RF Blockset

For Use with Simulink®

- Modeling
- Simulation
- Implementation

User's Guide

Version 1

The MathWorks

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

Trademarks

Patents

Revision History

June 2004	Online only	New for Version 1.0 (Release 14)
August 2004	Online only	Revised for Version 1.0.1 (Release 14+)
March 2005	Online only	Revised for Version 1.1 (Release 14SP2)
September 2005	Online only	Revised for Version 1.2 (Release 14SP3)

Contents

1 Getting Started

2 Creating an RF Model

3 Working with RF Models

4 Plotting Network Parameters

5 Block Reference

Index

Getting Started

What Is the RF Blockset?

The RF Blockset is a tool for design, analysis, and simulation of the RF (Radio Frequency) portion of wireless communications systems. It uses complex 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®, Signal Processing 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 RF 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 the cascade of components 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 a Simulink signal into the RF Blockset physical modeling environment and back into a Simulink signal. 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.

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

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`.

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

Key Blockset Concepts

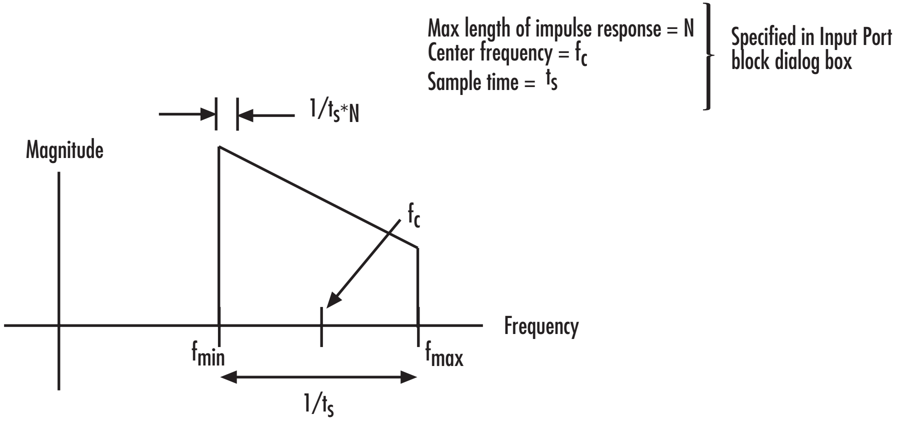
The RF Blockset enables you to simulate the frequency response, noise, and nonlinearity in an RF system. In this section, you learn more about how the blockset works.

This section contains the following topic:

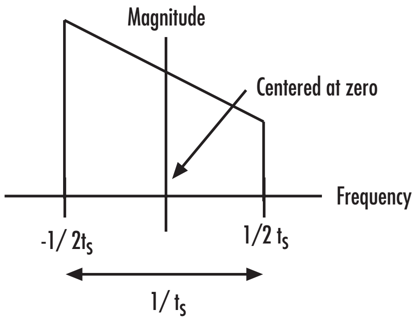
- “Baseband-Equivalent Modeling” on page 1-6 — Understand how the RF Blockset performs the two-step mathematical transformation to create a complex baseband-equivalent model in the time domain
- “Modeling Paradigm” on page 1-8 — Learn how the RF Blockset uses a baseband-equivalent modeling paradigm to efficiently represent the behavior of RF signals and systems in the time domain
- “Noise” on page 1-18 — Understand how the RF Blockset models noise
- “Mixers” on page 1-27 — Learn how RF Blockset mathematical and physical mixer blocks model both mixers and local oscillators
- “Interpretation of Simulink Signals” on page 1-28 — Understand how the RF Blockset interprets these input and output signals

Baseband-Equivalent Modeling

When you run your RF model, the RF Blockset takes the frequency domain parameters you specified in the physical blocks and performs a two-step mathematical transformation to create a complex baseband-equivalent model in the time domain. This process is represented graphically in the figures below:



Passband Spectrum of a Modulated RF Carrier



Baseband-Equivalent Model

The two-step mathematical transformation used to create a complex baseband-equivalent model is reviewed in detail in the next two topics.

Step 1 — Calculating the Baseband Transfer Function

The Output Port block first uses the interpolated S-parameters calculated by the individual blocks and the simulation frequencies it has previously computed to calculate the passband transfer function. The passband transfer function is calculated as

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

where V_S and V_L are the source and load voltages and f represents the 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. The Output Port block derives the Y-parameters from the individual blocks' S-parameters.

The Output Port block then translates the passband transfer function to its equivalent baseband transfer function as $H(f - f_c)$, where f_c is the specified center frequency.

Step 2 — Calculating the Baseband-Equivalent Impulse Response

The Output Port block 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 equal to the maximum length specified. For a detailed example of this process, see "Modeling Paradigm" on page 1-8.

Modeling Paradigm

The RF Blockset uses a baseband-equivalent modeling paradigm to efficiently represent the behavior of RF signals and systems in the time domain. The example in this topic illustrates how the RF Blockset works.

- "Modeling an LC Bandpass Filter" on page 1-9 — Create an LC bandpass filter with a bandwidth of 200 MHz centered at 700 MHz to attenuate a three-tone source signal.

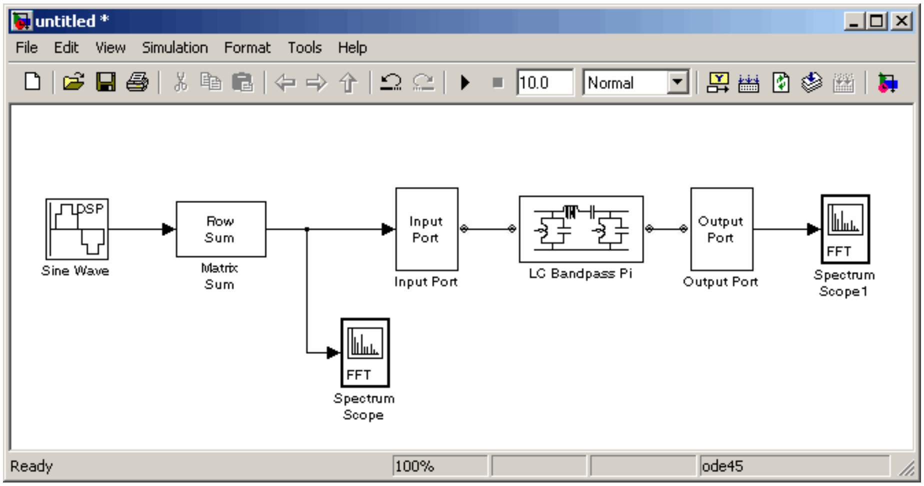
Modeling an LC Bandpass Filter

Here, you create an LC bandpass filter with a bandwidth of 200 MHz centered at 700 MHz. You use this filter to attenuate a three-tone source signal.

