCD-parameters for each circuit element at each of the modeling frequencies, and converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

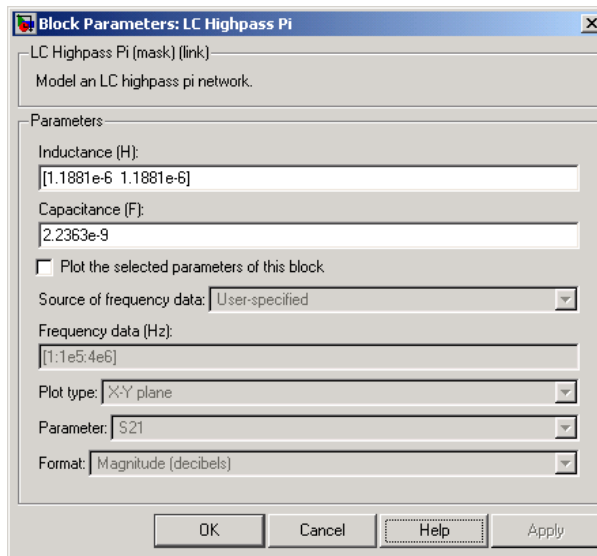
See the Output Port block reference page for information about determining the modeling frequencies.

The LC highpass pi network object is a two-port network as shown in the circuit diagram below.



$[L_1, L_2, L_3, \dots]$ is the value of the 'L' property, and $[C_1, C_2, \dots]$ is the value of the 'C' property.

Dialog Box



Inductance (H)

Vector containing the inductances, in order from source to load, of all inductors in the network. The inductance vector must contain at least two elements. All values must be strictly positive.

Capacitance (F)

Vector containing the capacitances, in order from source to load, of all capacitors in the network. Its length must be equal to or one less than the length of the vector you provide in the **Inductance** parameter. All values must be strictly positive.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

Examples

See the LC Bandpass Pi block for an example of an LC filter.

LC Highpass Pi

References

[1] Ludwig, Reinhold and Pavel Bretchko, *RF Circuit Design: Theory and Applications*, Prentice-Hall, 2000.

[2] Zverev, Anatol I., *Handbook of Filter Synthesis*, John Wiley & Sons, 1967.

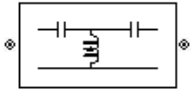
See Also

General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Tee, LC Lowpass Pi, LC Lowpass Tee, Series RLC, Shunt RLC

Purpose Model LC highpass tee network

Library Ladder Filters sublibrary of the Physical library

Description



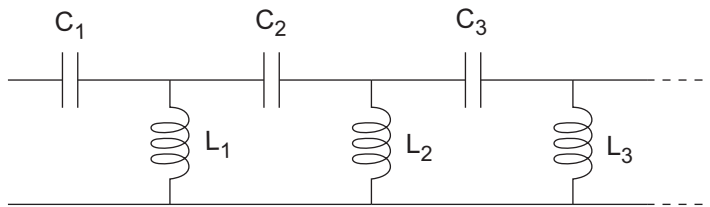
The LC Highpass Tee block models the LC highpass tee network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor in the network, the block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. For each series circuit, $A = 1$, $B = Z$, $C = 0$, and $D = 1$, where Z is the impedance of the series circuit. For each shunt, $A = 1$, $B = 0$, $C = Y$, and $D = 1$, where Y is the admittance of the shunt circuit.

The LC Highpass Tee block then cascades the ABCD-parameters for each circuit element at each of the modeling frequencies, and converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

See the Output Port block reference page for information about determining the modeling frequencies.

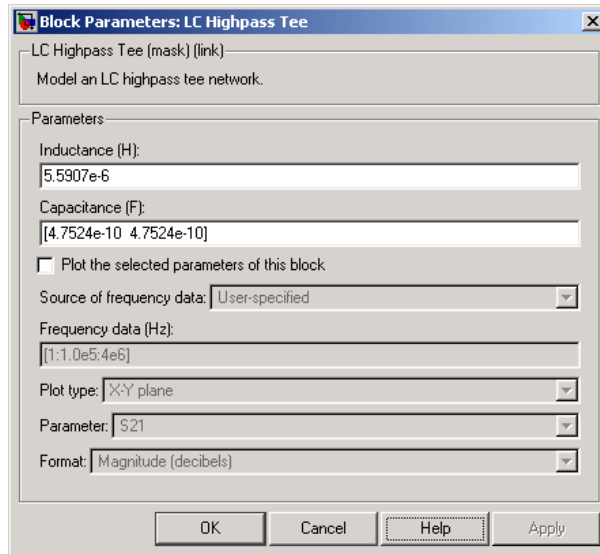
The LC highpass tee network object is a two-port network as shown in the circuit diagram below.



$[L_1, L_2, L_3, \dots]$ is the value of the 'L' property, and $[C_1, C_2, C_3, \dots]$ is the value of the 'C' property.

LC Highpass Tee

Dialog Box



Inductance (H)

Vector containing the inductances, in order from source to load, of all inductors in the network. All values must be strictly positive. The vector cannot be empty.

Capacitance (F)

Vector containing the capacitances, in order from source to load, of all capacitors in the network. The capacitance vector must contain at least two elements. Its length must be equal to or one greater than the length of the vector you provide in the **Inductance** parameter. All values must be strictly positive.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

Examples

See the LC Bandpass Pi block for an example of an LC filter.

References

[1] Ludwig, Reinhold and Pavel Bretchko, *RF Circuit Design: Theory and Applications*, Prentice-Hall, 2000.

[2] Zverev, Anatol I., *Handbook of Filter Synthesis*, John Wiley & Sons, 1967.

See Also

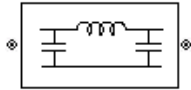
General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Pi, LC Lowpass Pi, LC Lowpass Tee, Series RLC, Shunt RLC

LC Lowpass Pi

Purpose Model LC lowpass pi network

Library Ladder Filters sublibrary of the Physical library

Description



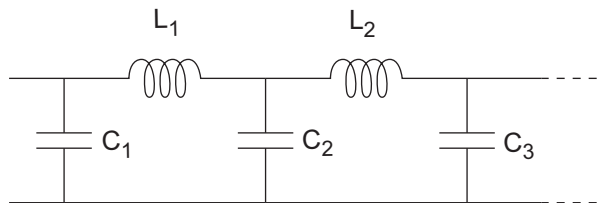
The LC Lowpass Pi block models the LC lowpass pi network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor in the network, the block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. For each series circuit, $A = 1$, $B = Z$, $C = 0$, and $D = 1$, where Z is the impedance of the series circuit. For each shunt, $A = 1$, $B = 0$, $C = Y$, and $D = 1$, where Y is the admittance of the shunt circuit.

The LC Lowpass Pi block then cascades the ABCD-parameters for each circuit element at each of the modeling frequencies, and converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

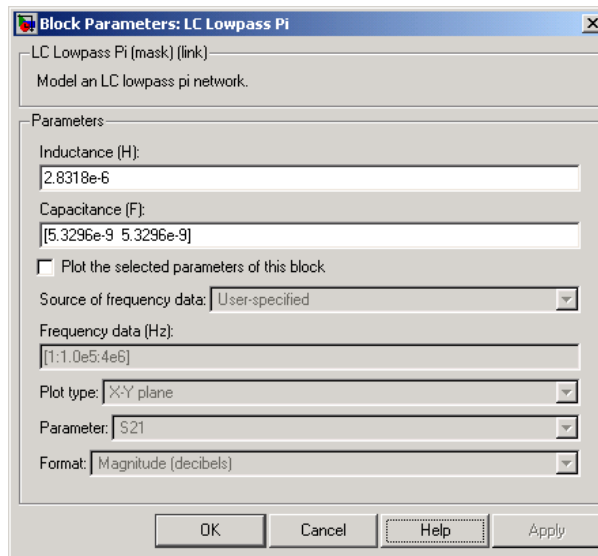
See the Output Port block reference page for information about determining the modeling frequencies.

The LC lowpass pi network object is a two-port network as shown in the circuit diagram below.



$[L_1, L_2, \dots]$ is the value of the 'L' property, and $[C_1, C_2, C_3, \dots]$ is the value of the 'C' property.

Dialog Box



Inductance (H)

Vector containing the inductances, in order from source to load, of all inductors in the network. All values must be strictly positive. The vector cannot be empty.

Capacitance (F)

Vector containing the capacitances, in order from source to load, of all capacitors in the network. The capacitance vector must contain at least two elements. Its length must be equal to or one greater than the length of the vector you provide in the **Inductance** parameter. All values must be strictly positive.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

Examples

See the LC Bandpass Pi block for an example of an LC filter.

LC Lowpass Pi

References

[1] Ludwig, Reinhold and Pavel Bretchko, *RF Circuit Design: Theory and Applications*, Prentice-Hall, 2000.

[2] Zverev, Anatol I., *Handbook of Filter Synthesis*, John Wiley & Sons, 1967.

See Also

General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Pi, LC Highpass Tee, LC Lowpass Tee, Series RLC, Shunt RLC

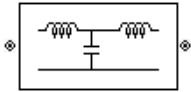
Purpose

Model LC lowpass tee network

Library

Ladders Filters sublibrary of the Physical library

Description



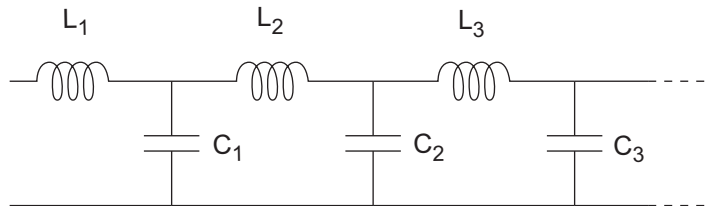
The LC Lowpass Tee block models the LC lowpass tee network described in the block dialog box in terms of its frequency-dependent S-parameters.

For each inductor and capacitor in the network, the block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. For each series circuit, $A = 1$, $B = Z$, $C = 0$, and $D = 1$, where Z is the impedance of the series circuit. For each shunt, $A = 1$, $B = 0$, $C = Y$, and $D = 1$, where Y is the admittance of the shunt circuit.

The LC Lowpass Tee block then cascades the ABCD-parameters for each circuit element at each of the modeling frequencies, and converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

See the Output Port block reference page for information about determining the modeling frequencies.

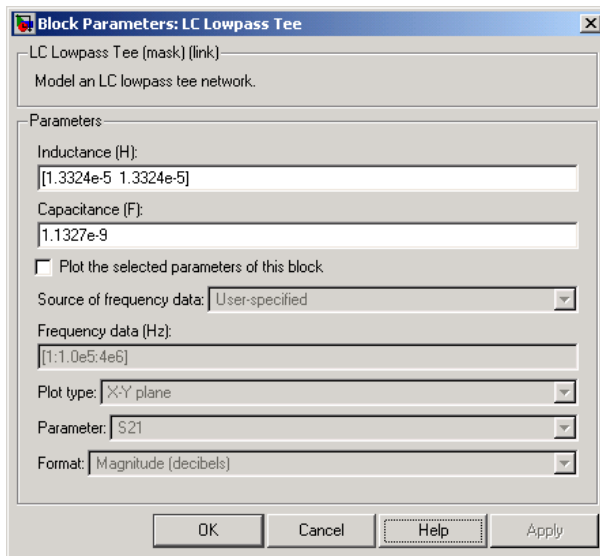
The LC lowpass tee network object is a two-port network as shown in the circuit diagram below.



$[L_1, L_2, L_3, \dots]$ is the value of the 'L' property, and $[C_1, C_2, C_3, \dots]$ is the value of the 'C' property.

LC Lowpass Tee

Dialog Box



Inductance (H)

Vector containing the inductances, in order from source to load, of all inductors in the network. The inductance vector must contain at least two elements. All values must be strictly positive.

Capacitance (F)

Vector containing the capacitances, in order from source to load, of all capacitors in the network. Its length must be equal to or one less than the length of the vector you provide in the **Inductance** parameter. All values must be strictly positive.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

Examples

See the LC Bandpass Pi block for an example of an LC filter.

References

[1] Ludwig, Reinhold and Pavel Bretchko, *RF Circuit Design: Theory and Applications*, Prentice-Hall, 2000.

[2] Zverev, Anatol I., *Handbook of Filter Synthesis*, John Wiley & Sons, 1967.

See Also

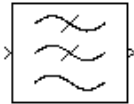
General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Pi, LC Highpass Tee, LC Lowpass Pi, Series RLC, Shunt RLC

Lowpass RF Filter

Purpose Standard lowpass RF filters in baseband-equivalent complex form

Library Mathematical

Description The Lowpass RF Filter block lets you design standard analog lowpass filters, implemented in baseband-equivalent complex form. The following table describes the available design methods.



| Design Method | Description |
|---------------|--|
| Butterworth | The magnitude response of a Butterworth filter is maximally flat in the passband and monotonic overall. |
| Chebyshev I | The magnitude response of a Chebyshev I filter is equiripple in the passband and monotonic in the stopband. |
| Chebyshev II | The magnitude response of a Chebyshev II filter is monotonic in the passband and equiripple in the stopband. |
| Elliptic | The magnitude response of an elliptic filter is equiripple in both the passband and the stopband. |
| Bessel | The delay of a Bessel filter is maximally flat in the passband. |

The block input must be a discrete-time complex signal.

Note This block assumes a nominal impedance of 1 ohm.

Select the design of the filter from the **Design method** list in the dialog box. For each design method, the block enables you to specify the filter design parameters shown in the following table.

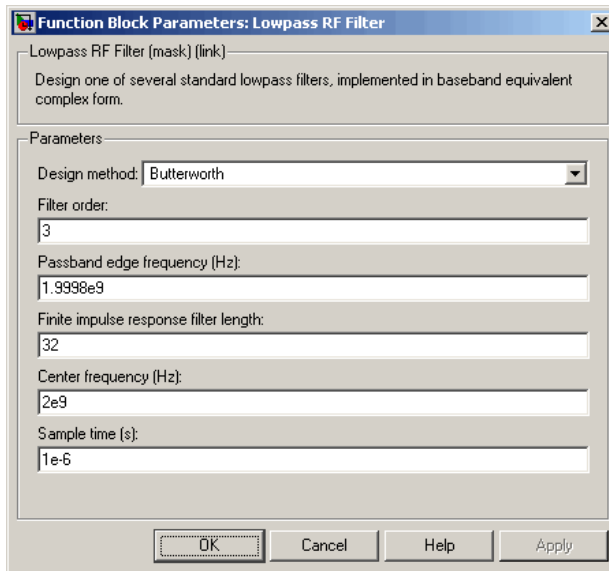
| Design Method | Filter Design Parameters |
|----------------------|---|
| Butterworth | Order, passband edge frequency |
| Chebyshev I | Order, passband edge frequency, passband ripple |
| Chebyshev II | Order, stopband edge frequency, stopband attenuation |
| Elliptic | Order, passband edge frequency, passband ripple, stopband attenuation |
| Bessel | Order, passband edge frequency |

The Lowpass RF Filter block designs the filters using the Signal Processing Toolbox filter design functions `buttap`, `cheb1ap`, `cheb2ap`, `ellipap`, and `besselap`.

Note Some RF blocks require the sample time to perform baseband modeling calculations. To ensure the accuracy of these calculations, the Input Port block, as well as the mathematical RF blocks, compare the input sample time to the sample time you provide in the mask. If they do not match, or if the input sample time is missing because the blocks are not connected, an error message appears.

Lowpass RF Filter

Dialog Box



The parameters displayed in the dialog box vary for different design methods. Only some of these parameters are visible in the dialog box at any one time.

Parameters that are tunable can be changed while the model is running.

Design method

Filter design method. The design method can be Butterworth, Chebyshev I, Chebyshev II, Elliptic, or Bessel. Tunable.

Filter order

Order of the filter.

Passband edge frequency (Hz)

Passband edge frequency for Butterworth, Chebyshev I, elliptic, and Bessel designs. Tunable.

Stopband edge frequency (Hz)

Stopband edge frequency for Chebyshev II designs. Tunable.

Passband ripple in dB

Passband ripple for Chebyshev I and elliptic designs. Tunable.

Stopband attenuation in dB

Stopband attenuation for Chebyshev II and elliptic designs. Tunable.

Finite impulse response filter length

Desired length of the baseband-equivalent impulse response for the filter.

Center frequency (Hz)

Center of the modeling frequencies.

Sample time (s)

Time interval between consecutive samples of the input signal.

See Also

Amplifier, Bandpass RF Filter, Bandstop RF Filter, Highpass RF Filter, Mixer

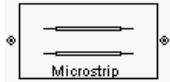
buttap, cheb1ap, cheb2ap, ellipap, besslap (Signal Processing Toolbox)

Microstrip Transmission Line

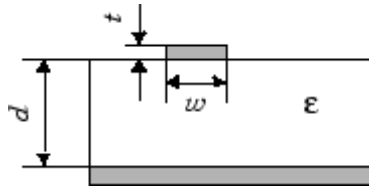
Purpose Model microstrip transmission line

Library Transmission Lines sublibrary of the Physical library

Description



The Microstrip Transmission Line block models the microstrip transmission line described in the block dialog in terms of its frequency-dependent S-parameters. A microstrip transmission line is shown here in cross-section. Its physical characteristics include the microstrip width (w), the microstrip thickness (t), the substrate height (d), and the relative permittivity constant (ϵ).



The block lets you model the transmission line as a stub or as a stubless line.

Stubless Transmission Line

If you model a microstrip transmission line as a stubless line, the Microstrip Transmission Line block calculates the frequency-dependent S-parameters using the physical length of the transmission line, D , and the complex propagation constant, k .

$$S_{11} = 0$$

$$S_{12} = e^{-kD}$$

$$S_{21} = e^{-kD}$$

$$S_{22} = 0$$

$k = \alpha_a + i\beta$, where α_a is the attenuation coefficient and β is the wave number. The attenuation coefficient α_a is related to the loss, α , by

$$\alpha_a = -\ln 10 \frac{\alpha}{20}$$

where α is the reduction in signal strength, in dB, per unit length. α combines both conductor loss and dielectric loss and is derived from the physical parameters specified in the Microstrip Transmission Line block dialog box. The wave number β is related to the phase velocity, V_p , by

$$\beta = \frac{2\pi f}{V_p}$$

where $V_p = c / \sqrt{\epsilon_{\text{eff}}}$, and ϵ_{eff} is the frequency dependent effective dielectric constant. f is the vector of modeling frequencies determined by the Output Port block. The phase velocity V_p is also known as the wave propagation velocity.

The Microstrip Transmission Line block normalizes the resulting S-parameters to a reference impedance of 50 ohms.

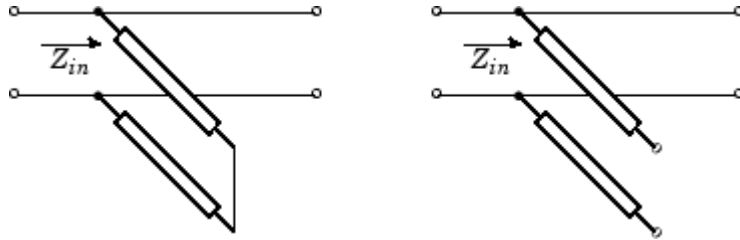
Shunt and Series Stubs

If you model the transmission line as a shunt or series stub, the Microstrip Transmission Line block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. It then uses the `abcd2s` function to convert the ABCD-parameters to S-parameters.

Shunt ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to Shunt, the two-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.

Microstrip Transmission Line



Z_{in} is the input impedance of the shunt circuit. The ABCD-parameters for the shunt stub are calculated as

$$A = 1$$

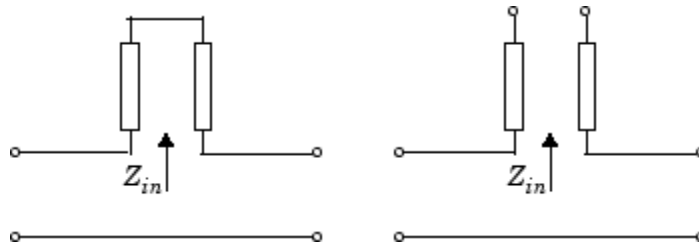
$$B = 0$$

$$C = 1/Z_{in}$$

$$D = 1$$

Series ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to Series, the two-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



Z_{in} is the input impedance of the series circuit. The ABCD-parameters for the series stub are calculated as

$$\begin{aligned}A &= 1 \\B &= Z_{in} \\C &= 0 \\D &= 1\end{aligned}$$

Dialog Box

Block Parameters: Microstrip Transmission Line

Microstrip Transmission Line (mask) (link)
Model a microstrip transmission line.

