

# RF Blockset

For Use with Simulink®

- Modeling
- Simulation
- Implementation

## How to Contact The MathWorks:



www.mathworks.com	Web
comp.soft-sys.matlab	Newsgroup



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



508-647-7000	Phone
--------------	-------



508-647-7001	Fax
--------------	-----



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

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

### *RF Blockset User's Guide*

© COPYRIGHT 2004 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.

MATLAB, Simulink, Stateflow, Handle Graphics, and Real-Time Workshop are registered trademarks, and TargetBox is a trademark of The MathWorks, Inc.

Smith is a registered trademark of Analog Instruments Company, New Providence, NJ.

Touchstone is a U.S. registered trademark of Agilent Technologies, Inc.

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

Printing History: June 2004	Online only	New for Version 1.0 (Release 14)
August 2004	Online only	Revised for Version 1.0.1 (Release 14+)

## Introduction

**1**

<b>What Is the RF Blockset?</b> .....	<b>1-2</b>
<b>RF Blockset Libraries</b> .....	<b>1-3</b>
<b>Data Visualization</b> .....	<b>1-4</b>
<b>Data Format Support</b> .....	<b>1-5</b>
<b>Required Products</b> .....	<b>1-6</b>
<b>Demos</b> .....	<b>1-7</b>

## Building an RF Model

**2**

<b>RF Models</b> .....	<b>2-2</b>
Example of an RF Model .....	<b>2-2</b>
Physical and Mathematical RF Blocks .....	<b>2-2</b>
<b>Creating an RF Model</b> .....	<b>2-3</b>
RF Blockset Libraries .....	<b>2-3</b>
Setting RF Blockset Block Parameters .....	<b>2-6</b>

## Plotting Network Parameters

**3**

<b>Generating a Plot</b> .....	<b>3-2</b>
Block Parameters .....	<b>3-2</b>
Example of a Z Smith Chart .....	<b>3-5</b>

Examples of Other Plot Types . . . . . 3-7

**Modifying a Plot . . . . . 3-10**

**Block Reference**

**4**

---

**Blocks — Categorical List . . . . . 4-2**

    Mathematical . . . . . 4-2

    Physical . . . . . 4-4

**Blocks — Alphabetical List . . . . . 4-11**

**Index**

---

# Introduction

---

What Is the RF Blockset? (p. 1-2)

Introduces the RF Blockset and describes its capabilities.

RF Blockset Libraries (p. 1-3)

Summarizes the RF Blockset libraries.

Data Visualization (p. 1-4)

Lists the plots you can generate.

Data Format Support (p. 1-5)

Lists the data formats that the RF Blockset supports.

Required Products (p. 1-6)

Documents products required for use of the RF Blockset.

Demos (p. 1-7)

Tells you where to find RF Blockset demos.

## What Is the RF Blockset?

The RF Blockset is a tool for design, analysis, and simulation of RF (Radio Frequency) communications systems. It uses baseband equivalent behavioral modeling to model and analyze RF systems in the time domain.

RF technology is used to design and test RF circuits for cable television, wireless LAN, and other wireless applications such as broadcasting, radar, satellite communications, microwave relay, and mobile telephony.

The RF Blockset enables you to assemble complex RF systems from libraries of RF components such as ladder filters, transmission lines, black box elements, amplifiers, and mixers. You can include Simulink®, DSP Blockset, and Communications Blockset blocks in your model, as well as blocks from other MathWorks blocksets.

The RF Blockset contains

- **Physical blocks.** These blocks describe network components in terms of their geometry, physical interactions, and measured data. Internally, the RF Blockset characterizes these RF components in terms of their network parameters, and constructs the behavioral model for an RF subsystem from those parameters.
- **Mathematical blocks.** These are Simulink style blocks that describe components in mathematical terms. Each block models the component's behavior according to predefined mathematical relationships.

Two specialized blocks let you include mathematical and physical blocks in the same model by providing transitions from the mathematical environment into the RF Blockset physical modeling environment and back again into the mathematical environment. These blocks also enable you to specify certain parameters that are common to a physical modeling subsystem.

The RF Blockset works with the RF Toolbox. You can create complex topologies with the RF Toolbox, then read them into RF Blockset blocks for inclusion in an RF model.

## RF Blockset Libraries

The RF Blockset has two main libraries, Mathematical and Physical.

The Physical library includes the following sublibraries. Use blocks from these sublibraries to model the physical or electrical structure of portions of RF systems.

- Amplifiers — RF amplifiers described by S-, Y-, or Z-parameters, noise figure, and IP3, or a data file containing these parameters
- Ladder Filters — RF filters whose network parameters can be calculated from their topologies
- Mixers — RF mixers described by S-, Y-, or Z-parameters and phase noise, or a data file containing these parameters
- Transmission Lines — RF filters whose network parameters can be calculated from their geometry
- Black Box Elements — Passive RF components described by S-, Y-, or Z-parameters, or a data file containing these parameters
- Input/Output Ports — Blocks that connect mathematical portions of the model to the physical portions. Mathematical portions of the model may also include blocks from Simulink and other blocksets that describe components in mathematical terms.

The Mathematical library includes amplifier, mixer, and filter blocks. These blocks provide mathematical equivalents of the RF components.

See [Blocks — by Category](#) for more information about these libraries.

## **Data Visualization**

The RF Blockset enables you to plot the network parameters of the component blocks in the Physical library. For each such block, you can generate an X-Y plane plot, polar plane plot, or Smith® chart of selected network parameters in a specified frequency range. The dialog box for each block contains the parameters you need to specify the plot.

The RF Blockset also provides a composite plot which includes four separate plots in one figure. This predefined combination of plots differs based on the type of block.



## Data Format Support

The RF Blockset supports the Touchstone® S2P, Y2P, Z2P, and H2P data file formats. It also introduces the MathWorks AMP format for amplifier data. The AMP format is intended for use as input to the General Amplifier block.

For information about the Touchstone file formats, see  
[http://www.eda.org/pub/ibis/connector/touchstone\\_spec11.pdf](http://www.eda.org/pub/ibis/connector/touchstone_spec11.pdf).

For information about the AMP file format see “AMP File Format” in the RF Toolbox documentation.

## **Required Products**

The RF Blockset requires MATLAB®, Simulink, the RF Toolbox, and the Signal Processing Blockset. You may also find the Communications Blockset especially useful and it is highly recommended.

Note that the Signal Processing Blockset requires the Signal Processing Toolbox and the Communications Blockset requires the Communications Toolbox.

## Demos

Demos of the RF Blockset capabilities are available on the Demos tab of the MATLAB Help browser. These demos show examples of linear filtering and nonlinear amplification, as well as transmitters and receivers.



# Building an RF Model

---

RF Models (p. 2-2)

Introduces the different block types needed to build an RF Blockset model and defines a requirement for connecting the different types.

Creating an RF Model (p. 2-3)

Describes the mechanics of creating an RF model

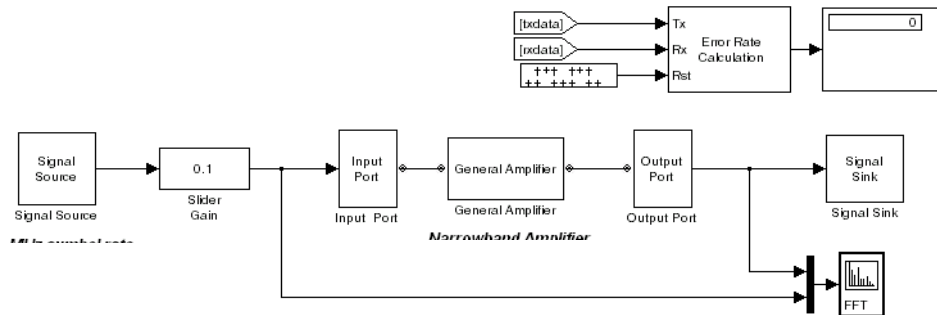
## RF Models

This section describes an RF model, the different styles of blocks in the RF Blockset, and a requirement in connecting different styles of blocks.

- “Example of an RF Model” on page 2-2
- “Physical and Mathematical RF Blocks” on page 2-2

### Example of an RF Model

An RF model consists of a block diagram that contains one or more schematics, each of which is a set of connected blocks representing an RF system. For example, the following model contains two schematics. The lower schematic displays the FFT of the input and output signals of a physical RF component – an amplifier.



### Physical and Mathematical RF Blocks

The RF Blockset includes both physical modeling blocks and mathematical modeling blocks. Physical blocks describe RF components in terms of their geometry, topologies, or network parameters. Mathematical blocks, which are Simulink style blocks, describe components in mathematical terms.

In the diagram above, the Input Port, General Amplifier, and Output Port blocks represent a physical component network. Other blocks represent the mathematical portions of the model. Note that the RF Blockset requires that you use the Input Port and Output Port blocks to connect the physical and mathematical portions of the model. Subsequent topics explain the use of these blocks in more detail.

## Creating an RF Model

This section describes the mechanics of creating an RF model.

- “RF Blockset Libraries” on page 2-3
- “Connecting RF Blocks” on page 2-4
- “Setting RF Blockset Block Parameters” on page 2-6

### RF Blockset Libraries

The RF Blockset consists of two main libraries, Physical and Mathematical. Blocks in the Physical library model RF network components in terms of their geometry, physical interactions, and measured data. Blocks in the Mathematical library model component behavior according to predefined mathematical relationships.

An RF model can contain blocks from both the Physical and Mathematical libraries. It can also include Simulink, Signal Processing Blockset, and Communications Blockset blocks, as well as blocks from other MathWorks blocksets.

### Physical Library

The RF Blockset provides the following libraries of physical modeling blocks. Use these blocks to model the physical structure of portions of a component network. You may find it convenient to encapsulate these parts of the network into subsystems.

- Amplifiers
- Black Box Elements
- Ladder Filters
- Mixers
- Transmission Lines

The Physical Library also contains the Input/Output Connectors library. This library contains the Input Port and Output Port blocks that you use to connect the physical and mathematical portions of a model.

### Mathematical Library

The RF Blockset includes the following libraries of mathematical modeling blocks. These blocks provide mathematical equivalents of RF components.

- Amplifiers
- Filters
- Mixers

See [Blocks — by Category](#) for more information about these libraries.

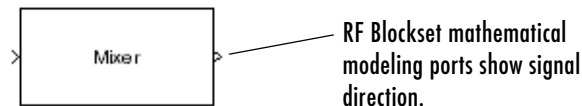
### Connecting RF Blocks

To create an RF model, first open a model window and open the RF Blockset library by typing `rf1ib` at the Command Window prompt. Then drag instances of RF Blockset blocks into the model window and draw lines to connect the blocks.

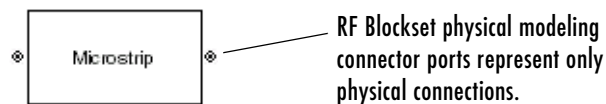
In general, you connect RF Blockset blocks in the same way you connect Simulink blocks. However, RF Blockset physical modeling blocks and mathematical blocks have different styles of ports and you cannot connect them directly.

### Block Ports

Just as for standard Simulink blocks, the RF Blockset mathematical modeling blocks have input and output ports. The ports show the direction of the signal at the port.



In contrast, the RF Blockset physical modeling blocks have specialized *connector ports*. These ports represent only physical connections; they do not imply signal direction.





## Connecting the Blocks

Just as for standard Simulink blocks, the *signal lines* that you draw between the ports of RF Blockset mathematical modeling blocks represent inputs to and outputs from the mathematical functions represented by the blocks.

In contrast, the lines you draw between RF Blockset physical modeling blocks, called *connection lines*, represent physical connections among the components represented by the blocks. You can draw connection lines only between the connector ports of physical modeling blocks. You cannot branch these connection lines. Connection lines appear as solid black when connected and as dashed red lines when either end is unconnected.

---

**Note** Physical blocks must be fully connected before you run the model. If they are not, an error is produced.

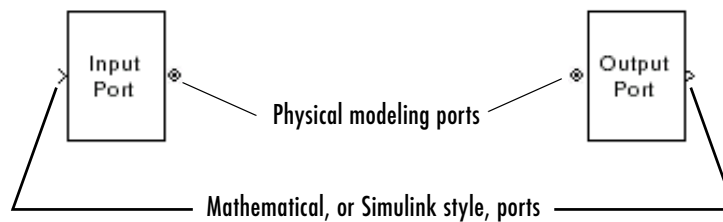
---

You can freely connect Simulink, Signal Processing, and RF Blockset mathematical modeling blocks. However, you cannot directly connect any of these blocks to an RF Blockset physical modeling block. The RF Blockset provides special blocks to connect the physical and mathematical parts of the model.

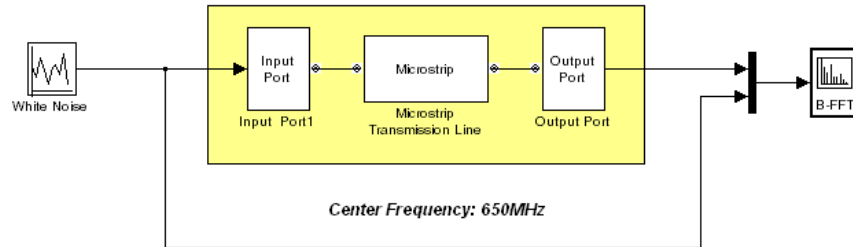
## Connecting Physical and Mathematical Parts of a Model

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

The Input Port and Output Port blocks have one of each kind of connector port. The Input Port block has a standard Simulink style input port and one physical modeling port. The Output Port block has one physical modeling port and a standard Simulink style output port.



A simple RF model of a coaxial transmission line would might look like 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 both Signal Processing Blockset blocks.



The Input Port block dialog box also enables you to provide the parameter data needed to calculate the baseband equivalent impulse response for the bracketed physical subsystem. The Output Port block dialog box enables you to specify the load impedance for the physical subsystem. See the Input Port and Output Port reference pages for more information about these blocks.

### Setting RF Blockset Block Parameters

You can set RF Blockset block parameters using either the block dialog boxes or the Simulink `set_param` and `get_param` commands. Open the dialogs by double-clicking the block.

# Plotting Network Parameters

---

Plotting Network Parameters contains these sections.

Generating a Plot (p. 3-2)

Tells you how to generate plots and describes the available plots.

Modifying a Plot (p. 3-10)

Tells you how to modify existing plots.

## Generating a Plot

Blocks in the RF Physical Library that model physical components provide a plotting capability that enables you to plot the network parameters of those components. The Output Port block provides a plotting capability only after you have run the model of which it is a part.

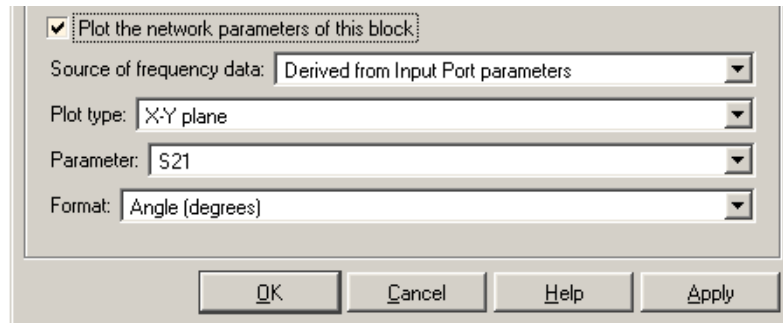
You can request a plot or change a plot whenever the model is not running. While the model is running, any existing plots continue to be displayed, but are not updated.

This section contains these topics.

- “Block Parameters” on page 3-2
- “Example of a Z Smith Chart” on page 3-5
- “Examples of Other Plot Types” on page 3-7

### Block Parameters

Blocks that enable you to request a plot provide plotting parameters in their block dialog boxes. Choose the plot parameters to define the plot you want, then click **Apply**.



#### Plot the network parameters of this block

Check this box to produce a plot. The plot is generated when you click **Apply** or **OK**.

**Source of frequency data**

Source of the frequencies at which the data is plotted. Choices vary for different blocks. All blocks can derive the frequencies from the Input Port block parameters, or can accept a user-specified vector of frequencies. Some blocks can use the specified mask **Frequency** parameter, or extract the frequencies from the object specified in the **RFDATA object** mask parameter. The Output Port block can use the derived simulation frequencies, once you have run the model.

**Frequency range (Hz)**

Specify, as a vector, the range of frequencies you want to plot. The frequencies must be strictly positive. This field becomes visible if you select User-specified as the source of frequency data.

**Plot type**

Type of plot. See table below.

**Parameter**

Network parameter to be plotted. The available choices vary with the type of plot. See table below.

**Format**

Format in which to plot the selected network parameter. The available choices vary based on the selected parameter. See table below.

The following table outlines the available choices.

<b>Plot Type</b>	<b>Parameters</b>	<b>Formats</b>
Composite Data	Generates four separate plots in one figure. The predefined combination of plots differs based on the type of block.	

<b>Plot Type</b>	<b>Parameters</b>	<b>Formats</b>
X-Y Plane	S11, S12, S21, S22	Magnitude (decibels) Magnitude (linear) Angle (degrees) Real Imaginary
	VSWRIn, VSWRout	Magnitude (decibels) None
	OIP3 (Output Port block only)	dBm W mW
	NF (Output Port block only)	Magnitude (decibels) None
	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)	dB None
	AM/PM (General Amplifier block with power data only)	Angle (degrees) Angle (radians)
	PhaseNoise (Mixer blocks only)	dBc/Hz
Polar Plane	S11, S12, S21, S22	None
Z Smith Chart	S11, S22	None
Y Smith Chart	S11, S22	None
ZY Smith Chart	S11, S22	None

---

**Note** To display the legend(s) for a plot, type `legend show` at the command line.

---

## Example of a Z Smith Chart

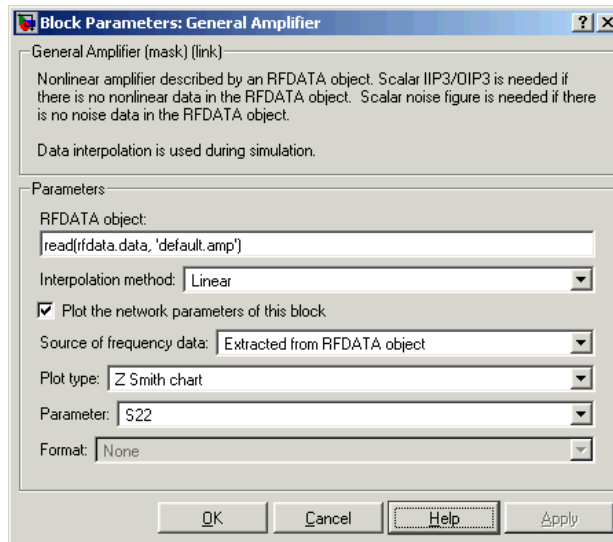
This example uses the RF Toolbox `read` function to create a data (`rfddata.data`) object that describes the nonlinear amplifier in the file `default.amp`. The file contains S-parameters for frequencies from 1.0 GHz to 2.9 GHz at intervals of 0.01 GHz. The General Amplifier block uses linear interpolation to model the network described in the object.

---

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

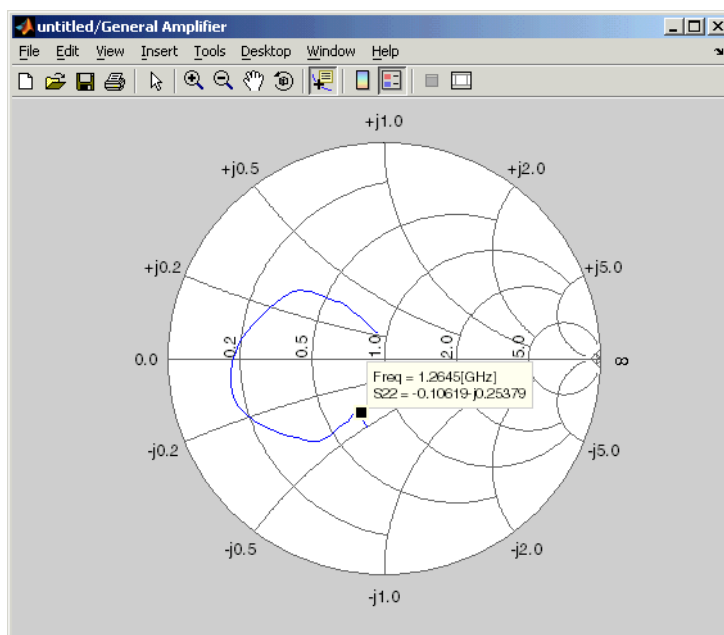
See “AMP File Format” in the RF Toolbox documentation for information about `.amp` files .

---



The plot parameters in the dialog request a Z Smith chart of the S22 parameters using frequency data extracted from the rfddata object.






---

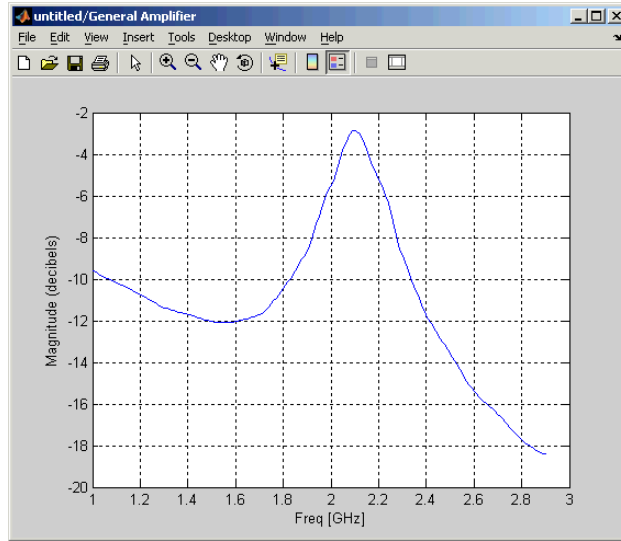
**Note** To display data tips for a plotted line, check **Data Cursor** on the **Tool** menu, and then click the data cursor on the plotted line to see the frequency and the plotted parameter at that point. See “Data Cursor — Displaying Data Values Interactively” in the MATLAB documentation for more information.

---

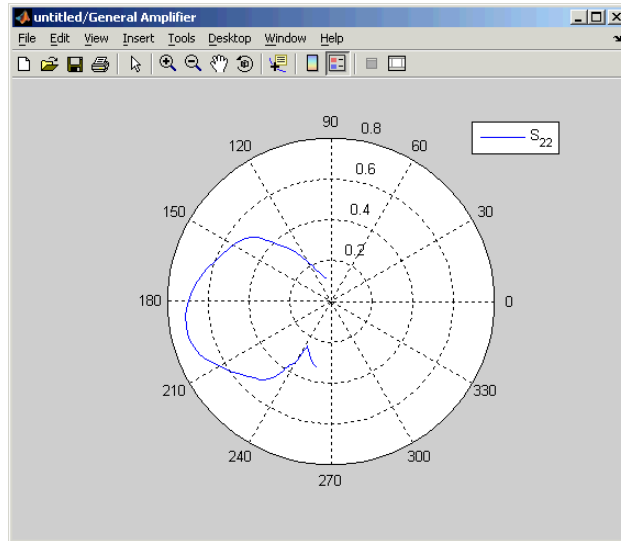
## Examples of Other Plot Types

The following figures show an instance of each of the other plot types for the same data, at the same frequency range.

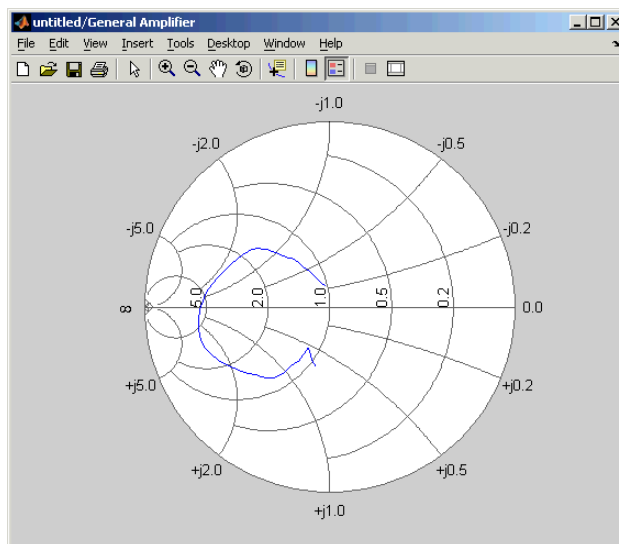
## X-Y Plane Plot



## Polar-Plane Plot

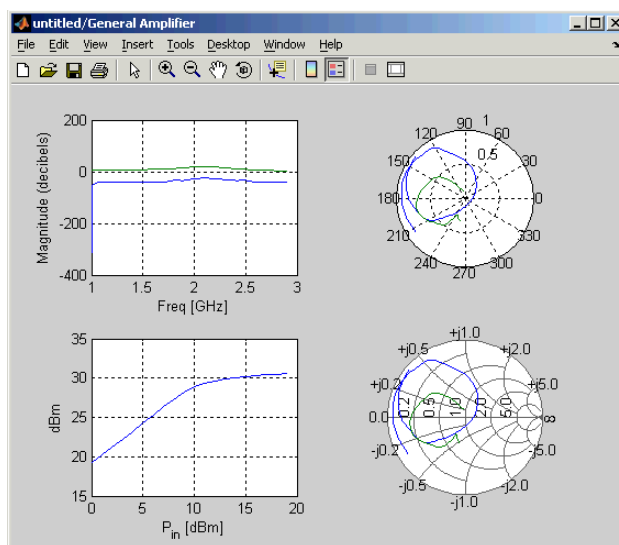


## Y Smith Chart



## Composite Data Chart

The actual combination of plots differs according to the type of block.



## Modifying a Plot

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

Block Parameter	Result of Change
Frequency range	The plot is redrawn using the new frequency range.
Plot type	The figure is redrawn in the new plot type. Other plot parameters retain their values if those values are valid for the new plot type. Else they revert to their default values.
Parameter	The new parameter is added to the existing plot.
Format	The plot is redrawn using the new format.