1 Create a new Simulink model, and click-and-drag the following blocks into it.

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

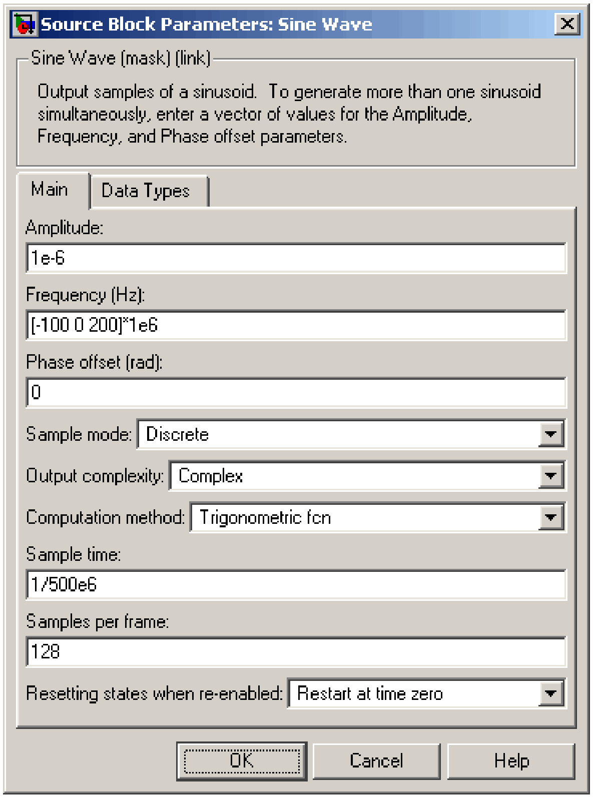
2 Connect the blocks so that your model resembles the following figure.



3 Use the Sine Wave and Matrix Sum blocks to build a three-tone source signal. The actual signal you want to model has one tone at 700 MHz, the center of the filter, one tone at 600 MHz, the lower edge of the passband, and a third tone at 900 MHz, outside the passband. However, because the input to the Input Port block must be the baseband-equivalent signal of the actual RF signal you want to model, you create a three-tone source signal with three tones at -100 MHz, 0 MHz, and 200 MHz, respectively.

Set the Sine Wave block parameters as follows:

- **Amplitude** = `1e-6`
- **Frequency (Hz)** = `[-100 0 200]*1e6`
- **Output complexity** = `Complex`
- **Sample time** = `1/500e6`
- **Sample per frame** = `128`



Sine Wave (mask) (link)

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

Amplitude: 1e-6

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

Phase offset (rad): 0

Sample mode: Discrete

Output complexity: Complex

Computation method: Trigonometric fcn

Sample time: 1/500e6

Samples per frame: 128

Resetting states when re-enabled: Restart at time zero

Because the baseband-equivalent of a real passband signal is complex, you set the **Output complexity** parameter to `Complex`.

The output of the Sine Wave block is a three-channel signal, where each channel corresponds to a different frequency. Use the Matrix Sum block to combine the three channels by setting its **Sum along** parameter to `Rows`.

4 Use the Spectrum Scope block to view the three-tone source signal. Set the block parameters as follows:

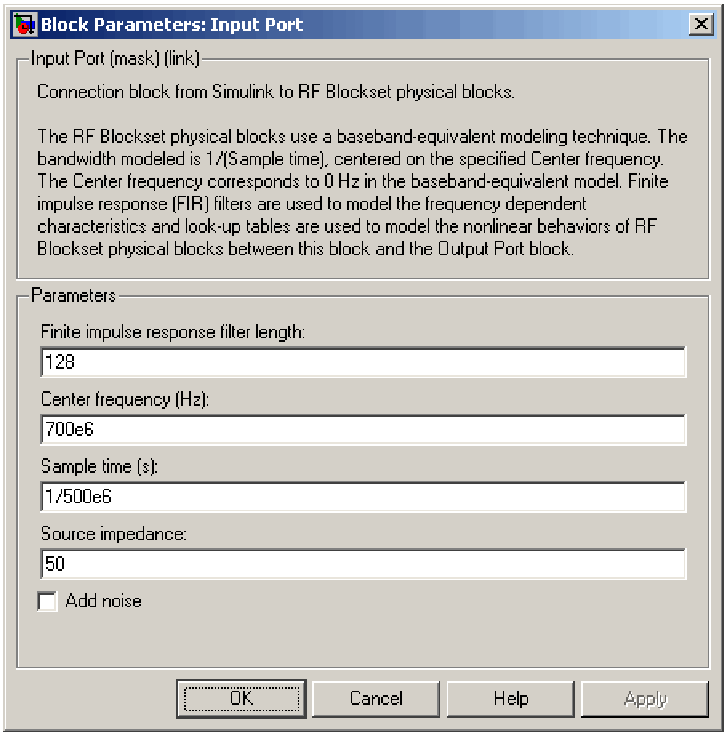
- **Axis Properties** pane, **Frequency range** = `[-Fs/2...Fs/2]`

- **Axis Properties** pane, **Minimum Y-limit** = `-160`
- **Axis Properties** pane, **Maximum Y-limit** = `-100`

Setting the **Scope position** parameter enables you to see both scope windows when you run the model.

5 Use the Input Port block to specify the portion of the RF spectrum that the physical blocks model. Set the block parameters as follows:

- **Center frequency** = `700e6`
- **Sample time(s)** = `1/500e6`
- Clear the **Add noise** check box.



Block Parameters: Input Port

Input Port (mask) (link)

Connection block from Simulink to RF Blockset physical blocks.

The RF Blockset physical blocks use a baseband-equivalent modeling technique. The bandwidth modeled is 1/(Sample time), centered on the specified Center frequency. The Center frequency corresponds to 0 Hz in the baseband-equivalent model. Finite impulse response (FIR) filters are used to model the frequency dependent characteristics and look-up tables are used to model the nonlinear behaviors of RF Blockset physical blocks between this block and the Output Port block.

Parameters

Finite impulse response filter length: 128

Center frequency (Hz): 700e6

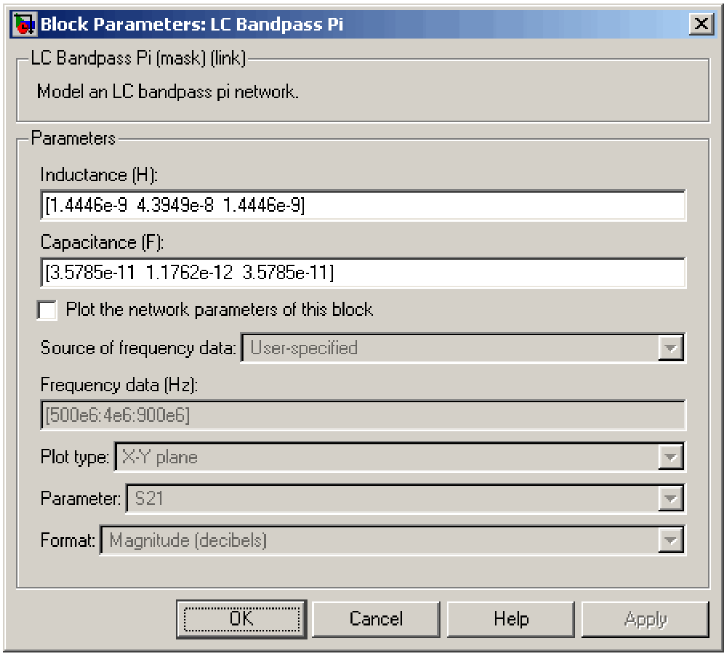
Sample time (s): 1/500e6

Source impedance: 50

□ Add noise

The Input Port block cannot inherit a sample time so you must specify it in the block parameters dialog box. Set the **Sample time(s)** parameter to the same value as the **Sample time** parameter in the Sine Wave block. By specifying a sample time of `1/500e6` second, you establish a modeling bandwidth of 500 MHz. For more information about the **Source impedance** parameter, see “Interpretation of Simulink Signals” on page 1-28.

6 Use the LC Bandpass Pi block to attenuate the source signal. Use the default parameters for inductance and capacitance.



Block Parameters: LC Bandpass Pi

LC Bandpass Pi (mask) (link)

Model an LC bandpass pi network.