Parameters

Strip width (m): 0.6e-3

Substrate height (m): 0.635e-3

Strip thickness (m): 0.005e-3

Relative permittivity constant: 9.8

Conductivity in conductor (S/m): inf

Loss tangent in dielectric: 0

Transmission line length (m): 0.01

Stub mode: Not a stub

Plot the selected parameters of this block

Source of frequency data: User-specified

Frequency data (Hz): [1e9;1.0e6;3e9]

Plot type: X-Y plane

Parameter: S21

Format: Angle (degrees)

OK Cancel Help Apply

Strip width (m)

Width of the microstrip transmission line.

Microstrip Transmission Line

Substrate height (m)

Thickness of the dielectric on which the microstrip resides.

Strip thickness (m)

Physical thickness of the microstrip.

Relative permittivity constant

Relative permittivity of the dielectric expressed as the ratio of the permittivity of the dielectric to permittivity in free space ϵ_0 .

Conductivity in conductor (S/m)

Conductivity of the conductor in siemens per meter.

Loss tangent in dielectric

Loss angle tangent of the dielectric.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

Type of stub. Choices are Not a stub, Shunt, or Series.

Termination of stub

Stub termination for stub modes Shunt and Series. Choices are Open or Short. This parameter becomes visible only when **Stub mode** is set to Shunt or Series.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

References

[1] Gupta, K.C., G. Ramesh, I. Bahl, and P. Bhartia, *Microstrip Lines and Slotlines*, Second Edition, Artech House, 1996. pp. 102-109.

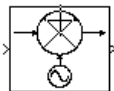
See Also

Coaxial Transmission Line, Coplanar Waveguide Transmission Line, General Passive Network, Transmission Line, Parallel-Plate Transmission Line, Two-Wire Transmission Line

Purpose Complex baseband model of mixer with phase noise

Library Mathematical

Description



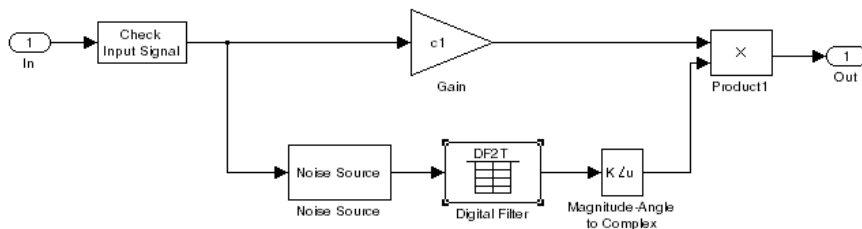
The Mixer block generates a complex baseband model of a mixer, with phase noise whose spectrum is characterized by a $1/f$ slope. The level of the spectrum is specified by the noise power contained in one hertz bandwidth offset from the carrier by a certain frequency.

Note This block assumes a nominal impedance of 1 ohm.

The block applies the phase noise to the signal as follows:

- 1 Generates additive white Gaussian noise (AWGN) and filters it with a digital filter.
- 2 Adds the resulting noise to the angle component of the input signal.

You can view the block's implementation of phase noise by right-clicking the block and selecting **Look under mask** from the pop-up menu. This displays the following figure.

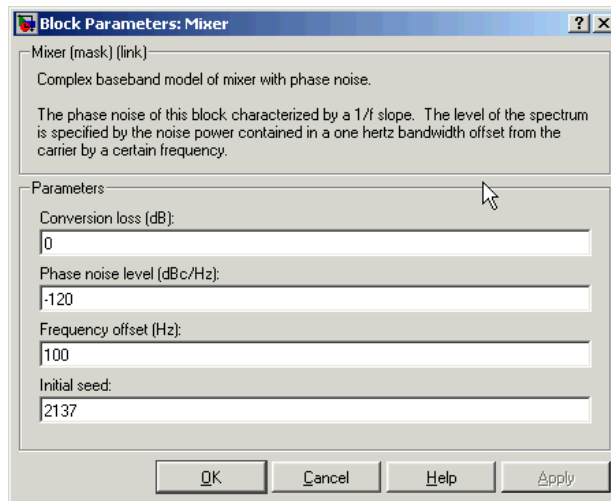


You can view the construction of the Noise Source subsystem by double-clicking it.

Mixer

Note Some RF blocks require the sample time to perform baseband modeling calculations. To ensure the accuracy of these calculations, the Input Port block, as well as the mathematical RF blocks, compare the input sample time to the sample time you provide in the mask. If they do not match, or if the input sample time is missing because the blocks are not connected, an error message appears.

Dialog Box



You can change parameters that are marked as tunable while the model is running.

Conversion loss (dB)

Scalar specifying the conversion loss for the mixer. Tunable.

Phase noise level (dBc/Hz)

Scalar specifying the phase noise level in decibels relative to the carrier, per hertz. Tunable.

Frequency offset (Hz)

Scalar specifying the frequency offset. Tunable.

Initial seed

Nonnegative integer specifying the initial seed for the random number generator the block uses to generate noise.

References

[1] Kasdin, N.J., "Discrete Simulation of Colored Noise and Stochastic Processes and $1/f^\alpha$; Power Law Noise Generation," The Proceedings of the IEEE, May, 1995, Vol. 83, No. 5.

See Also

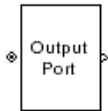
Amplifier, Bandpass RF Filter, Bandstop RF Filter, Highpass RF Filter, Lowpass RF Filter

Output Port

Purpose Connection block from RF physical blocks to Simulink environment

Library Input/Output Ports sublibrary of the Physical library

Description The Output Port block produces the baseband-equivalent time-domain response of an input signal traveling through a series of RF physical components. The Output Port block



- 1** Partitions the RF physical components into linear and nonlinear subsystems.
- 2** Extracts the complex impulse response of the linear subsystem for baseband-equivalent modeling of the RF linear system.
- 3** Extracts the nonlinear AMAM/AMPM modeling for RF nonlinearity.

The Output Port block also serves as a connecting port from an RF physical part of the model to the Simulink, or mathematical, part of the model. For more information about how the Output Port block converts the RF Blockset physical modeling environment signals to mathematical Simulink signals, see “Converting to and from Simulink Signals” on page A-15. For more information about connecting mathematical and physical parts of a model, see Chapter 2, “Modeling an RF System”.

Note Some RF blocks require the sample time to perform baseband modeling calculations. To ensure the accuracy of these calculations, the Input Port block, as well as the mathematical RF blocks, compare the input sample time to the sample time you provide in the mask. If they do not match, or if the input sample time is missing because the blocks are not connected, an error message appears.

Linear Subsystem

For the linear subsystem, the Output Port block uses the Input Port block parameters and the interpolated S-parameters calculated by each

of the cascaded physical blocks to calculate the baseband-equivalent impulse response. Specifically, it

- 1 Determines the modeling frequencies f as an N -element vector. The modeling frequencies are a function of the center frequency f_c , the sample time t_s , and the finite impulse response filter length N , all of which you specify in the Input Port block dialog box.

The n th element of f , f_n , is given by

$$f_n = f_{\min} + \frac{n-1}{t_s N} \quad n = 1, \dots, N$$

where

$$f_{\min} = f_c - \frac{1}{2t_s}$$

- 2 Calculates the passband transfer function for the frequency range as

$$H(f) = \frac{V_L(f)}{V_S(f)}$$

where V_S and V_L are the source and load voltages, and f represents the modeling frequencies. More specifically,

$$H(f) = \frac{S_{21} * (1 + \Gamma_l) * (1 - \Gamma_s)}{2 * (1 - S_{22} * \Gamma_l) * (1 - \Gamma_{in} * \Gamma_s)}$$

Output Port

where

$$\Gamma_l = \frac{Z_l - Z_o}{Z_l + Z_o}$$

$$\Gamma_s = \frac{Z_s - Z_o}{Z_s + Z_o}$$

$$\Gamma_{in} = S_{11} + \left(S_{12} * S_{21} * \frac{\Gamma_l}{(1 - S_{22} * \Gamma_l)} \right)$$

and

- Z_s is the source impedance.
- Z_L is the load impedance.
- S_{ij} are the S-parameters of a two-port network.

The passband transfer function is shown in the following figure.

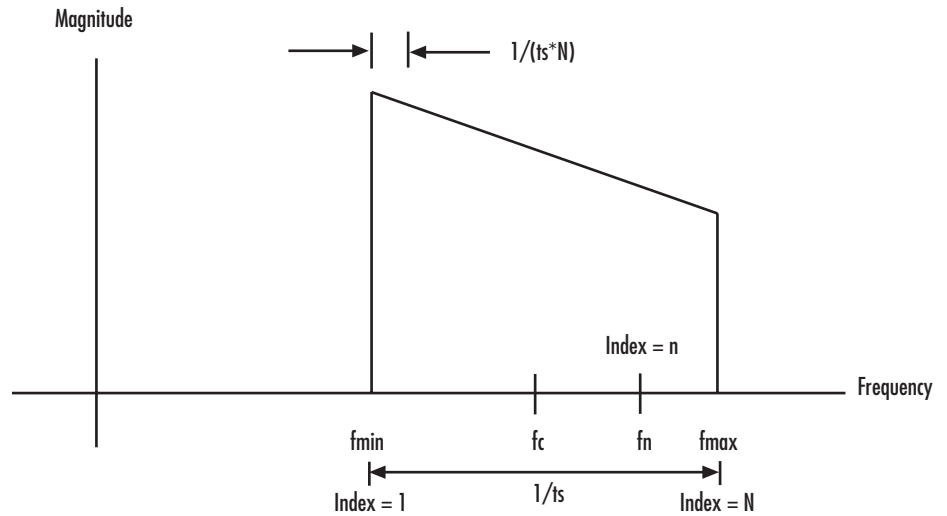
Passband Spectrum of a Modulated RF Carrier

Specify these parameters in the Input Port dialog box:

Finite impulse response filter length = N

Center frequency = f_c

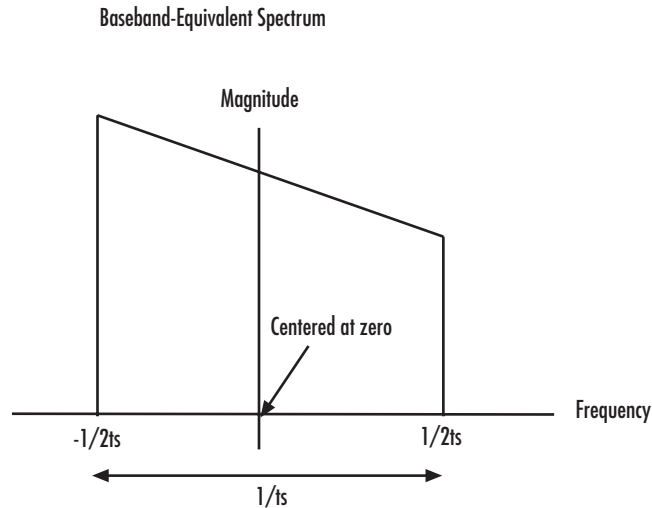
Sample time = t_s



- 3 Translates the passband transfer function to baseband as $H(f - f_c)$, where f_c is the specified center frequency.

Output Port

The baseband transfer function is shown in the following figure.

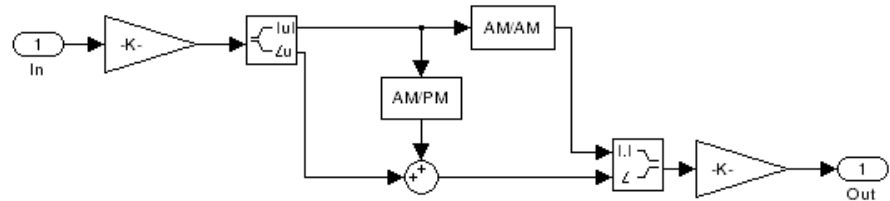


- 4 Obtains the baseband-equivalent impulse response by calculating the inverse FFT of the baseband transfer function. For faster simulation, the block calculates the IFFT using the next power of 2 greater than the specified finite impulse response filter length. Then, it truncates the impulse response to a length equal to the filter length specified.

For the linear subsystem, the Output Port block uses the calculated impulse response as input to the Signal Processing Blockset Digital Filter block to determine the output.

Nonlinear Subsystem

The nonlinear subsystem is implemented by AM/AM and AM/PM nonlinear models, as shown in the figure below.



The nonlinearities of AM/AM and AM/PM conversions are extracted from the power data of an amplifier or mixer by the equations

$$AM_{out} = \sqrt{R_l \cdot P_{out}}$$

$$PM_{out} = \text{Phase}$$

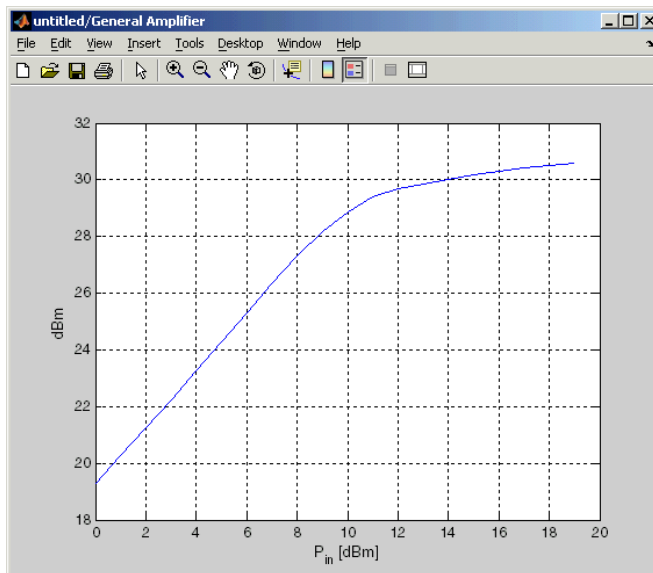
$$AM_{in} = \sqrt{R_s \cdot P_{in}}$$

where AM_{in} is the AM of the input voltage, AM_{out} and PM_{out} are the AM and PM of the output voltage, R_s , is the source resistance (50 ohms), R_l is the load resistance (50 ohms), P_{in} is the input power, P_{out} is the output power, and $Phase$ is the phase shift between the input and output voltage.

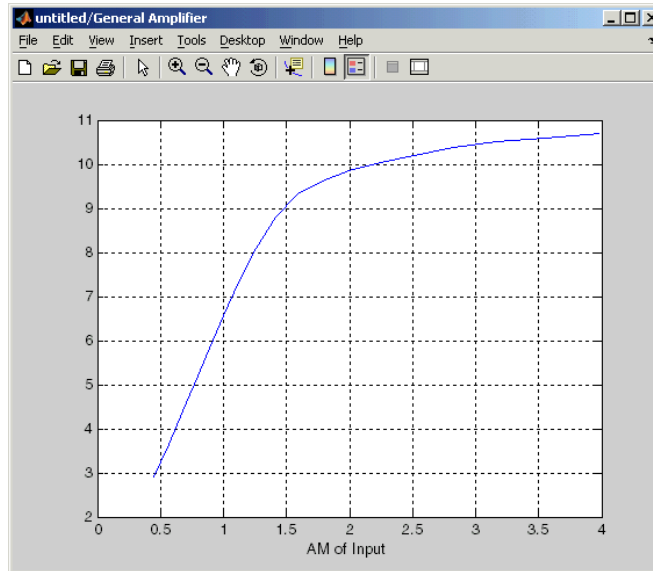
Note You can provide power data via a .amp file. See “AMP File Format” in the RF Toolbox documentation for information about this format.

Output Port

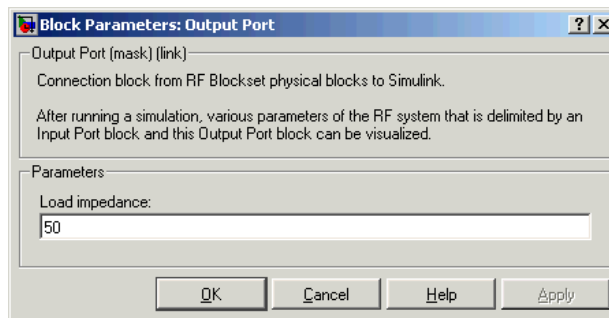
The following figure shows the original power data of an amplifier.



This figure shows the extracted AM/AM nonlinear conversion.



Dialog Box



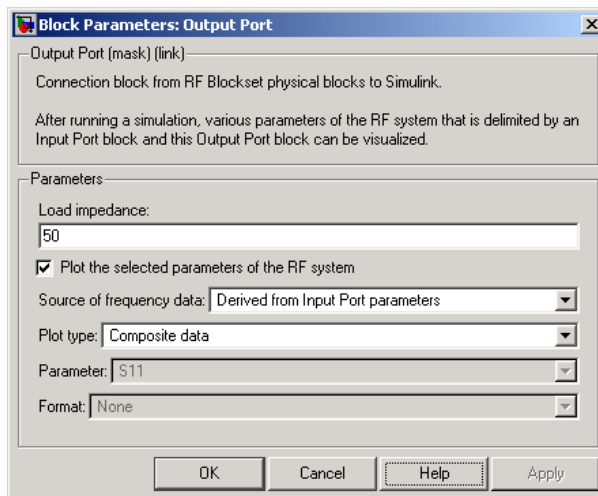
Load impedance

Load impedance of the RF network described in the physical model to which it connects.

Output Port

Plot the selected parameters of the RF system

This parameter and the associated plotting parameters shown below become visible if you display the Output Port mask after you run the model. For more information about plotting, see Chapter 3, “Plotting Model Data”.



See Also

Input Port
s2y (RF Toolbox)

Purpose

Model parallel-plate transmission line

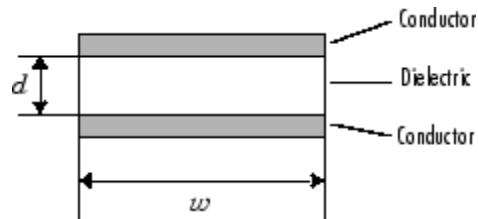
Library

Transmission Lines sublibrary of the Physical library

Description



The Parallel-Plate Transmission Line block models the parallel-plate transmission line described in the block dialog box in terms of its frequency-dependent S-parameters. A parallel-plate transmission line is shown here in cross-section. Its physical characteristics include the plate width w and the plate separation d .



The block lets you model the transmission line as a stub or as a stubless line.

Stubless Transmission Line

If you model a parallel-plate transmission line as a stubless line, the Parallel-Plate Transmission Line block calculates the frequency-dependent S-parameters using the physical length of the transmission line, D , and the complex propagation constant, k .

$$S_{11} = 0$$

$$S_{12} = e^{-kD}$$

$$S_{21} = e^{-kD}$$

$$S_{22} = 0$$

k can be expressed in terms of the resistance (R), inductance (L), conductance (G), and capacitance (C) per unit length (meters) as

Parallel-Plate Transmission Line

$$k = k_r + jk_i = \sqrt{(R + j2\pi fL)(G + j2\pi fC)}$$

where

$$R = \frac{2}{\omega \sigma_{\text{cond}} \delta}$$

$$L = \mu \frac{d}{\omega}$$

$$G = \sigma_{\text{diel}} \frac{\omega}{d}$$

$$C = \varepsilon \frac{\omega}{d}$$

In these equations, σ_{cond} is the conductivity in the conductor and σ_{diel} is the conductivity in the dielectric. μ is the permeability of the dielectric, ε is its permittivity, and skin depth δ is calculated as $1/\sqrt{\pi f \mu \sigma_{\text{cond}}}$. f is the vector of modeling frequencies for the specified parameters. See the Output Port block reference page for information about determining the modeling frequencies.

The Parallel-Plate Transmission Line block normalizes the resulting S-parameters to a reference impedance of 50 ohms.

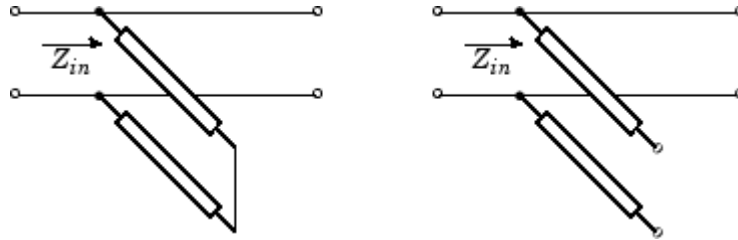
Shunt and Series Stubs

If you model the transmission line as a shunt or series stub, the Parallel-Plate Transmission Line block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. It then uses the `abcd2s` function to convert the ABCD-parameters to S-parameters.