### Example

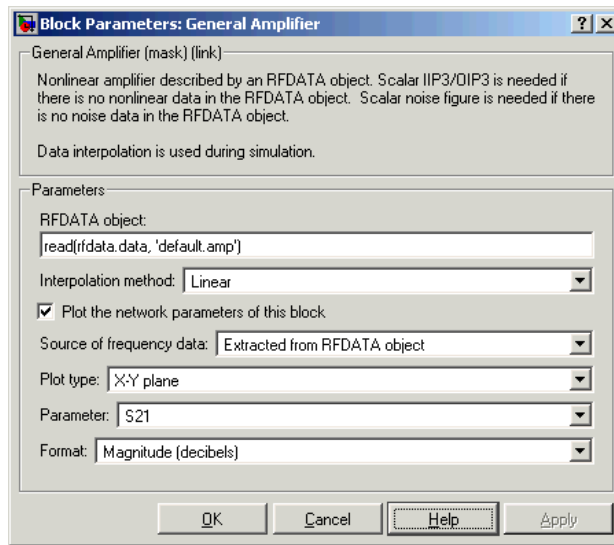
Start by generating an X-Y plane plot of the S21 parameters for the nonlinear amplifier described in the file `default.amp`, using frequency data extracted from the data (`rfddata.data`) object.

---

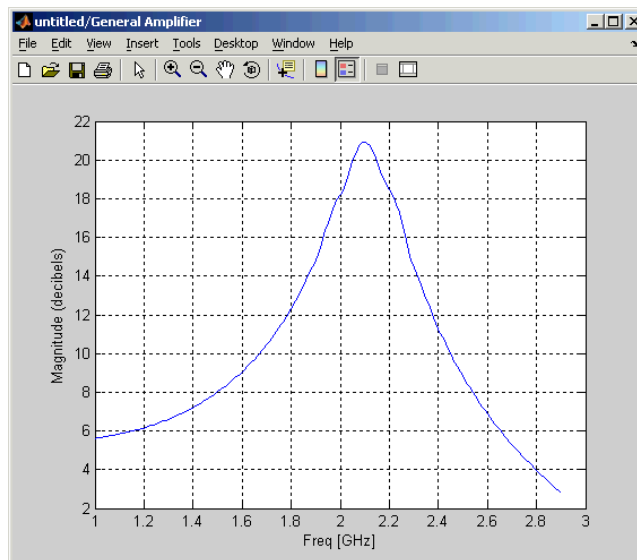
**Note** The General Amplifier block models the nonlinear amplifier described by an `rfddata.data` object. See the RF Toolbox `rfddata` reference page for information about `rfddata` objects.

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

---

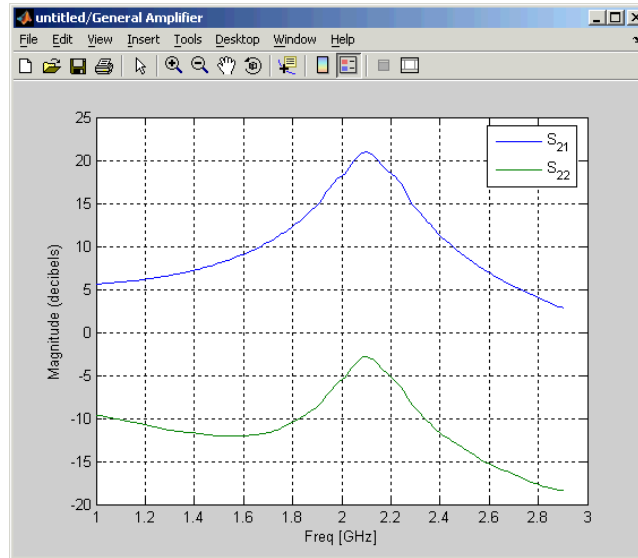


The block dialog above produces the plot below.

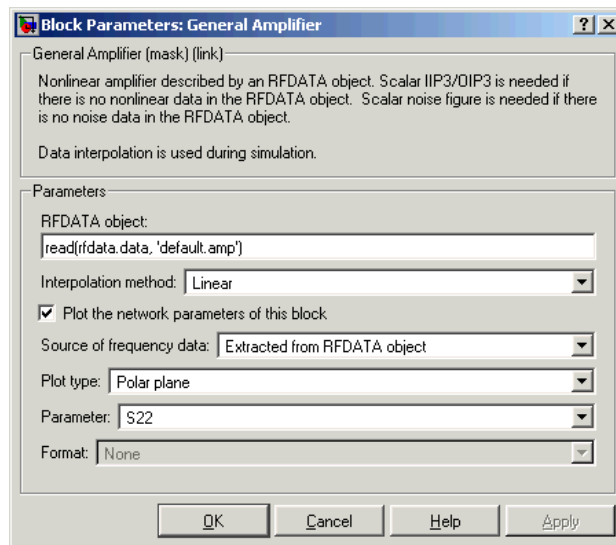


Add the S22 parameter to the plot by changing the value of **Parameter** to S22.

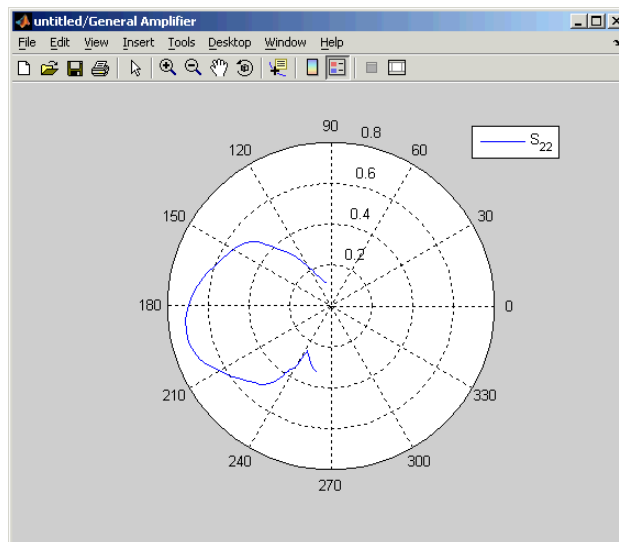
**Note** To display the legend for a plot, type `legend show` at the command line.



Change the **Plot type** to Polar plane. Note that the value of **Parameter** remains as S<sub>22</sub>, the last parameter selected for the previous plot.



This produces the following plot.







# Block Reference

---

- |                                      |                                                          |
|--------------------------------------|----------------------------------------------------------|
| Blocks — Categorical List (p. 4-2)   | Lists the RF Blockset blocks according to their purpose. |
| Blocks — Alphabetical List (p. 4-11) | Lists the RF Blockset blocks alphabetically.             |

## Blocks — Categorical List

This chapter contains detailed descriptions of all RF Blockset blocks. It first shows the libraries and lists their contents, and then presents the block reference entries in alphabetical order.

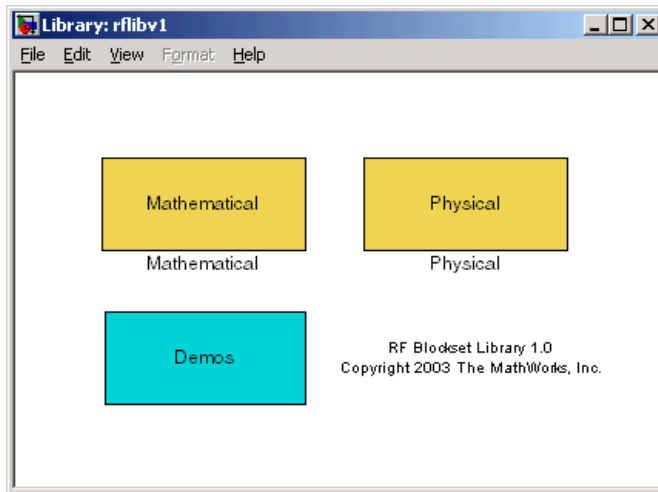
The main library is divided into these sublibraries:

- “Mathematical” on page 4-2 — Contains Simulink-style blocks that describe components in mathematical terms.
- “Physical” on page 4-4 — Contains blocks that model physical components in terms of their geometry and physical characteristics.

You can open the main library by typing

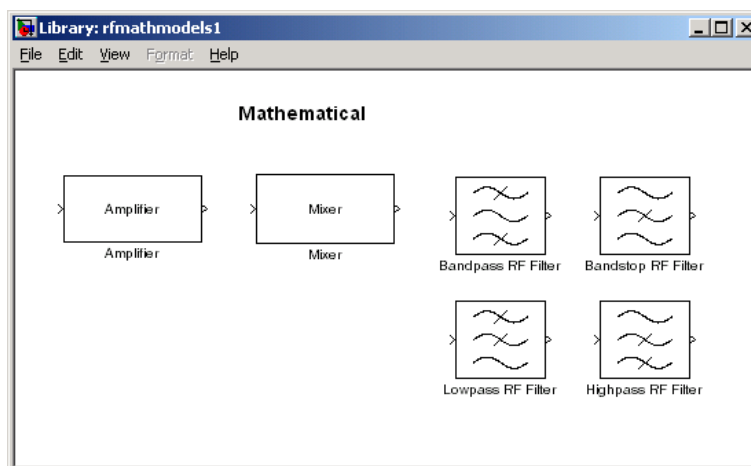
```
rflib
```

at the MATLAB prompt. Each yellow icon in the window represents a library. Double-click an icon to open the library.



### Mathematical

You can open the Mathematical library by double-clicking its icon in the main RF Blockset library.

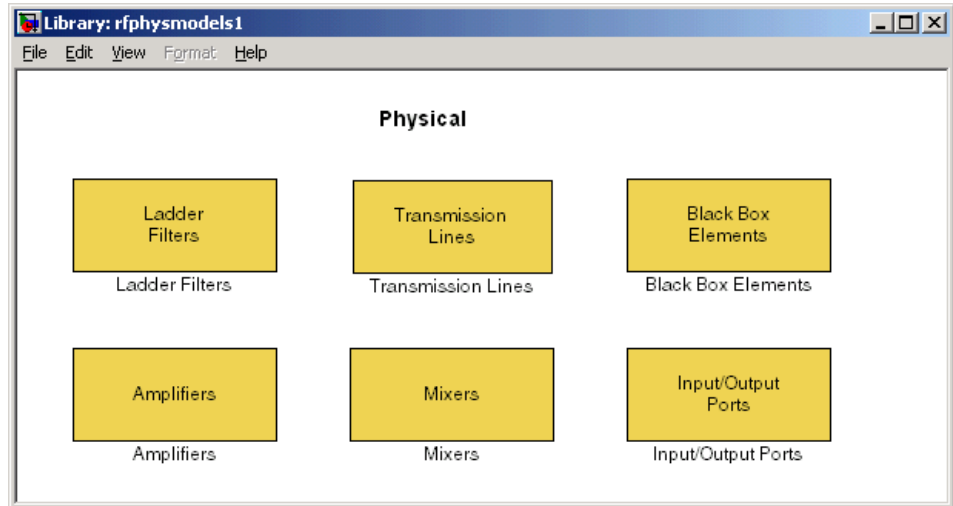


The table below lists the blocks in the Mathematical library. For information about a specific block, see its reference page.

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

## Physical

You can open the Physical library by double-clicking its icon in the main RF Blockset library.

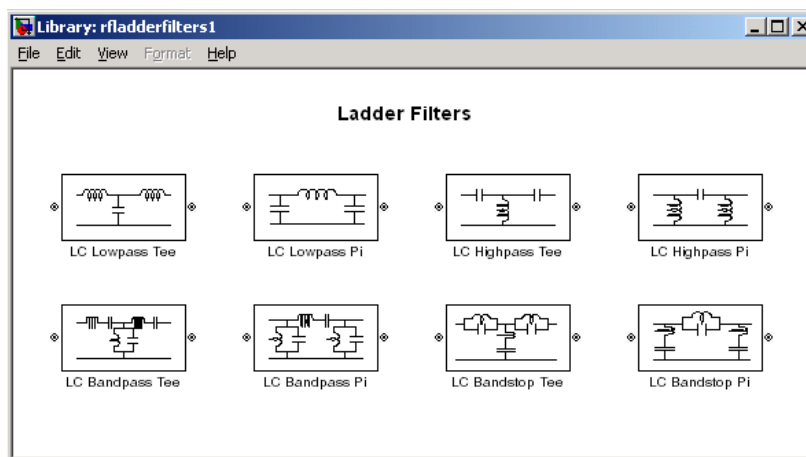


The Physical library contains these sublibraries:

- “Ladder Filters” on page 4-5
- “Transmission Lines” on page 4-6
- “Black Box Elements” on page 4-7
- “Amplifiers” on page 4-8
- “Mixers” on page 4-9
- “Input/Output Ports” on page 4-9

## Ladder Filters

The Ladder Filters library contains RF filters whose network parameters can be calculated from their topologies. You can open the Ladder Filters sublibrary by double-clicking its icon in the Physical library.

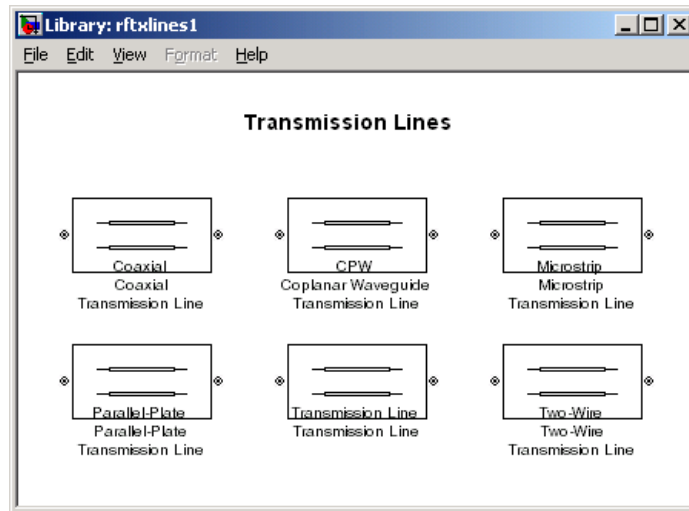


The table below lists the blocks in the Ladder Filters sublibrary. For information about a specific block, see its reference page.

Block Name	Purpose
LC Lowpass Pi	Model an LC lowpass pi network
LC Lowpass Tee	Model an LC lowpass tee network
LC Highpass Pi	Model an LC highpass pi network
LC Highpass Tee	Model an LC highpass tee network
LC Bandpass Pi	Model an LC bandpass pi network
LC Bandpass Tee	Model an LC bandpass tee network
LC Bandstop Pi	Model an LC bandstop pi network
LC Bandstop Tee	Model an LC bandstop tee network

## Transmission Lines

The Transmission Lines library contains RF filters whose network parameters can be calculated from their geometry. You can open the Transmission Lines sublibrary by double-clicking its icon in the Physical library.

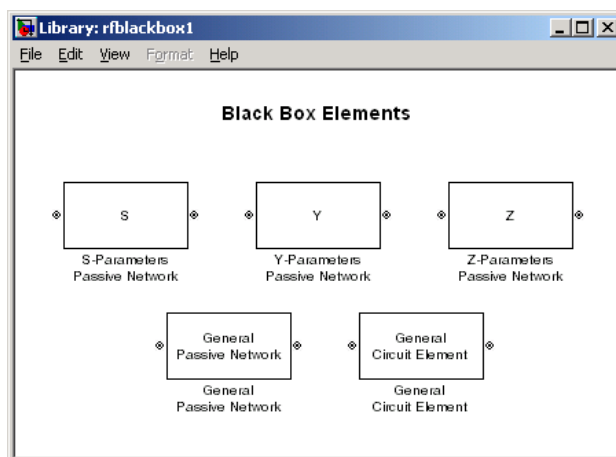


The table below lists the blocks in the Transmission Lines sublibrary. For information about a specific block, see its reference page.

Block Name	Purpose
Coaxial Transmission Line	Model a coaxial transmission line
Coplanar Waveguide Transmission Line	Model a coplanar waveguide transmission line
Transmission Line	Model an abstract transmission line
Microstrip Transmission Line	Model a microstrip transmission line
Parallel-Plate Transmission Line	Model a parallel-plate transmission line
Two-Wire Transmission Line	Model a two-wire transmission line

## Black Box Elements

Blocks in the Black Box Elements library describe passive RF components by their S-, Y-, or Z-parameters, or a data file containing these parameters. You can open the Black Box Elements sublibrary by double-clicking its icon in the Physical library.

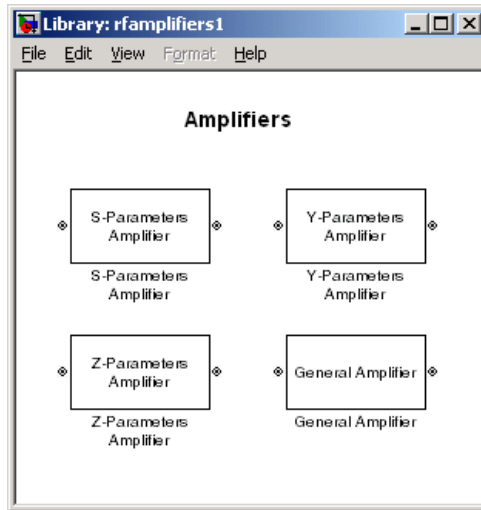


The table below lists the blocks in the Black Box Elements sublibrary. For information about a specific block, see its reference page.

Block Name	Purpose
S-Parameters Passive Network	Model a passive network using its S-parameters
Y-Parameters Passive Network	Model a passive network using its Y-parameters
Z-Parameters Passive Network	Model a passive network using its Z-parameters
General Passive Network	Model a two-port passive network described by an RFDATA object
General Circuit Element	Model a two-port passive network described by an RFCKT object

## Amplifiers

Blocks in the Amplifiers library describe RF amplifiers by S-, Y-, or Z-parameters, or an object containing these parameters. You can open the Amplifiers sublibrary by double-clicking its icon in the Physical library.



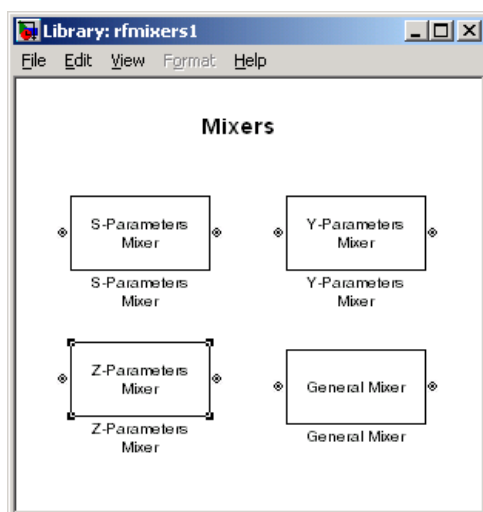
The table below lists the blocks in the Amplifiers sublibrary. For information about a specific block, see its reference page.

Block Name	Purpose
S-Parameters Amplifier	Model a nonlinear amplifier using its S-parameters
Y-Parameters Amplifier	Model a nonlinear amplifier using its Y-parameters
Z-Parameters Amplifier	Model a nonlinear amplifier using its Z-parameters
General Amplifier	Model a nonlinear amplifier described by an RFDATA object.



## Mixers

Blocks in the Mixers library describe RF mixers by S-, Y-, or Z-parameters, or an object containing these parameters. You can open the Mixers sublibrary by double-clicking its icon in the Physical library.



The table below lists the blocks in the Mixers sublibrary. For information about a specific block, see its reference page.

Block Name	Purpose
General Mixer	Model a Mixer described by an <code>rfdata.data</code> object
S-Parameters Mixer	Model a mixer using its S-parameters
Y-Parameters Mixer	Model a mixer using its Y-parameters
Z-Parameters Mixer	Model a mixer using its Z-parameters

## Input/Output Ports

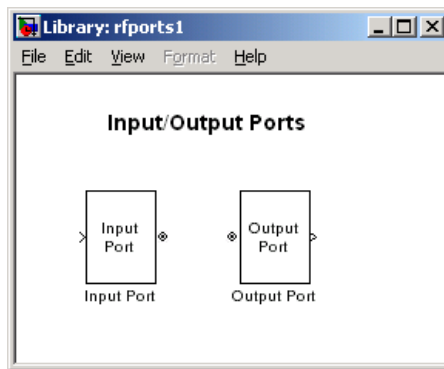
Blocks in the Input/Output Ports sublibrary enable you to transition from the mathematical environment into the physical modeling environment and from

the physical modeling environment back into the mathematical environment. You can open the Input/Output Ports sublibrary by double-clicking its icon in the Physical library.

---

**Note** For information about using these blocks, see “Connecting RF Blocks” on page 2-4.

---



The table below lists the blocks in the Input/Output Ports sublibrary. For information about a specific block, see its reference page.

Block Name	Purpose
Input Port	Provides a connection from the Simulink or mathematical environment to the RF physical modeling environment. It also enables you to specify certain parameters that are common to the RF physical blocks that are bracketed by the Input Port and Output Port blocks.
Output Port	Provides a connection from the RF physical modeling environment to the Simulink or mathematical environment. It also enables you to specify the load impedance for the physical part of the model between the Input Port and Output Port blocks.

## **Blocks — Alphabetical List**

This section contains block reference pages listed alphabetically.

# 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.

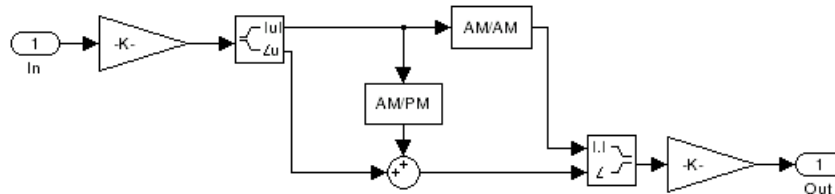


## Modeling Nonlinearity

Specify the method of your choice using the **Method** 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 add 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.

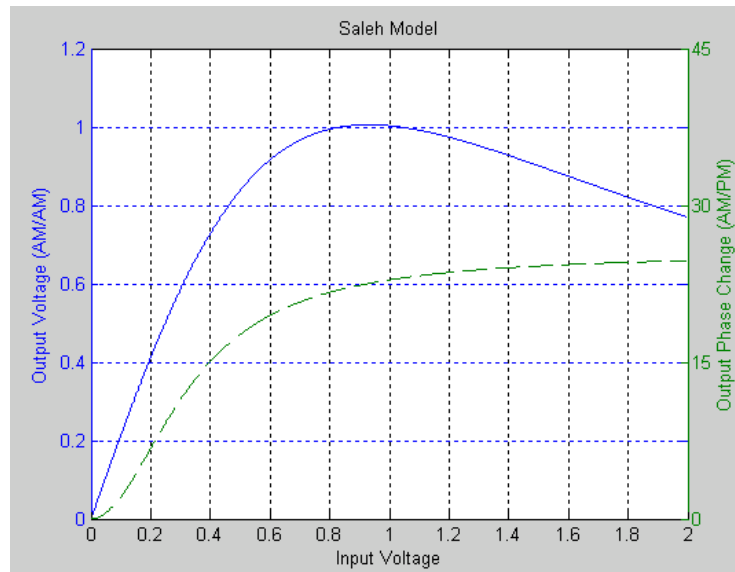
**AM/AM and AM/PM Conversions.** The subsystems for the nonlinear methods implement the AM/AM and AM/PM conversions differently, according to the interpolation method you specify. If you want to see exactly how the Amplifier block implements the conversions for a specific method, you can view the AM/AM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

- 1 Right-click on the Amplifier block.
- 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. This displays a subsystem similar to the one shown in the preceding figure.
- 4 Double-click on 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

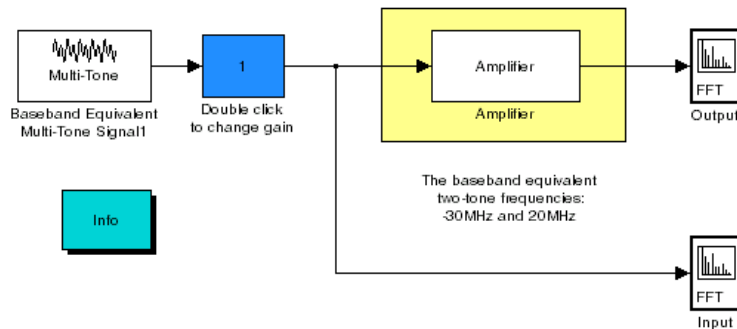
- Output voltage against input voltage for the AM/AM conversion
- Output phase against input voltage for the AM/PM conversion

# Amplifier



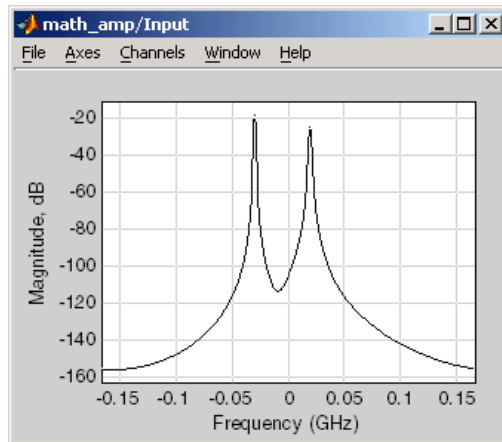
## Effects of the Amplifier Block

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

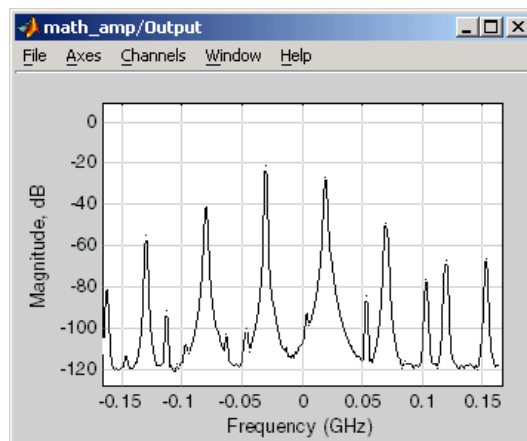


This demo uses a baseband equivalent multi-tone signal as input to the Amplifier block. A Simulink Slider Gain block enables you to vary the gain

between 1 and 10. The following figure shows the input signal with gain set to the default 1.