Parameters

Inductance (H): [1.4446e-9 4.3949e-8 1.4446e-9]

Capacitance (F): [3.5785e-11 1.1762e-12 3.5785e-11]

□ Plot the network parameters of this block

Source of frequency data: User-specified

Frequency data (Hz): [500e6:4e6:900e6]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

Note If, on the LC Bandpass Pi block dialog box, you select the **Plot the network parameters of this block** check box, the plot displays the actual RF behavior of the block at the selected passband and *not* the complex baseband-equivalent response.

7 Use the Output Port block to specify the system's load impedance and to transition from the physical portion of the model to a Simulink signal. Use the default parameters.

Note After you run the model, the **Plot the network parameters of the RF system** check box appears on the Output Port block dialog box. If you select this check box, the plot displays the actual RF behavior of the system and *not* the behavior of the baseband equivalent model.

---

8 Use the Spectrum Scope1 block to view the attenuated three-tone source signal. Set the block parameters as follows:

- **Axis Properties** pane, **Frequency range** = `[-Fs/2...Fs/2]`
- **Axis Properties** pane, **Minimum Y-limit** = `-160`
- **Axis Properties** pane, **Maximum Y-limit** = `-105`

9 Run the model.

The RF Blockset takes the frequency domain parameters you specified in the physical blocks and performs a two-step mathematical transformation to create a complex baseband-equivalent model in the time domain. This process is called the complex baseband-equivalent model technique. It is analogous to the techniques used for the blocks in the Communications Blockset Channels and RF Impairments libraries. This technique assumes that the frequencies of interest lie between f_{min} and f_{max} and are centered around a center frequency, f_c. For efficient simulation, $f_{max} - f_{min} \ll f_c$. It also assumes that the out-of-band energy is negligible. For more information about this technique, see "Baseband-Equivalent Modeling" on page 1-6.

The RF Blockset uses the complex baseband-equivalent model technique because it improves simulation speed. To model a system in the time domain, Simulink would need to use a small step size.

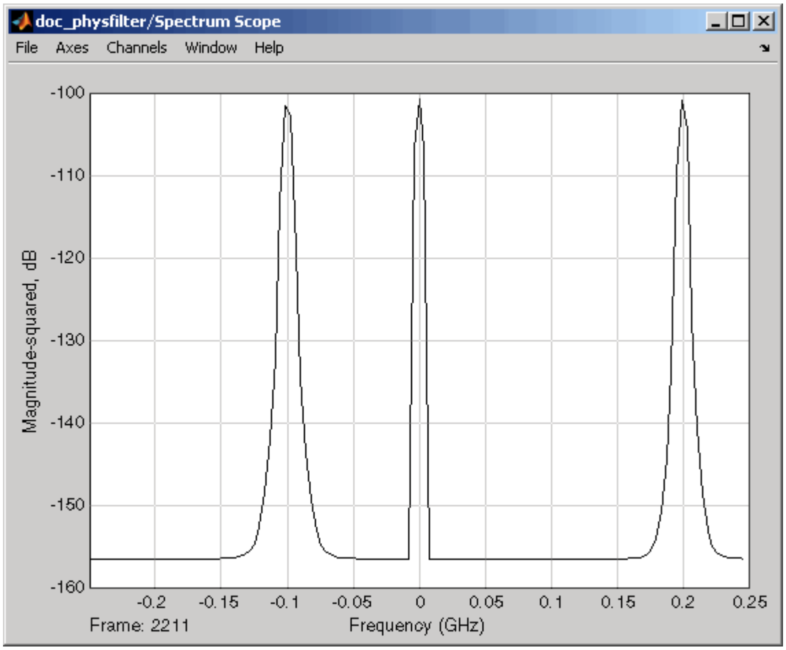
$$t_s = \frac{1}{2f_{max}}$$

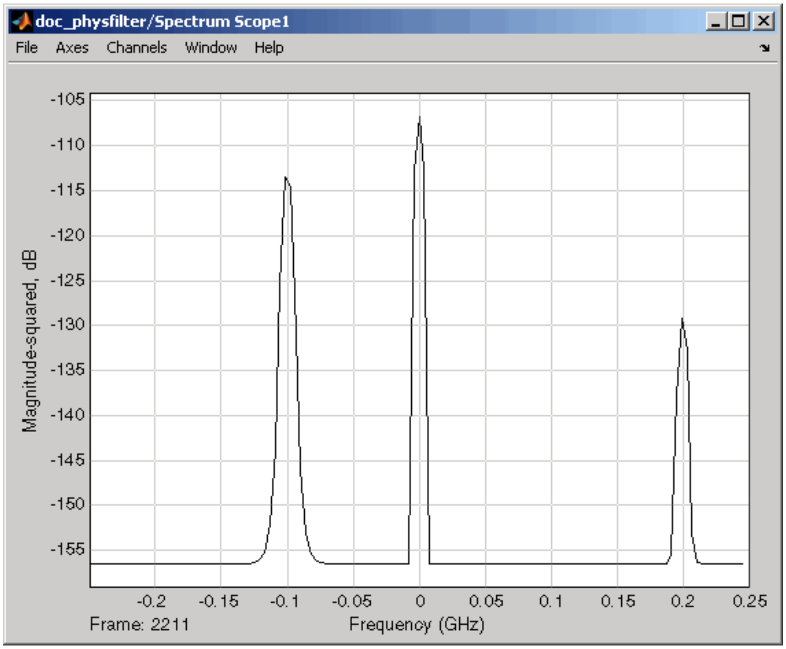
Consequently, the simulation speed would be slow. By creating a complex baseband-equivalent model in the time domain, the time step for the model can be much larger.

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

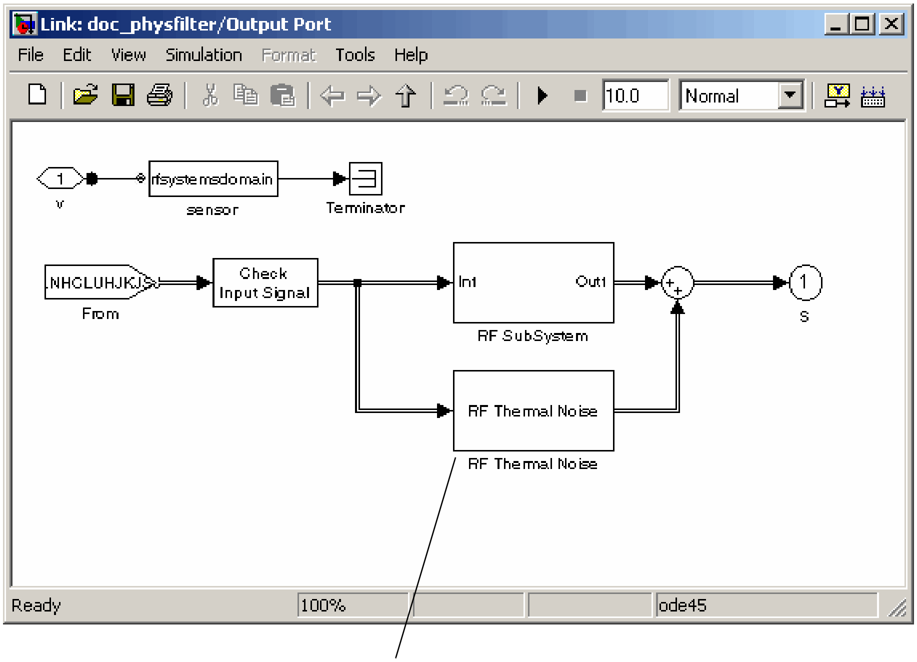
As a result, the model runs much faster.

You can view the source signal and the attenuated signal in the Spectrum Scope windows while the model is running.