Shunt ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to Shunt, the two-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.

Parallel-Plate Transmission Line



Z_{in} is the input impedance of the shunt circuit. The ABCD-parameters for the shunt stub are calculated as

$$A = 1$$

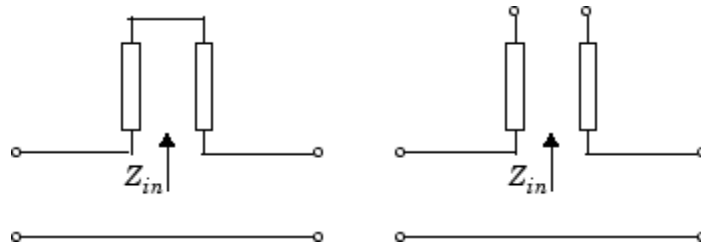
$$B = 0$$

$$C = 1/Z_{in}$$

$$D = 1$$

Series ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to Series, the two-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



Z_{in} is the input impedance of the series circuit. The ABCD-parameters for the series stub are calculated as

Parallel-Plate Transmission Line

$$\begin{aligned}A &= 1 \\B &= Z_{in} \\C &= 0 \\D &= 1\end{aligned}$$

Dialog Box

Block Parameters: Parallel-Plate Transmission Line

Parallel-Plate Transmission Line (mask) (link)
Model a parallel-plate transmission line.

Parameters

Plate width (m): 5e-3

Plate separation (m): 1e-3

Relative permeability constant: 1

Relative permittivity constant: 2.3

Conductivity in conductor (S/m): inf

Conductivity in dielectric (S/m): 0

Transmission line length (m): 0.01

Stub mode: Not a stub

Plot the selected parameters of this block

Source of frequency data: User-specified

Frequency data (Hz): [1e9:1.0e6:3e9]

Plot type: X-Y plane

Parameter: S21

Format: Angle (degrees)

OK Cancel Help Apply

Plate width (m)

Physical width of the parallel-plate transmission line.

Plate separation (m)

Thickness of the dielectric separating the plates.

Relative permeability constant

Relative permeability of the dielectric expressed as the ratio of the permeability of the dielectric to permeability in free space μ_0 .

Relative permittivity constant

Relative permittivity of the dielectric expressed as the ratio of the permittivity of the dielectric to permittivity in free space ϵ_0 .

Conductivity in conductor (S/m)

Conductivity of the conductor in siemens per meter.

Conductivity in dielectric (S/m)

Conductivity of the dielectric in siemens per meter.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

Type of stub. Choices are Not a stub, Shunt, or Series.

Termination of stub

Stub termination for stub modes Shunt and Series. Choices are Open or Short. This parameter becomes visible only when **Stub mode** is set to Shunt or Series.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

References

[1] Ludwig, Reinhold and Pavel Bretchko, *RF Circuit Design: Theory and Applications*, Prentice-Hall, 2000.

See Also

Coaxial Transmission Line, Coplanar Waveguide Transmission Line, General Passive Network, Transmission Line, Microstrip Transmission Line, Two-Wire Transmission Line

RLCG Transmission Line

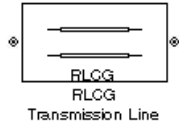
Purpose

Model RLCG transmission line

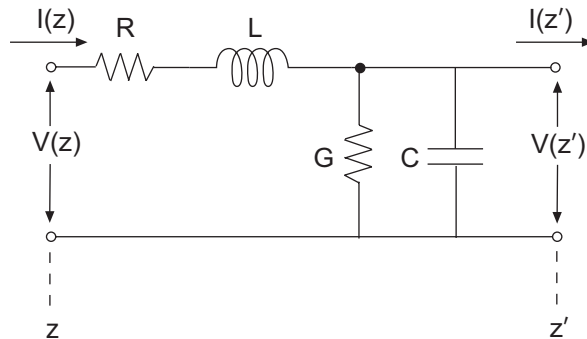
Library

Transmission Lines sublibrary of the Physical library

Description



The RLCG Transmission Line block models the RLCG transmission line described in the block dialog box in terms of its frequency-dependent resistance, inductance, capacitance, and conductance. The transmission line, which can be lossy or lossless, is treated as a two-port linear network.



where $z' = z + \Delta z$.

The block lets you model the transmission line as a stub or as a stubless line.

Stubless Transmission Line

If you model a RLCG transmission line as a stubless line, the RLCG Transmission Line block calculates the frequency-dependent S-parameters using the physical length of the transmission line, D , and the complex propagation constant, k .

$$S_{11} = 0$$

$$S_{12} = e^{-kD}$$

$$S_{21} = e^{-kD}$$

$$S_{22} = 0$$

k can be expressed in terms of the resistance (R), inductance (L), conductance (G), and capacitance (C) per unit length (meters) as

$$k = k_r + jk_i = \sqrt{(R + j2\pi fL)(G + j2\pi fC)}$$

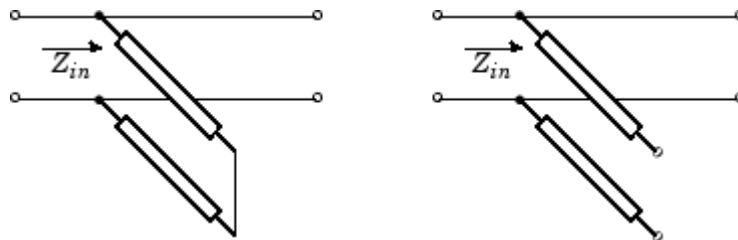
The RLCG Transmission Line block normalizes the resulting S-parameters to a reference impedance of 50 ohms.

Shunt and Series Stubs

If you model the transmission line as a shunt or series stub, the RLCG Transmission Line block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. It then uses the `abcd2s` function to convert the ABCD-parameters to S-parameters.

Shunt ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to Shunt, the two-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.



Z_{in} is the input impedance of the shunt circuit. The ABCD-parameters for the shunt stub are calculated as

RLCG Transmission Line

$$A = 1$$

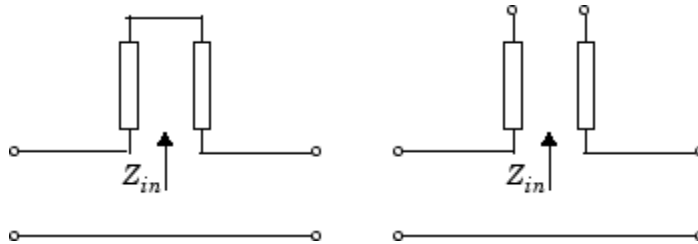
$$B = 0$$

$$C = 1/Z_{in}$$

$$D = 1$$

Series ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to **Series**, the two-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



Z_{in} is the input impedance of the series circuit. The ABCD-parameters for the series stub are calculated as

$$A = 1$$

$$B = Z_{in}$$

$$C = 0$$

$$D = 1$$

Dialog Box

Block Parameters: RLCG Transmission Line

RLCG Transmission Line (mask) (link)
Model a RLCGI transmission line.

Parameters

Resistance per length (ohms/m):
0

Inductance per length (H/m):
0

Capacitance per length (F/m):
0

Conductance per length (S/m):
0

Frequency (Hz):
1e9

Interpolation method: Linear

Transmission line length (m):
0.01

Stub mode: Not a stub

Plot the selected parameters of this block

Source of frequency data: User-specified

Frequency data (Hz):
[1e9; 1.0e6; 3e9]

Plot type: X-Y plane

Parameter: S21

Format: Angle (degrees)

OK Cancel Help Apply

Resistance per length (ohms/m)

Vector of resistance values in ohms per meter.

Inductance per length (H/m)

Vector of inductance values in henries per meter.

Capacitance per length (F/m)

Vector of capacitance values in farads per meter.

Conductance per length (S/m)

Vector of conductance values in siemens per meter.

RLCG Transmission Line

Frequency (Hz)

Vector of frequency values at which the resistance, inductance, capacitance, and conductance values are known.

Interpolation method

Specify the interpolation method the block uses to calculate the parameter values at the modeling frequencies. Your choices are Linear, Spline, or Cubic.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

Type of stub. Your choices are Not a stub, Shunt, or Series.

Termination of stub

Stub termination for stub modes Shunt and Series. Choices are Open or Short. This parameter becomes visible only when **Stub mode** is set to Shunt or Series.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

References

[1] Ludwig, Reinhold and Pavel Bretchko, *RF Circuit Design: Theory and Applications*, Prentice-Hall, 2000.

See Also

Coaxial Transmission Line, Coplanar Waveguide Transmission Line, General Passive Network, Parallel-Plate Transmission Line, Transmission Line, Microstrip Transmission Line, Two-Wire Transmission Line

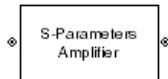
Purpose

Model nonlinear amplifier using its S-parameters

Library

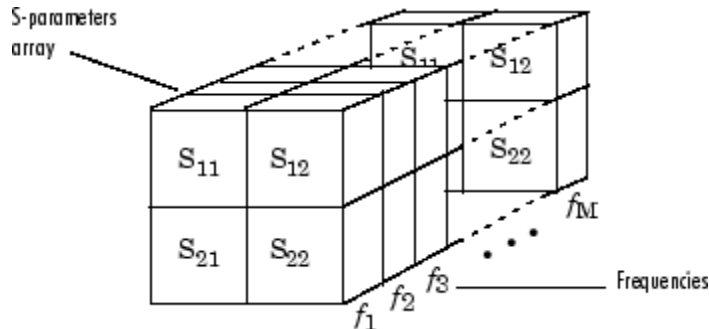
Amplifiers sublibrary of the Physical library

Description



The S-Parameters Amplifier block models the nonlinear amplifier described in the block dialog box, in terms of its frequency-dependent S-parameters, their frequencies, and the reference impedance of the S-parameters. The block also takes into account the IP3 value and the noise figure.

In the **S-parameters** field of the block dialog box, provide the S-parameters for each of M frequencies as a 2-by-2-by- M array. In the **Frequency** field, specify the frequencies for the S-parameters as an M -element vector. The elements of the frequencies vector must be in the same order as the S-parameters. All frequencies must be positive. For example, the following figure shows the correspondence between the S-parameters array and the vector of frequencies.



The S-Parameters Amplifier block interpolates the given S-parameters to determine their values at the modeling frequencies. The modeling frequencies are determined by the Output Port block. See Appendix A, “RF Blockset Algorithms” for more details.

S-Parameters Amplifier

Dialog Box

Block Parameters: S-Parameters Amplifier

S-Parameters Amplifier (mask) (link)

Nonlinear amplifier described by the frequency-dependent S-Parameters (2x2xM array), the Frequency (vector of length M), and the Reference impedance (scalar or vector of length M), as well as scalars IIP3/OIP3 and noise figure. M is the number of frequencies.

Data interpolation is used during simulation.

Parameters:

S-Parameters: [0,0,1,0]

Frequency (Hz): 2.0e9

Reference impedance: 50

Interpolation method: Linear

OIP3 type: OIP3

OIP3 (dBm): inf

Noise figure (dB): 0

Plot the selected parameters of this block

Source of frequency data: Same as the Frequency parameter

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

OK Cancel Help Apply

S-Parameters

S-parameters for a nonlinear amplifier in a 2-by-2-by-M array. M is the number of S-parameters.

Frequency (Hz)

Frequencies of the S-parameters as an M-element vector. The order of the frequencies must correspond to the order of the S-parameters in **S-Parameters**. All frequencies must be positive.

Reference impedance

Reference impedance of the S-parameters as a scalar or a vector of length M. The value of this parameter can be real or complex. If you provide a scalar, that value is applied to all frequencies.

Interpolation method

Method used to interpolate the given S-parameters over the range of frequencies. Interpolation can be Cubic, Linear (default), or Spline.

IP3 type

Type of third-order intercept point. The value can be IIP3 (input intercept point) or OIP3 (output intercept point).

IIP3 (dBm)

Input power intercept point as a scalar value. This field becomes visible if you select IIP3 as the IP3 type.

OIP3 (dBm)

Output power intercept point as a scalar value. This field becomes visible if you select OIP3 as the IP3 type.

Noise figure (dB)

Scalar ratio of the available signal-to-noise power ratio at the input to the available signal-to-noise power ratio at the output, $((S_i/N_i)/(S_o/N_o))$.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

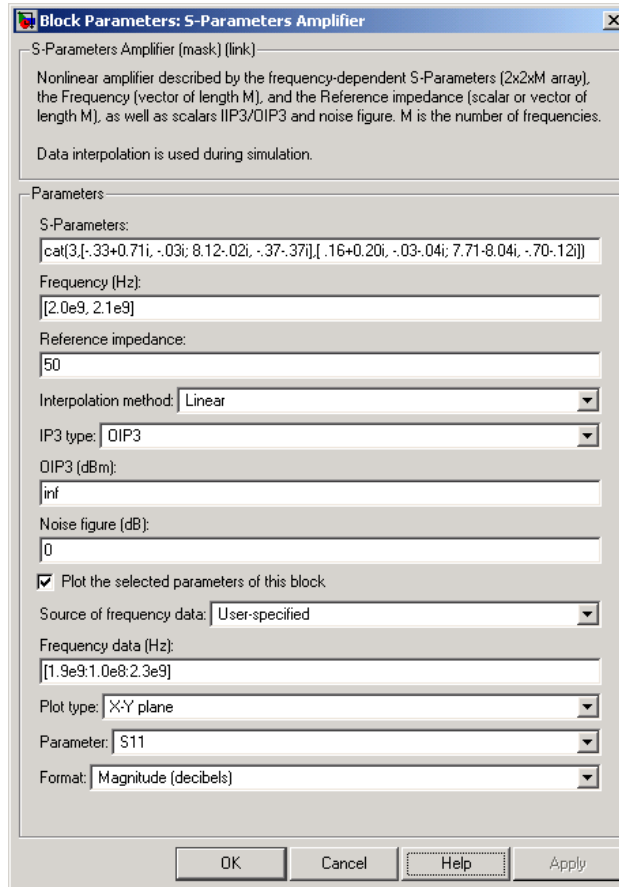
Examples

Plotting Parameters with the S-Parameters Amplifier Block

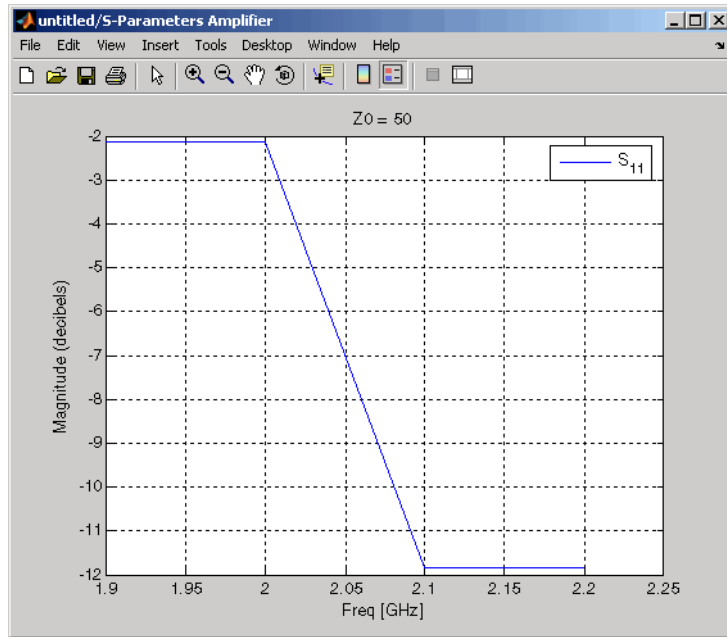
The following example specifies S-parameters $[-.33+.71i, -.03i; 8.12-.02i, -.37-.37i]$ and $[0.16+.20i, -.03-.04i; 7.71-8.04i, -.70-.12i]$ at frequencies 2.0 GHz and 2.1 GHz respectively, with a reference impedance of 50 ohms. It uses the MATLAB `cat` function to create the 2-by-2-by-2 S-parameters array.

S-Parameters Amplifier

```
cat(3, [-.33+0.71i,      -.03i;  8.12-.02i,  -.37-.37i], ...  
      [ .16+0.20i,  -.03-.04i;  7.71-8.04i,  -.70-.12i])
```



The plot parameters in the dialog box request an X-Y Plane plot of the S11 parameters in the frequency range 1.8 to 2.3 GHz.



See Also

General Amplifier, Output Port, Y-Parameters Amplifier, Z-Parameters Amplifier

interp1 (MATLAB)

S-Parameters Mixer

Purpose Model mixer using its S-parameters

Library Mixer sublibrary of the Physical library

Description



The S-Parameters Mixer block models the nonlinear mixer described in the block dialog box, in terms of its frequency-dependent S-parameters, their frequencies, and the reference impedance of the S-parameters. The block also takes phase noise into account.

The S21 parameter values describe the conversion gain as a function of frequency, referred to the mixer input frequency. The other S-parameters also refer to the mixer input frequency.

The S-Parameters Mixer block interpolates the given S-parameters to determine their values at the modeling frequencies. The modeling frequencies are determined by the Output Port block. See Appendix A, “RF Blockset Algorithms” for more details.

The RF Blockset computes the normalized reflected amplitude at the mixer input ($E_{r,1}$) and at the mixer output ($E_{r,2}$) from the interpolated S-parameters as

$$\begin{aligned}E_{r,1}(f_{in}) &= S_{11}(f_{in})E_{i1}(f_{in}) + S_{12}(f_{in})E_{i2}(f_{out}) \\E_{r,2}(f_{out}) &= S_{21}(f_{in})E_{i1}(f_{in}) + S_{22}(f_{in})E_{i2}(f_{out})\end{aligned}$$

where

- f_{in} and f_{out} are the mixer input and output frequencies, respectively.
- E_{i1} and E_{i2} are the normalized incident amplitudes at the mixer input and output, respectively.

The interpolated S21 parameter values describe the conversion gain as a function of frequency, referred to the mixer input frequency.

Dialog Box

Block Parameters: S-Parameters Mixer

S-Parameters Mixer (mask) [link]

2-port mixer described by the frequency-dependent S-Parameters (2x2xM array), the Frequency (vector of length M), and the Reference impedance (scalar or vector of length M), as well as scalars IIP3/OIP3, Noise figure and Phase noise. M is the number of frequencies.

Data interpolation is used during simulation.

Parameters

S-Parameters:
[0,0,1,0]

Frequency (Hz):
2.0e9

Reference impedance:
50

Interpolation method: Linear

Type: Downconverter

LO frequency (Hz):
0.9e9

IP3 type: OIP3

OIP3 (dBm):
inf

Noise figure (dB):
0

Phase noise frequency offset (Hz):
[0.1 1 10 100]*1e3

Phase noise level (dBc/Hz):
[-70 -120 -140 -150]

Plot the selected parameters of this block.

Source of frequency data: Same as the Frequency parameter

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

OK Cancel Help Apply

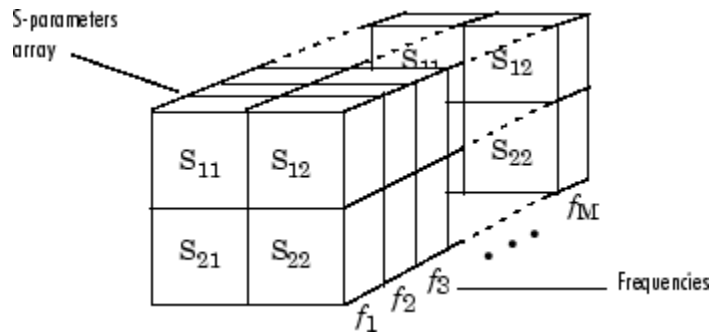
S-Parameters

S-parameters for a nonlinear mixer in a 2-by-2-by-M array. M is the number of S-parameters.

S-Parameters Mixer

Frequency (Hz)

Frequencies of the S-parameters as an M-element vector. The order of the frequencies must correspond to the order of the S-parameters in **S-Parameters**. All frequencies must be positive. The following figure shows the correspondence between the S-parameters array and the vector of frequencies.



Reference impedance

Reference impedance of the S-parameters as a scalar or a vector of length M. The value of this parameter can be real or complex. If you provide a scalar, that value is applied to all frequencies.

Interpolation method

Method used to interpolate the given S-parameters over the range of frequencies. Interpolation can be Linear (default), Spline, or Cubic.

Type

Type of mixer. Choices are Downconverter (default) and Upconverter.

LO frequency (Hz)

Local oscillator frequency. If you choose Downconverter, the RF

Blockset computes the mixer output frequency, f_{out} , from the

mixer input frequency, f_{in} , and the local oscillator frequency, f_{lo} , as $f_{out} = f_{in} - f_{lo}$. If you choose Upconverter, $f_{out} = f_{in} + f_{lo}$.