The next figure shows the same signal after it passes through the Amplifier block, with the **Method** parameter set to Ghorbani model. The demo uses the default Amplifier block **AM/AM parameters** [8.1081 1.5413 6.5202 -0.0718] and the **AM/PM parameters** to [4.6645 2.0965 10.88 -0.003], with no output scaling. The demo also specifies thermal noise as Noise figure, for which it uses the default 3.01 dB.



## Parameters for Selected Methods

The following sections discuss parameters specific to the Saleh, Ghorbani, and Rapp models.

---

**Note** The Amplifier block also enables you to model nonlinearity using linear, cubic polynomial, and hyperbolic tangent methods.

---

**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, alpha and beta, 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 alpha and beta, 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  $O_{\text{sat}}$  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** 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.

# Amplifier

---

- Noise temperature — specifies the noise in degrees 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 degrees Kelvin. In terms of noise factor,

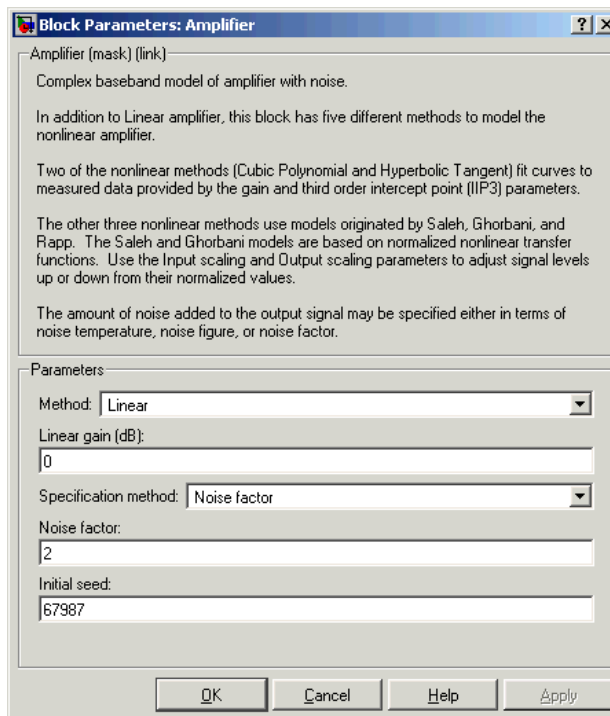
$$\text{Noise figure} = 10\log(\text{Noise factor})$$

---

**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 with 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 is produced.

---

## Dialog Box



The parameters displayed in the dialog box vary for different methods of modeling nonlinearity. Only some of the parameters listed below 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. 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.

## **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 degrees 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 degrees 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

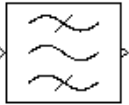
Bandpass RF Filter, Bandstop RF Filter, Highpass RF Filter, Lowpass RF Filter, Mixer

# Bandpass RF Filter

**Purpose** Standard bandpass RF filters in baseband equivalent complex form

**Library** Mathematical

## Description



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

<b>Design Method</b>	<b>Description</b>
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.

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.

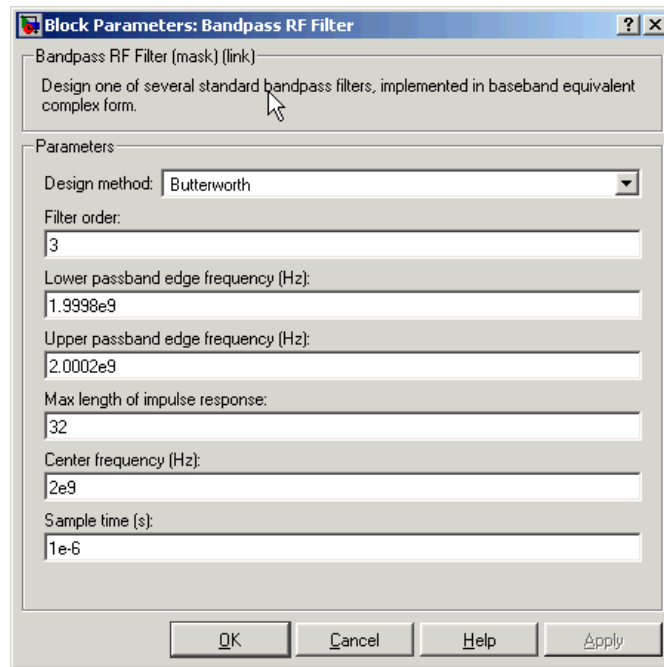
<b>Design Method</b>	<b>Filter Design Parameters</b>
Butterworth	Order, lower passband edge frequency, upper passband edge frequency
Chebyshev I	Order, lower passband edge frequency, upper passband edge frequency, passband ripple
Chebyshev II	Order, lower stopband edge frequency, upper stopband edge 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 with 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 is produced.

# Bandpass RF Filter

## Dialog Box



The parameters displayed in the dialog box vary for different design methods. Only some of the parameters listed below 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.

**Max length of impulse response**

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

**Center frequency (Hz)**

Center of the simulation frequencies.

**Sample time (s)**

Time interval between consecutive samples of the input signal.

**See Also**

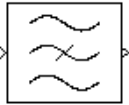
Amplifier, Bandstop RF Filter, Highpass RF Filter, Lowpass RF Filter, Mixer  
buttap, cheb1ap, cheb2ap, ellipap, besselap (Signal Processing Toolbox)

# Bandstop RF Filter

**Purpose** Standard bandstop RF filters in baseband equivalent complex form

**Library** Mathematical

## Description



The Bandstop RF Filter block enables you to 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.

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.

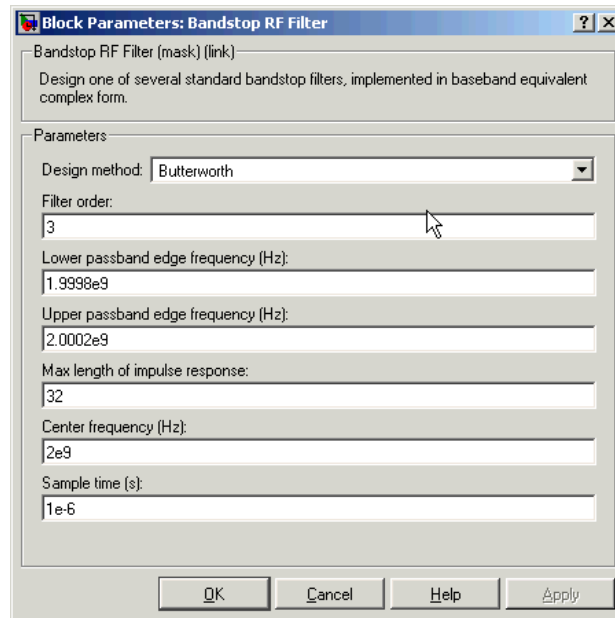
<b>Design Method</b>	<b>Filter Design Parameters</b>
Butterworth	Order, lower passband edge frequency, upper passband edge frequency
Chebyshev I	Order, lower passband edge frequency, upper passband edge frequency, passband ripple
Chebyshev II	Order, lower stopband edge frequency, upper stopband edge 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 with 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 is produced.

# Bandstop RF Filter

## Dialog Box



The parameters displayed in the dialog box vary for different design methods. Only some of the parameters listed below 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.

**Max length of impulse response**

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

**Center frequency (Hz)**

Center of the simulation frequencies.

**Sample time (s)**

Time interval between consecutive samples of the input signal.

**See Also**

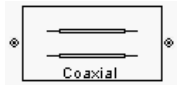
Amplifier, Bandpass RF Filter, Highpass RF Filter, Lowpass RF Filter, Mixer  
buttap, cheb1ap, cheb2ap, ellipap, besselap (Signal Processing Toolbox)

# Coaxial Transmission Line

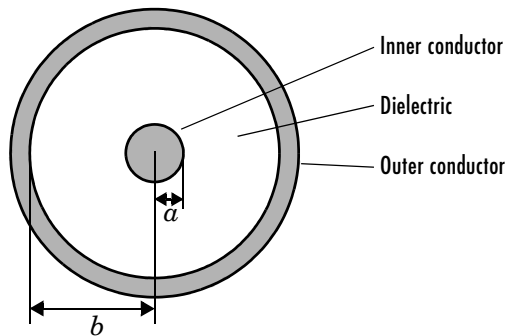
**Purpose** Model a coaxial transmission line

**Library** Transmission Lines sublibrary of the Physical library

## Description



The Coaxial Transmission Line block models the coaxial transmission line described in the block dialog 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 enables you to 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 simulation 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,  $\sigma_{\text{cond}}$  is the conductivity in the conductor and  $\sigma_{\text{diel}}$  is the conductivity in the dielectric.  $\mu$  is the permeability of the dielectric,  $\epsilon$  is its permittivity, and skin depth  $\delta$  is calculated as  $1/\sqrt{\pi f \mu \sigma_{\text{cond}}}$ .  $f$  is a vector of simulation frequencies for the specified parameters. See the Output Port block for information about determining the simulation frequencies.

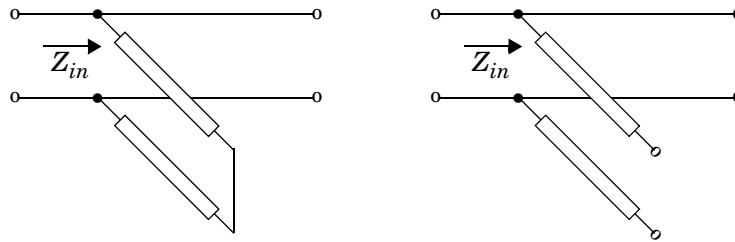
The Coaxial Transmission Line block normalizes the resulting S-parameters to a reference impedance of 50 ohms.

## 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 for each frequency contained in the simulation 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 2-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

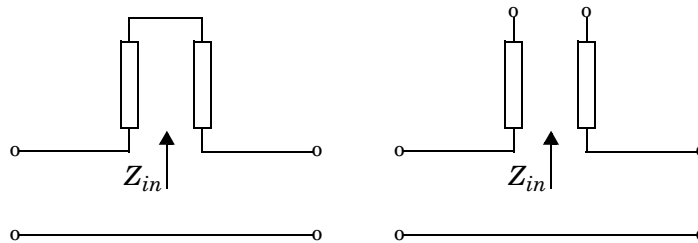
$$A = 1$$

$$B = 0$$

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

$$D = 1$$

**Series ABCD-Parameters.** When you set the **Stub mode** parameter in the mask dialog box to **Series**, the 2-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.

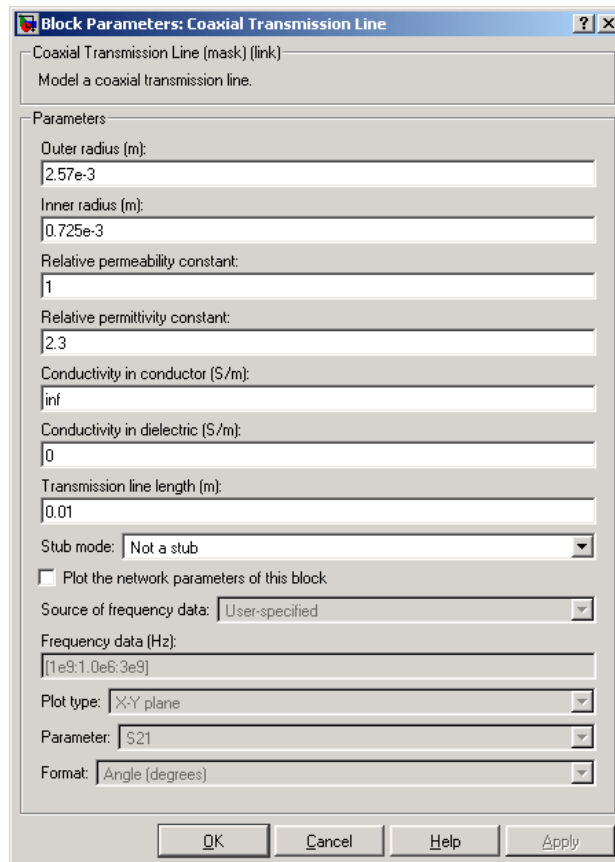


$Z_{in}$  is the input impedance of the series circuit. The ABCD-parameters for the series stub are calculated as



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

## Dialog Box



### Outer radius (m)

Radius of the outer conductor of the coaxial transmission line.

### Inner radius (m)

Radius of the inner conductor of the coaxial transmission line.

# 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  $\mu_0$ .

## **Relative permittivity constant**

Relative permittivity of the dielectric expressed as the ratio of the permittivity of the dielectric to permittivity in free space  $\epsilon_0$ .

## **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.

---

**Note** For information about plotting, see “Plotting Network Parameters” on page 3-1.

---

## **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 a coplanar waveguide transmission line

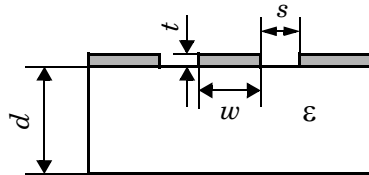
## Library

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 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 ( $\epsilon$ ).



The block enables you to 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  $\alpha_a$  is the attenuation coefficient and  $\beta$  is the wave number. The attenuation coefficient  $\alpha_a$  is related to the loss,  $\alpha$ , by

# Coplanar Waveguide Transmission Line

---

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

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

The wave number  $\beta$  is related to the phase velocity,  $V_p$ , by

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

$V_p = c / \sqrt{\epsilon_{\text{eff}}}$  where  $\epsilon_{\text{eff}}$  is the frequency dependent effective dielectric constant.  $f$  is a vector of simulation frequencies for the specified parameters. See the Output Port block for information about determining the simulation frequencies. The phase velocity  $V_p$  is also known as the wave propagation velocity.

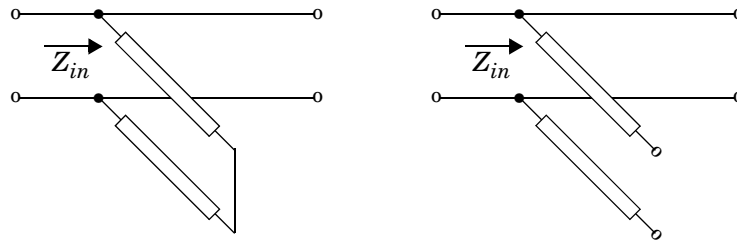
The Coplanar Waveguide Transmission Line block normalizes the resulting S-parameters to a reference impedance of 50 ohms.

## 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 for each frequency contained in the vector of simulation 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 2-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

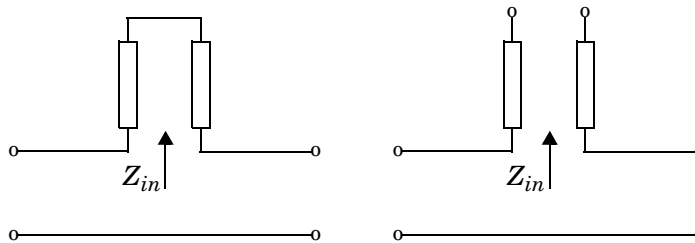
$$A = 1$$

$$B = 0$$

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

$$D = 1$$

**Series ABCD-Parameters.** When you set the **Stub mode** parameter in the mask dialog box to **Series**, the 2-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: 3.8

Conductivity in conductor (S/m): inf

Loss tangent in dielectric: 0

Transmission line length (m): 0.01

Stub mode: Not a stub

Plot the network 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

### Conductor width (m)

Physical width of the conductor.

# Coplanar Waveguide Transmission Line

---

**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  $\epsilon_0$ .

**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.

---

**Note** For information about plotting, see “Plotting Network Parameters” on page 3-1.

---

**References**

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

**See Also**

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

# General Amplifier

**Purpose** Model a nonlinear amplifier described by an `rfdata` object

**Library** Amplifiers sublibrary of the Physical library

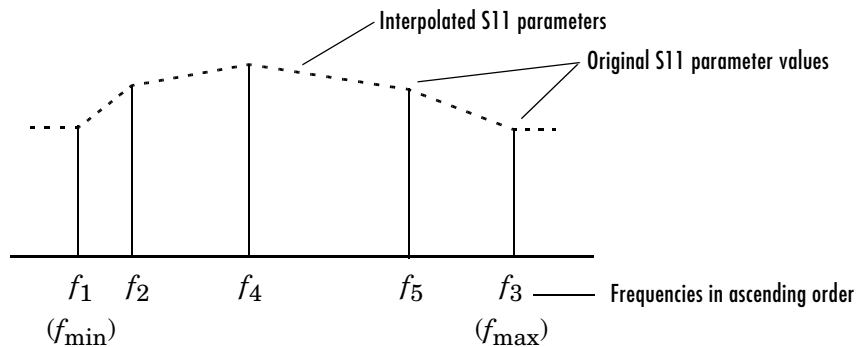
**Description** The General Amplifier block models the nonlinear amplifier described by an `rfdata.data` object. See the RF Toolbox `rfdata` reference page for information about `rfdata` objects.



## Network Parameters

If network parameter data and their corresponding frequencies exist as S-parameters in the `rfdata.data` object, the General Amplifier block interpolates the S-parameters to determine the S-parameters at the simulation frequencies. If the `rfdata.data` object contains network Y- or Z-parameters, the block first converts them to S-parameters.

Specifically, the block orders the S-parameters according to ascending order of their frequencies,  $f_n$ . It then 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 five different frequencies.



The **Interpolation** field in the block dialog 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 diagram above, the General Amplifier block uses the parameter values at  $f_{\min}$  for all simulation frequencies smaller than  $f_{\min}$ . It uses the parameters values at  $f_{\max}$  for all simulation frequencies greater than  $f_{\max}$ . In both cases, the results may not be accurate.

## Nonlinearity

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

If the `rfdata.data` object contains no power data, the General Amplifier block dialog enables you to enter either the OIP3 or IIP3 as a scalar value for nonlinearity.

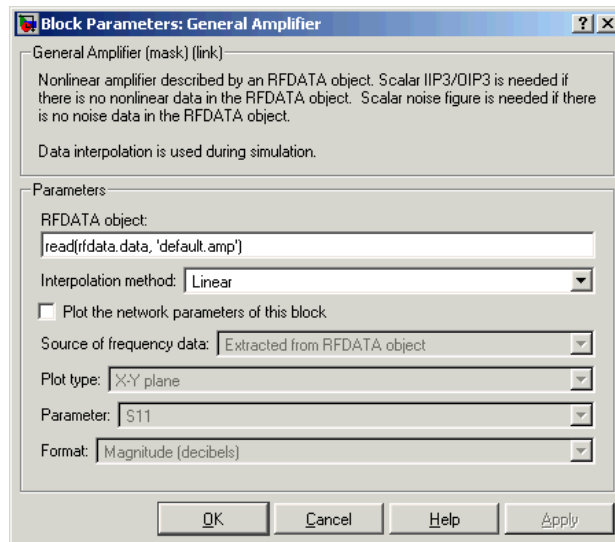
## Active Noise

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 simulation 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 Amplifier block dialog enables you to enter a value for the noise figure.

# General Amplifier

## 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 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.

### 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 `rfddata.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 network parameters, see “Plotting Network Parameters” on page 3-1. Use `RFTool` or the RF Toolbox plotting functions to plot other data.

---

## Examples

### Creating a General Amplifier Block from File Data

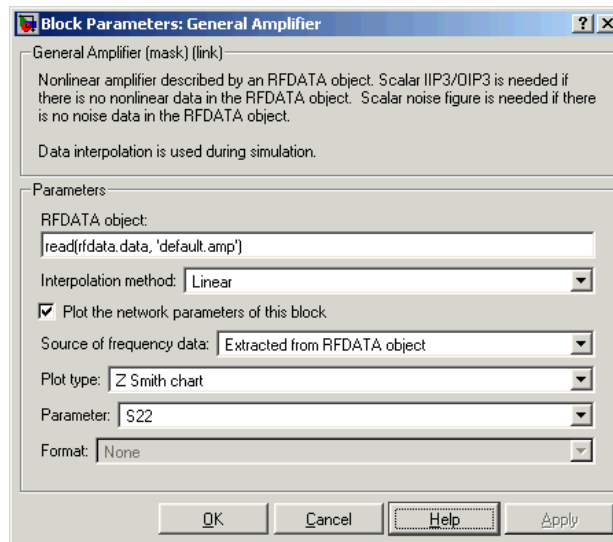
This example uses the RF Toolbox `read` function to create an `rfdata.data` object that describes the nonlinear amplifier in the file `default.amp`. The file, which is read into an RF Toolbox data (`rfdata.data`) object, contains S-parameters for frequencies from 1.0 GHz to 2.9 GHz at intervals of 0.01 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.

---

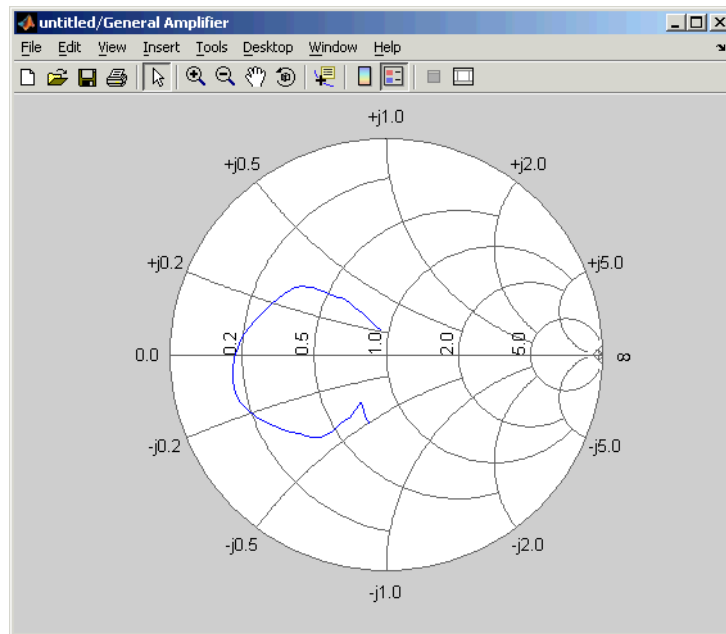
**Note** See “AMP File Format” in the RF Toolbox documentation for information about `.amp` files.

---

# General Amplifier



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



## See Also

S-Parameters Amplifier, Y-Parameters Amplifier, Z-Parameters Amplifier

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

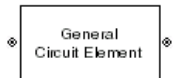
`interp1` (MATLAB)

# General Circuit Element

**Purpose** Model a two-port network described by an rfckt object

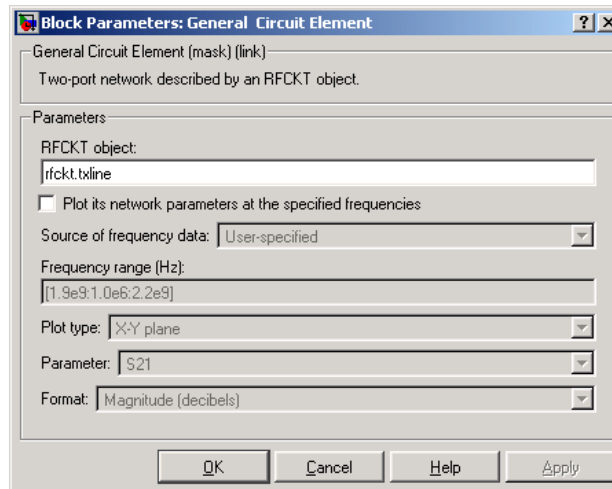
**Library** Black Box Elements sublibrary of the Physical library

**Description** The General Circuit Element block models the two-port network described by an rfckt object. See the RF Toolbox rfckt reference page for information about these objects and a list of all rfckt objects.



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

## Dialog Box



## RFCKT object

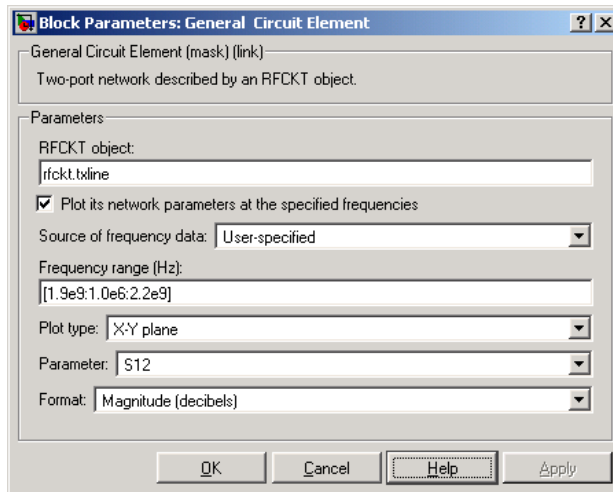
An RF Toolbox circuit (rfckt) object. You can specify the object as (1) the handle of a circuit object previously created using the RF Toolbox, (2) an RF Toolbox command such as rfckt.txline, rfckt.coaxial, or rfckt.cascade which creates a default circuit object of the specified type, or (3) a MATLAB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

**Note** For information about plotting, see “Plotting Network Parameters” on page 3-1.

## Examples

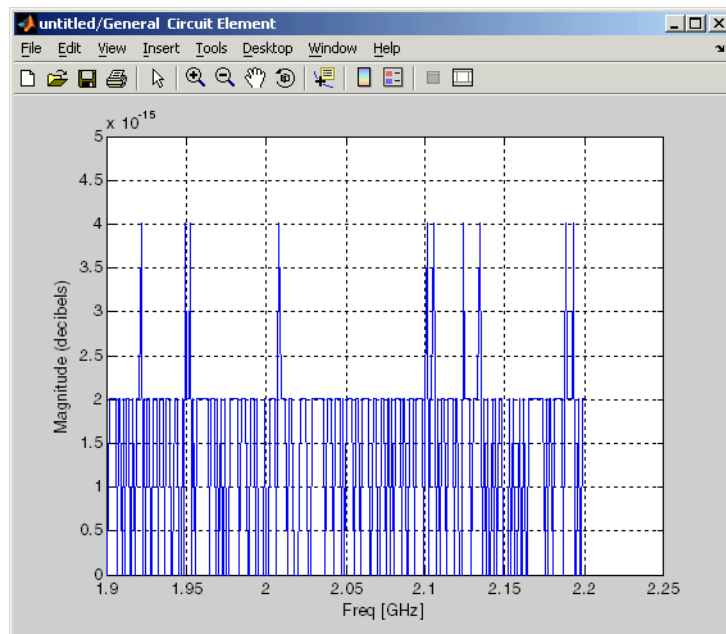
### Creating a General Circuit Element from an RF Toolbox Object

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



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

# General Circuit Element



## See Also

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

rfckt (RF Toolbox)

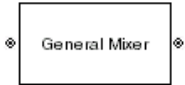
interp1 (MATLAB)



**Purpose** Model a mixer described by an `rfdata` object

**Library** Mixer sublibrary of the Physical library

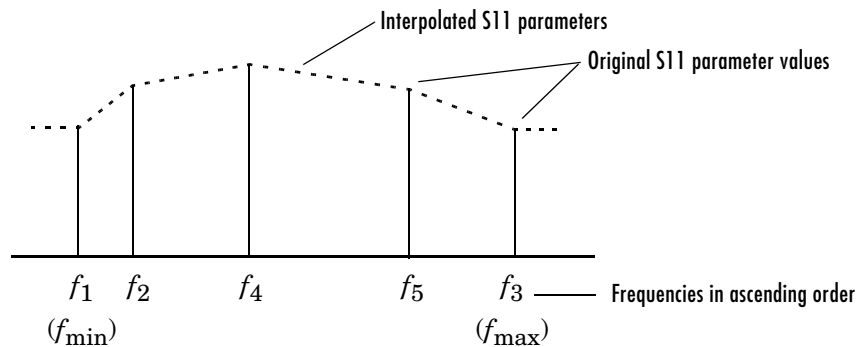
**Description** The General Mixer block models the mixer described by an `rfdata.data` object. See the RF Toolbox `rfdata` reference page for information about `rfdata` objects.