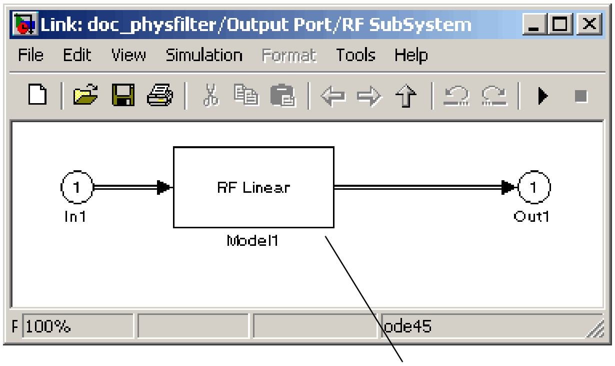


10 Once you have run your simulation or clicked the **Update diagram** button, you can view the baseband-equivalent model by right-clicking the Output Port block and selecting **Look Under Mask**. The following figure is an example of what you might see.



Block models system noise.

In the previous figure, you see a block that models thermal noise. If you right-click on the RF SubSystem block and select **Look Under Mask**, you see the blocks that model the linear and nonlinear components of the RF system. In this example, the blockset only needed to model linear components.


Link: doc_physfilter/Output Port/RF SubSystem
File Edit View Simulation Format Tools Help
1
In1
RF Linear
Model1
1
Out1
F 100%
ode45

Block models linear behavior.

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

You have now used the RF Blockset to model an LC bandpass filter. You also learned more about how the RF Blockset works.

Note Some RF Blockset mathematical blocks are dependent on the center frequency. To obtain accurate results, you must keep track of the center frequency when you are working with these blocks. All RF Blockset physical blocks keep track of the center frequency for you.

Noise

The RF Blockset physical blocks model thermal noise. Typically, you can specify output-referred noise in one of three ways:

- `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 the standard reference noise temperature of 290 degrees kelvin. In terms of noise factor,

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

Calculating Noise Figure

For RF Blockset physical amplifier and mixer blocks, noise can be characterized by noise figure values or the spot noise data, which is more accurate. If you specify the noise figure values directly, the RF Blockset interpolates to find the noise figure values at the simulation frequencies and uses these for its calculations. If you specify spot noise data using a Touchstone or AMP data file, the RF Blockset takes the minimum noise figure, $NF_{\min}$, equivalent noise resistance, R_n, and optimal source admittance, Y_{opt}, values in the file and interpolates to find the values at the simulation frequencies. Then it uses the following equation to calculate the noise correlation matrix, C_A.

$$C_A = 2kT\begin{bmatrix} R_n & \dfrac{NF_{\min}-1}{2} - R_n {Y_{opt}}^* \\ \dfrac{NF_{\min}-1}{2} - R_n Y_{opt} & R_n \left|Y_{opt}\right|^2 \end{bmatrix}$$

In the above equation, k is the Boltzmann constant and T is the noise temperature. Once you know the noise correlation matrix, you can use the following equation to calculate the noise factor, F.

$$F = 1 + \frac{z^+ C_A z}{2kT\,\mathrm{Re}\{Z_s\}}$$

$$z = \begin{bmatrix} 1 \\ {Z_s}^* \end{bmatrix}$$

In the above equations, Z_s is the nominal impedance, which is 50 ohms, and z^+ is the Hermitian conjugation of z. Finally, you can use the noise factor to calculate the noise figure, NF.

$$NF = 10\log(F)$$

For the other blocks, such as transmission lines and filters, the RF Blockset calculates the noise figure values using the network parameters. So, there are no noise parameters on the block dialog boxes.

For more information about these calculation techniques, see Hillbrand, Herbert and Peter H. Russer, "An Efficient Method for Computer Aided Noise Analysis of Linear Amplifier Networks," *IEEE Transactions on Circuits and Systems*, Vol. CAS-23, No. 4, April 1976, pp. 235-238.

Calculating Noise Power

The RF Blockset uses the noise temperature, which you specify in the block parameters dialog box, and the modeling bandwidth, which it calculates from the model's sample time and center frequency, to calculate the noise power:

$$\text{Noise power} = \text{kTB}$$

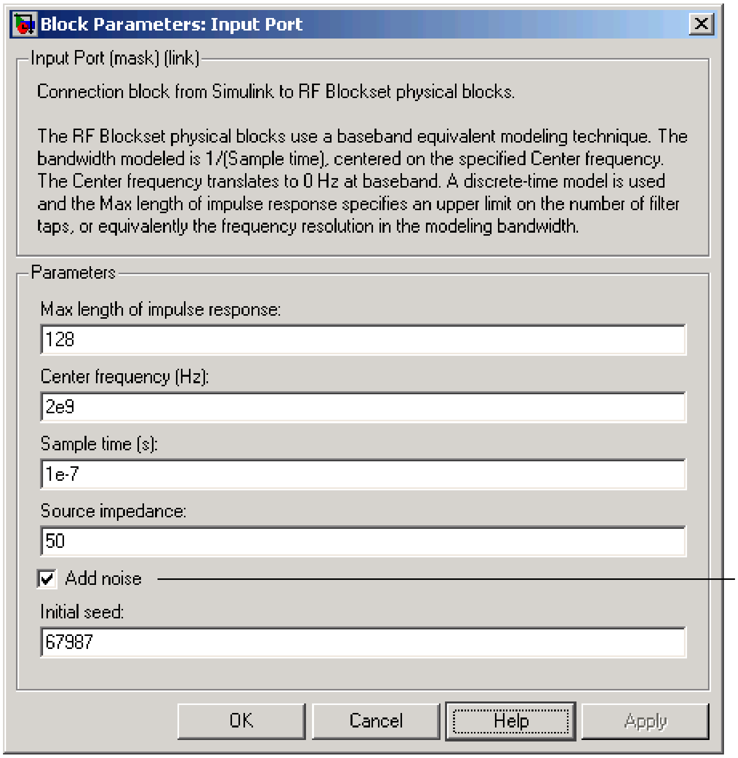
Here, k is the Boltzmann constant, which is 1.38e-23 Joules/kelvin, T is the noise temperature in kelvin and B is the bandwidth in hertz. It uses this noise power to determine the amplitude of the noise that it adds to the system using a Gaussian distributed pseudorandom number generator.

For more information, see the `rfdata.noise` and `rfdata.nf` reference pages.

Adding Noise to Your System

To add noise to your RF system, follow these steps:

1. Enter values for the **Noise figure (dB)** parameters in the physical blocks.
2. On the Input Port block dialog box, select the **Add noise** check box.



Block Parameters: Input Port

Input Port (mask) (link)

Connection block from Simulink to RF Blockset physical blocks.

The RF Blockset physical blocks use a baseband equivalent modeling technique. The bandwidth modeled is 1/(Sample time), centered on the specified Center frequency. The Center frequency translates to 0 Hz at baseband. A discrete-time model is used and the Max length of impulse response specifies an upper limit on the number of filter taps, or equivalently the frequency resolution in the modeling bandwidth.