Note The mixer output frequency must be positive. This means that if you choose a downconverting mixer, f_{in} must be greater than f_{lo} . Otherwise, an error appears.

IP3 type

Type of third-order intercept point. The value can be IIP3 (input intercept point) or OIP3 (output intercept point).

IIP3 (dBm)

Input power intercept point as a scalar value. This field becomes visible if you select IIP3 as the IP3 type.

OIP3 (dBm)

Output power intercept point as a scalar value. This field becomes visible if you select OIP3 as the IP3 type.

Noise figure (dB)

Scalar ratio of the available signal-to-noise power ratio at the input to the available signal-to-noise power ratio at the output ($S_i/N_i)/(S_o/N_o$).

Phase noise frequency offset (Hz)

Vector specifying the frequency offset.

Phase noise level (dBc/Hz)

Vector specifying the phase noise level.

Note For information about plotting the mixer parameters, see Chapter 3, “Plotting Model Data”.

See Also

General Mixer, Output Port, Y-Parameters Mixer, Z-Parameters Mixer

S-Parameters Passive Network

Purpose Model passive network using its S-parameters

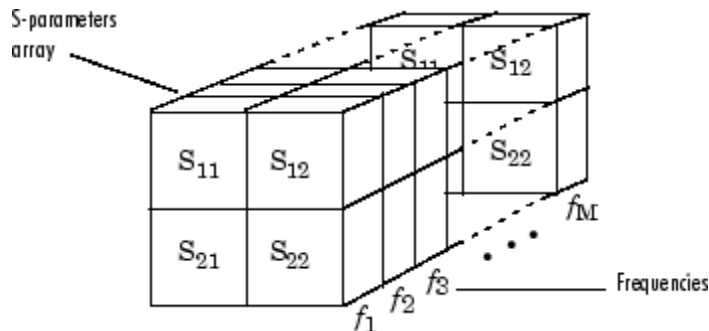
Library Black Box Elements sublibrary of the Physical library

Description



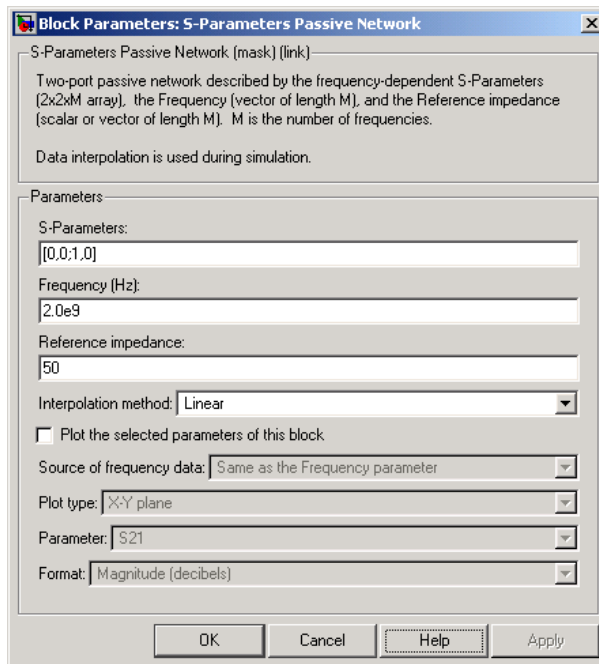
The S-Parameters Passive Network block models the two-port passive network described in the block dialog box, in terms of its S-parameters, their frequencies, and the reference impedance of the S-parameters.

In the **S-Parameters** field of the block dialog box, provide the S-parameters for each of M frequencies as a 2-by-2-by- M array. In the **Frequency** field, specify the frequencies for the S-parameters as an M -element vector. The elements of the vector must be in the same order as the S-parameters. All frequencies must be positive. For example, the following figure shows the correspondence between the S-parameters array and the vector of frequencies.



The S-Parameters Passive Network block interpolates the given S-parameters to determine their values at the modeling frequencies. The modeling frequencies are determined by the Output Port block. See Appendix A, “RF Blockset Algorithms” for more details.

Dialog Box



S-Parameters

S-parameters for a two-port passive network in a 2-by-2-by-M array. M is the number of S-parameters.

Frequency (Hz)

Frequencies of the S-parameters as an M-element vector. The order of the frequencies must correspond to the order of the S-parameters in **S-Parameters**. All frequencies must be positive.

Reference impedance

Reference impedance of the network as a scalar or a vector of length M. The value of this parameter can be real or complex. If you provide a scalar, that value is applied to all frequencies.

S-Parameters Passive Network

Interpolation method

Method used to interpolate the given S-parameters over the range of frequencies. Interpolation can be Cubic, Linear (default), or Spline.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

Examples

Plotting Parameters with the S-Parameters Passive Network Block

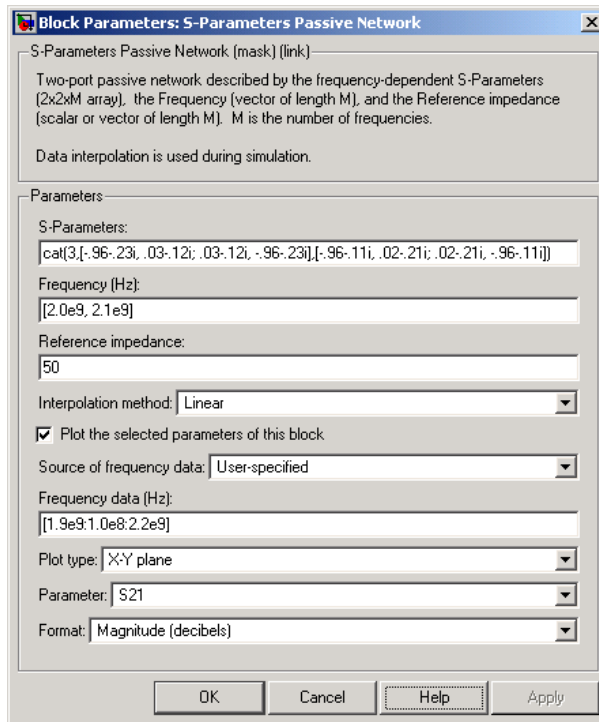
The following example specifies S-parameters $[-.96-.23i, .03-.12i; .03-.12i, -.96-.23i]$ and $[-.96-.11i, .02-.21i; .02-.21i, -.96-.11i]$ at frequencies 2.0 GHz and 2.1 GHz respectively. It uses the MATLAB `cat` function to create the 2-by-2-by-2 S-parameters array.

```
cat(3, [-.96-.23i, .03-.12i; .03-.12i, -.96-.23i], ...
      [-.96-.11i, .02-.21i; .02-.21i, -.96-.11i])
```

You could also use the MATLAB `reshape` function. The following statement produces the same result as the one shown above.

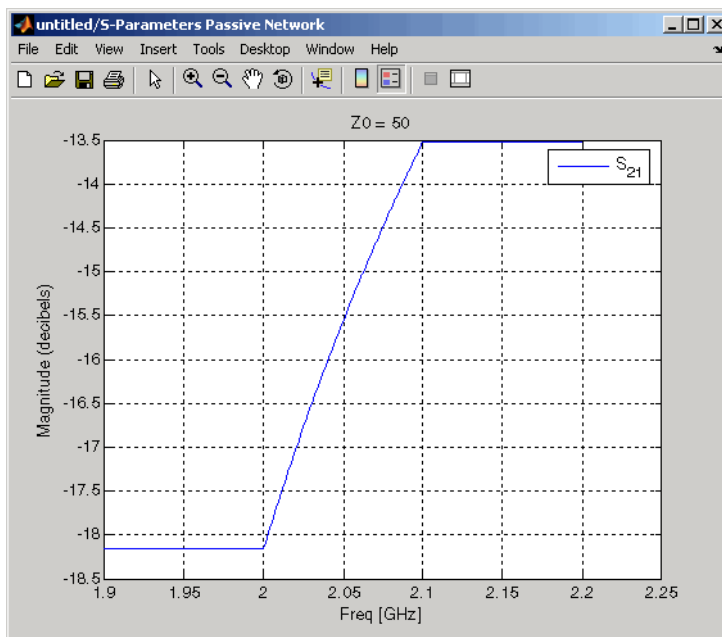
```
reshape([- .96-.23i;.03-.12i;.03-.12i;-.96-.23i;...
        -.96-.11i;.02-.21i;.02-.21i;-.96-.11i],2,2,2)
```


S-Parameters Passive Network



The plot parameters in the dialog box request an X-Y Plane plot of the S21 S-parameter magnitudes, in decibels, in the frequency range 1.9 to 2.2 GHz.

S-Parameters Passive Network



See Also

General Circuit Element, General Passive Network, Output Port, Y-Parameters Passive Network, Z-Parameters Passive Network

interp1 (MATLAB)

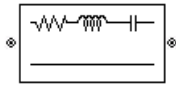
Purpose

Model series RLC network

Library

Ladders Filters sublibrary of the Physical library

Description



The Series RLC block models the series RLC network described in the block dialog box, in terms of its frequency-dependent S-parameters.

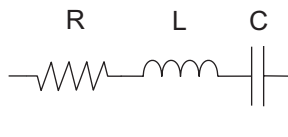
For the given resistance, inductance, and capacitance, the block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies, and then converts the ABCD-parameters to S-parameters using the RF Toolbox `abcd2s` function. See the Output Port block reference page for information about determining the modeling frequencies.

For this circuit, $A = 1$, $B = Z$, $C = 0$, and $D = 1$, where

$$Z = \frac{-LC\omega^2 + jRC\omega + 1}{jC\omega}$$

and $\omega = 2\pi f$.

The series RLC object is a two-port network as shown in the circuit diagram below.



Series RLC

Dialog Box

Block Parameters: Series RLC

Series RLC (mask) (link)

A series combination of resistive (R), inductive (L), and capacitive (C) elements.

This block can model a combination of any, all, or none of these three elements. Elements will be included or excluded depending upon the specified parameter values.

To exclude the resistive element, set the resistance value to 0 Ohms.
To exclude the inductive element, set the inductance value to 0 Henries.
To exclude the capacitive element, set the capacitance value to Inf Farads.

If all three elements are excluded, which is the default setting, the block will become a direct connection from input to output and have no effect on the signal.

Parameters

Resistance (ohms):
0

Inductance (H):
0

Capacitance (F):
Inf

Plot the selected parameters of this block

Source of frequency data: User-specified

Frequency data (Hz):
[1:1.0e5:4e6]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

OK Cancel Help Apply

Resistance (ohms)

Scalar value for the resistance. The value must be nonnegative.

Inductance (H)

Scalar value for the inductance. The value must be nonnegative.

Capacitance (F)

Scalar value for the capacitance. The value must be nonnegative.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

See Also

General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Pi, LC Highpass Tee, LC Lowpass Pi, LC Lowpass Tee, Shunt RLC

Shunt RLC

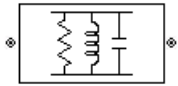
Purpose

Model shunt RLC network

Library

Ladders Filters sublibrary of the Physical library

Description



The Shunt RLC block models the shunt RLC network described in the block dialog box, in terms of its frequency-dependent S-parameters.

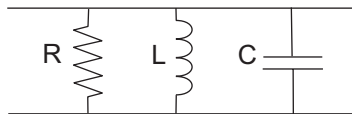
For the given resistance, inductance, and capacitance, the block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies, and then converts the ABCD-parameters to S-parameters using the RF Toolbox `abcd2s` function. See the Output Port block reference page for information about determining the modeling frequencies.

For this circuit, $A = 1$, $B = 0$, $C = Y$, and $D = 1$, where

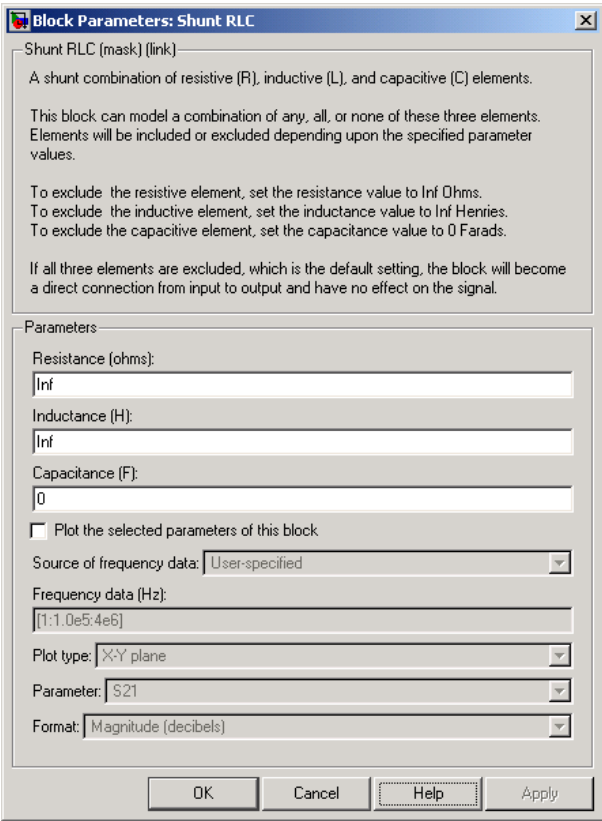
$$Y = \frac{-LC\omega^2 + j(L/R)\omega + 1}{jL\omega}$$

and $\omega = 2\pi f$.

The shunt RLC object is a two-port network as shown in the circuit diagram below.



Dialog Box



Resistance (ohms)

Scalar value for the resistance. The value must be nonnegative.

Inductance (H)

Scalar value for the inductance. The value must be nonnegative.

Capacitance (F)

Scalar value for the capacitance. The value must be nonnegative.

Shunt RLC

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

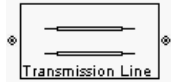
See Also

General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Pi, LC Highpass Tee, LC Lowpass Pi, LC Lowpass Tee, Series RLC

Purpose Model general transmission line

Library Transmission Lines sublibrary of the Physical library

Description



The Transmission Line block models the transmission line described in the block dialog box in terms of its physical parameters. The transmission line, which can be lossy or lossless, is treated as a two-port linear network.

The block enables you to model the transmission line as a stub or as a stubless line.

Stubless Transmission Line

If you model the transmission line as a stubless line, the Transmission Line block calculates the frequency-dependent S-parameters using the physical length of the transmission line, D , and the complex propagation constant, k .

$$S_{11} = 0$$

$$S_{12} = e^{-kD}$$

$$S_{21} = e^{-kD}$$

$$S_{22} = 0$$

$k = \alpha_a + i\beta$, where α_a is the attenuation coefficient and β is the wave number. The attenuation coefficient α_a is related to the loss, α , by

$$\alpha_a = -\ln 10 \frac{\alpha}{20}$$

and the wave number β is related to the phase velocity, V_p , by

$$\beta = \frac{2\pi f}{V_p}$$

Transmission Line

where f is the vector of modeling frequencies determined by the Output Port block. The phase velocity V_p is also known as the wave propagation velocity.

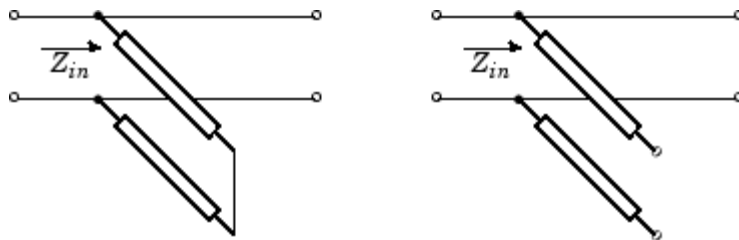
The Transmission Line block normalizes the resulting S-parameters to a reference impedance of 50 ohms.

Shunt and Series Stubs

If you model the transmission line as a shunt or series stub, the Transmission Line block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. It then uses the `abcd2s` function to convert the ABCD-parameters to S-parameters.

Shunt ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to Shunt, the two-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.



Z_{in} is the input impedance of the shunt circuit. The ABCD-parameters for the shunt stub are calculated as

$$A = 1$$

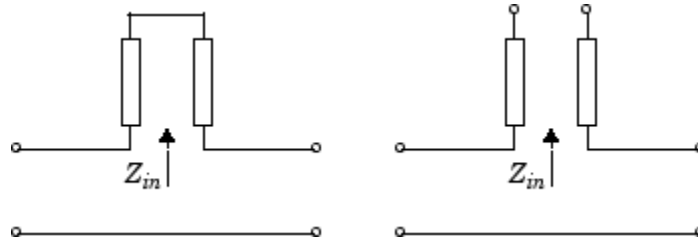
$$B = 0$$

$$C = 1/Z_{in}$$

$$D = 1$$

Series ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to Series, the two-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



Z_{in} is the input impedance of the series circuit. The ABCD-parameters for the series stub are calculated as

$$A = 1$$

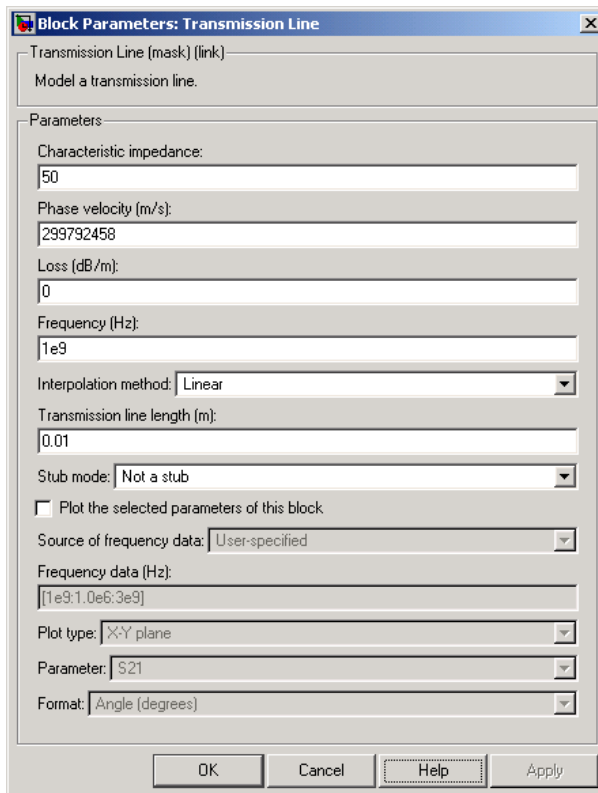
$$B = Z_{in}$$

$$C = 0$$

$$D = 1$$

Transmission Line

Dialog Box



Characteristic impedance

Characteristic impedance of the transmission line. The value can be complex.

Phase velocity (m/s)

Propagation velocity of a uniform plane wave on the transmission line.

Loss (dB/m)

Reduction in strength of the signal as it travels over the transmission line. Must be positive.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

Type of stub. Choices are Not a stub, Shunt, or Series.

Termination of stub

Stub termination for stub modes Shunt and Series. Choices are Open or Short. This parameter becomes visible only when **Stub mode** is set to Shunt or Series.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

References

[1] Ludwig, Reinhold and Pavel Bretchko, *RF Circuit Design: Theory and Applications*, Prentice-Hall, 2000.

See Also

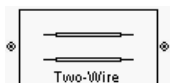
Coaxial Transmission Line, Coplanar Waveguide Transmission Line, General Passive Network, Microstrip Transmission Line, Parallel-Plate Transmission Line, Two-Wire Transmission Line

Two-Wire Transmission Line

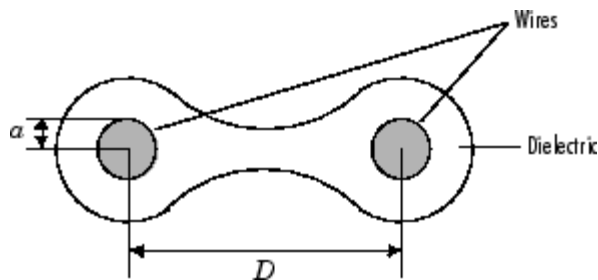
Purpose Model two-wire transmission line

Library Transmission Lines sublibrary of the Physical library

Description



The Two-Wire Transmission Line block models the two-wire transmission line described in the block dialog box in terms of its frequency-dependent S-parameters. A parallel-plate transmission line is shown here in cross-section. Its physical characteristics include the radius of the wires a , and the distance between the wire centers D .



The block enables you to model the transmission line as a stub or as a stubless line.