## Network Parameters

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 the S-parameters at the simulation frequencies. If the `rfdata.data` object contains network Y- or Z-parameters, the block first converts them to S-parameters.

Specifically, the block orders the S-parameters according to ascending order of their frequencies,  $f_n$ . It then 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 five different frequencies.



The **Interpolation** field in the block dialog 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.

# General Mixer

---

As shown in the diagram above, the General Mixer block uses the parameter values at  $f_{\min}$  for all simulation frequencies smaller than  $f_{\min}$ . It uses the parameters values at  $f_{\max}$  for all simulation frequencies greater than  $f_{\max}$ . In both cases, the results may not be accurate.

## Active Noise

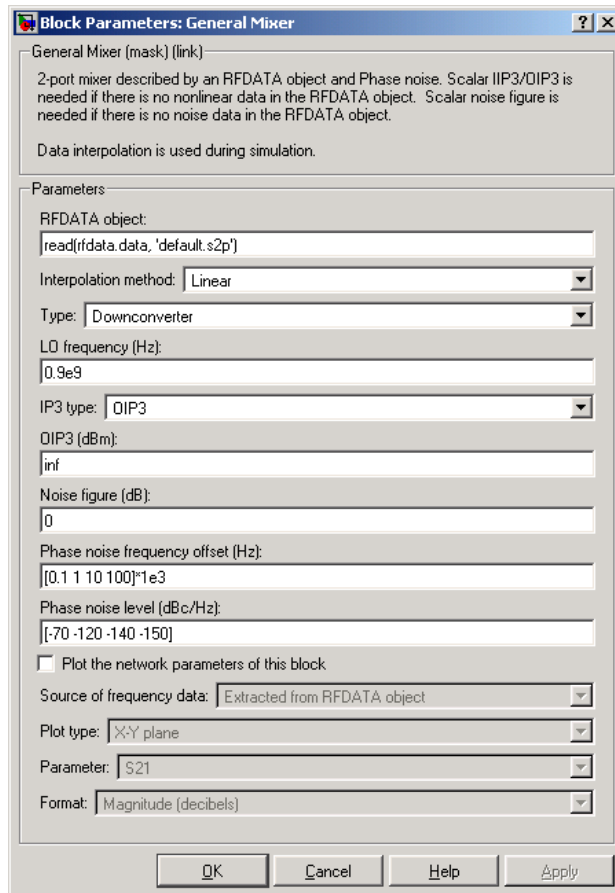
If active spot noise data exists in the `rfdata.data` object, the block uses the data to calculate the noise figure. It first interpolates the noise data for the simulation frequencies, using the specified **Interpolation method**. It then calculates the noise figure using the resulting values.

If the `rfdata.data` object contains no noise data, the General Mixer block dialog enables you to 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.

## 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`, which creates a default data object, or (3) a MATLAB expression that generates such an object. See the RF Toolbox for more information about data objects.

# General Mixer

---

## Interpolation method

For network data, the method used to interpolate the parameters contained in the `rfdata.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,  $f_{out} = f_{in} - f_{lo}$ .  
If you choose Upconverter,  $f_{out} = f_{in} + f_{lo}$ .

## 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 `rfdata.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 `rfdata.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 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 network parameters, see “Plotting Network Parameters” on page 3-1. Use RFTool or the RF Toolbox plotting functions to plot other data.

---

## See Also

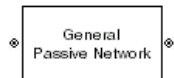
S-Parameters Mixer, Y-Parameters Mixer, Z-Parameters Mixer  
rfdata, rfdata.data (RF Toolbox)

# General Passive Network

**Purpose** Model a two-port passive network described by an `rfdata` object

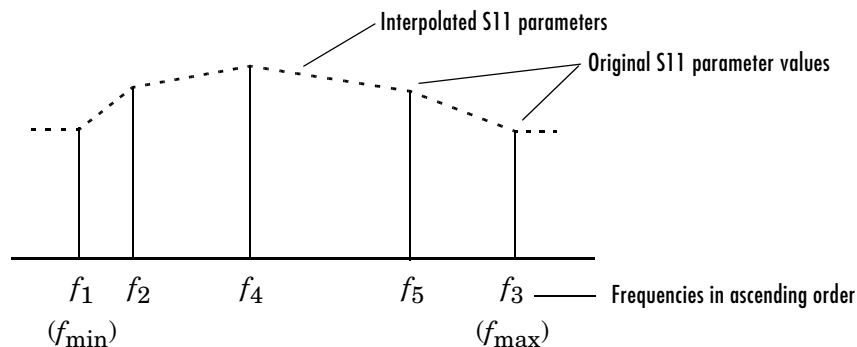
**Library** Black Box Elements sublibrary of the Physical library

## Description



The General Passive Network block models the two-port passive network described by an `rfdata.data` object. See the RF Toolbox `rfdata` reference page for information about these objects.

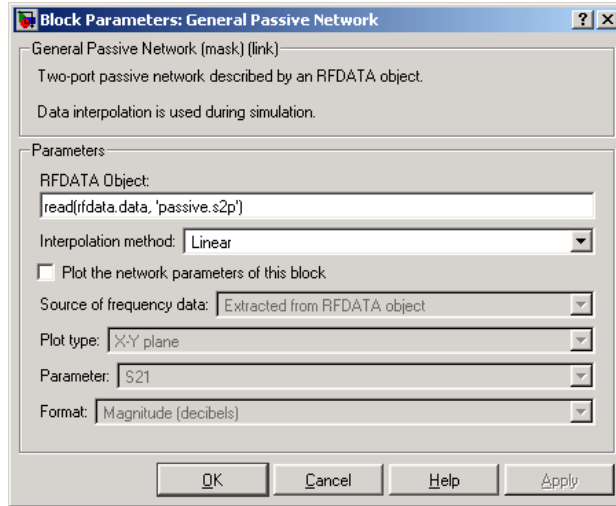
The block orders the parameters contained in the `rfdata` object according to the ascending order of their frequencies,  $f_n$ . It then interpolates the parameters, using the MATLAB `interp1` function, to determine the parameters at the simulation frequencies. For example, the curve in the following diagram illustrates the result of interpolating the S11 parameters at five different frequencies.



The **Interpolation** field in the block dialog 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 diagram above, the General Passive Network block uses the parameter values at  $f_{\min}$  for all simulation frequencies smaller than  $f_{\min}$ . It uses the parameters values at  $f_{\max}$  for all simulation frequencies greater than  $f_{\max}$ . In both cases, the results may not be accurate.

## Dialog Box



### RFDATA object

An RF Toolbox data (`rfdata.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 `rfdata.data`, which creates a default data object, or (3) a MATLAB expression that generates such an object. See the RF Toolbox for more information about data objects.

### Interpolation method

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

---

**Note** For information about plotting, see “Plotting Network Parameters” on page 3-1.

---

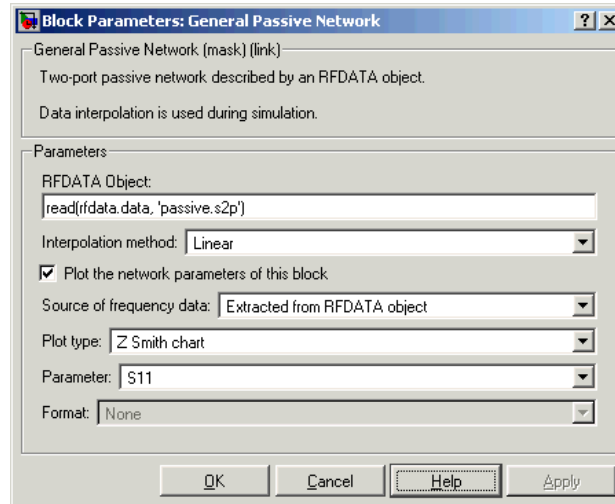
## Examples

### Create a General Passive Network Block from File Data

This example uses the RF Toolbox `read` function to create an `rfdata.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.

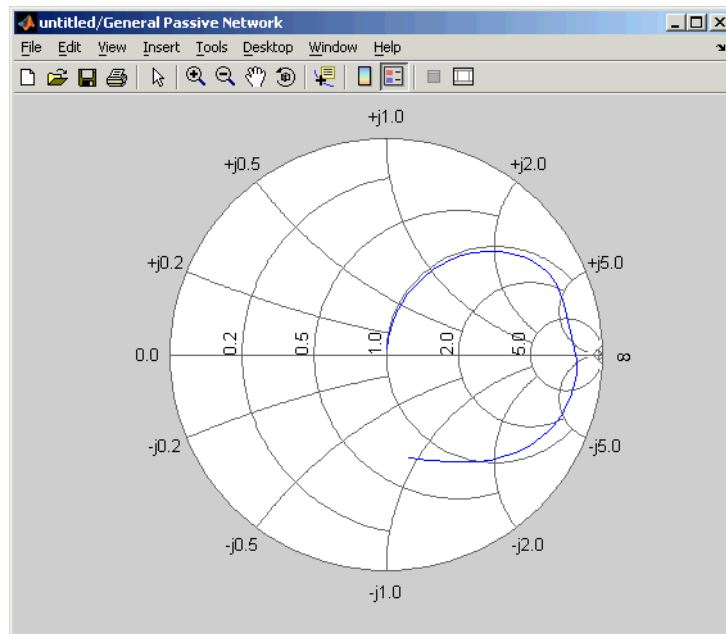
# General Passive Network

The General Passive Network block uses linear interpolation to model the network described in the object.



The plot parameters in the dialog request a Z Smith chart of the S11 parameters using the frequencies taken from the **RFDATA object** parameter.





## See Also

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

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

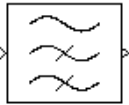
`interp1` (MATLAB)

# Highpass RF Filter

**Purpose** Standard highpass RF filters in baseband equivalent complex form

**Library** Mathematical

## Description



The Highpass RF Filter block enables you to 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.

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

Design Method	Filter Design Parameters
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`.

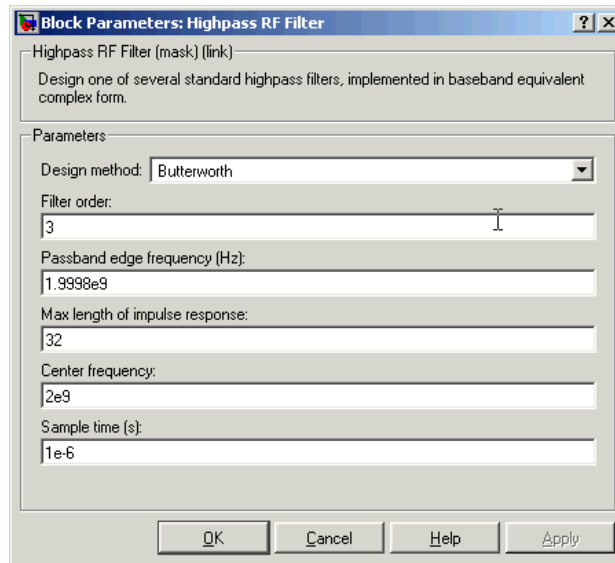
---

**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 with 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 is produced.

---

# Highpass RF Filter

## Dialog Box



The parameters displayed in the dialog box vary for different design methods. Only some of the parameters listed below 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.

**Max length of impulse response**

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

**Center frequency (Hz)**

Center of the simulation 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, besse1ap (Signal Processing Toolbox)

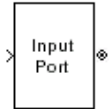
# Input Port

---

**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 enables you to provide the parameter data needed to calculate the simulation frequencies and the baseband equivalent impulse response for the physical subsystem.

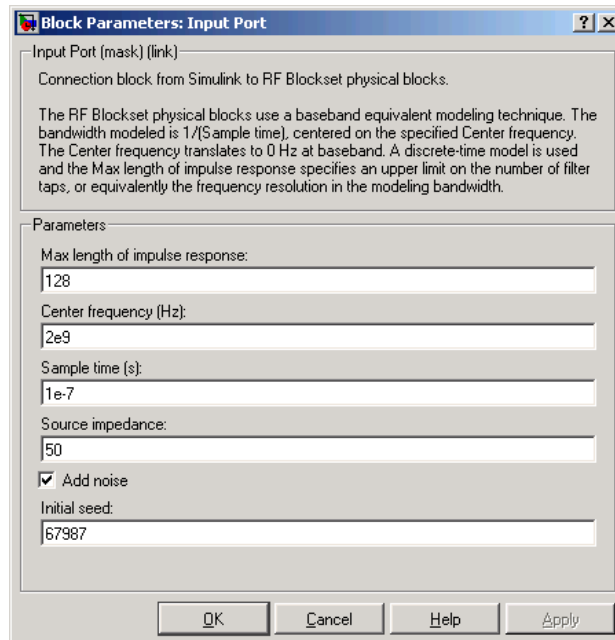
For more information about connecting mathematical and physical parts of a model, see “Building an RF Model” on page 2-1.

---

**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 with 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 is produced.

---

## Dialog Box



### Max length of impulse response

Desired length of the baseband equivalent impulse response for the physical model.

### Center frequency (Hz)

Center of the simulation frequencies. See the Output Port block for information about calculating the simulation frequencies.

### 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.

# Input Port

---

## **Add noise**

If you check 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 check 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 check the **Add noise** parameter.

## **See Also**

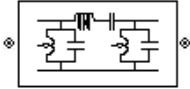
Output Port



**Purpose** Model an LC bandpass pi network

**Library** Ladder Filters sublibrary of the Physical library

## Description

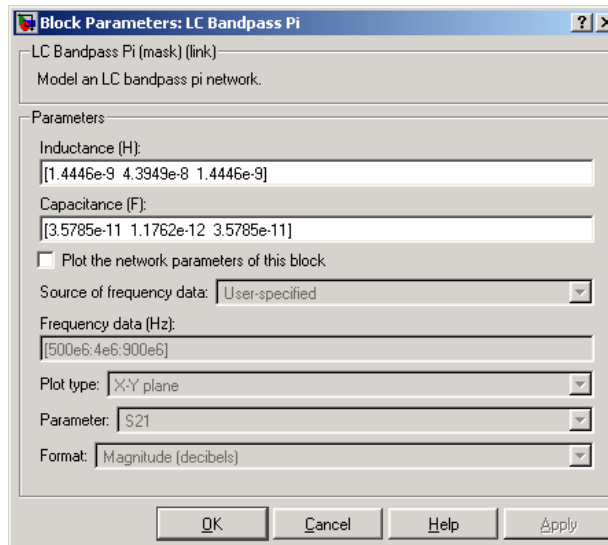


The LC Bandpass Pi block models the LC bandpass pi network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor pair in the network, the block first calculates the ABCD-parameters for each frequency contained in vector of simulation 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. See the Output Port block for information about the simulation frequencies.

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

## Dialog Box



# LC Bandpass Pi

## 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.

---

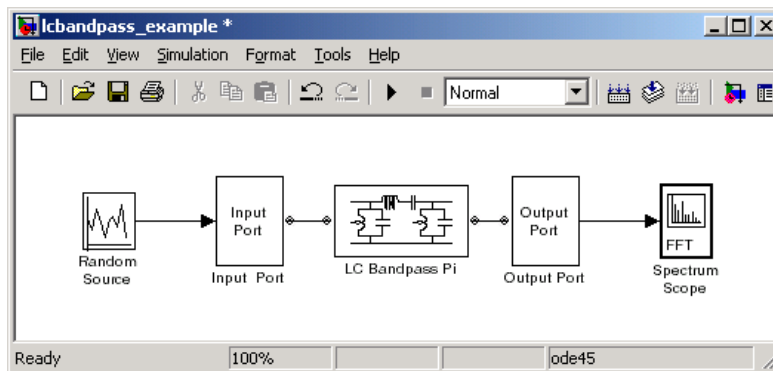
**Note** For information about plotting, see “Plotting Network Parameters” on page 3-1.

---

## Examples

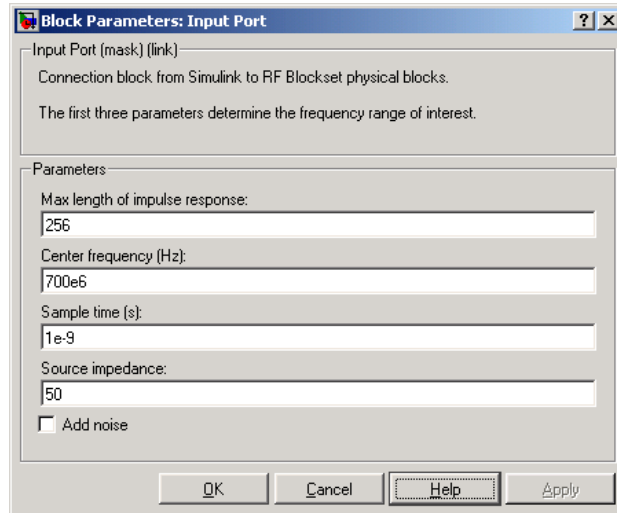
### Using a Ladder Filter Block to Filter Gaussian Noise

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.



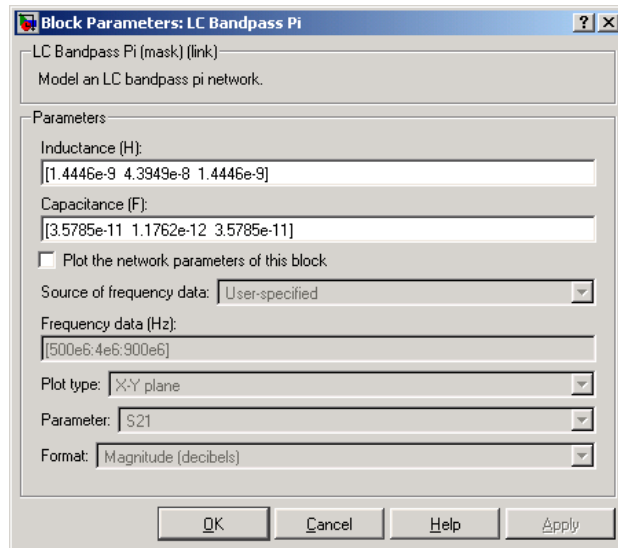
The Random Source block (Signal Processing Blockset) produces frame-based output at 512 samples per frame. Its **Sample time** parameter is set to 1.0e-5. This sample time must match the sample time for the physical part of the model, which you provide in the Input Port block diagram.

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



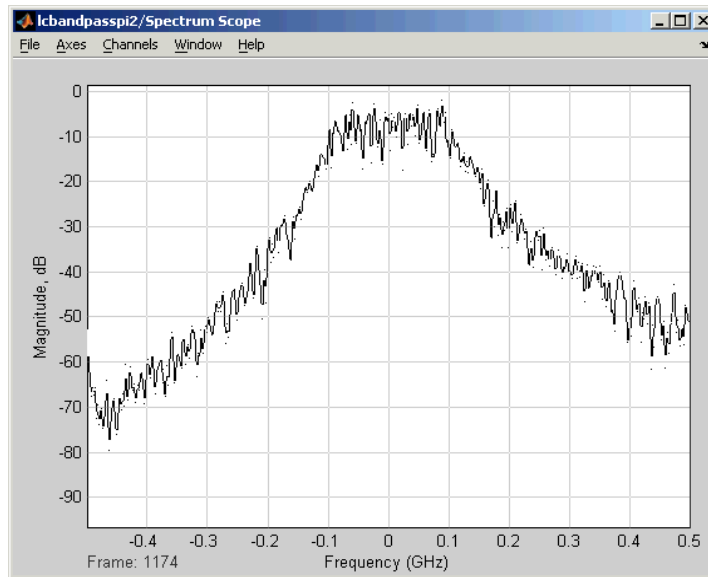
The LC Bandpass Pi block provides the inductances for three inductors, in order from source to load, [1.4446e-9, 4.3949e-8, 1.4446e-9]. Similarly, it provides the capacitances for three capacitors [3.5785e-11, 1.1762e-12, 3.5785e-11].

# LC Bandpass Pi



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.

The Axis Properties of the Spectrum Scope block have been adjusted to show the frequencies above and below the carrier.



## 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

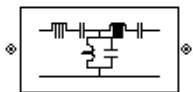
General Passive Network, LC Bandpass Tee, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Pi, LC Highpass Tee, LC Lowpass Pi, LC Lowpass Tee

# LC Bandpass Tee

**Purpose** Model an LC bandpass tee network

**Library** Ladder Filters sublibrary of the Physical library

## Description

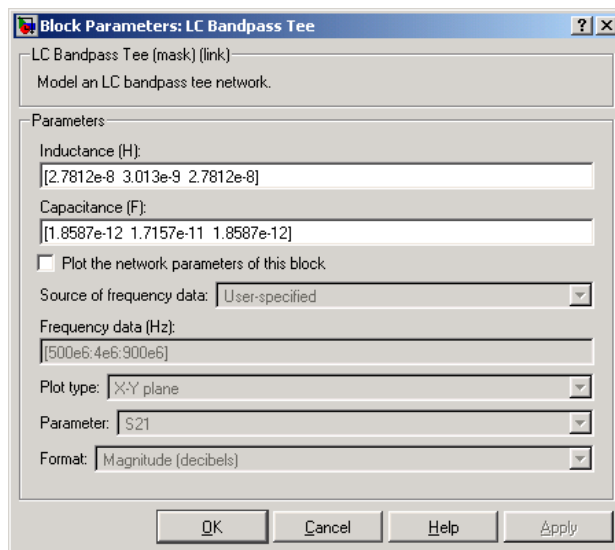


The LC Bandpass Tee block models the LC bandpass tee network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor pair in the network, the block first calculates the ABCD-parameters for each frequency contained in the vector of simulation 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. See the Output Port block for information about the simulation frequencies.

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

## 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 “Plotting Network Parameters” on page 3-1.

---

## **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**

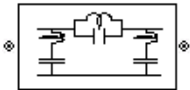
General Passive Network, LC Bandpass Pi, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Pi, LC Highpass Tee, LC Lowpass Pi, LC Lowpass Tee

# LC Bandstop Pi

**Purpose** Model an LC bandstop pi network

**Library** Ladder Filters sublibrary of the Physical library

## Description

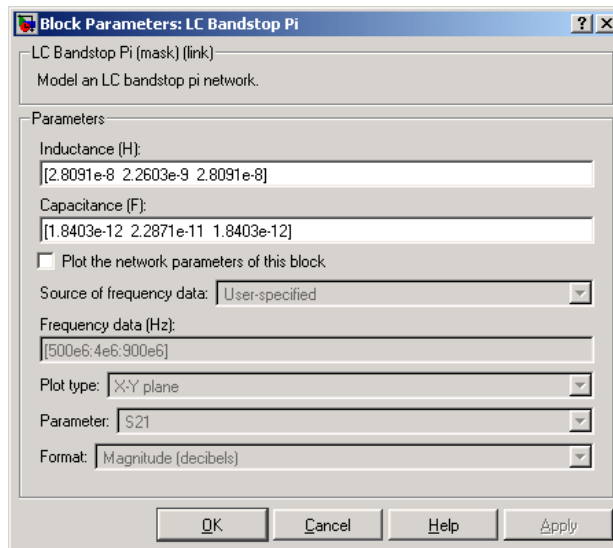


The LC Bandstop Pi block models the LC bandstop pi network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor pair in the network, the block first calculates the ABCD-parameters for each frequency contained in the vector of simulation 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. See the Output Port block for information about the simulation frequencies.

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

## 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 “Plotting Network Parameters” on page 3-1.

---

**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**

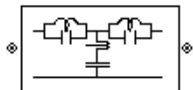
General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Tee, LC Highpass Pi, LC Highpass Tee, LC Lowpass Pi, LC Lowpass Tee

# LC Bandstop Tee

**Purpose** Model an LC bandstop tee network

**Library** Ladder Filters sublibrary of the Physical library

## Description

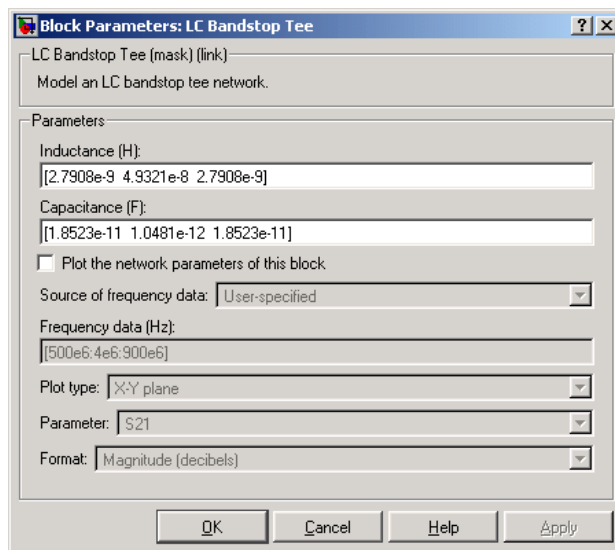


The LC Bandstop Tee block models the LC bandstop tee network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor pair in the network, the block first calculates the ABCD-parameters for each frequency contained in the vector of simulation 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. See the Output Port block for information about the simulation frequencies.