Parameters

Max length of impulse response: 128

Center frequency (Hz): 2e9

Sample time (s): 1e-7

Source impedance: 50

☑ Add noise

Initial seed: 67987

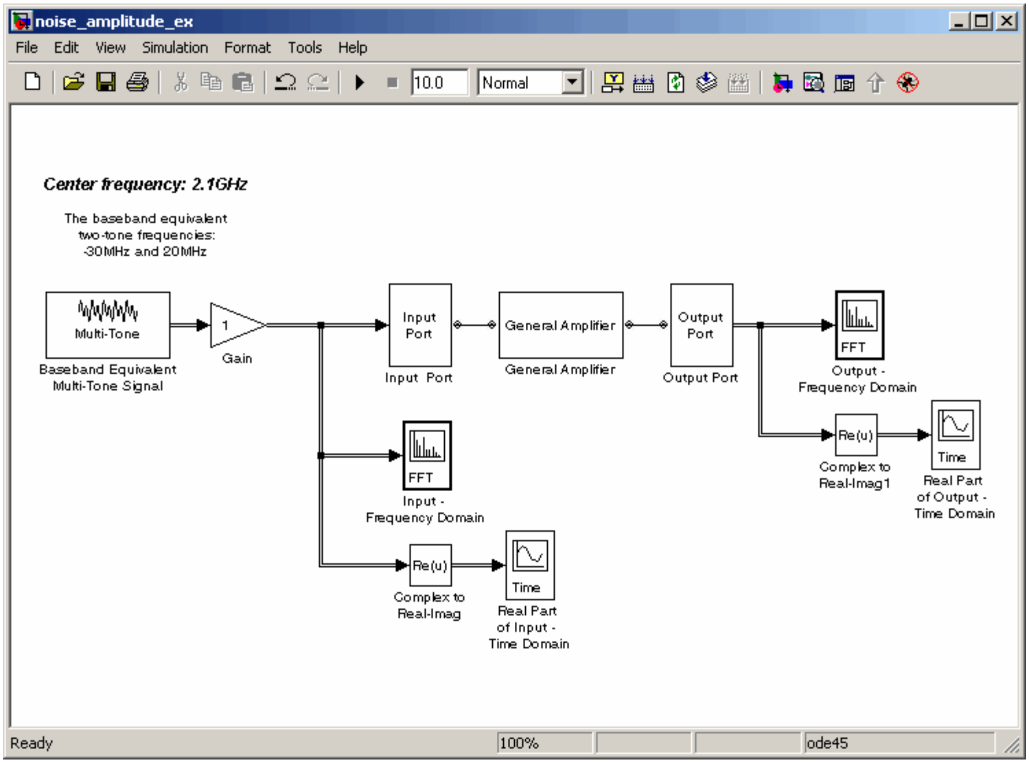
Select this check box to take the noise data in the physical blocks into account. This check box is selected by default.

3 Run the model.

Noise Amplitude

The noise in communications systems has a very small amplitude, typically between 1e-6 to 1e-12 Watts. In contrast, the default signal power of a Communications Blockset modulator block is 1 Watt at a nominal 1 ohm. As a result, you won’t see the noise the RF Blockset adds to your signal unless you attenuate the signal amplitude to obtain the desired signal to noise ratio.

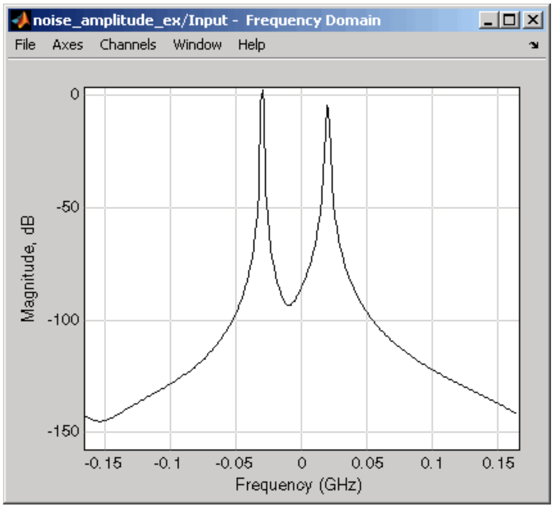
For example, suppose you have the following model that contains a multitone test signal source.



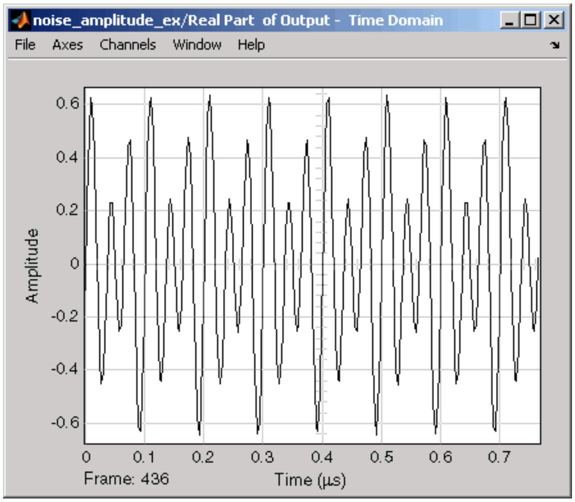
Center frequency: 2.1GHz

The baseband equivalent
two-tone frequencies:
-30MHz and 20MHz

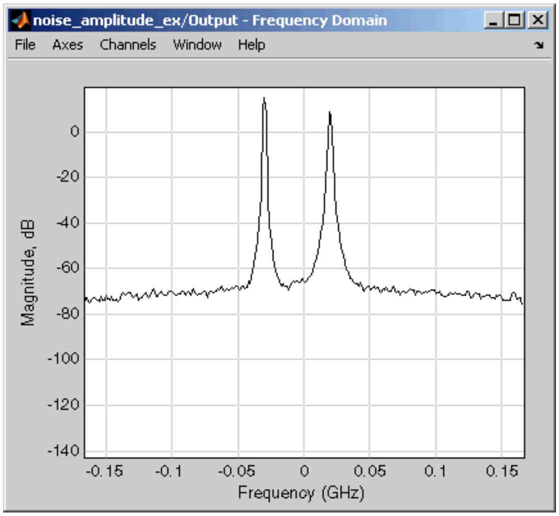
When you run this model, you can see the original signal in the frequency domain in the Input - Frequency Domain window.


noise_amplitude_ex/Input - Frequency Domain
File Axes Channels Window Help
0
-50
-100
-150
Magnitude, dB
-0.15
-0.1
-0.05
0
0.05
0.1
0.15
Frequency (GHz)

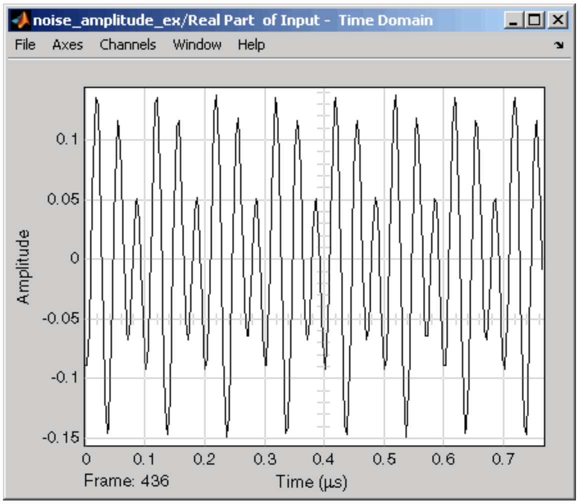
You can see the real part of the complex-valued original signal in the time domain in the Real Part of Input - Time Domain window.


noise_amplitude_ex/Real Part of Output - Time Domain
File Axes Channels Window Help
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
Amplitude
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Frame: 436
Time (µs)

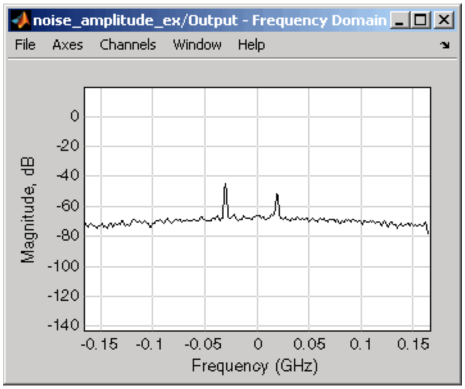
In the model, the Input Port block adds noise to the signal and the General Amplifier block amplifies it. You can see the amplified, noisy signal in the frequency domain in the Output - Frequency Domain window.