Stubless Transmission Line

If you model a parallel-plate transmission line as a stubless line, the Two-Wire Transmission Line block calculates the frequency-dependent S-parameters using the physical length of the transmission line, D , and the complex propagation constant, k .

$$S_{11} = 0$$

$$S_{12} = e^{-kD}$$

$$S_{21} = e^{-kD}$$

$$S_{22} = 0$$

k can be expressed in terms of the resistance (R), inductance (L), conductance (G), and capacitance (C) per unit length (meters) as

$$k = k_r + jk_i = \sqrt{(R + j2\pi fL)(G + j2\pi fC)}$$

where

$$R = \frac{1}{\pi a \sigma_{\text{cond}} \delta}$$

$$L = \frac{\mu}{\pi} \text{acosh}\left(\frac{D}{2a}\right)$$

$$G = \frac{\pi \sigma_{\text{diel}}}{\text{acosh}(D/(2a))}$$

$$C = \frac{\pi \epsilon}{\text{acosh}(D/(2a))}$$

In these equations, σ_{cond} is the conductivity in the conductor and σ_{diel} is the conductivity in the dielectric. μ is the permeability of the dielectric, ϵ is its permittivity, and skin depth δ is calculated as $1/\sqrt{\pi f \mu \sigma_{\text{cond}}}$. f is the vector of modeling frequencies for the specified parameters. See the Output Port block reference page for information about determining the modeling frequencies.

The Two-Wire Transmission Line block normalizes the resulting S-parameters to a reference impedance of 50 ohms.

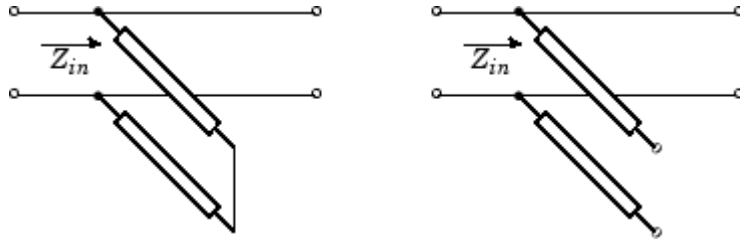
Shunt and Series Stubs

If you model the transmission line as a shunt or series stub, the Two-Wire Transmission Line block first calculates the ABCD-parameters at each frequency contained in the vector of modeling frequencies. It then uses the `abcd2s` function to convert the ABCD-parameters to S-parameters.

Shunt ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to Shunt, the two-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.

Two-Wire Transmission Line



Z_{in} is the input impedance of the shunt circuit. The ABCD-parameters for the shunt stub are calculated as

$$A = 1$$

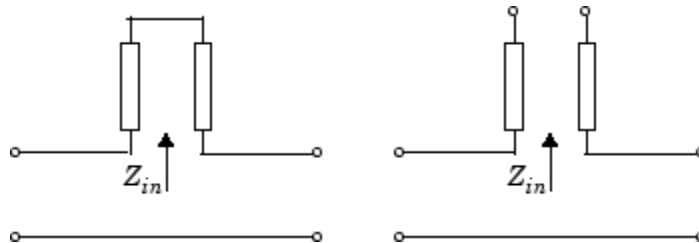
$$B = 0$$

$$C = 1/Z_{in}$$

$$D = 1$$

Series ABCD-Parameters

When you set the **Stub mode** parameter in the mask dialog box to Series, the two-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



Z_{in} is the input impedance of the series circuit. The ABCD-parameters for the series stub are calculated as

Two-Wire Transmission Line

$$\begin{aligned}A &= 1 \\B &= Z_{in} \\C &= 0 \\D &= 1\end{aligned}$$

Dialog Box

Block Parameters: Two-Wire Transmission Line

Two-Wire Transmission Line (mask) (link)
Model a two-wire transmission line.

Parameters

Wire radius (m):
0.67e-3

Wire separation (m):
1.62e-3

Relative permeability constant:
1

Relative permittivity constant:
2.3

Conductivity in conductor (S/m):
inf

Conductivity in dielectric (S/m):
0

Transmission line length (m):
0.01

Stub mode: Not a stub

Plot the selected parameters of this block.

Source of frequency data: User-specified

Frequency data (Hz):
[1e9;1.0e6;3e9]

Plot type: X-Y plane

Parameter: S21

Format: Angle (degrees)

OK Cancel Help Apply

Wire radius (m)

Radius of the conducting wires of the two-wire transmission line.

Two-Wire Transmission Line

Wire separation (m)

Physical distance between the wires.

Relative permeability constant

Relative permeability of the dielectric expressed as the ratio of the permeability of the dielectric to permeability in free space μ_0 .

Relative permittivity constant

Relative permittivity of the dielectric expressed as the ratio of the permittivity of the dielectric to permittivity in free space ϵ_0 .

Conductivity in conductor (S/m)

Conductivity of the conductor in siemens per meter.

Conductivity in dielectric (S/m)

Conductivity of the dielectric in siemens per meter.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

Type of stub. Choices are Not a stub, Shunt, or Series.

Termination of stub

Stub termination for stub modes Shunt and Series. Choices are Open or Short. This parameter becomes visible only when **Stub mode** is set to Shunt or Series.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

References

[1] Ludwig, Reinhold and Pavel Bretchko, *RF Circuit Design: Theory and Applications*, Prentice-Hall, 2000.

See Also

Coaxial Transmission Line, Coplanar Waveguide Transmission Line, General Passive Network, Transmission Line, Microstrip Transmission Line, Parallel-Plate Transmission Line

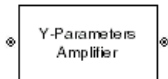
Purpose

Model nonlinear amplifier using its Y-parameters

Library

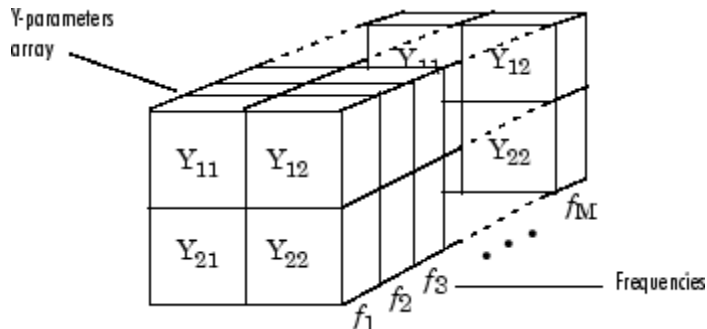
Amplifiers sublibrary of the Physical library

Description



The Y-Parameters Amplifier block models the nonlinear amplifier described in the block dialog box, in terms of its frequency-dependent Y-parameters and their frequencies. The block also takes into account the IP3 value and the noise figure.

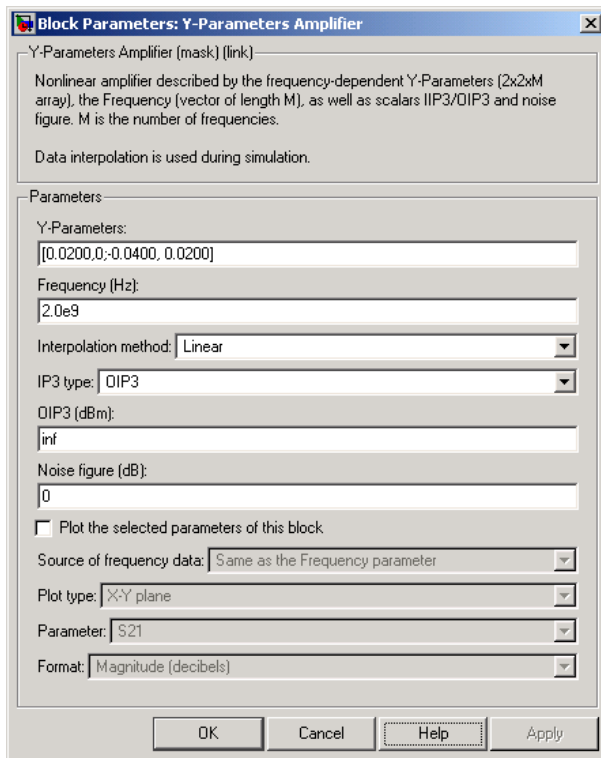
In the **Y-Parameters** field of the block dialog box, provide the Y-parameters for each of M frequencies as a 2-by-2-by- M array. In the **Frequency** field, specify the frequencies for the Y-parameters as an M -element vector. The elements of the frequencies vector must be in the same order as the Y-parameters. All frequencies must be positive. For example, the following figure shows the correspondence between the Y-parameters array and the vector of frequencies.



The Y-Parameters Amplifier block uses the RF Toolbox `y2s` function to convert the Y-parameters to S-parameters, and then interpolates the resulting S-parameters to determine their values at the modeling frequencies. The modeling frequencies are determined by the Output Port block. See Appendix A, “RF Blockset Algorithms” for more details.

Y-Parameters Amplifier

Dialog Box



Y-Parameters

Y-parameters for a nonlinear amplifier in a 2-by-2-by-M array. M is the number of Y-parameters.

Frequency (Hz)

Frequencies of the Y-parameters as an M-element vector. The order of the frequencies must correspond to the order of the Y-parameters in **Y-Parameters**. All frequencies must be positive.

Interpolation method

Method used to interpolate the S-parameters, as derived from the Y-parameters, over the range of frequencies. Interpolation can be Cubic, Linear (default), or Spline.

IP3 type

Type of third-order intercept point. The value can be IIP3 (input intercept point) or OIP3 (output intercept point).

IIP3 (dBm)

Input power intercept point as a scalar value. This field becomes visible if you select IIP3 as the IP3 type.

OIP3 (dBm)

Output power intercept point as a scalar value. This field becomes visible if you select OIP3 as the IP3 type.

Noise figure (dB)

Scalar ratio of the available signal-to-noise power ratio at the input to the available signal-to-noise power ratio at the output, $(S_i/N_i)/(S_o/N_o)$.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

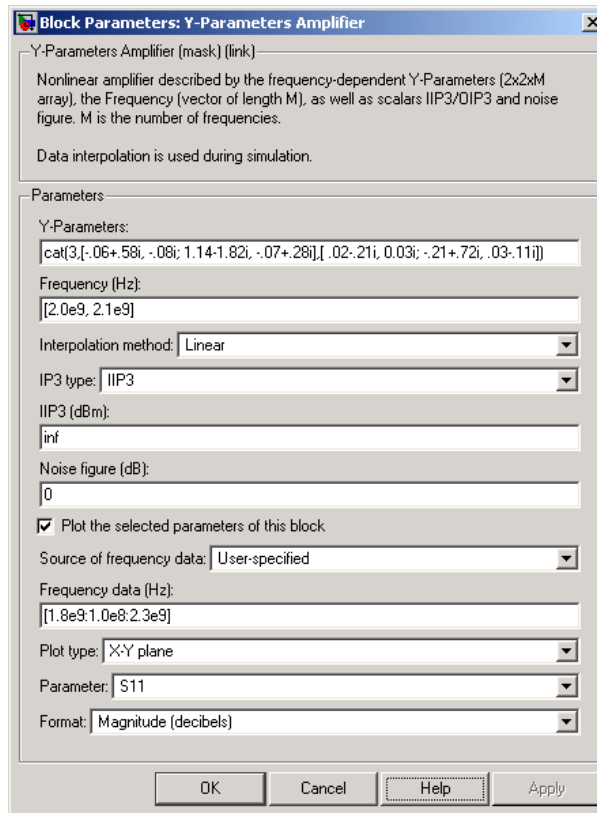
Examples

Plotting Parameters with the Y-Parameters Amplifier Block

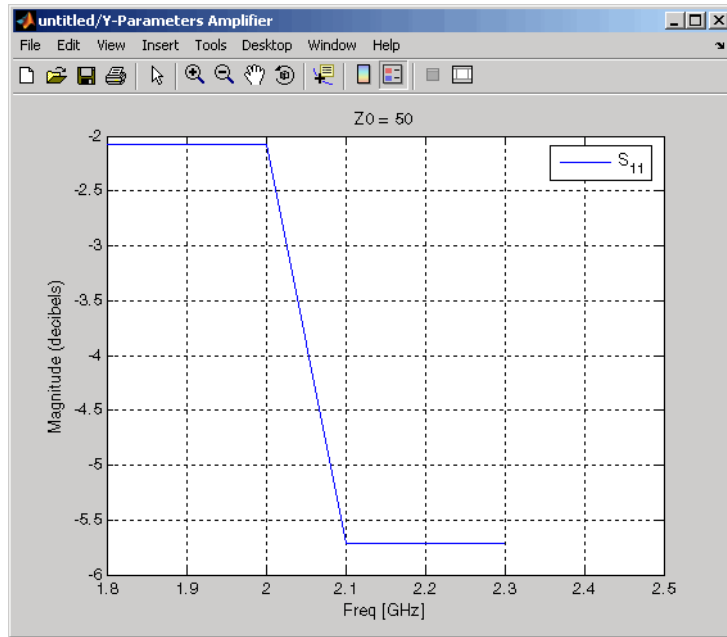
The following example specifies Y-parameters $[-.06+.58i, -.08i; 1.14-1.82i, -.07+.28i]$ and $[.02-.21i, 0.03i; -.21+.72i, .03-.11i]$ at frequencies 2.0 GHz and 2.1 GHz respectively. It uses the MATLAB `cat` function to create the 2-by-2-by-2 Y-parameters array

```
cat(3,[-.06+.58i, -.08i; 1.14-1.82i, -.07+.28i],...  
      [.02-.21i, 0.03i; -.21+.72i, .03-.11i])
```

Y-Parameters Amplifier



The plot parameters in the dialog box request an X-Y plane plot of the S11 parameters in the frequency range 1.8 to 2.3 GHz.



See Also

General Amplifier, Output Port, S-Parameters Amplifier, Z-Parameters Amplifier

y2s (RF Toolbox)

interp1 (MATLAB)

Y-Parameters Mixer

Purpose Model mixer using its Y-parameters

Library Mixer sublibrary of the Physical library

Description



The Y-Parameters Mixer block models the nonlinear mixer described in the block dialog box in terms of its frequency-dependent Y-parameters, their frequencies, and the reference impedance of the Y-parameters. The block also takes phase noise into account.

The Y-parameter values all refer to the mixer input frequency.

The Y-Parameters Mixer block uses the RF Toolbox `y2s` function to convert the Y-parameters to S-parameters and then interpolates the resulting S-parameters to determine their values at the modeling frequencies. The modeling frequencies are determined by the Output Port block. See Appendix A, “RF Blockset Algorithms” for more details.

The RF Blockset computes the normalized reflected amplitude at the mixer input (E_{r1}) and at the mixer output (E_{r2}) from the interpolated S-parameters as

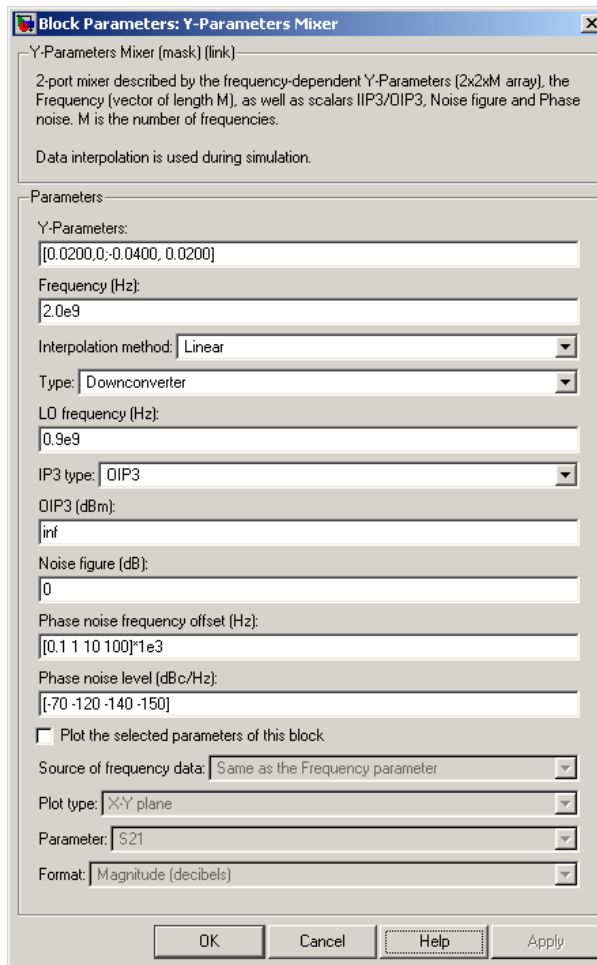
$$\begin{aligned} E_{r1}(f_{in}) &= S_{11}(f_{in})E_{i1}(f_{in}) + S_{12}(f_{in})E_{i2}(f_{out}) \\ E_{r2}(f_{out}) &= S_{21}(f_{in})E_{i1}(f_{in}) + S_{22}(f_{in})E_{i2}(f_{out}) \end{aligned}$$

where

- f_{in} and f_{out} are the mixer input and output frequencies, respectively.
- E_{i1} and E_{i2} are the normalized incident amplitudes at the mixer input and output, respectively.

The interpolated S21 parameter values describe the conversion gain as a function of frequency, referred to the mixer input frequency.

Dialog Box



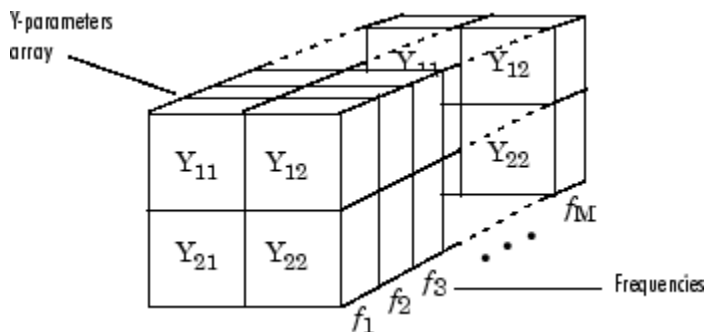
Y-Parameters

Y-parameters for a nonlinear mixer in a 2-by-2-by- M array. M is the number of Y-parameters.

Y-Parameters Mixer

Frequency (Hz)

Frequencies of the Y-parameters as an M-element vector. The order of the frequencies must correspond to the order of the Y-parameters in **Y-Parameters**. All frequencies must be positive. The following figure shows the correspondence between the Y-parameters array and the vector of frequencies.



Interpolation method

Method used to interpolate the S-parameters, as derived from the Y-parameters, over the range of frequencies. Interpolation can be Linear (default), Spline, or Cubic.

Type

Type of mixer. Choices are Downconverter (default) and Upconverter.

LO frequency (Hz)

Local oscillator frequency. If you choose Downconverter, the RF

Blockset computes the mixer output frequency, f_{out} , from the

mixer input frequency, f_{in} , and the local oscillator frequency, f_{lo} , as $f_{out} = f_{in} - f_{lo}$. If you choose Upconverter, $f_{out} = f_{in} + f_{lo}$.

Note The mixer output frequency must be positive. This means that if you choose a downconverting mixer, f_{in} must be greater than f_{lo} . Otherwise, an error appears.

IP3 type

Type of third-order intercept point. The value can be IIP3 (input intercept point) or OIP3 (output intercept point).

IIP3 (dBm)

Input power intercept point as a scalar value. This field becomes visible if you select IIP3 as the IP3 type.

OIP3 (dBm)

Output power intercept point as a scalar value. This field becomes visible if you select OIP3 as the IP3 type.

Noise figure (dB)

Scalar ratio of the available signal-to-noise power ratio at the input to the available signal-to-noise power ratio at the output, $(S_i/N_i)/(S_o/N_o)$.

Phase noise frequency offset (Hz)

Vector specifying the frequency offset.

Phase noise level (dBc/Hz)

Vector specifying the phase noise level.

Note For information about plotting the mixer parameters, see Chapter 3, “Plotting Model Data”.