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

## 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 “Plotting Network Parameters” on page 3-1.

---

**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**

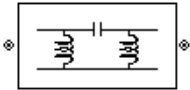
General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Pi, LC Highpass Pi, LC Highpass Tee, LC Lowpass Pi, LC Lowpass Tee

# LC Highpass Pi

**Purpose** Model an LC highpass pi network

**Library** Ladder Filters sublibrary of the Physical library

## Description

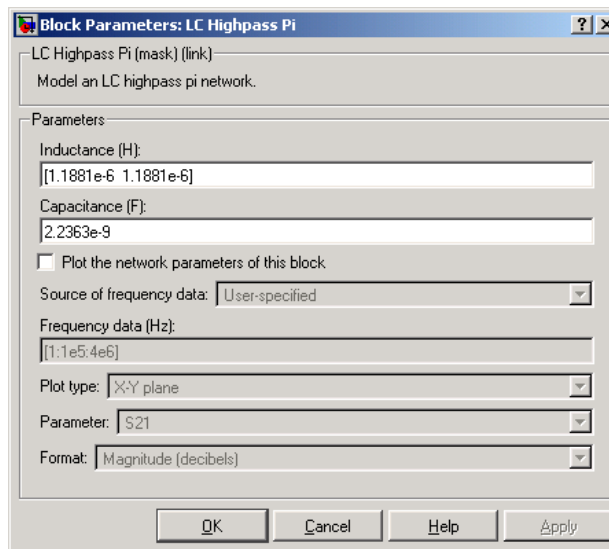


The LC Highpass Pi block models the LC highpass pi network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor in the network, the block first calculates the ABCD-parameters for each frequency contained in the vector of simulation 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. See the Output Port block for information about the simulation frequencies.

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

## 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 “Plotting Network Parameters” on page 3-1.

---

## **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**

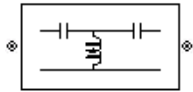
General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Tee, LC Lowpass Pi, LC Lowpass Tee

# LC Highpass Tee

**Purpose** Model an LC highpass tee network

**Library** Ladder Filters sublibrary of the Physical library

## Description

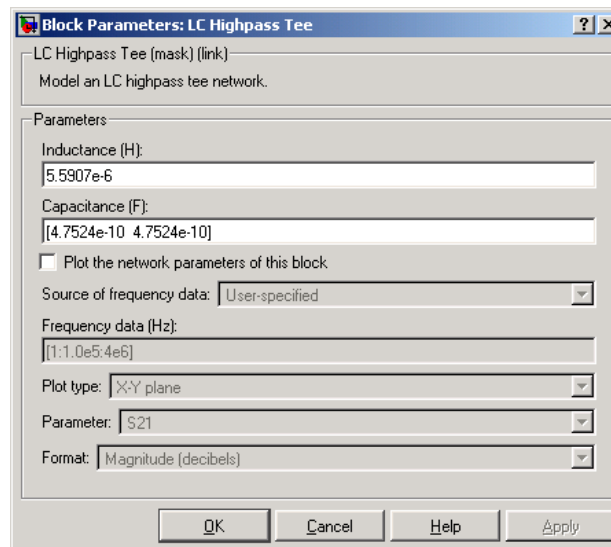


The LC Highpass Tee block models the LC highpass tee network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor in the network, the block first calculates the ABCD-parameters for each frequency contained in the vector of simulation 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. See the Output Port block for information about the simulation frequencies.

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

## 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 “Plotting Network Parameters” on page 3-1.

---

**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**

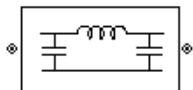
General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Pi, LC Lowpass Pi, LC Lowpass Tee

# LC Lowpass Pi

**Purpose** Model an LC lowpass pi network

**Library** Ladder Filters sublibrary of the Physical library

## Description

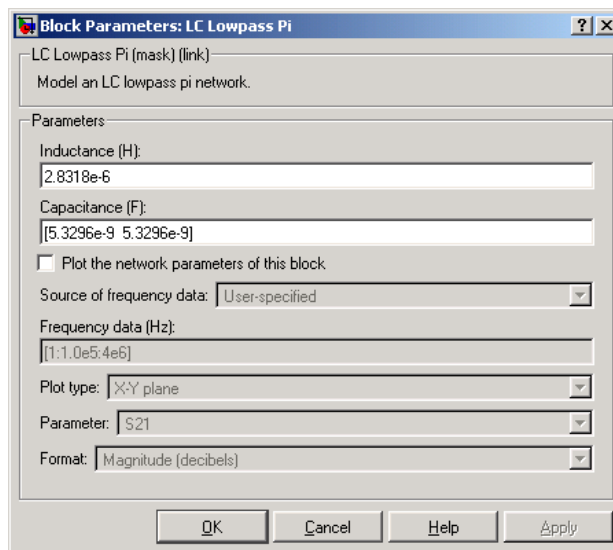


The LC Lowpass Pi block models the LC lowpass pi network described in the block dialog box, in terms of its frequency-dependent S-parameters.

For each inductor and capacitor in the network, the block first calculates the ABCD-parameters for each frequency contained in the vector of simulation 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. See the Output Port block for information about the simulation frequencies.

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

## 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 “Plotting Network Parameters” on page 3-1.

---

**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**

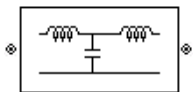
General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Pi, LC Highpass Tee, LC Lowpass Tee

# LC Lowpass Tee

**Purpose** Model an LC lowpass tee network

**Library** Ladders Filters sublibrary of the Physical library

## Description

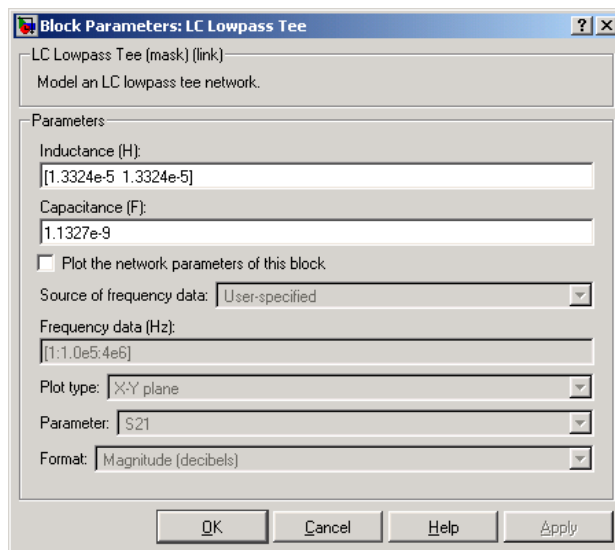


The LC Lowpass Tee block models the LC lowpass tee network described in the block dialog box in terms of its frequency-dependent S-parameters.

For each inductor and capacitor in the network, the block first calculates the ABCD-parameters for each frequency contained in the vector of simulation 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. See the Output Port block for information about the simulation frequencies.

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

## 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 “Plotting Network Parameters” on page 3-1.

---

**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**

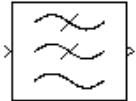
General Passive Network, LC Bandpass Pi, LC Bandpass Tee, LC Bandstop Pi, LC Bandstop Tee, LC Highpass Pi, LC Highpass Tee, LC Lowpass Pi

# Lowpass RF Filter

**Purpose** Standard lowpass RF filters in baseband equivalent complex form

**Library** Mathematical

## Description



The Lowpass RF Filter block enables you to 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.

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

Design Method	Filter Design Parameters
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`.

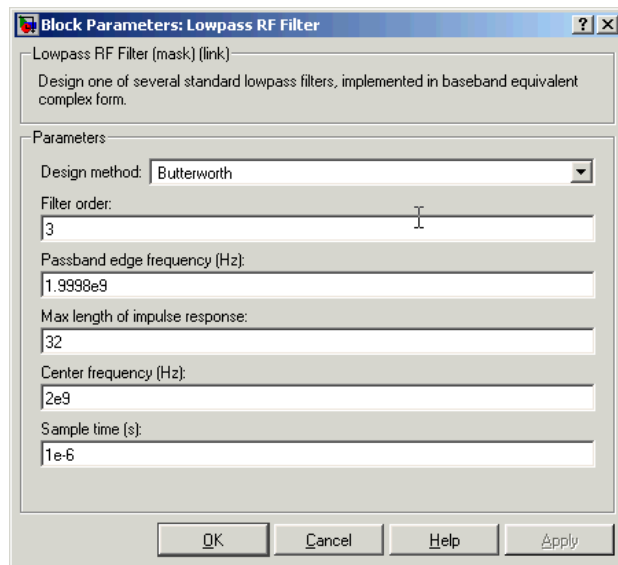
---

**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 with 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 is produced.

---

# Lowpass RF Filter

## Dialog Box



The parameters displayed in the dialog box vary for different design methods. Only some of the parameters listed below 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.

**Max length of impulse response**

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

**Center frequency (Hz)**

Center of the simulation frequencies.

**Sample time (s)**

Time interval between consecutive samples of the input signal.

**See Also**

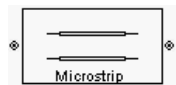
Amplifier, Bandpass RF Filter, Bandstop RF Filter, Highpass RF Filter, Mixer  
buttap, cheb1ap, cheb2ap, ellipap, besse1ap (Signal Processing Toolbox)

# Microstrip Transmission Line

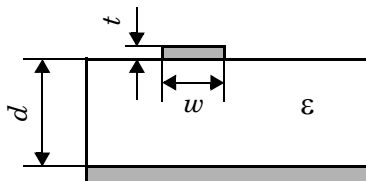
**Purpose** Model a microstrip transmission line

**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 ( $\epsilon$ ).



The block enables you to 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  $\alpha_a$  is the attenuation coefficient and  $\beta$  is the wave number. The attenuation coefficient  $\alpha_a$  is related to the loss,  $\alpha$ , by

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



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

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

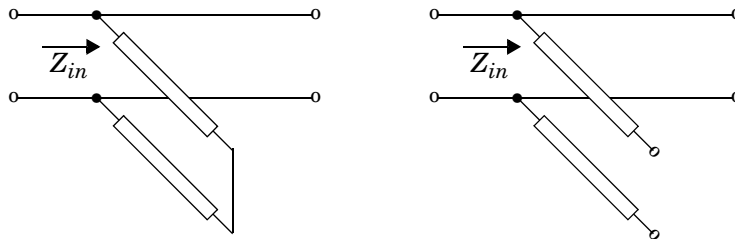
where  $V_p = c/\sqrt{\epsilon_{\text{eff}}}$ , and  $\epsilon_{\text{eff}}$  is the frequency dependent effective dielectric constant.  $f$  is the vector of simulation frequencies for the specified parameters. See the Output Port block for information about determining the simulation frequencies. The 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 for each frequency contained in the vector of simulation 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 2-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.



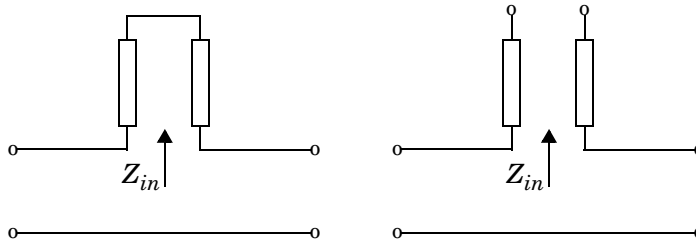
$Z_{in}$  is the input impedance of the shunt circuit. The ABCD-parameters for the shunt stub are calculated as

# Microstrip Transmission Line

---

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

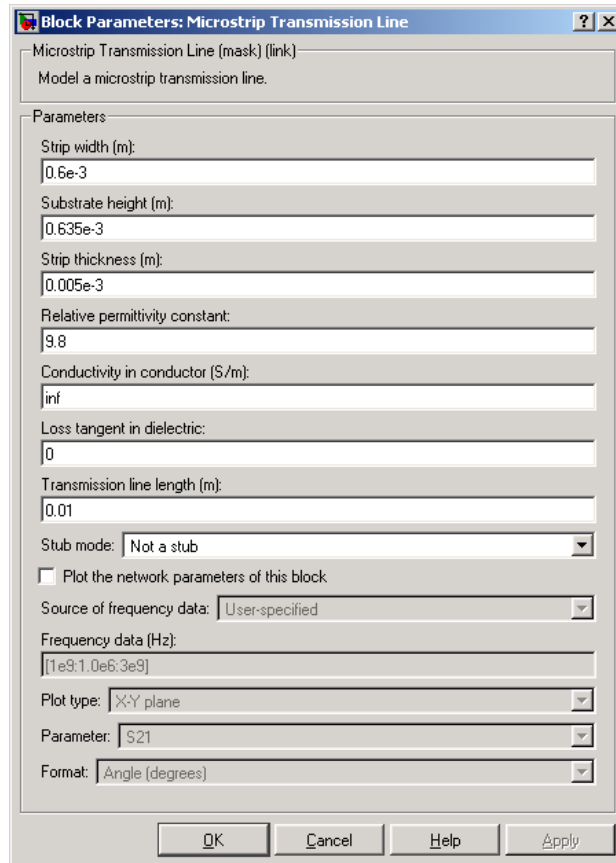
**Series ABCD-Parameters.** When you set the **Stub mode** parameter in the mask dialog box to **Series**, the 2-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



$Z_{in}$  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



### Strip width (m)

Width of the microstrip transmission line.

### Substrate height (m)

Thickness of the dielectric on which the microstrip resides.

### Strip thickness (m)

Physical thickness of the microstrip.

# Microstrip Transmission Line

---

## Relative permittivity constant

Relative permittivity of the dielectric expressed as the ratio of the permittivity of the dielectric to permittivity in free space  $\epsilon_0$ .

## 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.

---

**Note** For information about plotting, see “Plotting Network Parameters” on page 3-1.

---

## References

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

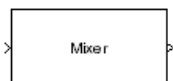
## See Also

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

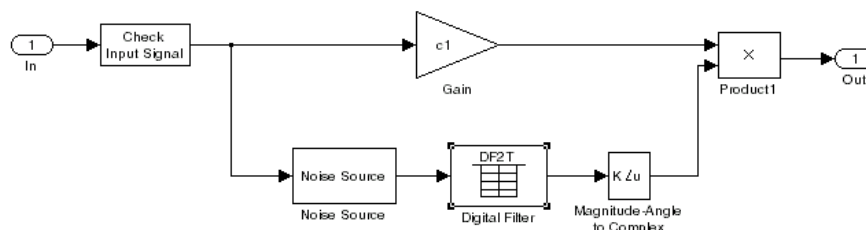
**Description**



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. 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 on 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.

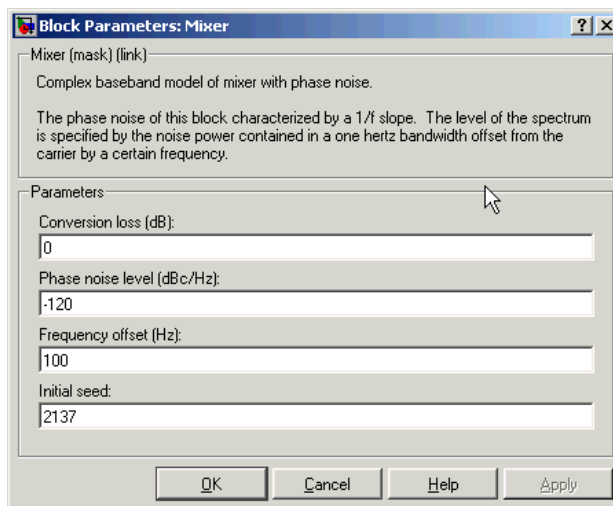
---

**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 with 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 is produced.

---

# Mixer

## 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.

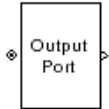
## **See Also**

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

**Purpose** Connection block from RF physical blocks to the 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 travelling 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 connecting mathematical and physical parts of a model, see “Building an RF Model” on page 2-1.

---

**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 with 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 is produced.

---

## 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 Combines the S-parameters from the individual physical blocks and converts the resulting S-parameters to Y-parameters. See the RF Toolbox function `s2y` for conversion details.

# Output Port

---

- 2** Determines the simulation frequencies  $f$  as an  $N$ -element vector. The simulation frequencies are a function of the center frequency  $f_c$ , the sample time  $t_s$ , and the maximum length of the impulse response  $N$ , 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

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

- 3** Calculates the passband transfer function for the frequency range.

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

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

$$H(f) = \frac{-Y_{21}(f)}{(1 + Z_S(f) \cdot Y_{11}(f))(Y_{22}(f) + Z_L(f)^{-1}) - (Z_S(f) \cdot Y_{12}(f) \cdot Y_{21}(f))}$$

where

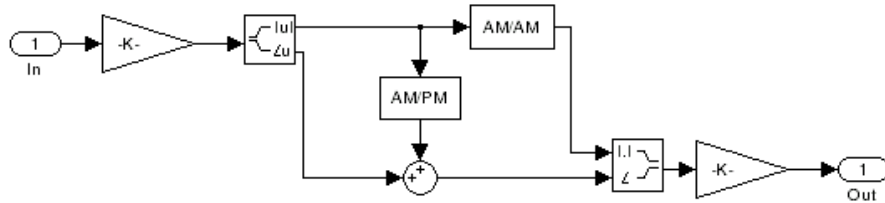
- $Z_S$  is the source impedance.
  - $Z_L$  is the load impedance.
  - $Y_{ij}$  are the Y-parameters of a two-port network.
- 4** Translates the passband transfer function to baseband as  $H(f-f_c)$ , where  $f_c$  is the specified center frequency.
- 5** Obtains the baseband equivalent impulse response by calculating the inverse FFT of the baseband transfer function. For reasons of efficiency, the block calculates the IFFT using the next power of 2 greater than the specified maximum impulse response length. It then truncates the impulse response to a length less than or equal to the max 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 following equations

$$AM_{\text{out}} = \sqrt{2 \cdot R_l \cdot P_{\text{out}}}$$

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

$$AM_{\text{in}} = \sqrt{4 \cdot R_s \cdot P_{\text{in}}}$$

where  $AM_{\text{in}}$  is the AM of the input voltage,  $AM_{\text{out}}$  and  $PM_{\text{out}}$  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),  $P_{\text{in}}$  is the input power,  $P_{\text{out}}$  is the output power, and Phase is the phase shift between the input and output voltage.

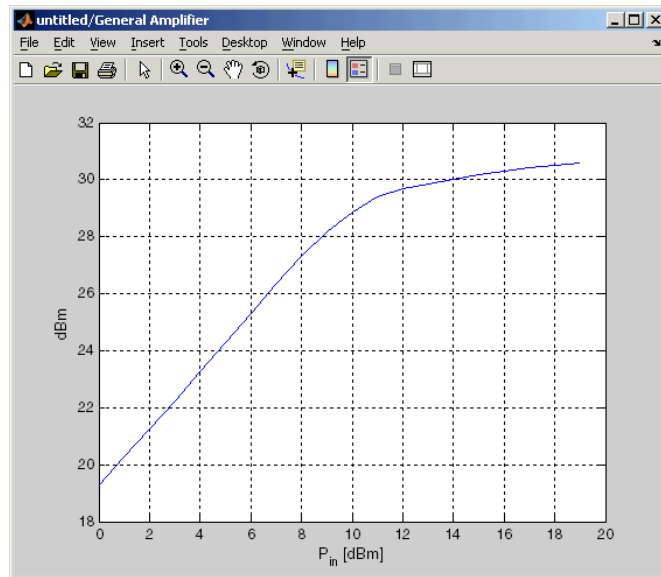
---

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

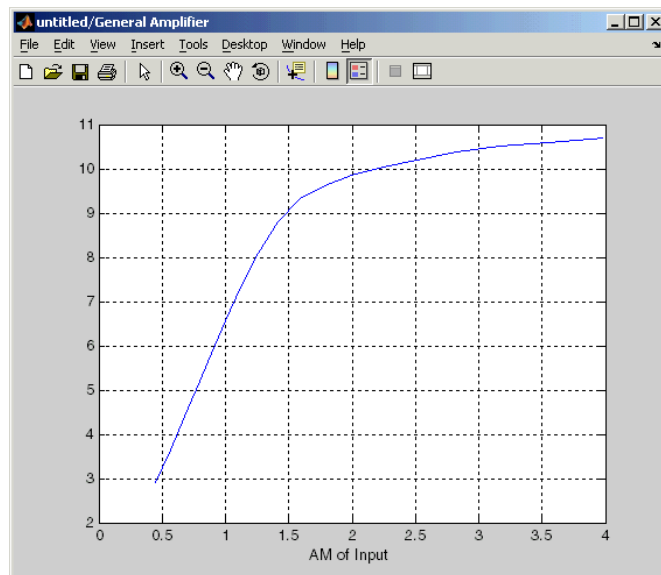
---

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

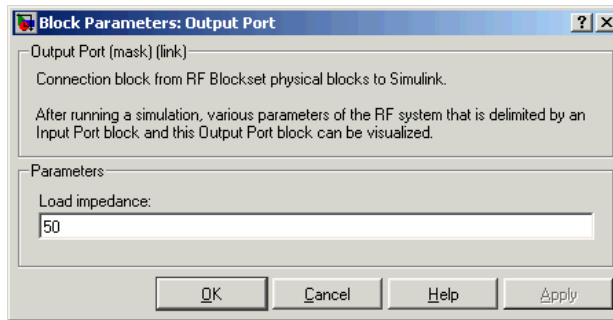
# Output Port



This figure shows the extracted AM/AM nonlinear conversion.



## Dialog Box

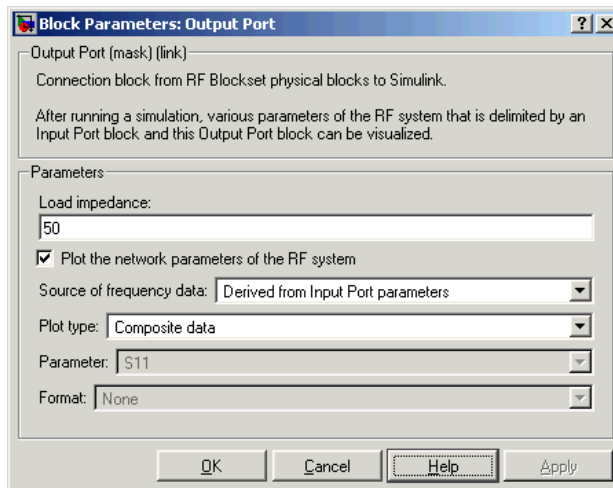


### Load impedance

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

### Plot the network 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. See “Plotting Network Parameters” on page 3-1 for more information about plotting.



## See Also

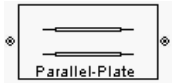
Input Port; s2y (RF Toolbox)

# Parallel-Plate Transmission Line

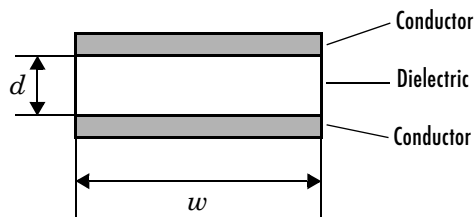
**Purpose** Model parallel-plate transmission line

**Library** Transmission Lines sublibrary of the Physical library

## Description



The Parallel-Plate Line block models the parallel-plate transmission line described in the block dialog 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 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 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

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

where

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

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

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

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

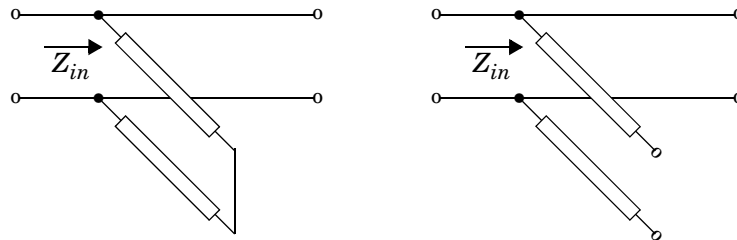
In these equations,  $\sigma_{\text{cond}}$  is the conductivity in the conductor and  $\sigma_{\text{diel}}$  is the conductivity in the dielectric.  $\mu$  is the permeability of the dielectric,  $\varepsilon$  is its permittivity, and skin depth  $\delta$  is calculated as  $1/\sqrt{\pi f \mu \sigma_{\text{cond}}}$ .  $f$  is the vector of simulation frequencies for the specified parameters. See the Output Port block for information about determining the simulation 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 for each frequency contained in the vector of simulation 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 2-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.



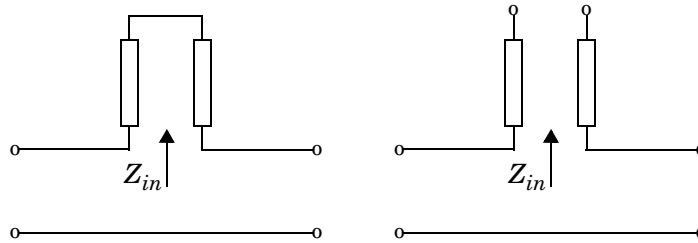
$Z_{in}$  is the input impedance of the shunt circuit. The ABCD-parameters for the shunt stub are calculated as

# Parallel-Plate Transmission Line

---

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

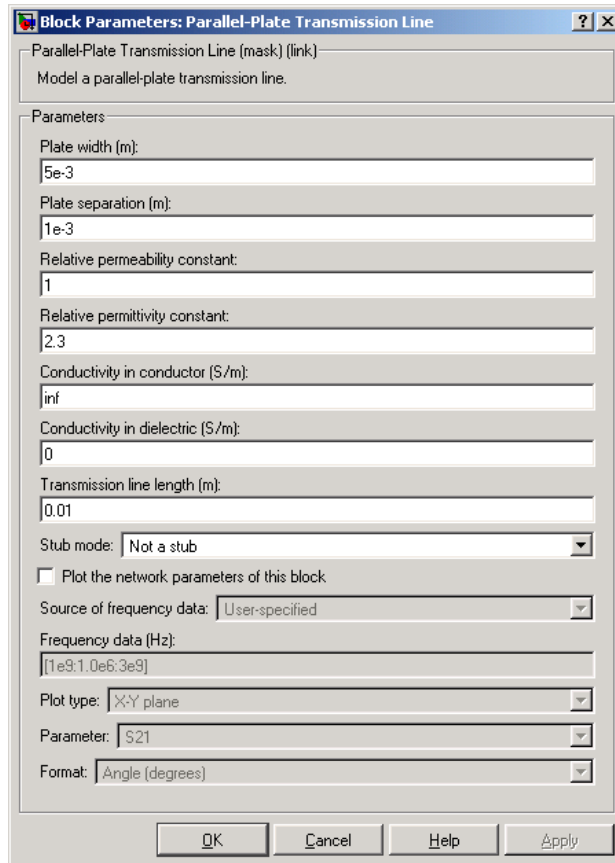
**Series ABCD-Parameters.** When you set the **Stub mode** parameter in the mask dialog box to **Series**, the 2-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



$Z_{in}$  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



### 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  $\mu_0$ .

# Parallel-Plate Transmission Line

---

## Relative permittivity constant

Relative permittivity of the dielectric expressed as the ratio of the permittivity of the dielectric to permittivity in free space  $\epsilon_0$ .