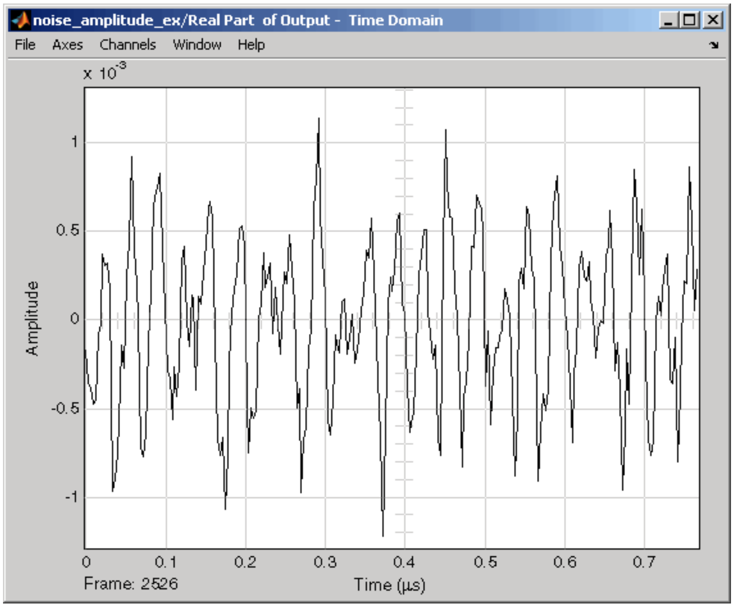
However, because the amplitude of the signal is large compared to the amplitude of the noise, you cannot see the noise in the Real Part of Output - Time Domain window.



To see the noise the time-domain output signal, you must attenuate the amplitude of the original signal. For example, suppose you set the **Gain** parameter to `1e-3`. This is equivalent to attenuating the input signal by 60 dB. After running the model again, you would see that the two signal peaks are not as pronounced in the Output - Frequency Domain window.

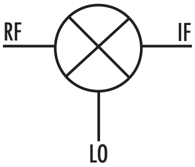


Also, you can see the noise the RF Blockset adds to your signal in the Real Part of Output - Time Domain window.

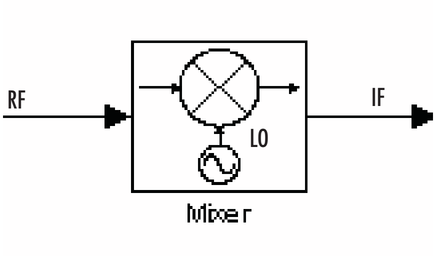


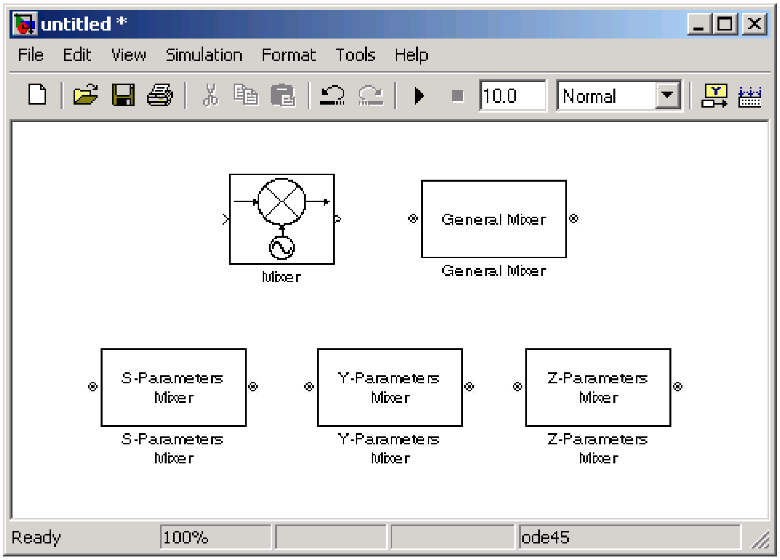
Mixers

Typically, the block diagram of a mixer has three ports as shown in the following representation of a downconversion mixer.



The RF Blockset mathematical and physical mixer blocks contain both a mixer and a local oscillator, so they have only two ports.

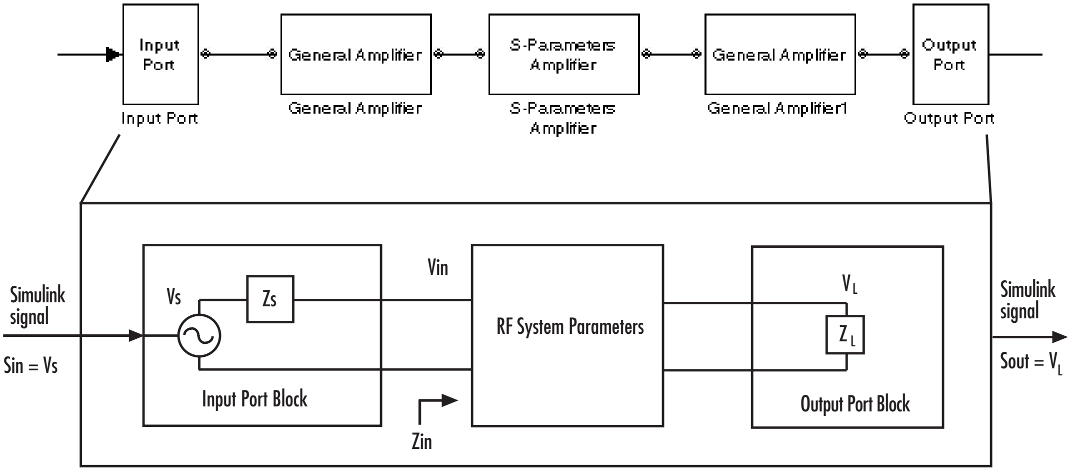




For the physical blocks, you can use the **LO frequency (Hz)** parameter to specify the local oscillator's frequency.

Interpretation of Simulink Signals

The RF Blockset interprets the Simulink signal input to the Input Port block as the voltage of the voltage source, as shown in the following figure. It interprets the output Simulink signal is the voltage over the load impedance.


Input Port
Input Port
General Amplifier
General Amplifier
S-Parameters Amplifier
S-Parameters Amplifier
General Amplifier
General Amplifier1
Output Port
Output Port
Simulink signal
Sin = Vs
Vs
Zs
Input Port Block
Vin
Zin
RF System Parameters
V_L
Z_L
Output Port Block
Simulink signal
Sout = V_L

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.

Creating an RF Model

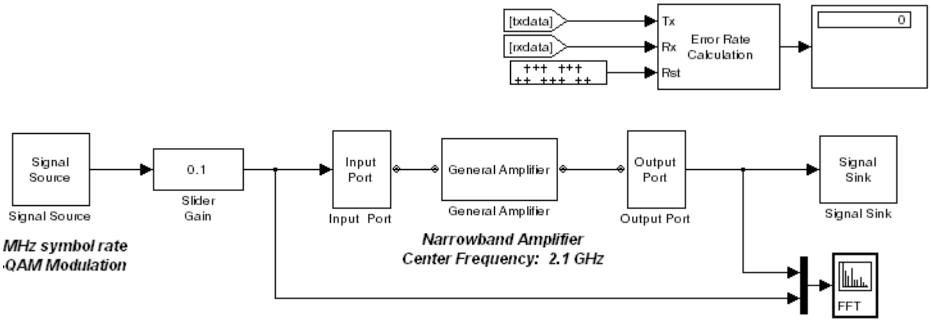
RF Models

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 a 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.