See Also

General Mixer, Output Port, S-Parameters Mixer, Z-Parameters Mixer

Y-Parameters Passive Network

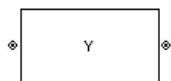
Purpose

Model passive network using its Y-parameters

Library

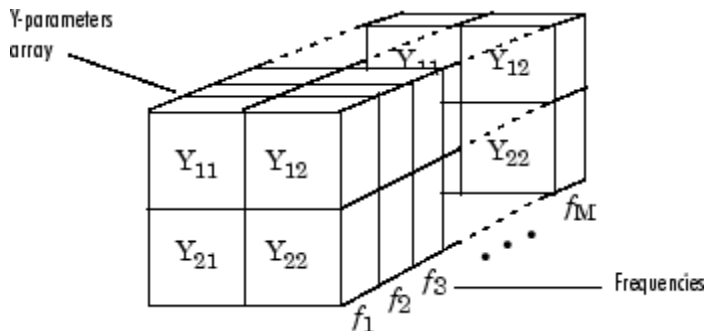
Black Box Elements sublibrary of the Physical library

Description



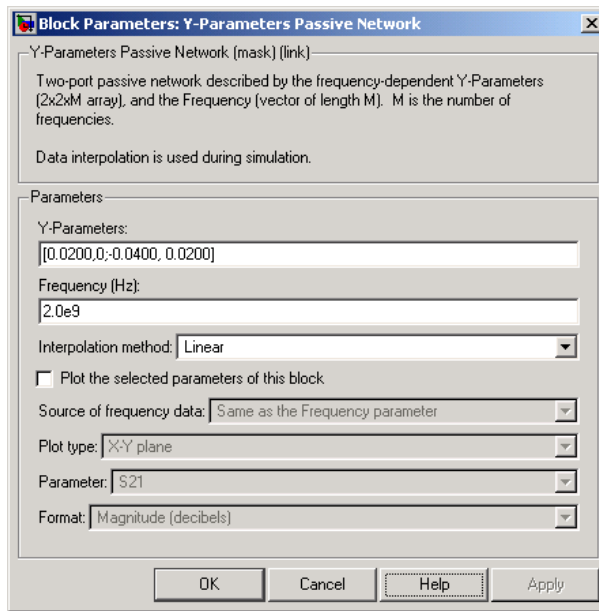
The Y-Parameters Passive Network block models the two-port passive network described in the block dialog box, in terms of its Y-parameters and their associated frequencies.

In the **Y-Parameters** field of the block dialog box, provide the Y-parameters for each of M frequencies as a 2-by-2-by-M array. In the **Frequency** field, specify the frequencies for the Y-parameters as an M-element vector. The elements of the vector must be in the same order as the Y-parameters. All frequencies must be positive. For example, the following figure shows the correspondence between the Y-parameters array and the vector of frequencies.



The Y-Parameters Passive Network block uses the RF Toolbox `y2s` function to convert the Y-parameters to S-parameters, and then interpolates the resulting S-parameters to determine their values at the modeling frequencies. The modeling frequencies are determined by the Output Port block. See Appendix A, “RF Blockset Algorithms” for more details.

Dialog Box



Y-Parameters

Y-parameters for a two-port passive network in a 2-by-2-by- M array. M is the number of Y-parameters.

Frequency (Hz)

Frequencies of the Y-parameters as an M -element vector. The order of the frequencies must correspond to the order of the Y-parameters in **Y-Parameters**. All frequencies must be positive.

Interpolation method

Method used to interpolate the S-parameters, as derived from the Y-parameters, over the range of frequencies. Interpolation can be Cubic, Linear (default), or Spline.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

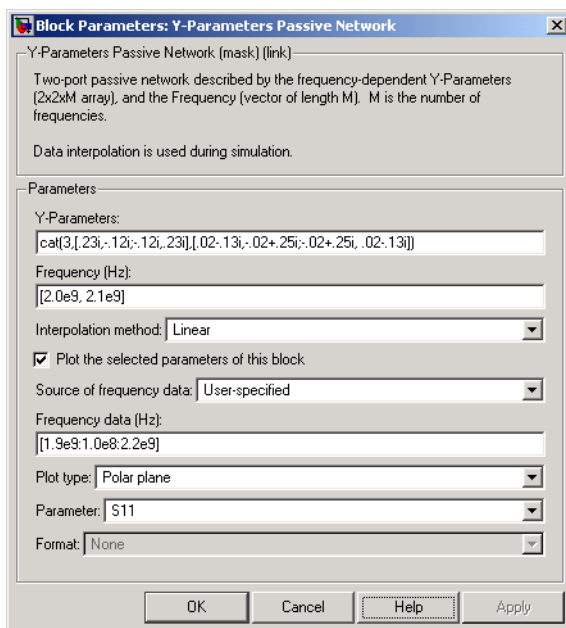
Y-Parameters Passive Network

Examples

Plotting Parameters with the Y-Parameters Passive Network Block

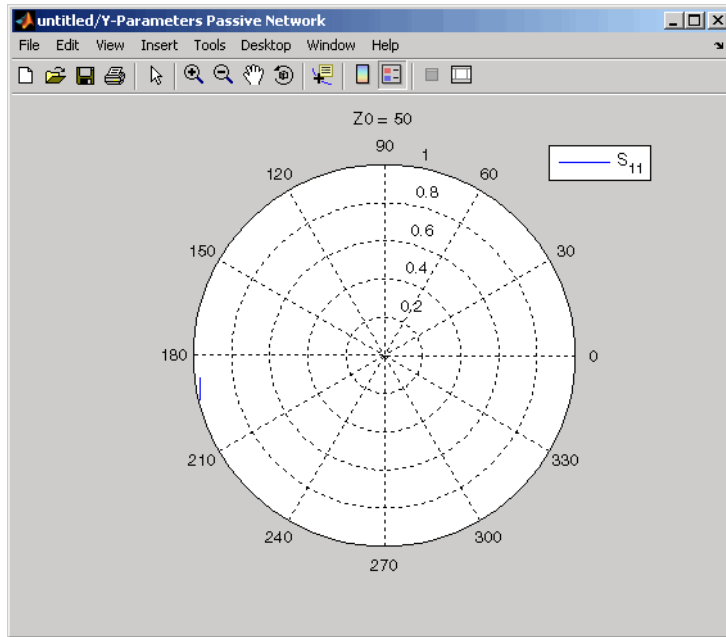
The following example specifies Y-parameters $[.23i, -.12i; -.12i, .23i]$ and $[.02-.13i, -.02+.25i; -.02+.25i, .02-.13i]$ at frequencies 2.0 GHz and 2.1 GHz respectively. It uses the MATLAB `cat` function to create the 2-by-2-by-2 Y-parameters array.

```
cat(3, [.23i, -.12i; -.12i, .23i], ...  
      [.02-.13i, -.02+.25i; -.02+.25i, .02-.13i])
```



The plot parameters in the dialog box request a polar plane plot of the S11 parameters in the frequency range 1.9 to 2.2 GHz.

Y-Parameters Passive Network



See Also

General Circuit Element, General Passive Network, Output Port, S-Parameters Passive Network, Z-Parameters Passive Network

y2s (RF Toolbox)

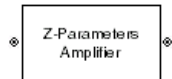
interp1 (MATLAB)

Z-Parameters Amplifier

Purpose Model nonlinear amplifier using its Z-parameters

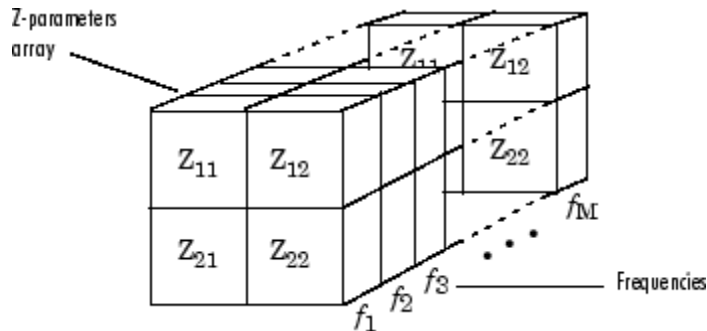
Library Amplifiers sublibrary of the Physical library

Description



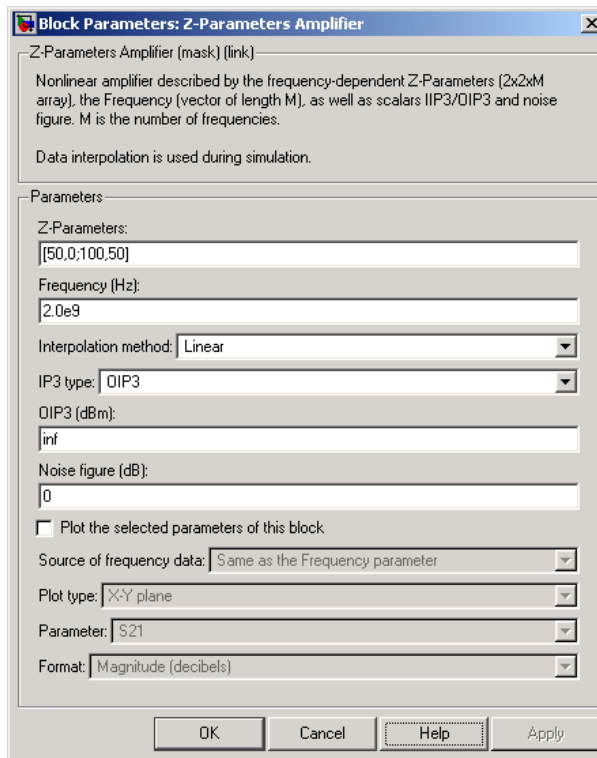
The Z-Parameters Amplifier block models the nonlinear amplifier described in the block dialog box, in terms of its frequency-dependent Z-parameters and their frequencies. The block also takes into account the IP3 value and the noise figure.

In the **Z-Parameters** field of the block dialog box, provide the Z-parameters for each of M frequencies as a 2-by-2-by-M array. In the **Frequency** field, specify the frequencies for the Z-parameters as an M-element vector. The elements of the frequencies vector must be in the same order as the Z-parameters. All frequencies must be positive. For example, the following figure shows the correspondence between the Z-parameters array and the vector of frequencies.



The Z-Parameters Amplifier block uses the RF Toolbox `y2s` function to convert the Z-parameters to S-parameters, and then interpolates the resulting S-parameters to determine their values at the modeling frequencies. The modeling frequencies are determined by the Output Port block. See Appendix A, “RF Blockset Algorithms” for more details.

Dialog Box



Z-Parameters

Z-parameters for a nonlinear amplifier in a 2-by-2-by-M array. M is the number of Z-parameters.

Frequency (Hz)

Frequencies of the Z-parameters as an M-element vector. The order of the frequencies must correspond to the order of the Z-parameters in **Z-Parameters**. All frequencies must be positive.

Interpolation method

Method used to interpolate the S-parameters, as derived from the Z-parameters, over the range of frequencies. Interpolation can be Cubic, Linear (default), or Spline.

Z-Parameters Amplifier

IP3 type

Type of third-order intercept point. The value can be IIP3 (input intercept point) or OIP3 (output intercept point).

IIP3 (dBm)

Input power intercept point as a scalar value. This field becomes visible if you select IIP3 as the IP3 type.

OIP3 (dBm)

Output power intercept point as a scalar value. This field becomes visible if you select OIP3 as the IP3 type.

Noise figure (dB)

Scalar ratio of the available signal-to-noise power ratio at the input to the available signal-to-noise power ratio at the output, $(S_i/N_i)/(S_o/N_o)$.

Note For information about plotting, see Chapter 3, “Plotting Model Data”.

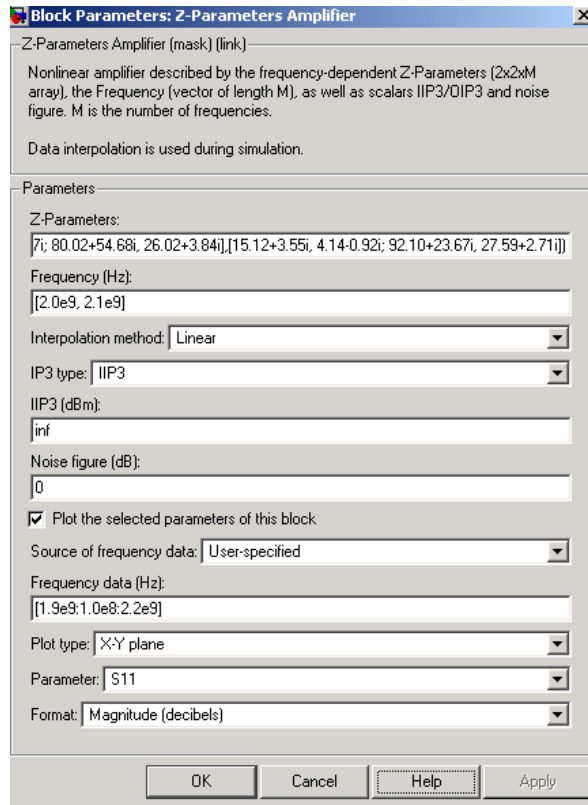
Examples

Plotting Parameters with the Z-Parameters Amplifier Block

The following example specifies Z-parameters [12.60+3.80i, 3.77-0.17i; 80.02+54.68i, 26.02+3.84i] and [15.12+3.55i, 4.14-0.92i; 92.10+23.67i, 27.59+2.71i] at frequencies 2.0 GHz and 2.1 GHz respectively. It uses the MATLAB `cat` function to create the 2-by-2-by-2 Z-parameters array.

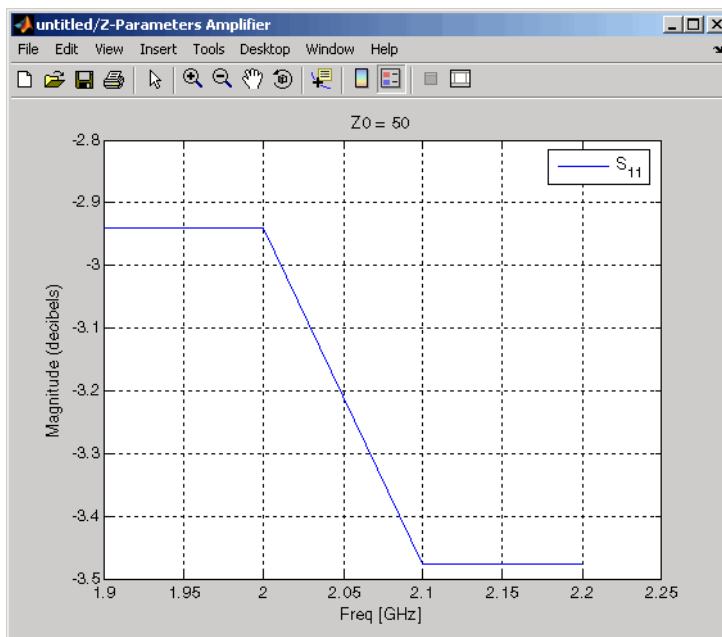
```
cat(3,[12.60+3.80i, 3.77-0.17i; 80.02+54.68i, 26.02+3.84i],...  
      [15.12+3.55i, 4.14-0.92i; 92.10+23.67i, 27.59+2.71i])
```

Z-Parameters Amplifier



The plot parameters in the dialog box request an X-Y plane plot of the S11 parameters in the frequency range 1.9 to 2.2 GHz.

Z-Parameters Amplifier



See Also

General Amplifier, Output Port, S-Parameters Amplifier, Y-Parameters Amplifier

z2s (RF Toolbox)

interp1 (MATLAB)

Purpose

Model mixer using its Z-parameters

Library

Mixer sublibrary of the Physical library

Description



The Z-Parameters Mixer block models the nonlinear mixer described in the block dialog box in terms of its frequency-dependent Z-parameters, their frequencies, and the reference impedance of the Z-parameters. The block also takes phase noise into account.

The Z-parameter values all refer to the mixer input frequency.

The Z-Parameters Mixer block uses the RF Toolbox `y2s` function to convert the Z-parameters to S-parameters and then interpolates the resulting S-parameters to determine their values at the modeling frequencies. The modeling frequencies are determined by the Output Port block. See Appendix A, “RF Blockset Algorithms” for more details.

The RF Blockset computes the normalized reflected amplitude at the mixer input (E_{r1}) and at the mixer output (E_{r2}) from the interpolated S-parameters as

$$\begin{aligned} E_{r1}(f_{in}) &= S_{11}(f_{in})E_{i1}(f_{in}) + S_{12}(f_{in})E_{i2}(f_{out}) \\ E_{r2}(f_{out}) &= S_{21}(f_{in})E_{i1}(f_{in}) + S_{22}(f_{in})E_{i2}(f_{out}) \end{aligned}$$

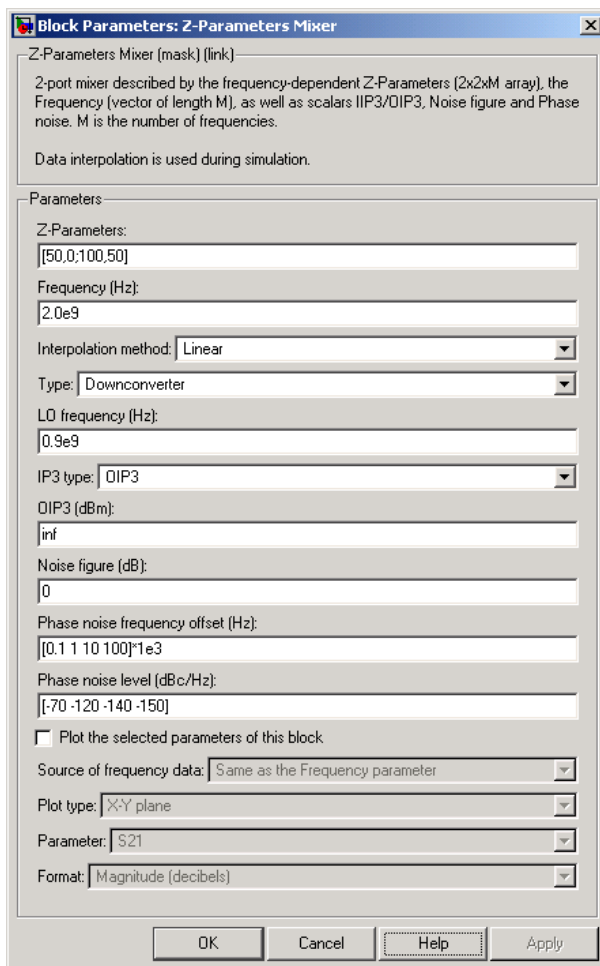
where

- f_{in} and f_{out} are the mixer input and output frequencies, respectively.
- E_{i1} and E_{i2} are the normalized incident amplitudes at the mixer input and output, respectively.

The interpolated S21 parameter values describe the conversion gain as a function of frequency, referred to the mixer input frequency.

Z-Parameters Mixer

Dialog Box

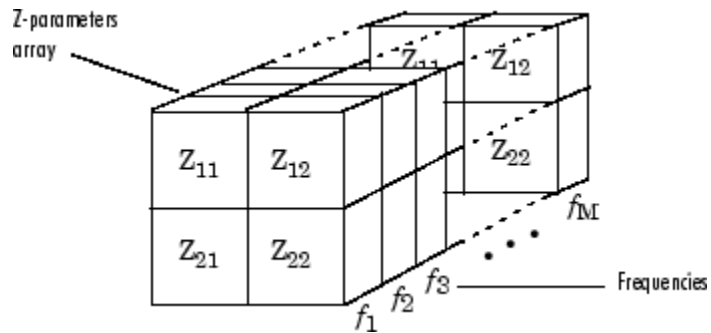


Z-Parameters

Z-parameters for a nonlinear mixer in a 2-by-2-by-M array. M is the number of Z-parameters.

Frequency (Hz)

Frequencies of the Z-parameters as an M-element vector. The order of the frequencies must correspond to the order of the Z-parameters in **Z-Parameters**. All frequencies must be positive. The following figure shows the correspondence between the Z-parameters array and the vector of frequencies.



Interpolation method

Method used to interpolate the S-parameters, as derived from the Z-parameters, over the range of frequencies. Interpolation can be Linear (default), Spline, or Cubic.

Type

Type of mixer. Choices are Downconverter (default) and Upconverter.

LO frequency (Hz)

Local oscillator frequency. If you choose Downconverter, the RF

Blockset computes the mixer output frequency, f_{out} , from the

mixer input frequency, f_{in} , and the local oscillator frequency, f_{lo} , as $f_{out} = f_{in} - f_{lo}$. If you choose Upconverter, $f_{out} = f_{in} + f_{lo}$.

Z-Parameters Mixer

Note The mixer output frequency must be positive. This means that if you choose a downconverting mixer, f_{in} must be greater than f_{lo} . Otherwise, an error appears.

IP3 type

Type of third-order intercept point. The value can be IIP3 (input intercept point) or OIP3 (output intercept point).

IIP3 (dBm)

Input power intercept point as a scalar value. This field becomes visible if you select IIP3 as the IP3 type.

OIP3 (dBm)

Output power intercept point as a scalar value. This field becomes visible if you select OIP3 as the IP3 type.

Noise figure (dB)

Scalar ratio of the available signal-to-noise power ratio at the input to the available signal-to-noise power ratio at the output, $(S_i/N_i)/(S_o/N_o)$.