## 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.

---

**Note** For information about plotting, see “Plotting Network Parameters” on page 3-1.

---

## 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



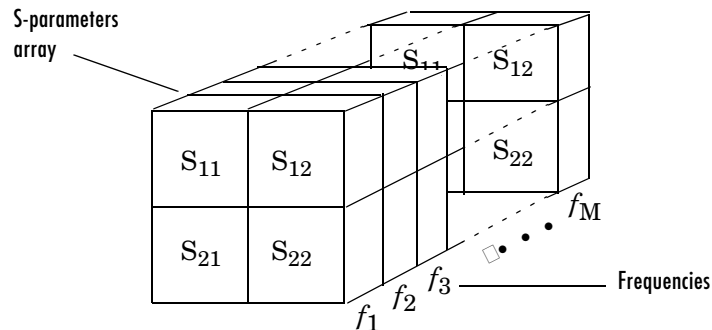
**Purpose** Model a nonlinear amplifier using its S-parameters

**Library** Amplifiers sublibrary of the Physical library

**Description** The S-Parameters Amplifier block models the nonlinear amplifier described in the block dialog, 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.



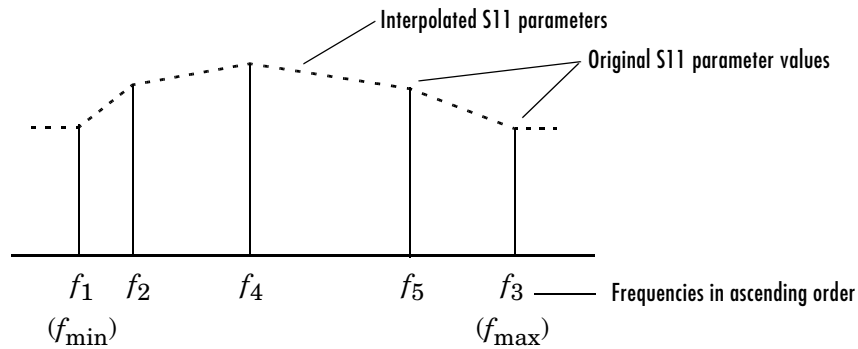
In the **S-parameters** field of the block dialog, provide the S-parameters for each of  $M$  frequencies as a 2-by-2-by- $M$  array. In the **Frequency** field, specify the frequencies for the S-parameters as an  $M$ -element vector. The elements of the frequencies vector must be in the same order as 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 block orders the S-parameters according to the ascending order of their frequencies, and then interpolates, using the MATLAB `interp1` function, to determine the S-parameters at the simulation frequencies. The curve in the following diagram illustrates the result of interpolating the S11 parameters at five different frequencies.

# S-Parameters Amplifier

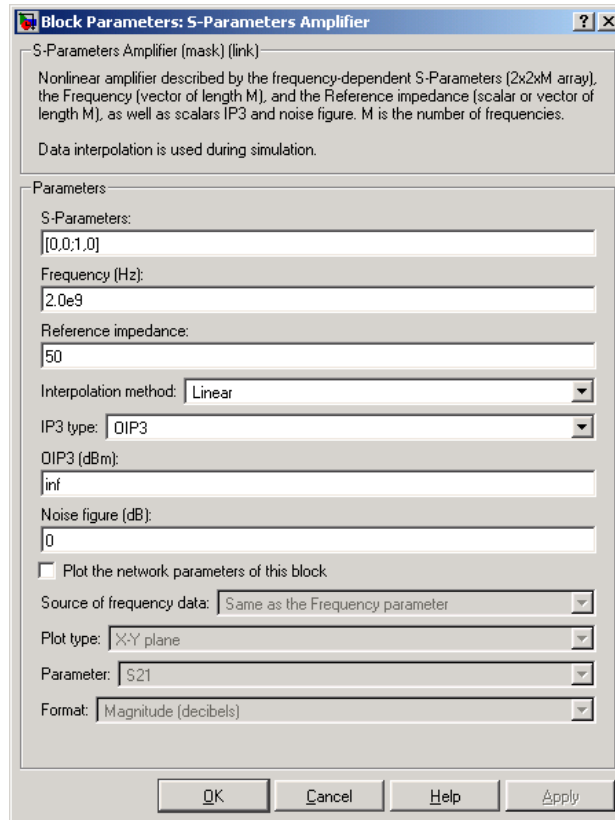
---



The **Interpolation** field in the block dialog 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 diagram above, the S-Parameters Amplifier block uses the S-parameters at  $f_{\min}$  for all simulation frequencies smaller than  $f_{\min}$ . It uses the S-parameters at  $f_{\max}$  for all simulation frequencies greater than  $f_{\max}$ . In both cases, the results may not be accurate.

## Dialog Box



### 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.

# S-Parameters Amplifier

---

## 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 “Plotting Network Parameters” on page 3-1.

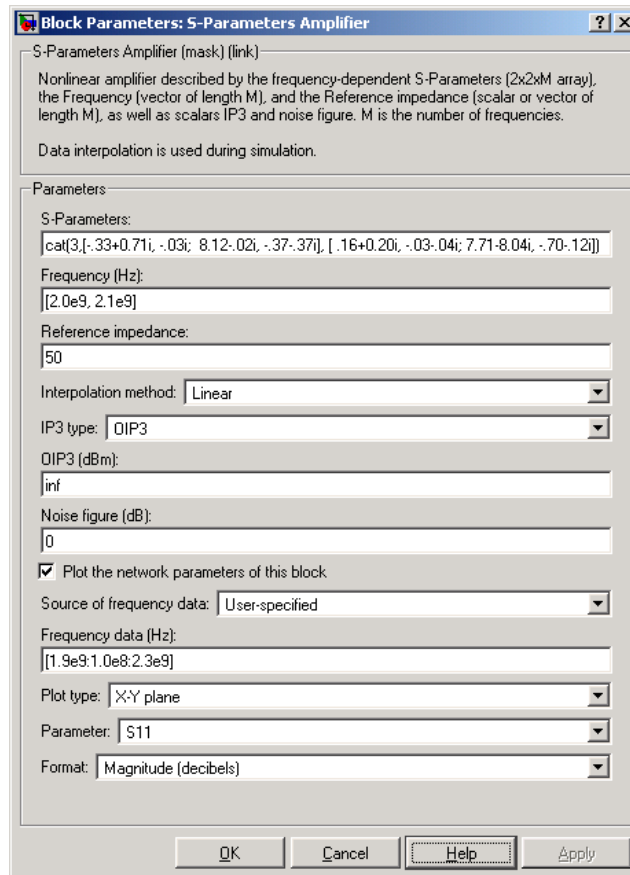
---

## Examples

### Plotting Parameters with the S-Parameters Amplifier Block

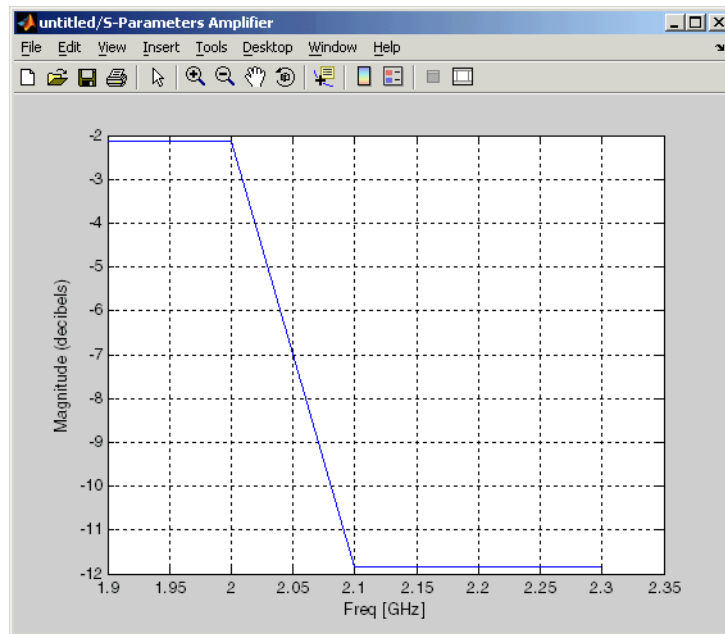
The following example specifies S-parameters  $[-.33+.71i, -.03i; 8.12-.02i, -.37-.37i]$  and  $[0.16+.20i, -.03-.04i; 7.71-8.04i, -.70-.12i]$  at frequencies 2.0 GHz and 2.1 GHz respectively, with a reference impedance of 50 ohms. It uses the MATLAB `cat` function to create the 2-by-2-by-2 S-parameters array.

```
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 request an X-Y Plane plot of the S11 parameters in the frequency range 1.8 GHz to 2.3 GHz.

# S-Parameters Amplifier



## See Also

General Amplifier, Y-Parameters Amplifier, Z-Parameters Amplifier  
interp1 (MATLAB)

**Purpose** Model a mixer using its S-parameters

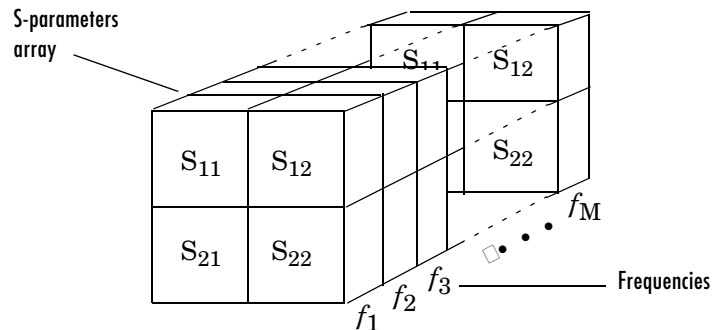
**Library** Mixer sublibrary of the Physical library

## Description



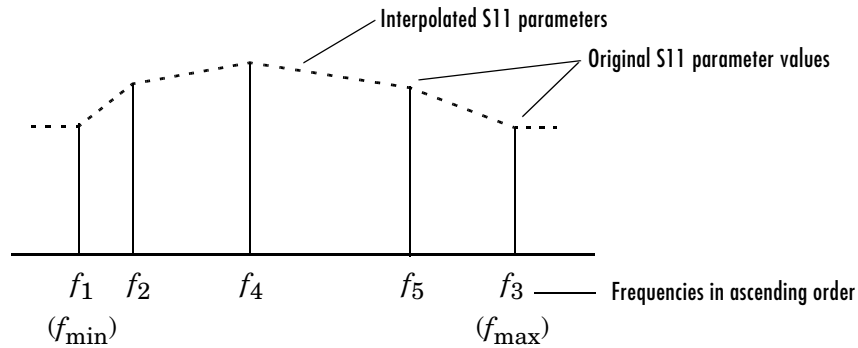
The S-Parameters Mixer block models the nonlinear mixer described in the block dialog, in terms of its frequency-dependent S-parameters, their frequencies, and the reference impedance of the S-parameters. The block also takes phase noise into account.

In the **S-parameters** field of the block dialog, 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 block orders the S-parameters according to the ascending order of their frequencies, and then interpolates, using the MATLAB `interp1` function, to determine the S-parameters at the simulation frequencies. The curve in the following diagram illustrates the result of interpolating the S11 parameters at five different frequencies.

# S-Parameters Mixer



The **Interpolation** field in the block dialog 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 diagram above, the S-Parameters Mixer block uses the S-parameters at  $f_{\min}$  for all simulation frequencies smaller than  $f_{\min}$ . It uses the S-parameters at  $f_{\max}$  for all simulation frequencies greater than  $f_{\max}$ . In both cases, the results may not be accurate.



## Dialog Box

**Block Parameters: S-Parameters Mixer** [?] [X]

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 IP3, 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 network 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.

## 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,  $f_{out} = f_{in} - f_{lo}$  .  
If you choose Upconverter,  $f_{out} = f_{in} + f_{lo}$  .

## 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 network parameters, see “Plotting Network Parameters” on page 3-1.

---

## See Also

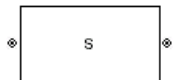
General Mixer, Y-Parameters Mixer, Z-Parameters Mixer

# S-Parameters Passive Network

**Purpose** Model a passive network using its S-parameters

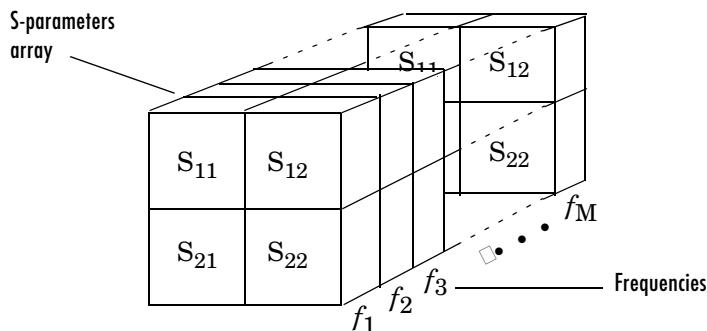
**Library** Black Box Elements sublibrary of the Physical library

## Description



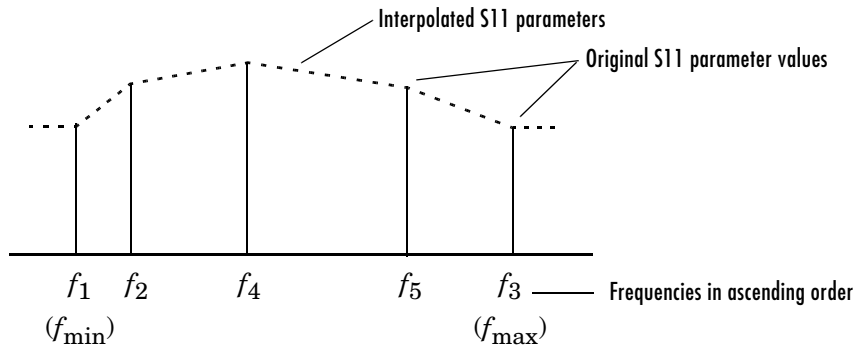
The S-Parameters Passive Network block models the two-port passive network described in the block dialog, in terms of its S-parameters, their frequencies, and the reference impedance of the S-parameters.

In the **S-Parameters** field of the block dialog, 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 block orders the S-parameters according to the ascending order of their frequencies, and then interpolates, using the MATLAB `interp1` function, to determine the S-parameters at the simulation frequencies. The curve in the following diagram illustrates the result of interpolating the S11 parameters at five different frequencies.

# S-Parameters Passive Network

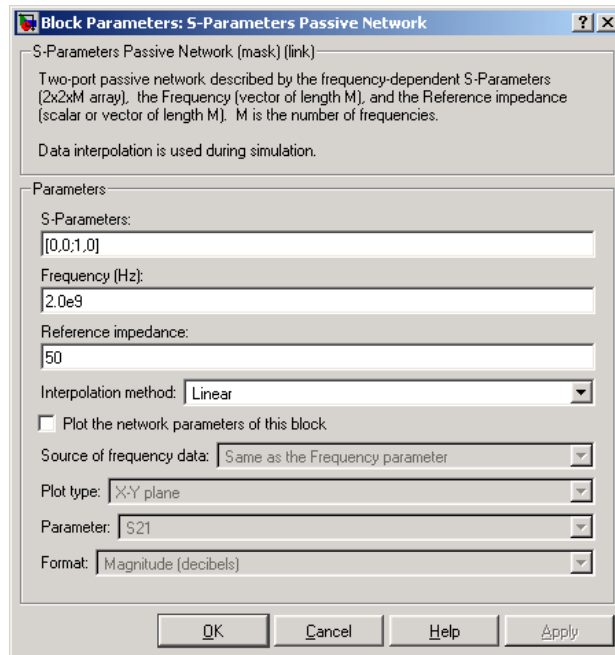


The **Interpolation** field in the block dialog enables you to specify the interpolation method as Cubic, Linear (default), or Spline. For more information about these methods, see “One-Dimensional Interpolation” and the `interp1` reference page in the MATLAB documentation.

As shown in the diagram above, the S-Parameters Passive Network block uses the S-parameters at  $f_{\min}$  for all simulation frequencies smaller than  $f_{\min}$ . It uses the S-parameters at  $f_{\max}$  for all simulation frequencies greater than  $f_{\max}$ . In both cases, the results may not be accurate.

# S-Parameters Passive Network

## Dialog Box



### S-Parameters

S-parameters for a 2-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.

### 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 “Plotting Network Parameters” on page 3-1.

---

## Examples

### Plotting Parameters with the S-Parameters Passive Network Block

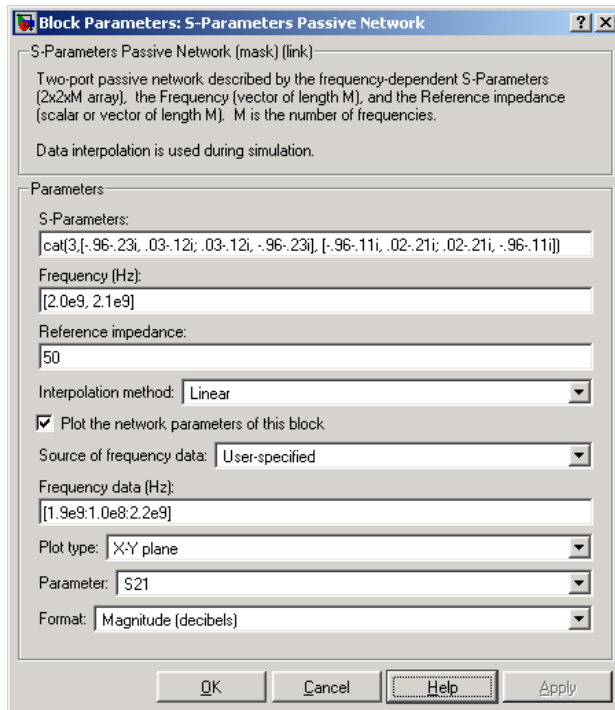
The following example specifies S-parameters  $[-.96-.23i, .03-.12i; .03-.12i, -.96-.23i]$  and  $[-.96-.11i, .02-.21i; .02-.21i, -.96-.11i]$  at frequencies 2.0 GHz and 2.1 GHz respectively. It uses the MATLAB `cat` function to create the 2-by-2-by-2 S-parameters array