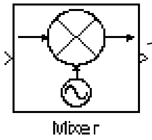
Connecting RF Blocks

To create an RF model, first open a model window and open the RF Blockset library by typing `rflib` 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.

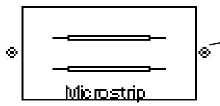
Block Ports

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.



RF Blockset mathematical modeling ports show signal direction.

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



RF Blockset physical modeling connector ports represent only physical connections.

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.

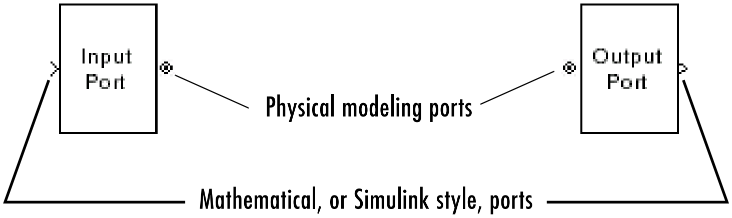
Note Physical blocks must be fully connected before you run the model. If they are not, an error is produced.

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.

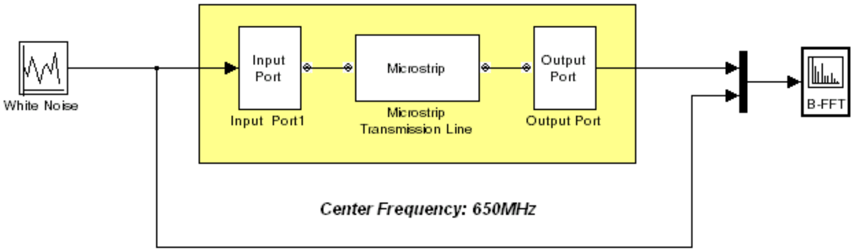
Connecting Physical and Mathematical Parts of a Model

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.

Working with RF Models

S-Parameters at Simulation Frequencies

In a physical system, each block provides network parameters at different frequencies. These are not necessarily the simulation frequencies for the physical system in which the block resides. A two-step process results in S-parameters at the simulation frequencies for each block:

1 "Determining the Simulation Frequencies" on page 3-2

2 "Interpolating the S-Parameters" on page 3-2

Determining the Simulation Frequencies

The Output Port block uses Input Port block parameters to determine the simulation frequencies f for the physical system bracketed between the Input Port block and the Output Port block. f is an N-element vector, where N is the maximum length of the impulse response. The simulation frequencies are also a function of the center frequency f_c and the sample time t_s.

The nth element of f, f_n, is given by

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

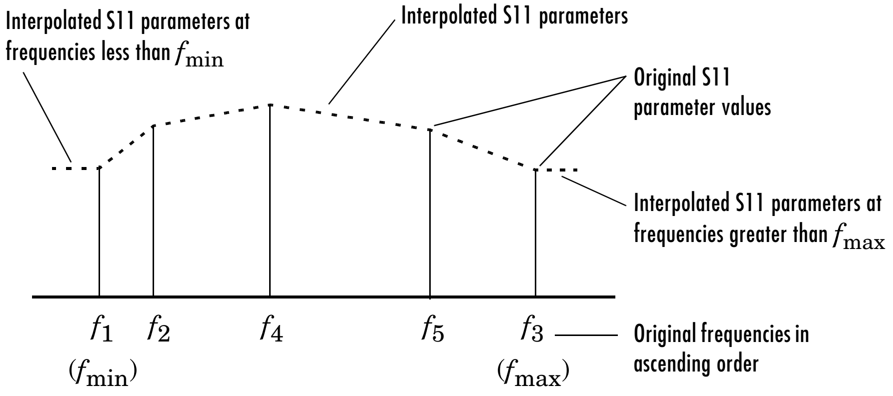
where

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

Interpolating the S-Parameters

Individual physical blocks calculate the S-parameters at the simulation frequencies determined by the Output Port block. Each block interpolates its S-parameters to determine the S-parameters at the simulation frequencies. If the block contains network Y- or Z-parameters, it first converts them to S-parameters.

Specifically, the block orders the S-parameters in the ascending order of their frequencies, f_n. 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 the original frequencies f_1 through f_5.


Interpolated S11 parameters at frequencies less than f_{min}
Interpolated S11 parameters
Original S11 parameter values
Interpolated S11 parameters at frequencies greater than f_{max}
f_1
(f_{min})
f_2
f_4
f_5
f_3
(f_{max})
Original frequencies in ascending order

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

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

Using File Data

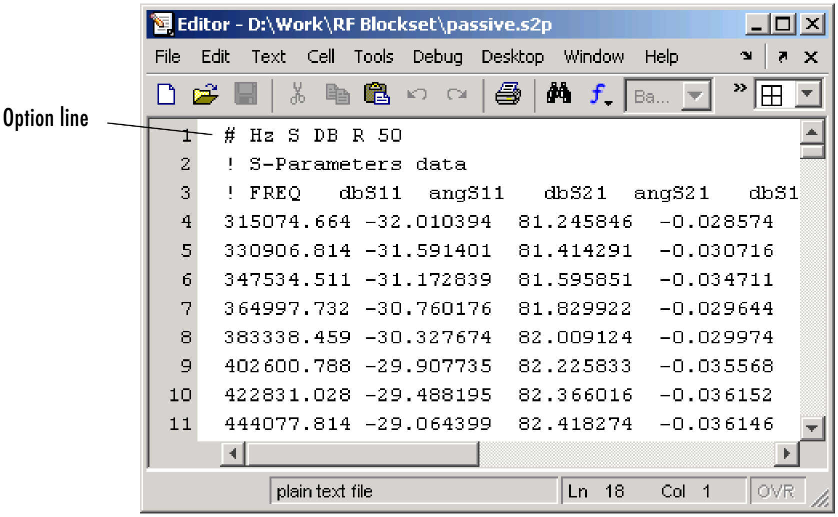
The RF Blockset can read data in Touchstone SnP, Y2P, Z2P, and H2P file formats. You can obtain this data from network analyzers, manufacturer Web sites, or other software packages. For Touchstone `.snp` files, `n` is an integer that represents the number of ports on the device to which the data corresponds. The RF Blockset only supports `.s2p` files. The scattering parameters in these files are represented by the complex variables S11, S12, S21, and S22.

In this example, you view the contents of `passive.s2p` and bring the data into an RF model using a General Passive Network block.