Phase noise frequency offset (Hz)

Vector specifying the frequency offset.

Phase noise level (dBc/Hz)

Vector specifying the phase noise level.

Note For information about plotting the mixer parameters, see Chapter 3, “Plotting Model Data”.

See Also

General Mixer, Output Port, S-Parameters Mixer, Y-Parameters Mixer

Z-Parameters Passive Network

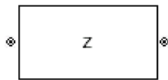
Purpose

Model passive network using its Z-parameters

Library

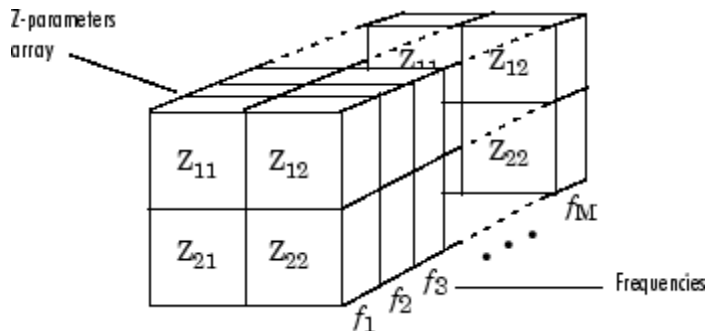
Black Box Elements sublibrary of the Physical library

Description



The Z-Parameters Passive Network block models the two-port passive network described in the block dialog box, in terms of its Z-parameters and their associated frequencies.

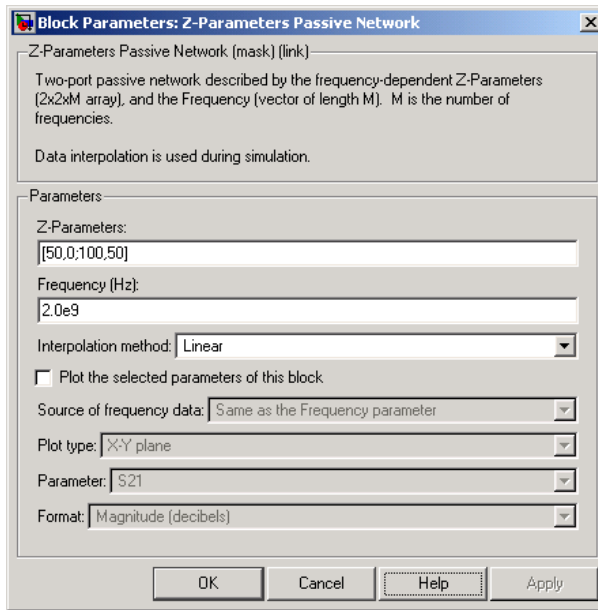
In the **Z-Parameters** field of the block dialog box, provide the Z-parameters for each of M frequencies as a 2-by-2-by-M array. In the **Frequency** field, specify the frequencies for the Z-parameters as an M-element vector. The elements of the vector must be in the same order as the Z-parameters. All frequencies must be positive. For example, the following figure shows the correspondence between the Z-parameters array and the vector of frequencies.



The Z-Parameters Passive Network block uses the RF Toolbox `y2s` function to convert the Z-parameters to S-parameters, and then interpolates the resulting S-parameters to determine their values at the modeling frequencies. The modeling frequencies are determined by the Output Port block. See Appendix A, “RF Blockset Algorithms” for more details.

Z-Parameters Passive Network

Dialog Box



Z-Parameters

Z-parameters for a two-port passive network in a 2-by-2-by-M array. M is the number of Z-parameters.

Frequency (Hz)

Frequencies of the Z-parameters as an M-element vector. The order of the frequencies must correspond to the order of the Z-parameters in **Z-Parameters**. All frequencies must be positive.

Interpolation method

Method used to interpolate the S-parameters, as derived from the Z-parameters, over the range of frequencies. Interpolation can be Cubic, Linear (default), or Spline.

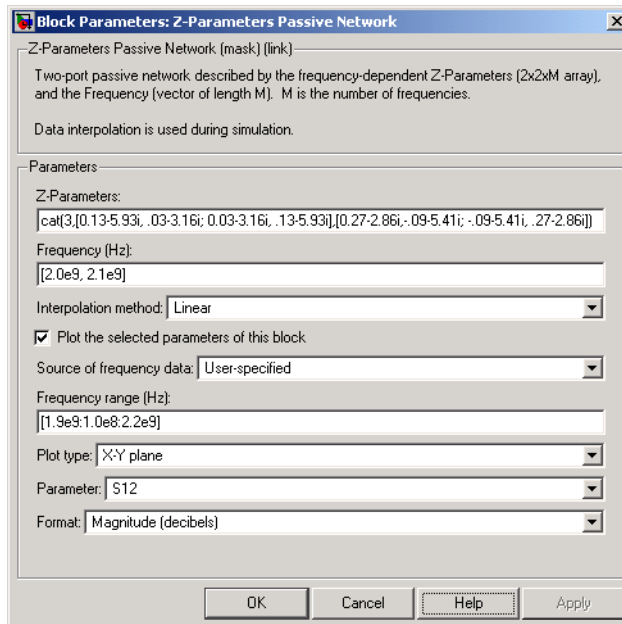
Note For information about plotting, see Chapter 3, “Plotting Model Data”.

Examples

Plotting Parameters with the Z-Parameters Passive Network Block

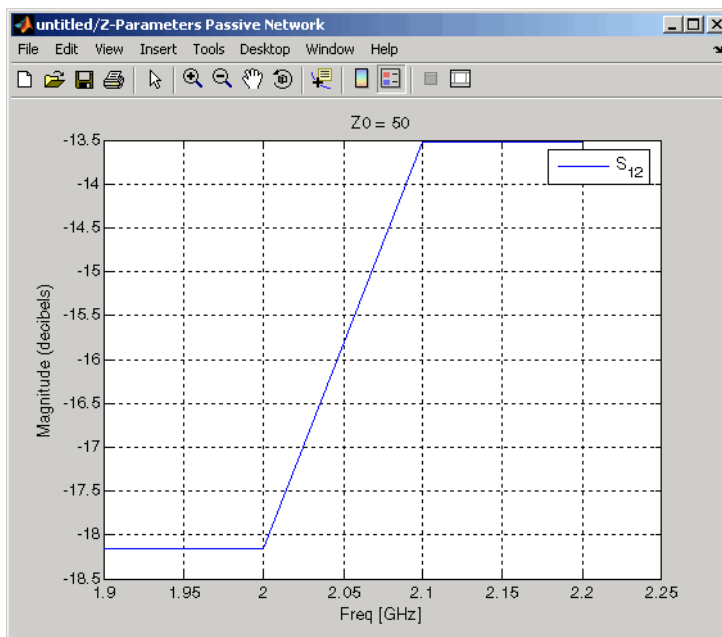
The following example specifies Z-parameters $[0.13 - 5.93i, .03-3.16i; 0.03-3.16i, .13-5.93i]$ and $[0.27-2.86i, -.09-5.41i; -.09-5.41i, .27-2.86i]$ at frequencies 2.0 GHz and 2.1 GHz respectively. It uses the MATLAB `cat` function to create the 2-by-2-by-2 Z-parameters array.

```
cat(3,[0.13-5.93i, .03-3.16i; 0.03-3.16i, .13-5.93i],...  
      [0.27-2.86i, -.09-5.41i; -.09-5.41i, .27-2.86i])
```



The plot parameters in the dialog box request an X-Y plane plot of the S12 parameters in the frequency range 1.9 to 2.2 GHz.

Z-Parameters Passive Network



See Also

General Circuit Element, General Passive Network, Output Port, S-Parameters Passive Network, Y-Parameters Passive Network

z2s (RF Toolbox)

interp1 (MATLAB)

RF Blockset Algorithms

Simulating an RF Model (p. A-2)

Gives an overview of how the RF Blockset simulates an RF model.

Determining the Modeling Frequencies (p. A-3)

Describes how the RF Blockset computes the modeling frequencies of a physical system.

Mapping Network Parameters to Modeling Frequencies (p. A-5)

Describes how the RF Blockset determines the values of the S-parameters of the physical blocks at the modeling frequencies.

Modeling Noise in an RF System (p. A-7)

Describes how the RF Blockset models thermal noise in an RF system.

Creating a Complex Baseband-Equivalent Model (p. A-10)

Describes how the RF Blockset uses the frequency-domain parameters of the RF blocks to create a baseband-equivalent model for time-domain simulation.

Converting to and from Simulink Signals (p. A-15)

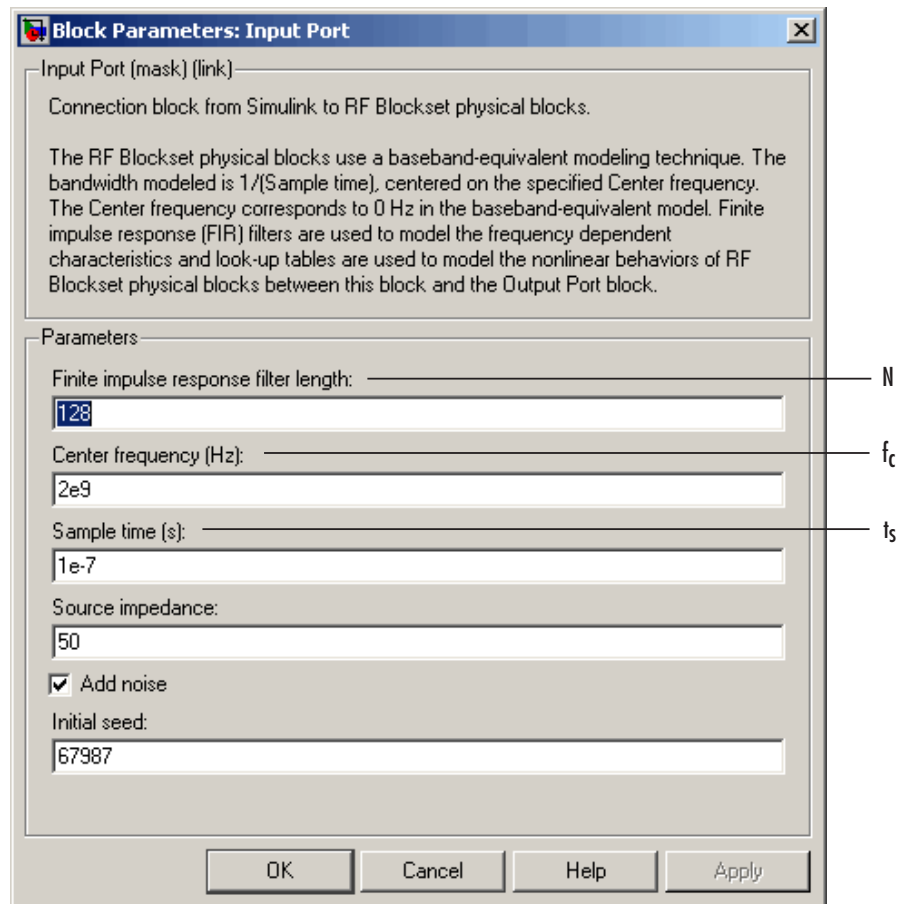
Explains how the Input Port and Output Port convert Simulink signals to and from the RF Blockset physical modeling environment during a simulation.

Simulating an RF Model

When you simulate a model that contains physical blocks, the RF Blockset determines the modeling frequencies of the physical system using the Input Port block parameters. The *modeling frequencies* are the frequencies at which the blockset takes information from the blocks to construct the baseband-equivalent model. Then, the RF Blockset determines the block parameter values at those frequencies and uses the information to create a baseband-equivalent model for time-domain simulation in Simulink.

Determining the Modeling Frequencies

When you simulate an RF model, the Output Port block uses Input Port block parameters to determine the modeling frequencies f for the physical system that is bracketed between the Input Port block and the Output Port block. f is an N -element vector, where N is the finite impulse response filter length. The modeling frequencies are a function of the center frequency f_c and the sample time t_s . The following figure shows the Input Port block parameters that determine the modeling frequencies.



f_n is the n th element of the vector of modeling frequencies, f , and is given by

$$f_n = f_{\min} + \frac{n-1}{t_s N} \quad n = 1, \dots, N$$

where

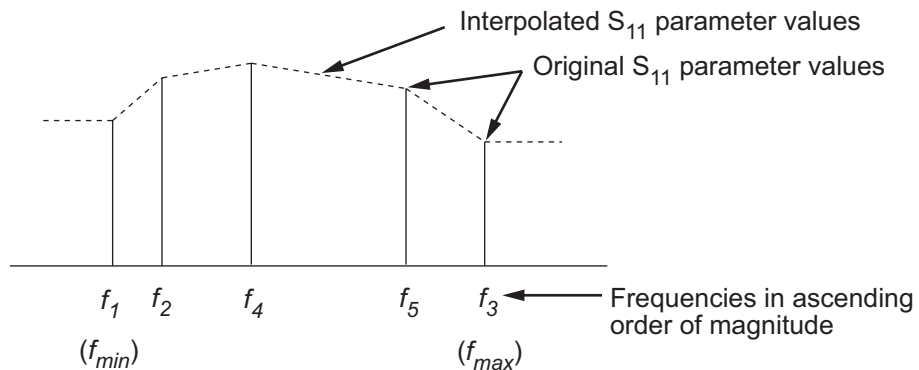
$$f_{\min} = f_c - \frac{1}{2t_s}$$

Mapping Network Parameters to Modeling Frequencies

In a physical system, each block provides network parameters at different frequencies. These frequencies are not necessarily the modeling frequencies for the physical system in which the block resides. To create a baseband-equivalent model, the RF Blockset must calculate the values of the S-parameters at the modeling frequencies.

Individual physical blocks calculate the S-parameters at the modeling frequencies determined by the Input Port block parameters. Each block interpolates its S-parameters to determine the S-parameters at the modeling frequencies. If the block contains network Y- or Z-parameters, it first converts them to S-parameters.

Specifically, the block orders the S-parameters in the ascending order of their frequencies, f_n . Then, it interpolates the S-parameters using the MATLAB `interp1` function. For example, the curve in the following diagram illustrates the result of interpolating the S₁₁ parameters at original frequencies f_1 through f_5 .



The **Interpolation** field in the individual block dialog boxes enables you to specify the interpolation method as Cubic, Linear (default), or Spline. For more information about these methods, see “One-Dimensional Interpolation” and the `interp1` reference page in the MATLAB documentation.

As shown in the previous diagram, each block uses the parameter values at f_{\min} for all modeling frequencies smaller than f_{\min} . The block uses the

parameter values at f_{\max} for all modeling frequencies greater than f_{\max} . In both cases, the results may not be accurate, so you need to specify network parameter values over a range of frequencies that is wide enough to account for the block behavior.

Modeling Noise in an RF System

The RF Blockset physical blocks model thermal noise. The Input Port block parameters specify whether to include noise in a simulation. When you include noise information in your model, the RF Blockset simulates the thermal noise of the physical system. This section explains how the RF Blockset simulates noise from user-specified information. For information on how to add noise to an RF model, see “Modeling Thermal Noise” on page 2-22.

In general, you can specify output-referred noise in one of three ways:

- **Noise temperature** — Specifies the noise in kelvin.
- **Noise factor** — Specifies the noise by the following equation

$$\text{Noise factor} = 1 + \frac{\text{Noise temperature}}{290}$$

- **Noise figure** — Specifies the noise in decibels relative to the standard reference noise temperature of 290 K. In terms of noise factor,

$$\text{Noise figure} = 10\log(\text{Noise factor})$$

These three specifications are equivalent, since you can compute each one from any of the others.

For RF Blockset physical amplifier and mixer blocks, you can specify noise using frequency-independent noise figure values. The General Amplifier and General Mixer blocks also provide the option to specify noise as frequency-dependent noise figure or as spot noise data.

For other RF blocks, such as transmission lines and filters, noise is only generated by the resistors in the network. The RF Blockset calculates this noise from basic resistor equations, so there are no noise parameters on the block dialog boxes.

When you run the simulation, the RF Blockset first computes the noise figure values at the modeling frequencies and then uses the noise figure information to calculate the output noise power.

This section contains the following topics:

- “Calculating Noise Figure at Modeling Frequencies” on page A-8
- “Calculating Output Noise Power” on page A-9

Calculating Noise Figure at Modeling Frequencies

To include noise information in a simulation, the RF Blockset must compute the noise figure values at the modeling frequencies.

If you specify the noise figure value directly, or if the blockset computes the noise figure value from the block resistance, the RF Blockset uses this value for the noise figure value at each of the modeling frequencies.

If you specify spot noise data using a Touchstone or AMP data file, the RF Blockset computes frequency-dependent noise figure information from the data in the file. It takes the minimum noise figure, NF_{min} , equivalent noise resistance, R_n , and optimal source admittance, Y_{opt} , values in the file and interpolates to find the values at the modeling frequencies. Then, it uses the following equation to calculate the noise correlation matrix, C_A .

$$C_A = 2kT \begin{bmatrix} R_n & \frac{NF_{min} - 1}{2} - R_n Y_{opt}^* \\ \frac{NF_{min} - 1}{2} - R_n Y_{opt} & R_n |Y_{opt}|^2 \end{bmatrix}$$

In the above equation, k is Boltzmann’s constant and T is the noise temperature in kelvin. The RF Blockset then calculates the noise factor, F , from the noise correlation matrix as follows:

$$F = 1 + \frac{z^+ C_A z}{2kT \operatorname{Re}\{Z_S\}}$$

$$z = \begin{bmatrix} 1 \\ Z_S^* \end{bmatrix}$$

In the two preceding equations, Z_S is the nominal impedance, which is 50 ohms, and z^+ is the Hermitian conjugation of z .

The RF Blockset obtains the noise figure, NF, from the noise factor:

$$NF = 10 \log(F)$$

For more information about these calculation techniques, see the following article:

Hillbrand, H. and P.H. Russer, "An Efficient Method for Computer Aided Noise Analysis of Linear Amplifier Networks," *IEEE Transactions on Circuits and Systems*, Vol. CAS-23, Number 4, pp. 235-238, 1976.

Calculating Output Noise Power

The RF Blockset uses noise power to determine the amplitude of the noise that it adds to the system using a Gaussian distributed pseudorandom number generator. It uses both the noise temperature and the modeling bandwidth to calculate the noise power:

$$\text{Noise power} = kTB$$

Here, k is Boltzmann's constant, which is $1.38e-23$ J/K, T is the noise temperature in kelvin and B is the bandwidth in hertz.

The RF Blockset computes noise temperature from the specified or calculated noise figure values, and it computes the modeling bandwidth from the model's sample time and center frequency.

For more information, see the `rfdata.noise` and `rfdata.nf` reference pages.

Creating a Complex Baseband-Equivalent Model

The RF Blockset simulates the physical system in the time domain using a baseband-equivalent model that it creates from the frequency-domain parameters of the physical blocks.

This section contains the following topics:

- “Baseband-Equivalent Modeling” on page A-10
- “Simulation Efficiency of a Complex Baseband-Equivalent Model” on page A-13

Baseband-Equivalent Modeling

To create a complex baseband-equivalent model in the time domain based on the network parameters of the physical system, the RF Blockset performs a mathematical transformation that consists of the following three steps:

- 1 “Calculating the Passband Transfer Function” on page A-10
- 2 “Calculating the Baseband Transfer Function” on page A-12
- 3 “Calculating the Baseband-Equivalent Impulse Response” on page A-13

Calculating the Passband Transfer Function

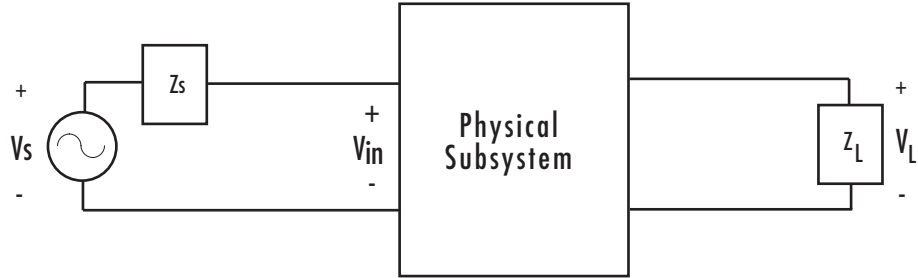
The RF Blockset calculates the passband transfer function from the physical block parameters at the modeling frequencies.

Note To learn how the RF Blockset uses the specified network parameters to compute the network parameters at the modeling frequencies, see “Mapping Network Parameters to Modeling Frequencies” on page A-5.