```
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. This statement would produce 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)
```

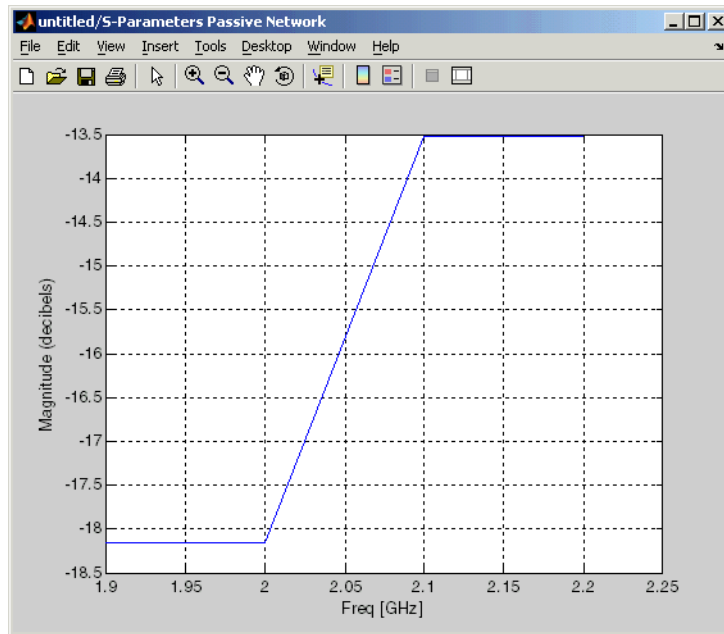
# S-Parameters Passive Network



The plot parameters in the dialog request an X-Y Plane plot of the S21 S-parameter magnitudes, in decibels, in the frequency range 1.9 GHz to 2.2 GHz.



# S-Parameters Passive Network



## See Also

General Circuit Element, General Passive Network, Y-Parameters Passive Network, Z-Parameters Passive Network

`interp1` (MATLAB)

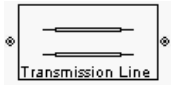
# Transmission Line

---

**Purpose** Model a 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 in terms of its physical parameters. The transmission line, which can be lossy or lossless, is treated as a 2-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  $\alpha_a$  is the attenuation coefficient and  $\beta$  is the wave number. The attenuation coefficient  $\alpha_a$  is related to the loss,  $\alpha$ , by

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

and the wave number  $\beta$  is related to the phase velocity,  $V_p$ , by

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

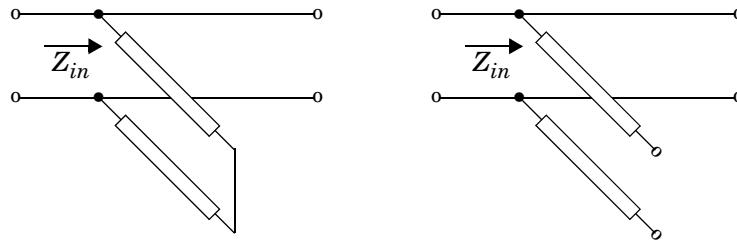
where  $f$  is the vector of simulation frequencies for the specified parameters. See the Output Port block for information about determining the simulation frequencies. The phase velocity  $V_p$  is also known as the wave propagation velocity.

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 for each frequency contained in the vector of simulation 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 2-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

$$A = 1$$

$$B = 0$$

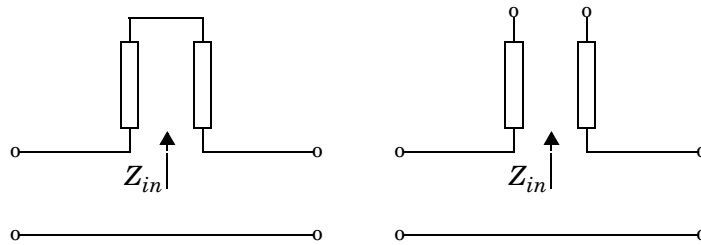
$$C = 1/Z_{in}$$

$$D = 1$$

**Series ABCD-Parameters.** When you set the **Stub mode** parameter in the mask dialog box to **Series**, the 2-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.

# Transmission Line

---



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

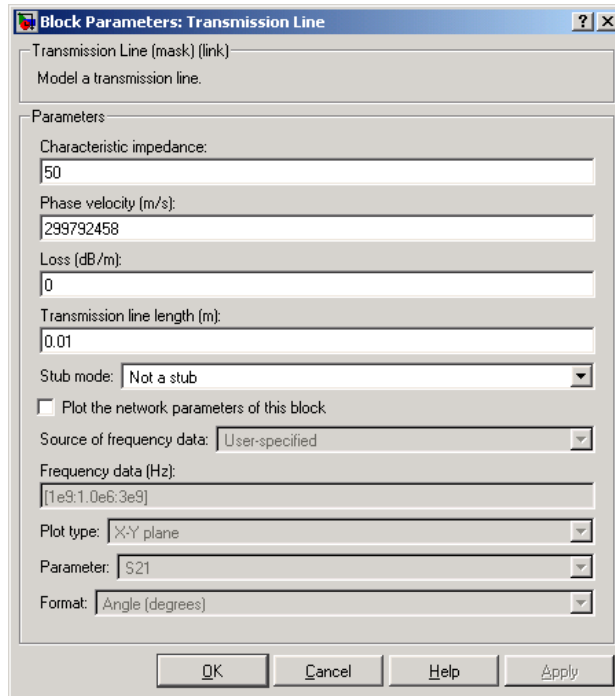
$$A = 1$$

$$B = Z_{in}$$

$$C = 0$$

$$D = 1$$

## Dialog Box



### 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.

# Transmission Line

---

## Termination of stub

Stub termination for stub modes Shunt and Series. Choices are Open or Short.

---

**Note** For information about plotting, see “Plotting Network Parameters” on page 3-1.

---

## 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

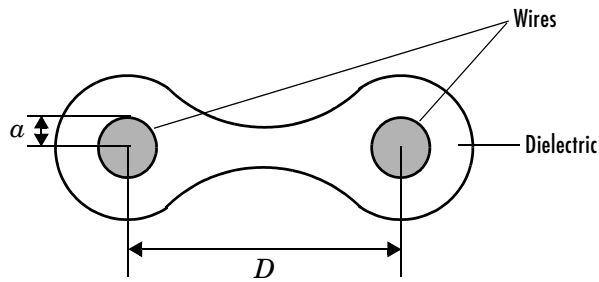
**Purpose** Model a two-wire transmission line

**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 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

# Two-Wire Transmission Line

---

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

where

$$R = \frac{1}{\pi a \sigma_{\text{cond}} \delta}$$

$$L = \frac{\mu}{\pi} \text{acosh}\left(\frac{D}{2a}\right)$$

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

$$C = \frac{\pi \varepsilon}{\text{acosh}(D/(2a))}$$

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

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

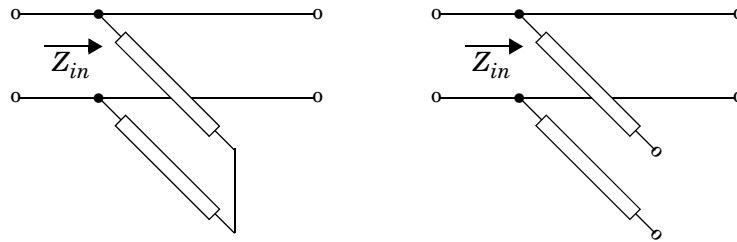
## Shunt and Series Stubs

If you model the transmission line as a shunt or series stub, the Two-Wire Transmission Line block first calculates the ABCD-parameters for each frequency contained in the vector of simulation 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 2-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

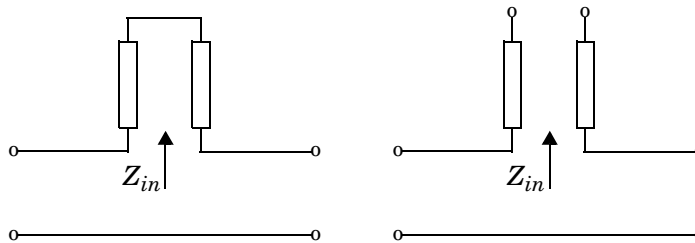
$$A = 1$$

$$B = 0$$

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

$$D = 1$$

**Series ABCD-Parameters.** When you set the **Stub mode** parameter in the mask dialog box to **Series**, the 2-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 network 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.

### 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  $\mu_0$ .

**Relative permittivity constant**

Relative permittivity of the dielectric expressed as the ratio of the permittivity of the dielectric to permittivity in free space  $\epsilon_0$ .

**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.

---

**Note** For information about plotting, see “Plotting Network Parameters” on page 3-1.

---

**References**

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

**See Also**

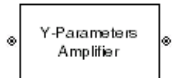
Coaxial Transmission Line, Coplanar Waveguide Transmission Line, General Passive Network, Transmission Line, Microstrip Transmission Line, Parallel-Plate Transmission Line

# Y-Parameters Amplifier

**Purpose** Model a nonlinear amplifier using its Y-parameters

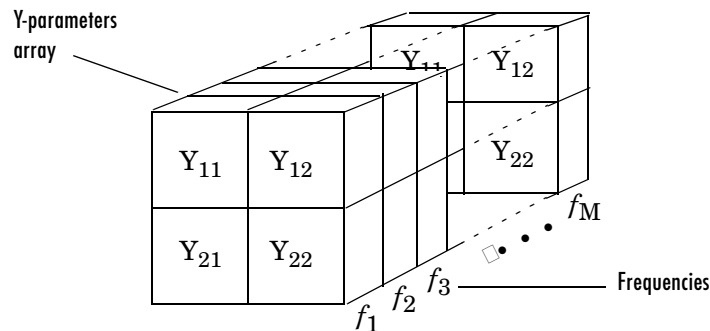
**Library** Amplifiers sublibrary of the Physical library

## Description

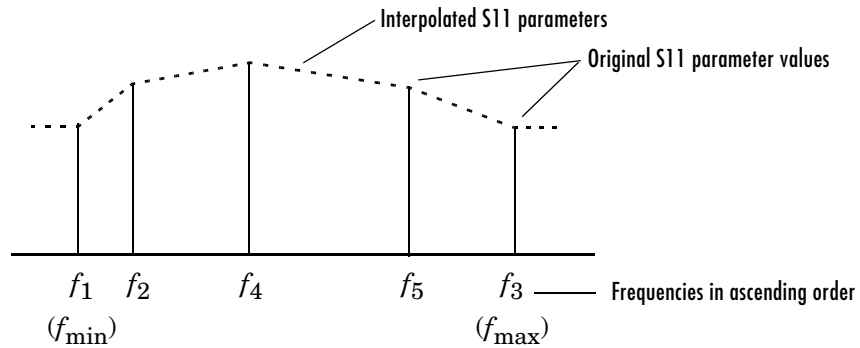


The Y-Parameters Amplifier block models the nonlinear amplifier described in the block dialog, in terms of its frequency-dependent Y-parameters and their frequencies. The block also takes into account the IP3 value and the noise figure.

In the **Y-Parameters** field of the block dialog, 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 block orders the Y-parameters according to the ascending order of their frequencies, and then uses the RF Toolbox `y2s` function to convert the Y-parameters to S-parameters. It interpolates the S-parameters using the MATLAB `interp1` function to determine the S-parameters at the simulation frequencies. The curve in the following diagram illustrates the result of interpolating the S11 parameters at five different frequencies.

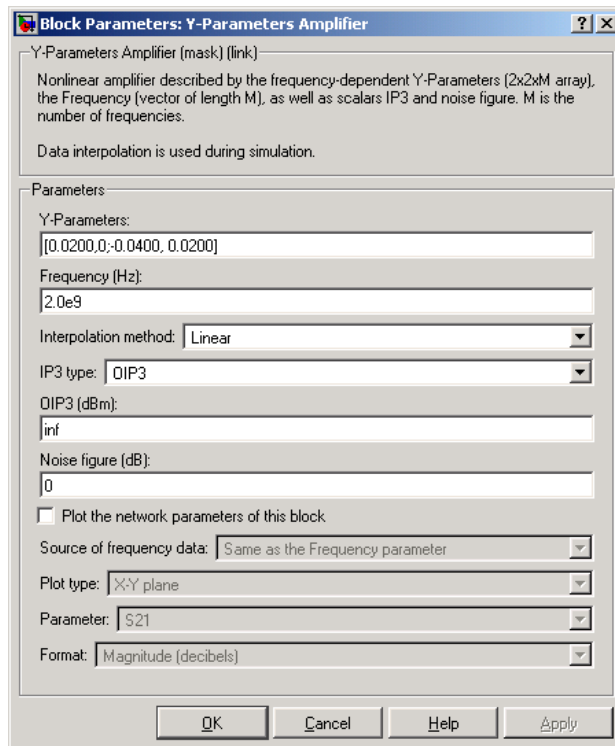


The **Interpolation** field in the block dialog enables you to specify the interpolation method as Cubic, Linear (default), or Spline. For more information about these methods, see “One-Dimensional Interpolation” and the `interp1` reference page in the MATLAB documentation.

As shown in the diagram above, the Y-Parameters Amplifier block uses the S-parameters at  $f_{\min}$  for all simulation frequencies smaller than  $f_{\min}$ . It uses the S-parameters at  $f_{\max}$  for all simulation frequencies greater than  $f_{\max}$ . In both cases, the results may not be accurate.

# 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 “Plotting Network Parameters” on page 3-1.

---

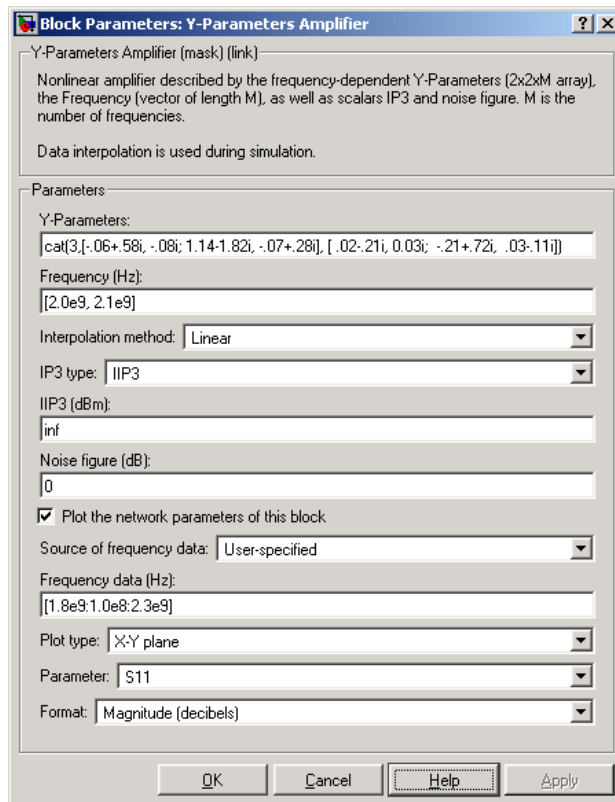
## Examples

### 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

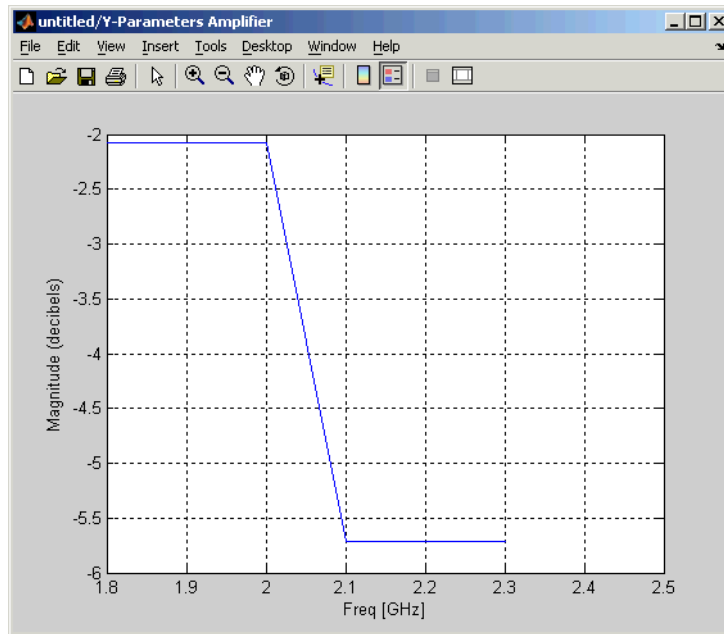
```
cat(3,[-.06+.58i, -.08i; 1.14-1.82i, -.07+.28i],...  
      [.02-.21i, 0.03i; -.21+.72i, .03-.11i])
```

# Y-Parameters Amplifier



The plot parameters in the dialog request an X-Y plane plot of the S11 parameters in the frequency range 1.8 GHz to 2.3 GHz.





## See Also

General Amplifier, S-Parameters Amplifier, Z-Parameters Amplifier

y2s (RF Toolbox)

interp1 (MATLAB)

# Y-Parameters Mixer

**Purpose** Model a mixer using its Y-parameters

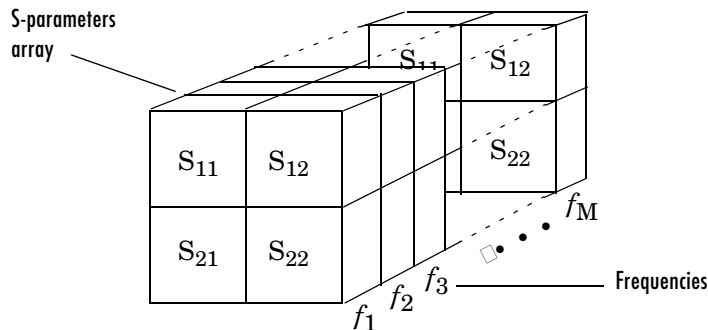
**Library** Mixer sublibrary of the Physical library

## Description

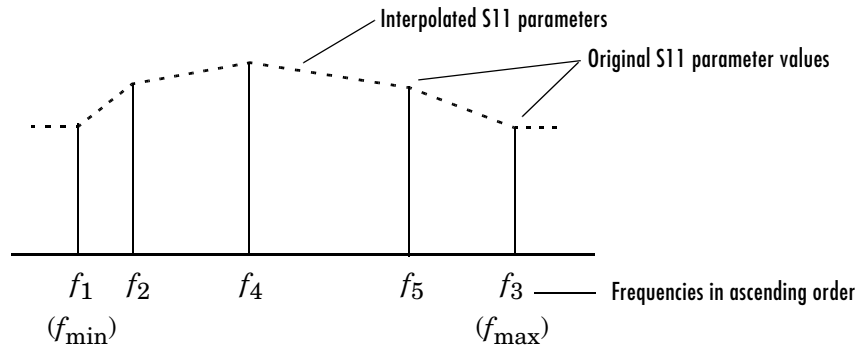


The Y-Parameters Mixer block models the nonlinear mixer described in the block dialog, in terms of its frequency-dependent Y-parameters, their frequencies, and the reference impedance of the Y-parameters. The block also takes phase noise into account.

In the **Y-parameters** field of the block dialog, 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 S-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 block orders the Y-parameters according to the ascending order of their frequencies, and then uses the RF Toolbox `y2s` function to convert the Y-parameters to S-parameters. It interpolates the S-parameters using the MATLAB `interp1` function to determine the S-parameters at the simulation frequencies. The curve in the following diagram illustrates the result of interpolating the S11 parameters at five different frequencies.

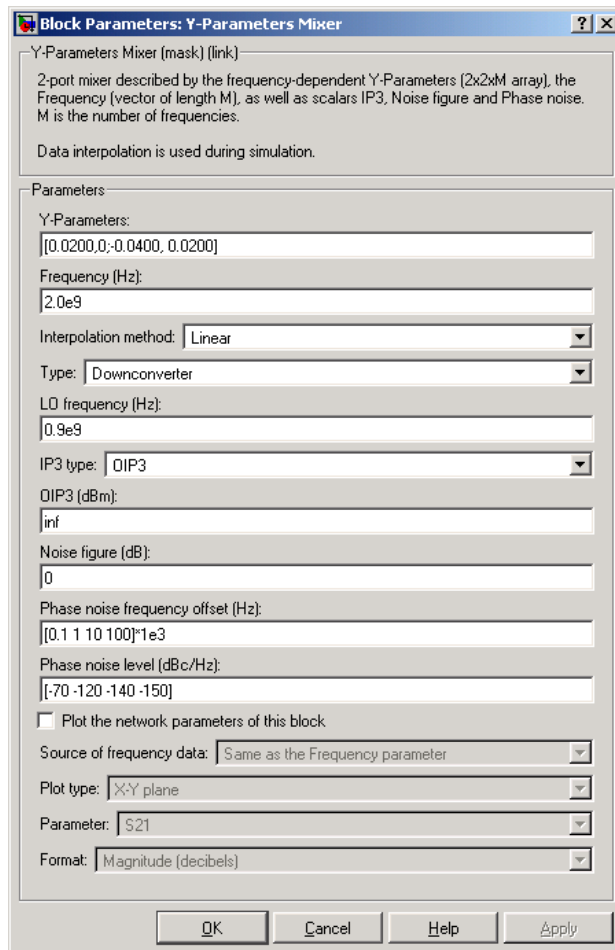


The **Interpolation** field in the block dialog enables you to specify the interpolation method as Cubic, Linear (default), or Spline. For more information about these methods, see “One-Dimensional Interpolation” and the `interp1` reference page in the MATLAB documentation.

As shown in the diagram above, the Y-Parameters Mixer block uses the S-parameters at  $f_{\min}$  for all simulation frequencies smaller than  $f_{\min}$ . It uses the S-parameters at  $f_{\max}$  for all simulation frequencies greater than  $f_{\max}$ . In both cases, the results may not be accurate.

# Y-Parameters Mixer

## 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.

### 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 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,  $f_{out} = f_{in} - f_{lo}$  .  
If you choose Upconverter,  $f_{out} = f_{in} + f_{lo}$  .

**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 network parameters, see “Plotting Network Parameters” on page 3-1.

---

**See Also**

General Mixer, S-Parameters Mixer, Z-Parameters Mixer

# Y-Parameters Passive Network

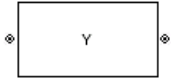
## Purpose

Model a passive network using its Y-parameters

## Library

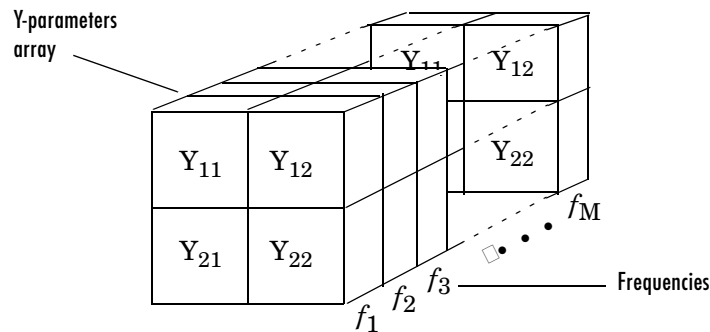
Black Box Elements sublibrary of the Physical library

## Description



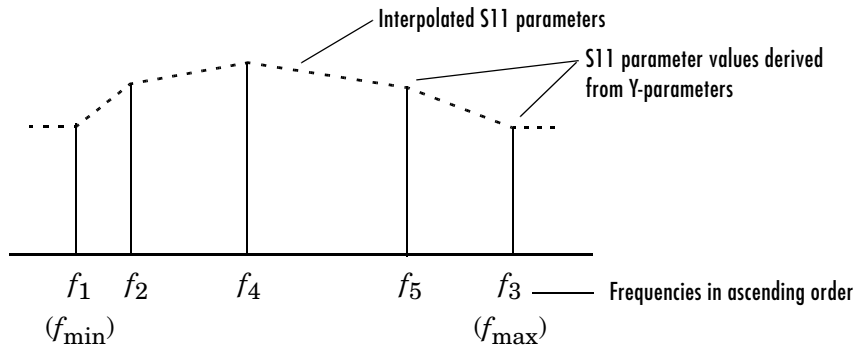
The Y-Parameters Passive Network block models the two-port passive network described in the block dialog, in terms of its Y-parameters and their associated frequencies.

In the **Y-Parameters** field of the block dialog, 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 block orders the Y-parameters according to the ascending order of their frequencies, and then uses the RF Toolbox `y2s` function to convert the Y-parameters to S-parameters. It interpolates the S-parameters, using the MATLAB `interp1` function, to determine the S-parameters at the simulation frequencies. The curve in the following diagram illustrates the result of interpolating the S11 parameters at five different frequencies.

# Y-Parameters Passive Network



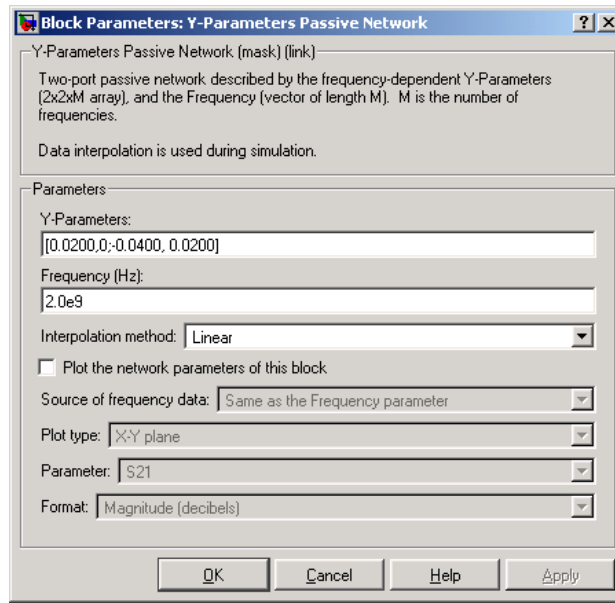
The **Interpolation** field in the block dialog enables you to specify the interpolation method as Cubic, Linear (default), or Spline. For more information about these methods, see “One-Dimensional Interpolation” and the `interp1` reference page in the MATLAB documentation.

As shown in the diagram above, the Y-parameters Passive Network block uses the S-parameters at  $f_{\min}$  for all simulation frequencies smaller than  $f_{\min}$ . It uses the S-parameters at  $f_{\max}$  for all simulation frequencies greater than  $f_{\max}$ . In both cases, the results may not be accurate.

# Y-Parameters Passive Network

---

## Dialog Box



### Y-Parameters

Y-parameters for a 2-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 “Plotting Network Parameters” on page 3-1.

---

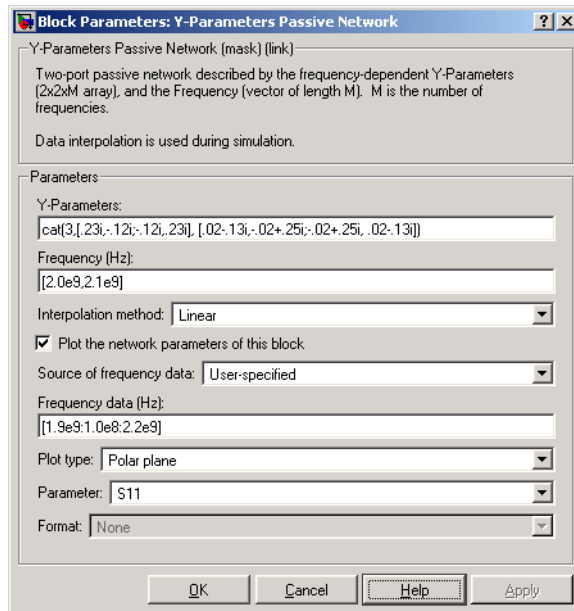


## Examples

### Plotting Parameters with the Y-Parameters Passive Network Block

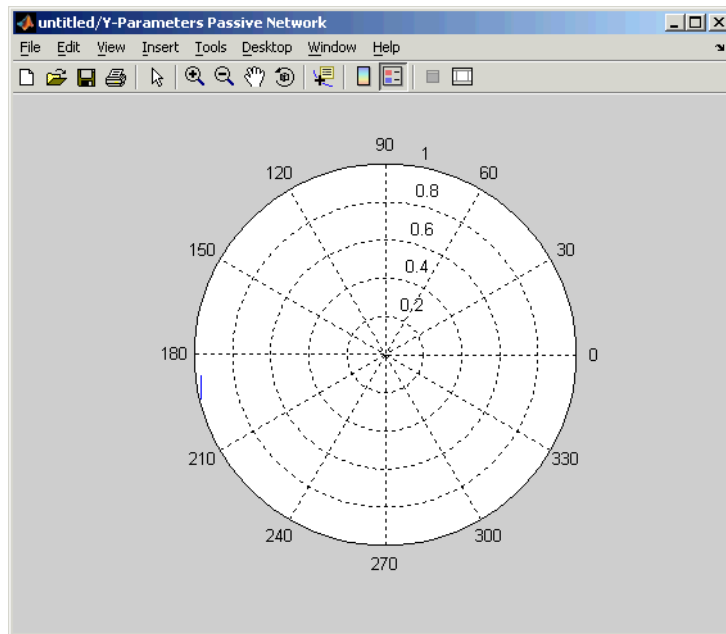
The following example specifies Y-parameters  $[\begin{smallmatrix} .23i & -.12i \\ -.12i & .23i \end{smallmatrix}]$  and  $[\begin{smallmatrix} .02-.13i & -.02+.25i \\ -.02+.25i & .02-.13i \end{smallmatrix}]$  at frequencies 2.0 GHz and 2.1 GHz respectively. It uses the MATLAB `cat` function to create 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 request a polar plane plot of the S11 parameters in the frequency range 1.9 GHz to 2.2 GHz.

# Y-Parameters Passive Network



## See Also

General Circuit Element, General Passive Network, S-Parameters Passive Network, Z-Parameters Passive Network

y2s (RF Toolbox)

interp1 (MATLAB)

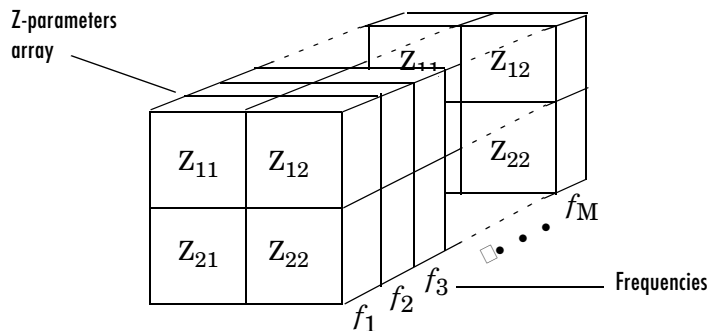
**Purpose** Model a nonlinear amplifier using its Z-parameters

**Library** Amplifiers sublibrary of the Physical library

**Description** The Z-Parameters Amplifier block models the nonlinear amplifier described in the block dialog, in terms of its frequency-dependent Z-parameters and their frequencies. The block also takes into account the IP3 value and the noise figure.

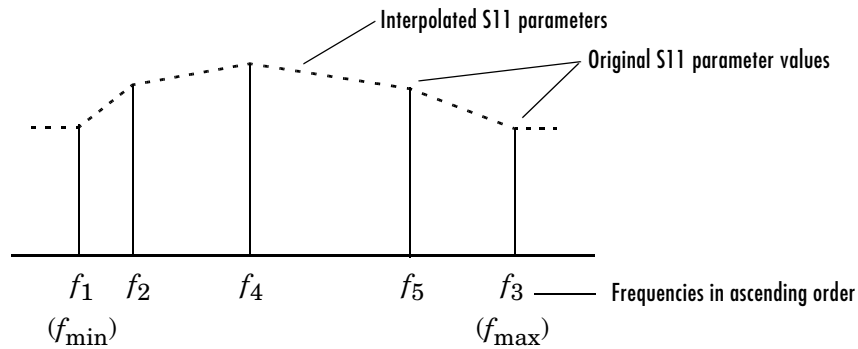


In the **Z-Parameters** field of the block dialog, provide the Z-parameters for each of M frequencies as a 2-by-2-by-M array. In the **Frequency** field, specify the frequencies for the Z-parameters as an M-element vector. The elements of the frequencies vector must be in the same order as the Z-parameters. All frequencies must be positive. For example, the following figure shows the correspondence between the Z-parameters array and the vector of frequencies.



The block orders the Z-parameters according to the ascending order of their frequencies, and then uses the RF Toolbox `z2s` function to convert the Z-parameters to S-parameters. It interpolates the S-parameters using the MATLAB `interp1` function to determine the S-parameters at the simulation frequencies. The curve in the following diagram illustrates the result of interpolating the S11 parameters at five different frequencies.

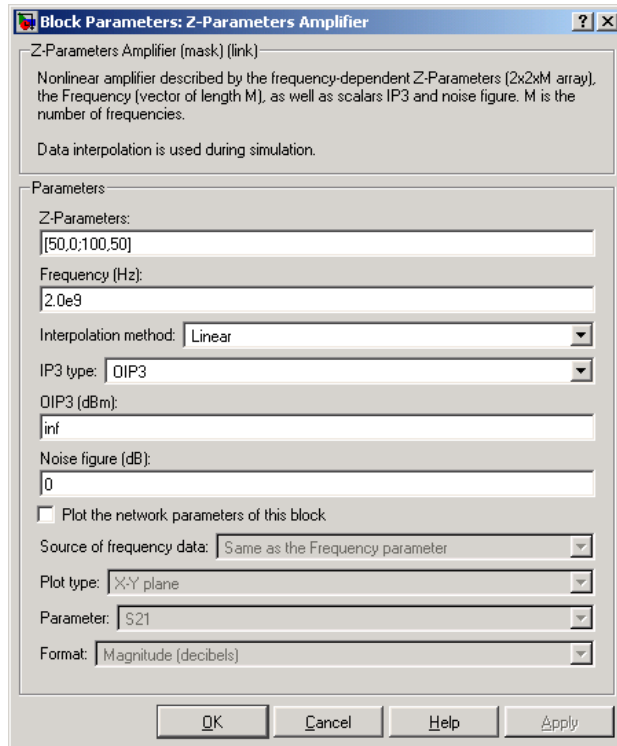
# Z-Parameters Amplifier



The **Interpolation** field in the block dialog 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 diagram above, the Z-Parameters Amplifier block uses the S-parameters at  $f_{\min}$  for all simulation frequencies smaller than  $f_{\min}$ . It uses the S-parameters at  $f_{\max}$  for all simulation frequencies greater than  $f_{\max}$ . In both cases, the results may not be accurate.

## 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 “Plotting Network Parameters” on page 3-1.

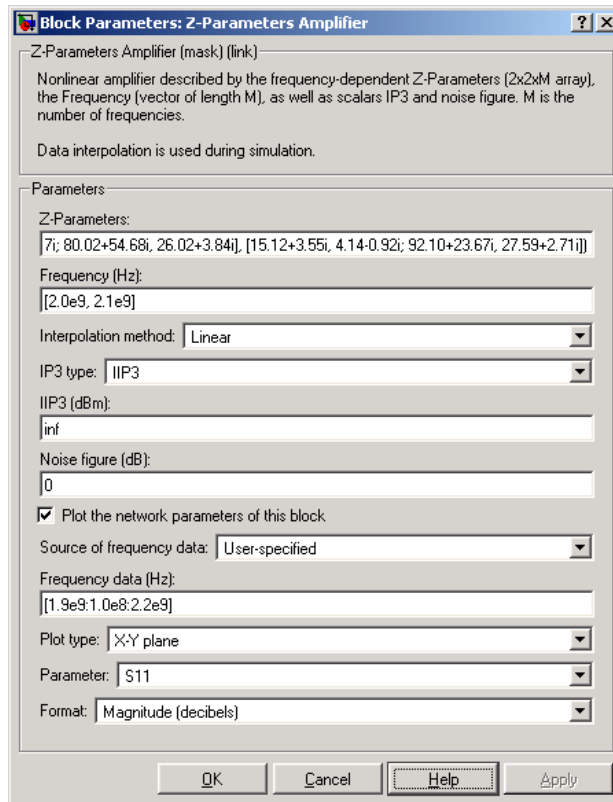
---

## Examples

### Plotting Parameters with the Z-Parameters Amplifier Block

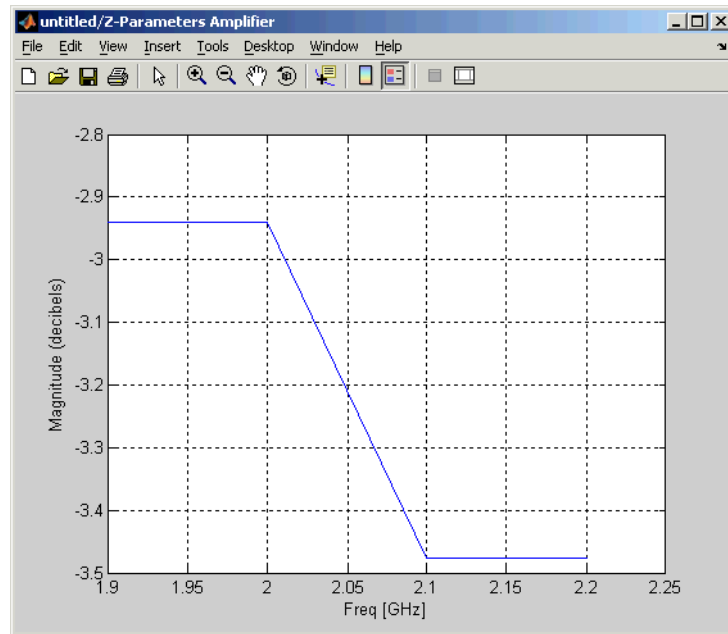
The following example specifies Z-parameters [12.60+3.80i, 3.77-0.17i; 80.02+54.68i, 26.02+3.84i] and [15.12+3.55i, 4.14-0.92i; 92.10+23.67i, 27.59+2.71i] at frequencies 2.0 GHz and 2.1 GHz respectively. It uses the MATLAB `cat` function to create the 2-by-2-by-2 Z-parameters array