The transfer function is defined as

$$H(f) = \frac{V_L(f)}{V_S(f)}$$

where V_S and V_L are the source and load voltages shown in the following figure, and f represents the modeling frequencies.



More specifically,

$$H(f) = \frac{S_{21} * (1 + \Gamma_l) * (1 - \Gamma_s)}{2 * (1 - S_{22} * \Gamma_l) * (1 - \Gamma_{in} * \Gamma_s)}$$

where

$$\Gamma_l = \frac{Z_l - Z_o}{Z_l + Z_o}$$

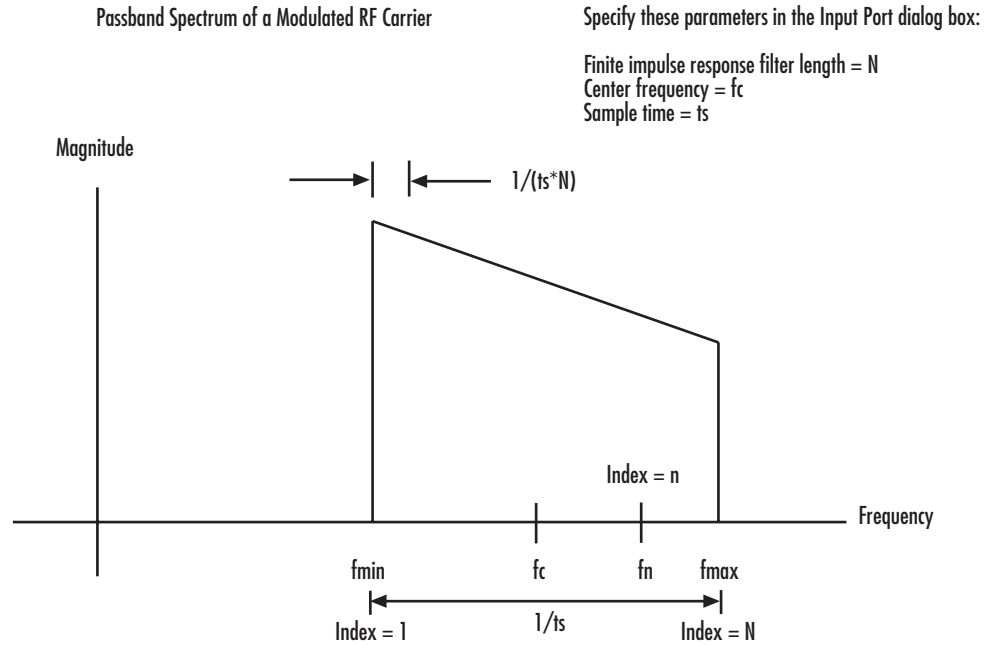
$$\Gamma_s = \frac{Z_s - Z_o}{Z_s + Z_o}$$

$$\Gamma_{in} = S_{11} + \left(S_{12} * S_{21} * \frac{\Gamma_l}{(1 - S_{22} * \Gamma_l)} \right)$$

and

- Z_S is the source impedance.
- Z_L is the load impedance.
- S_{ij} are the S-parameters of a two-port network.

The passband transfer function is shown in the following figure:



Calculating the Baseband Transfer Function

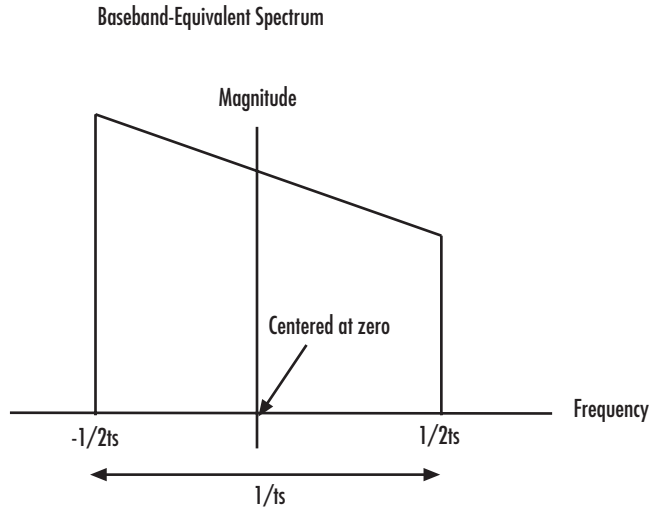
The RF Blockset calculates the baseband transfer function, $H_{baseband}(f)$, by translating the passband transfer function to its equivalent baseband transfer function:

$$H_{baseband}(f) = H_{passband}(f + f_c)$$

where f_c is the specified center frequency.

The resulting baseband-equivalent spectrum is centered at zero, so the RF Blockset can simulate the system using a much larger time step than Simulink can use for the same system. For information on why this translation allows for a larger time step, see “Simulation Efficiency of a Complex Baseband-Equivalent Model” on page A-13.

The baseband transfer function is shown in the following figure:



Calculating the Baseband-Equivalent Impulse Response

The RF Blockset calculates the baseband-equivalent impulse response by calculating the inverse FFT of the baseband transfer function. For faster simulation, the block calculates the IFFT using the next power of 2 greater than the specified finite impulse response filter length. Then, it truncates the impulse response to a length equal to the filter length specified.

Simulation Efficiency of a Complex Baseband-Equivalent Model

The baseband-equivalent modeling technique improves simulation speed by allowing the simulator to take larger time steps. To simulate a system in the time domain, Simulink would require a step size of:

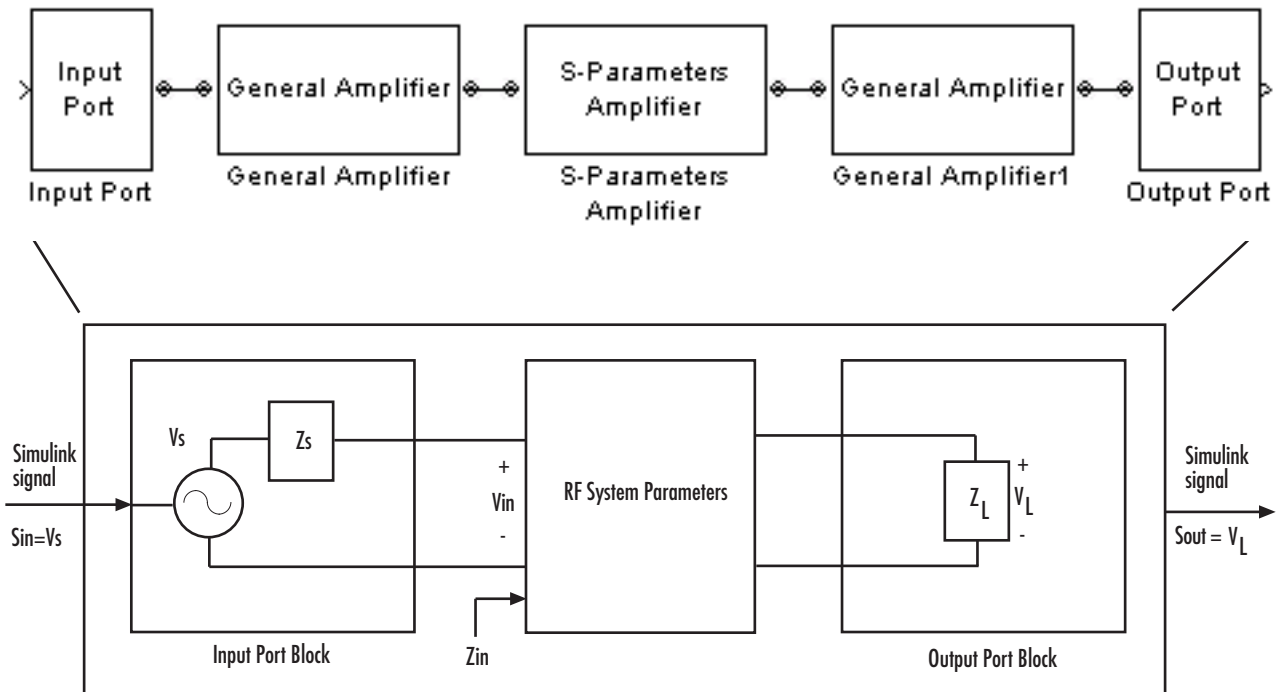
$$t_{step} = \frac{1}{2f_{max}}$$

Using the baseband-equivalent model of the same system, whose spectrum has been shifted down by f_c , allows for a much larger time step of:

$$t_{step} = \frac{1}{2(f_{\max} - f_c)} = \frac{1}{f_{\max} - f_{\min}}$$

Converting to and from Simulink Signals

When you simulate an RF model, the RF Blockset must convert the mathematical Simulink signals to and from the RF Blockset physical modeling environment. To perform this conversion, the RF Blockset interprets the Simulink signal that enters the Input Port block, S_{in} , as the voltage V_S across the source impedance Z_S . This process is shown in the following figure.

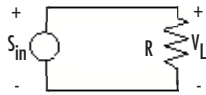


The RF Blockset interprets the output Simulink signal as the voltage V_L over the load impedance Z_L . You specify the source and load impedances in the Input Port and Output Port dialog boxes, respectively.

The RF Blockset interpretation of the input Simulink signal as the source voltage, V_S , produces different results than the interpretation that the input Simulink signal is the input voltage, V_{in} . When the source and load

impedances are the same, the former interpretation produces 6 dB of loss compared to the latter. You can see how this loss arises by analyzing the equivalent circuit model for each interpretation when there are no blocks between the Input Port and Output Port blocks.

The following figure shows the equivalent circuit model when the input Simulink signal, S_{in} , is taken to be the input voltage shown in the previous diagram.



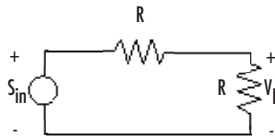
The load power is

$$P_L = \frac{V_L^2}{R} = \frac{S_{in}^2}{R}$$

In decibels, the load power is

$$10 \log(P_L) = 10 \log\left(\frac{S_{in}^2}{R}\right) \tag{A-1}$$

The following figure shows the equivalent circuit model when the input Simulink signal, S_{in} , is taken to be the source voltage shown in the previous diagram.



The load power is

$$P_L = \frac{V_L^2}{R} = \frac{\left(S_{in}/2\right)^2}{R}$$

In decibels, the load power is

$$10 \log(P_L) = 10 \log\left(\frac{S_{in}^2}{4R}\right) = 10 \log\left(\frac{S_{in}^2}{R}\right) - 6.02 \quad (\mathbf{A-2})$$

This load power is 6 dB less than the load power computed in Equation A-1.

Examples

Use this list to find examples in the documentation.

Examples

“Example — Modeling an LC Bandpass Filter” on page 1-12

“Example — Importing a Touchstone Data File into an RF Model” on page 2-8

“Example — Importing a Bandstop Filter into an RF Model” on page 2-15

“Example — Plotting Component Data on a Z Smith Chart” on page 3-15

“Example — Creating and Modifying Subsystem Plots” on page 3-22

A

- active noise
 - General Amplifier block 5-35
 - General Mixer block 5-44
- adding
 - noise 2-23
- adding physical and mathematical blocks 2-2
- algorithm
 - simulation A-2
- amplifier
 - modeling from file data 5-38
 - modeling from `rfdata` object 5-35
- Amplifier block 5-2
 - effects 5-5
 - modeling nonlinearity 5-2
 - methods 5-7
 - sample time comparison 5-11
 - thermal noise simulation 5-10
 - See also* General Amplifier block
- amplifier blocks
 - mathematical 4-2
 - physical 4-4

B

- Bandpass RF Filter block 5-16
 - sample time comparison 5-17
- Bandstop RF Filter block 5-20
 - sample time comparison 5-21
- baseband-equivalent model
 - creating A-10
 - simulation efficiency A-13
- baseband-equivalent time domain
 - modeling 5-98
 - linear subsystem 5-98
 - nonlinear subsystem 5-102
- Bessel filter design
 - Bandpass RF Filter block 5-16
 - Bandstop RF Filter block 5-20
 - Highpass RF Filter block 5-52

- Lowpass RF Filter block 5-86
- black box elements blocks 4-4
- block parameters
 - setting 2-6
- blocks
 - connecting 2-2
 - mathematical 4-2
 - physical 4-3
- Butterworth filter design
 - Bandpass RF Filter block 5-16
 - Bandstop RF Filter block 5-20
 - Highpass RF Filter block 5-52
 - Lowpass RF Filter block 5-86

C

- Chebyshev I filter design
 - Bandpass RF Filter block 5-16
 - Bandstop RF Filter block 5-20
 - Highpass RF Filter block 5-52
 - Lowpass RF Filter block 5-86
- Chebyshev II filter design
 - Bandpass RF Filter block 5-16
 - Bandstop RF Filter block 5-20
 - Highpass RF Filter block 5-52
 - Lowpass RF Filter block 5-86
- circuit element
 - modeling from `rfckt` object 5-42
- circuit objects
 - importing 2-14
- Coaxial Transmission Line block 5-24
 - shunt and series stubs 5-25
 - stubless 5-24
- composite data plots 3-5
- connecting blocks
 - RF Physical blocks to Simulink 2-2
 - Simulink to RF Physical blocks 2-2
- connector blocks 4-5
 - Input Port 5-56
 - Output Port 5-98

conversion

Simulink signal to RF signal A-15

Coplanar Waveguide Transmission Line

block 5-29

shunt and series stubs 5-30

stubless 5-29

D

data consistency

General Amplifier block 5-35

data files

importing 2-7

demos

in Help browser 1-4

E

Elliptic filter design

Bandpass RF Filter block 5-16

Bandstop RF Filter block 5-20

Highpass RF Filter block 5-52

Lowpass RF Filter block 5-86

example

creating and modifying plots 3-22

importing circuit objects 2-15

importing Touchstone file 2-8

F

filter blocks

mathematical 4-2

physical 4-3

filter design

physical blocks

LC Lowpass Tee block 5-83

filter design mathematical blocks

Bandpass RF Filter block 5-16

Bandstop RF Filter block 5-20

Highpass RF Filter block 5-52

Lowpass RF Filter block 5-86

filter design physical blocks

LC Bandpass Pi block 5-59

LC Bandpass Tee block 5-65

LC Bandstop Pi block 5-68

LC Bandstop Tee block 5-71

LC Highpass Pi block 5-74

LC Highpass Tee block 5-77

LC Lowpass Pi block 5-80

format

changing in plot 3-20

frequency range

changing in plot 3-20

G

General Amplifier block 5-35

active noise 5-35

generating from file data 5-38

nonlinearity 5-35

See also Amplifier block

General Circuit Element block 5-41

generating from rfckt object 5-42

General Mixer block 5-44

active noise 5-44

phase noise 5-45

See also Mixer block

General Passive Network block 5-49

generating from file data 5-50

Ghorbani model 5-9

H

Help browser

demos 1-4

Highpass RF Filter block 5-52

sample time comparison 5-53

I

importing

circuit objects 2-14

- data files 2-7
- Input Port block 5-56
 - parameter input for physical subsystem 5-56
 - sample time comparison 5-56
 - See also* Output Port block
- interpolation
 - network parameters A-5

L

- ladder filter blocks 4-3
- ladder filters
 - example to filter Gaussian noise 5-61
- LC Bandpass Pi block 5-59
 - example to filter Gaussian noise 5-61
- LC Bandpass Tee block 5-65
- LC Bandstop Pi block 5-68
- LC Bandstop Tee block 5-71
- LC Highpass Pi block 5-74
- LC Highpass Tee block 5-77
- LC Lowpass Pi block 5-80
- LC Lowpass Tee block 5-83
- libraries
 - RF physical and mathematical 1-6
- library
 - RF mathematical 1-9
 - RF physical 1-7
- link budget plots 3-7
- Lowpass RF Filter block 5-86
 - sample time comparison 5-87

M

- mathematical and physical models
 - connecting 2-2
- mathematical modeling blocks 1-9
- Microstrip Transmission Line block 5-90
 - shunt and series stubs 5-91
 - stubless 5-90

- Mixer block 5-95
 - sample time comparison 5-96
 - See also* General Mixer block
- mixer blocks
 - mathematical 4-2
 - physical 4-5
- model
 - adding RF components to 2-2
- modeling
 - amplifier from file data 5-38
 - amplifier from `rfdata` object 5-35
 - baseband-equivalent time domain 5-98
 - circuit element from `rfckt` object 5-42
 - passive network from file data 5-50
- modeling frequencies
 - calculating A-3
 - calculating noise A-8
 - S-parameter interpolation A-5
- models
 - physical and mathematical blocks 1-6

N

- network parameters
 - adding to a plot 3-20
 - interpolation at modeling frequencies A-5
- noise
 - active
 - General Amplifier block 5-35
 - General Mixer block 5-44
 - adding 2-23
 - calculating at modeling frequencies A-8
 - phase
 - General Mixer block 5-45
 - plotting 2-24
 - specifying mixer and amplifier 2-22
- noise output power
 - calculating A-9
- noise simulation A-7
- nonlinearity

- Amplifier block 5-2
- General Amplifier block 5-35

O

- opening
 - block libraries 1-6
- Output Port block 5-98
 - baseband-equivalent time domain modeling 5-98
 - plotting RF subsystem parameters 5-106
 - See also* Input Port block
- output power
 - calculating noise A-9

P

- Parallel-Plate Transmission Line block 5-107
 - shunt and series stubs 5-108
 - stubless 5-107
- parameter input
 - to physical subsystem 5-56
- parameters
 - adding to a plot 3-20
- passive network
 - modeling from file data 5-50
- phase noise
 - General Mixer block 5-45
- physical and mathematical models
 - connecting blocks 2-2
- physical modeling blocks 1-7
- physical subsystem
 - parameter input for 5-56
- plot types
 - modifying 3-20
- plots
 - budget 3-7
 - changing format of 3-20
 - changing frequency range of 3-20
 - composite 3-5

- creating 3-9
- displaying legend 3-15
- example of creating and modifying 3-22
- format 3-4
- modifying 3-20
- noise 2-24
- RF subsystem parameters 5-106
- Smith chart example 3-15
- types of 3-3
- updating 3-19
- using block parameters to specify 3-9
- X-Y plane 3-7
- product demos 1-4

R

- Rapp model 5-10
- RF block libraries
 - Mathematical 4-2
 - Physical 4-3
- RF Blockset
 - required and related products 1-3
- RF Blockset libraries 1-6
 - how to open 1-6
- RF models
 - connecting blocks 2-2
- rfckt object
 - modeling general circuit element 5-41
- rfddata object
 - modeling nonlinear amplifier 5-35
- RLCG Transmission Line block 5-112
 - shunt and series stubs 5-113
 - stubless 5-112

S

- S-Parameters Amplifier block 5-117
- S-Parameters Mixer block 5-122
- S-Parameters Passive Network block 5-126
- Saleh model 5-8

Series RLC block 5-131
 setting
 block parameters 2-6
 Shunt RLC block 5-134
 simulation
 noise A-7
 simulation algorithm
 overview A-2
 Simulink signal
 converting to and from A-15
 Smith chart
 example 3-15
 specifying
 mixer and amplifier thermal noise 2-22
 stubless transmission lines
 Coaxial Transmission Line block 5-24
 Coplanar Waveguide Transmission Line
 block 5-29
 Microstrip Transmission Line block 5-90
 Parallel-Plate Transmission Line
 block 5-107
 RLCG Transmission Line block 5-112
 Transmission Line block 5-137
 Two-Wire Transmission Line block 5-142
 stubs (shunt and series)
 Coaxial Transmission Line block 5-25
 Coplanar Waveguide Transmission Line
 block 5-30
 Microstrip Transmission Line block 5-91
 Parallel-Plate Transmission Line
 block 5-108
 RLCG Transmission Line block 5-113
 Transmission Line block 5-138
 Two-Wire Transmission Line block 5-143

T

thermal noise

 specifying mixer and amplifier 2-22
 time-domain modeling
 baseband-equivalent 5-98
 Touchstone file
 importing 2-8
 Transmission Line block 5-137
 shunt and series stubs 5-138
 stubless 5-137
 transmission line blocks 4-3
 Two-Wire Transmission Line block 5-142
 shunt and series stubs 5-143
 stubless 5-142

V

viewing
 demos 1-4

W

workflow
 RF Blockset 1-10
 workflow example
 RF Blockset 1-12

Y

Y-Parameters Amplifier block 5-147
 Y-Parameters Mixer block 5-152
 Y-Parameters Passive Network block 5-156

Z

Z-Parameters Amplifier block 5-160
 Z-Parameters Mixer block 5-165
 Z-Parameters Passive Network block 5-169