```
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])
```



The plot parameters in the dialog request an X-Y plane plot of the S11 parameters in the frequency range 1.9 GHz to 2.2 GHz.

# Z-Parameters Amplifier



## See Also

General Amplifier, S-Parameters Amplifier, Y-Parameters Amplifier

z2s (RF Toolbox)

interp1 (MATLAB)



**Purpose** Model a mixer using its Z-parameters

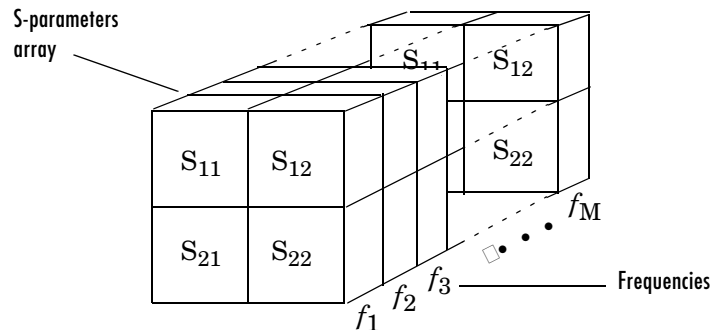
**Library** Mixer sublibrary of the Physical library

## Description



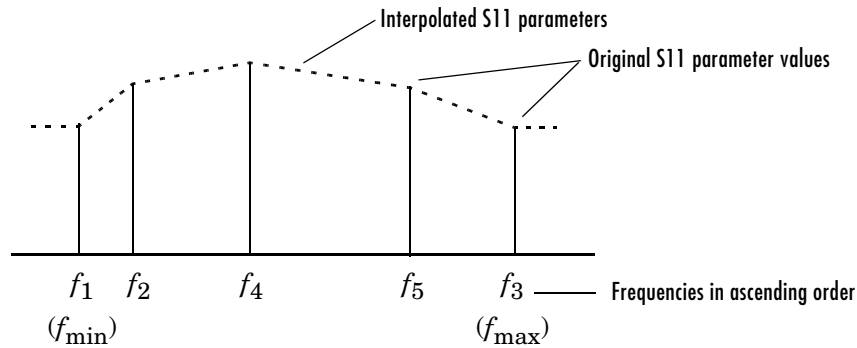
The Z-Parameters Mixer block models the nonlinear mixer described in the block dialog, in terms of its frequency-dependent Z-parameters, their frequencies, and the reference impedance of the Z-parameters. The block also takes phase noise into account.

In the **Z-parameters** field of the block dialog, 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 S-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 block orders the Z-parameters according to the ascending order of their frequencies, and then uses the RF Toolbox `y2s` function to convert the Z-parameters to S-parameters. It interpolates the S-parameters using the MATLAB `interp1` function to determine the S-parameters at the simulation frequencies. The curve in the following diagram illustrates the result of interpolating the S11 parameters at five different frequencies.

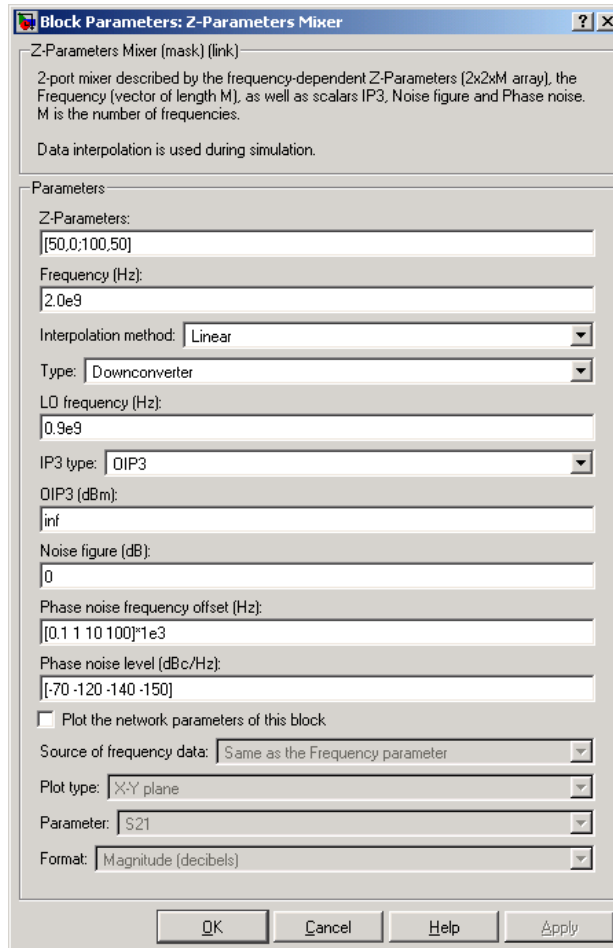
# Z-Parameters Mixer



The **Interpolation** field in the block dialog 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 diagram above, the Z-Parameters Mixer block uses the S-parameters at  $f_{\min}$  for all simulation frequencies smaller than  $f_{\min}$ . It uses the S-parameters at  $f_{\max}$  for all simulation frequencies greater than  $f_{\max}$ . In both cases, the results may not be accurate.

## 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.

# Z-Parameters Mixer

---

## 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,  $f_{out} = f_{in} - f_{lo}$ .  
If you choose Upconverter,  $f_{out} = f_{in} + f_{lo}$ .

## 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 network parameters, see “Plotting Network Parameters” on page 3-1.

---

## See Also

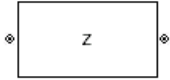
General Mixer, S-Parameters Mixer, Y-Parameters Mixer

# Z-Parameters Passive Network

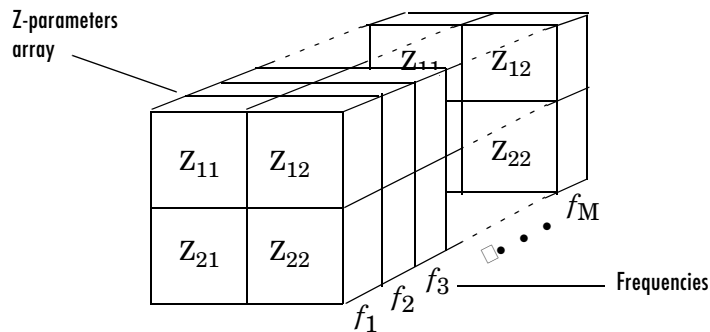
**Purpose** Model a passive network using its Z-parameters

**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, in terms of its Z-parameters and their associated frequencies.

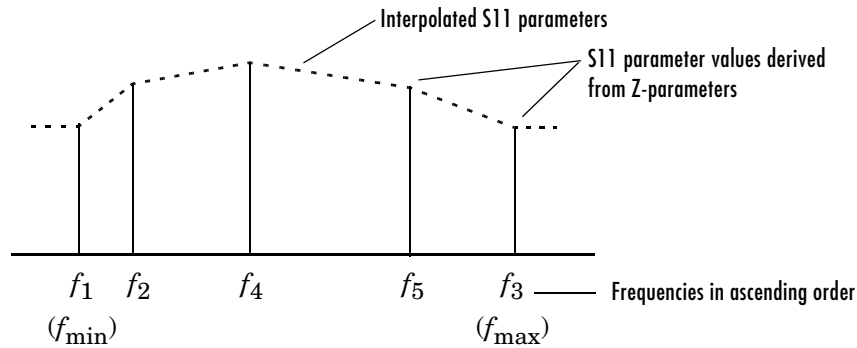


In the **Z-Parameters** field of the block dialog, provide the Z-parameters for each of M frequencies as a 2-by-2-by-M array. In the **Frequency** field, specify the frequencies for the Z-parameters as an M-element vector. The elements of the vector must be in the same order as the Z-parameters. All 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 orders the Z-parameters according to the ascending order of their frequencies, and then uses the RF Toolbox `z2s` function to convert the Z-parameters to S-parameters. It then interpolates the S-parameters, using the MATLAB `interp1` function, to determine the S-parameters at the simulation frequencies. The curve in the following diagram illustrates the result of interpolating the S11 parameters at five different frequencies.

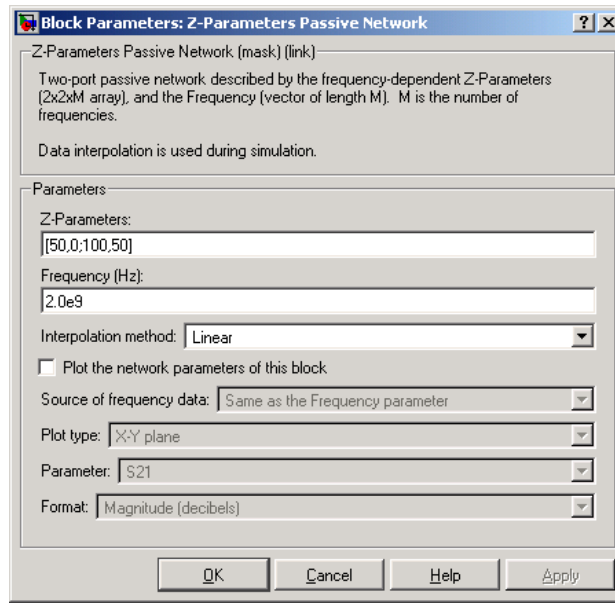
# Z-Parameters Passive Network



The **Interpolation** field in the block dialog 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 diagram above, the Z-parameters Passive Network block uses the S-parameters at  $f_{\min}$  for all simulation frequencies smaller than  $f_{\min}$ . It uses the S-parameters at  $f_{\max}$  for all simulation frequencies greater than  $f_{\max}$ . In both cases, the results may not be accurate.

## Dialog Box



### Z-Parameters

Z-parameters for a 2-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 “Plotting Network Parameters” on page 3-1.

---

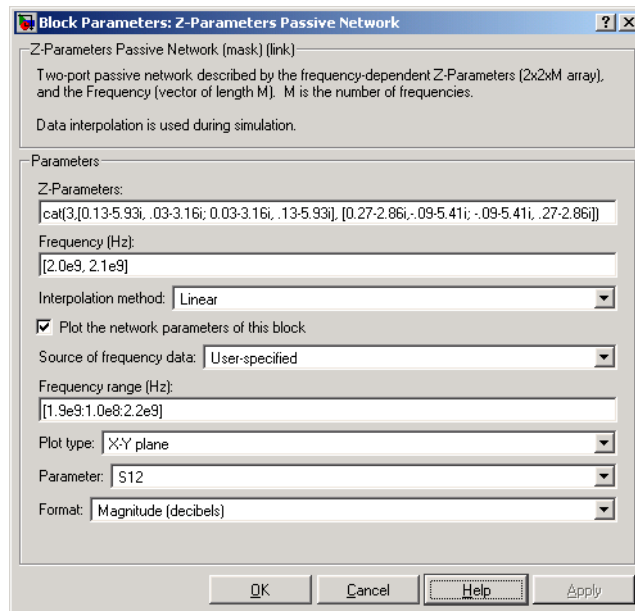
# Z-Parameters Passive Network

## 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.

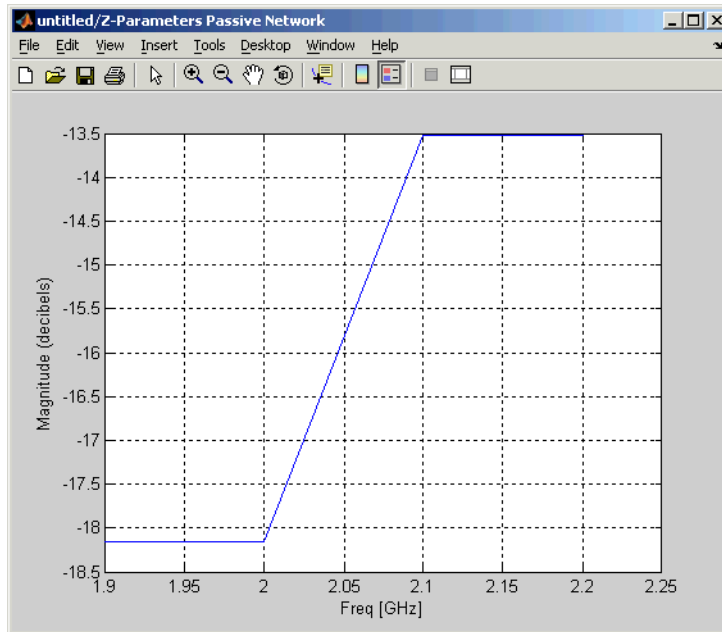
```
cat(3,[0.13-5.93i, .03-3.16i; 0.03-3.16i, .13-5.93i],...  
      [0.27-2.86i, -.09-5.41i; -.09-5.41i, .27-2.86i])
```



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



# Z-Parameters Passive Network



## See Also

General Circuit Element, General Passive Network, S-Parameters Passive Network, Y-Parameters Passive Network

z2s (RF Toolbox)

interp1 (MATLAB)

# Z-Parameters Passive Network

---

## A

- active noise
  - General Amplifier block 4-41
  - General Mixer block 4-50
- amplifier
  - modeling from file data 4-43
  - modeling from `rfd`data object 4-40
- Amplifier block 4-12
  - effects 4-14
  - modeling nonlinearity 4-12
    - methods 4-16
  - sample time comparison 4-18
  - thermal noise simulation 4-17
  - See also.* General Amplifier block

## B

- Bandpass RF Filter block 4-22
  - sample time comparison 4-23
- Bandstop RF Filter block 4-26
  - sample time comparison 4-27
- baseband-equivalent time domain modeling 4-95
  - linear subsystem 4-95
  - nonlinear subsystem 4-97
- Bessel filter design
  - Bandpass RF Filter block 4-22
  - Bandstop RF Filter block 4-26
  - Highpass RF Filter block 4-58
  - Lowpass RF Filter block 4-84
- block parameters
  - setting 2-6
- blocks
  - connecting 2-4
- Butterworth filter design
  - Bandpass RF Filter block 4-22
  - Bandstop RF Filter block 4-26
  - Highpass RF Filter block 4-58

- Lowpass RF Filter block 4-84

## C

- Chebyshev I filter design
  - Bandpass RF Filter block 4-22
  - Bandstop RF Filter block 4-26
  - Highpass RF Filter block 4-58
  - Lowpass RF Filter block 4-84
- Chebyshev II filter design
  - Bandpass RF Filter block 4-22
  - Bandstop RF Filter block 4-26
  - Highpass RF Filter block 4-58
  - Lowpass RF Filter block 4-84
- circuit element
  - modeling from `rfckt` object 4-46, 4-47
- Coaxial Transmission Line block 4-30
  - shunt and series stubs 4-31
  - stubless 4-30
- connecting blocks
  - RF Physical blocks to Simulink 2-4
  - Simulink to RF Physical blocks 2-4
- connector blocks
  - Input Port 4-62
  - Output Port 4-95
- Coplanar Waveguide Transmission Line block
  - 4-35
    - shunt and series stubs 4-36
    - stubless 4-35

## E

- Elliptic filter design
  - Bandpass RF Filter block 4-22
  - Bandstop RF Filter block 4-26
  - Highpass RF Filter block 4-58
  - Lowpass RF Filter block 4-84

**F**

## filter design

## mathematical blocks

Bandpass RF Filter block 4-22

Bandstop RF Filter block 4-26

Highpass RF Filter block 4-58

Lowpass RF Filter block 4-84

## physical blocks

LC Bandpass Pi block 4-65

LC Bandpass Tee block 4-70

LC Bandstop Pi block 4-72

LC Bandstop Tee block 4-74

LC Highpass Pi block 4-76

LC Highpass Tee block 4-78

LC Lowpass Pi block 4-80

LC Lowpass Tee block 4-82

## format

changing in plot 3-10

## frequency range

changing in plot 3-10

**G**

## General Amplifier block 4-40

active noise 4-41

generating from file data 4-43

nonlinearity 4-41

## General Circuit Element block 4-46

generating from `rfckt` object 4-47

## General Mixer block 4-49

active noise 4-50

phase noise 4-50

*See also.* Mixer block 4-93

## General Passive Network block 4-54

generating from file data 4-55

## Ghorbani model 4-16

**H**

## Highpass RF Filter block 4-58

sample time comparison 4-59

**I**

## Input Port block 4-62

parameter input for physical subsystem 4-62

sample time comparison 4-62

*See also.* Output Port Block

## interpolation of network parameters

General Amplifier block 4-40

General Mixer block 4-49

General Passive Network block 4-54

S-Parameters Amplifier block 4-105

S-Parameters Mixer block 4-111

S-Parameters Passive Network block 4-116

Y-Parameters Amplifier block 4-132

Y-Parameters Mixer block 4-138

Y-Parameters Passive Network block 4-142

Z-Parameters Amplifier block 4-147

Z-Parameters Mixer block 4-153

Z-Parameters Passive Network block 4-157

**L**

## ladder filters

example to filter Gaussian noise 4-66

## LC Bandpass Pi block 4-65

example to filter Gaussian noise 4-66

## LC Bandpass Tee block 4-70

## LC Bandstop Pi block 4-72

## LC Bandstop Tee block 4-74

## LC Highpass Pi block 4-76

## LC Highpass Tee block 4-78

## LC Lowpass Pi block 4-80

## LC Lowpass Tee block 4-82

Lowpass RF Filter block 4-84  
 sample time comparison 4-85

## M

mathematical and physical models  
 connecting 2-4  
 mathematical modeling blocks 2-2  
 Microstrip Transmission Line block 4-88  
 shunt and series stubs 4-89  
 stubless 4-88  
 Mixer block 4-93  
 sample time comparison 4-93  
*See also.* General Mixer block  
 modeling  
 amplifier from file data 4-43  
 amplifier from `rfd` object 4-40  
 baseband-equivalent time domain 4-95  
 circuit element from `rfckt` object 4-47  
 passive network from file data 4-55  
 models  
 building RF 2-1  
 physical and mathematical blocks 2-2

## N

network parameters  
 adding to a plot 3-10  
 interpolation  
 General Amplifier block 4-40  
 General Mixer block 4-49  
 General Passive Network block 4-54  
 General S-Parameters Passive Network  
 block 4-116  
 S-Parameters Amplifier block 4-105  
 S-Parameters Mixer block 4-111  
 Y-Parameters Amplifier block 4-132

Y-Parameters Mixer block 4-138  
 Y-Parameters Passive Network block 4-142  
 Z-Parameters Amplifier block 4-147  
 Z-Parameters Mixer block 4-153  
 Z-Parameters Passive Network block 4-157

plotting 3-1

noise

active

General Amplifier block 4-41

General Mixer block 4-50

phase

General Mixer block 4-50

nonlinearity

Amplifier block 4-12

General Amplifier block 4-41

## O

Output Port block 4-95  
 baseband-equivalent time domain modeling  
 4-95  
 plotting RF subsystem network parameters  
 4-99  
*See also.* Input Port Block

## P

Parallel-Plate Transmission Line block 4-100  
 shunt and series stubs 4-101  
 stubless 4-100  
 parameter input  
 to physical subsystem 4-62  
 passive network  
 modeling from file data 4-55  
 phase noise  
 General Mixer block 4-50

physical and mathematical models

connecting blocks 2-4

physical modeling blocks 2-2

physical subsystem

parameter input for 4-62

plot types

modifying 3-10

plots

displaying legend 3-12

examples 3-2

modifying 3-10

network parameters 3-1

RF subsystem 4-99

RF subsystem network parameters 4-99

types of 3-2

using block parameters to specify 3-2

## R

Rapp model 4-17

RF models

building 2-1

connecting blocks 2-4

description 2-2

rfckt object

modeling general circuit element 4-46

rfdata object

modeling nonlinear amplifier 4-40

## S

Saleh model 4-16

setting

block parameters 2-6

Smith chart

example 3-5

S-Parameters Amplifier block 4-105

S-Parameters Mixer block

interpolation of network parameters 4-111

S-Parameters Mixer block 4-111

S-Parameters Passive Network block 4-116

interpolation of network parameters 4-116

stubless transmission lines

Coaxial Transmission Line block 4-30

Coplanar Waveguide Transmission Line block  
4-35

Microstrip Transmission Line block 4-88

Parallel-Plate Transmission Line block 4-100

Transmission Line block 4-122

Two-Wire Transmission Line block 4-127

stubs (shunt and series)

Coaxial Transmission Line block 4-31

Coplanar Waveguide Transmission Line block  
4-36

Microstrip Transmission Line block 4-89

Parallel-Plate Transmission Line block 4-101

Transmission Line block 4-123

Two-Wire Transmission Line block 4-128

## T

time-domain modeling

baseband equivalent 4-95

Transmission Line block 4-122

shunt and series stubs 4-123

stubless 4-122

Two-Wire Transmission Line block 4-127

shunt and series stubs 4-128

stubless 4-127

## V

visualization

network parameters 3-1

**Y**

- Y-Parameters Amplifier block 4-132
  - interpolation of network parameters 4-132
- Y-Parameters Mixer block 4-138
  - interpolation of network parameters 4-138
- Y-Parameters Passive Network block 4-142
  - interpolation of network parameters 4-142

**Z**

- Z-Parameters Amplifier block 4-147
  - interpolation of network parameters 4-147
- Z-Parameters Mixer block 4-153
  - interpolation of network parameters 4-153
- Z-Parameters Passive Network block 4-157
  - interpolation of network parameters 4-157