1 Open the `passive.s2p` file. At the MATLAB command prompt, type

```
edit passive.s2p
```

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

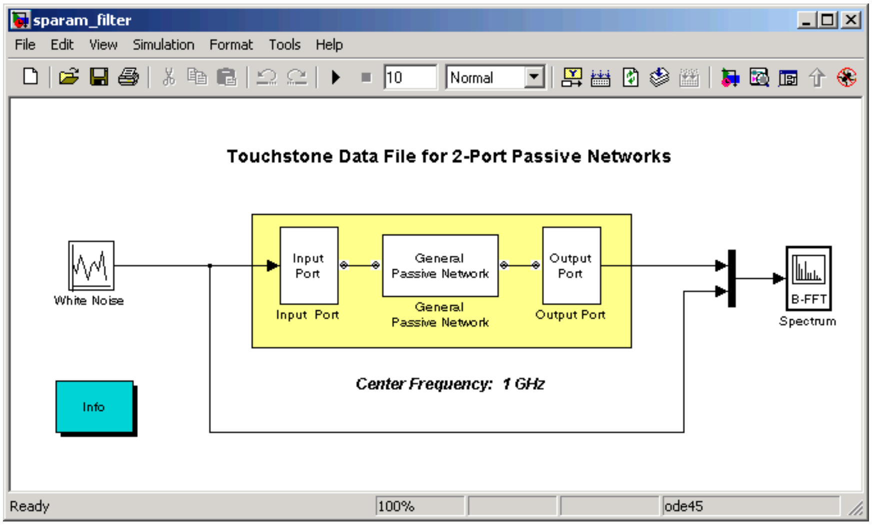

Editor - D:\Work\RF Blockset\passive.s2p
File Edit Text Cell Tools Debug Desktop Window Help
Ba...
plain text file
Ln 18 Col 1
OVR

Option line

```
# Hz S DB R 50
! S-Parameters data
! FREQ     dbS11   angS11    dbS21  angS21    dbS1
315074.664 -32.010394  81.245846  -0.028574
330906.814 -31.591401  81.414291  -0.030716
347534.511 -31.172839  81.595851  -0.034711
364997.732 -30.760176  81.829922  -0.029644
383338.459 -30.327674  82.009124  -0.029974
402600.788 -29.907735  82.225833  -0.035568
422831.028 -29.488195  82.366016  -0.036152
444077.814 -29.064399  82.418274  -0.036146
```

Note that the option line reveals that the frequency is in Hz, the file contains S-parameters, the parameters are given in terms of dB magnitude and phase, and the reference impedance is 50 ohms. For more information about the option line, see `http://www.eda.org/pub/ibis/connector/touchstone_spec11.pdf`.

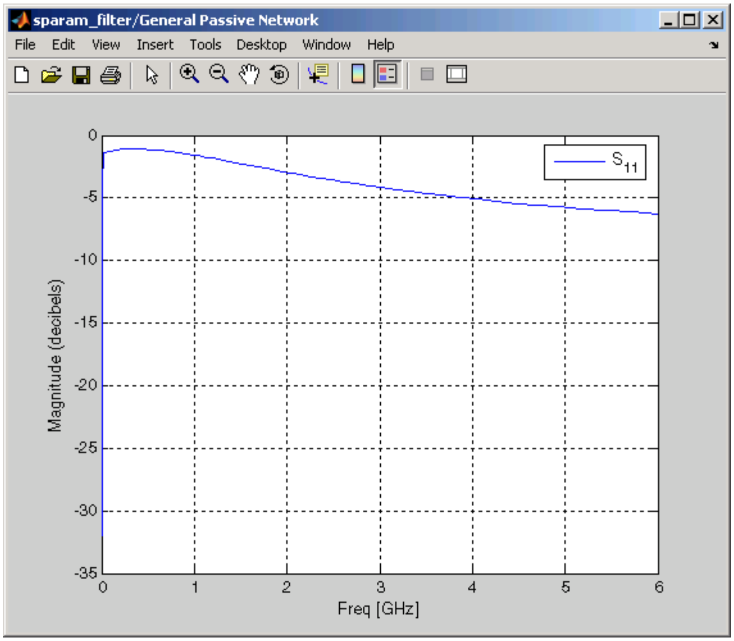
2 Open the RF Blockset demo called Touchstone Data File for 2-Port Passive Networks by typing `sparam_filter` at the MATLAB command prompt.



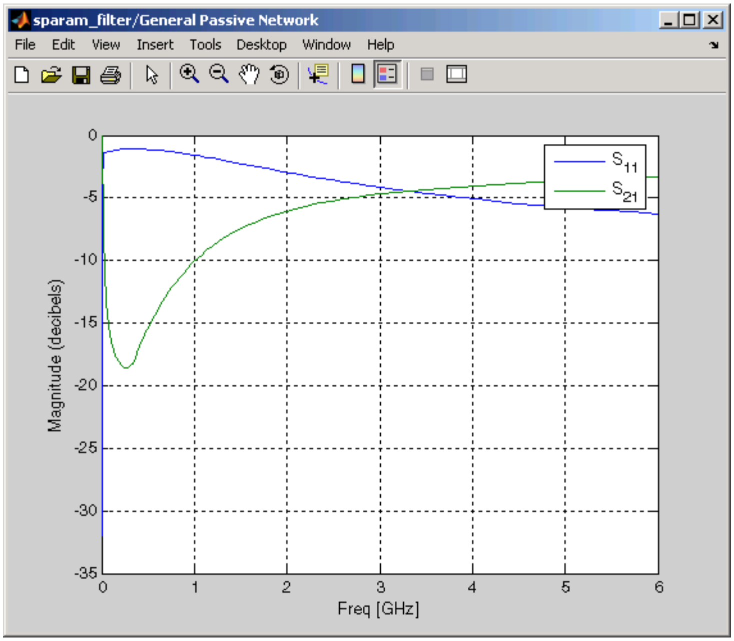
3 Use the General Passive Network block to import the data into the RF model. Double-click the General Passive Network block. The **RFDATA Object** parameter is set to `read(rfdata.data, 'passive.s2p')`. This creates an `rfdata.data` object and reads the data from `passive.s2p` into it. The block uses this data as part of its parameters.

4 Plot the data you imported into the RF model. Select the **Plot the network parameters of this block** check box. From the **Parameter** list, select `S11`. Click **Apply**.

The plot of the S11 parameter appears in the General Passive Network window. Note that the data is plotted at the RF frequencies and not at the baseband equivalent frequencies.

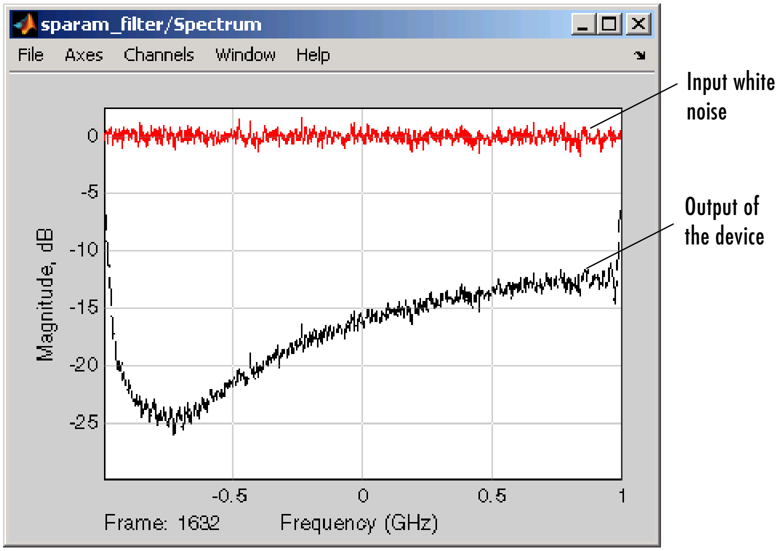

sparam_filter/General Passive Network
File Edit View Insert Tools Desktop Window Help
S11
0
-5
-10
-15
-20
-25
-30
-35
Magnitude (decibels)
0
1
2
3
4
5
6
Freq [GHz]

From the **Parameter** list, select `S21`. Click **Apply**. The block adds the S21 data to the plot. It continues to add to the plot until you close the plot window.


sparam_filter/General Passive Network
File Edit View Insert Tools Desktop Window Help
S_{11}
S_{21}
0
-5
-10
-15
-20
-25
-30
-35
Magnitude (decibels)
0
1
2
3
4
5
6
Freq [GHz]

5 Run the demo model.

The model plots the input white noise and the system output, which is approximately the same as S21, in the Spectrum window.



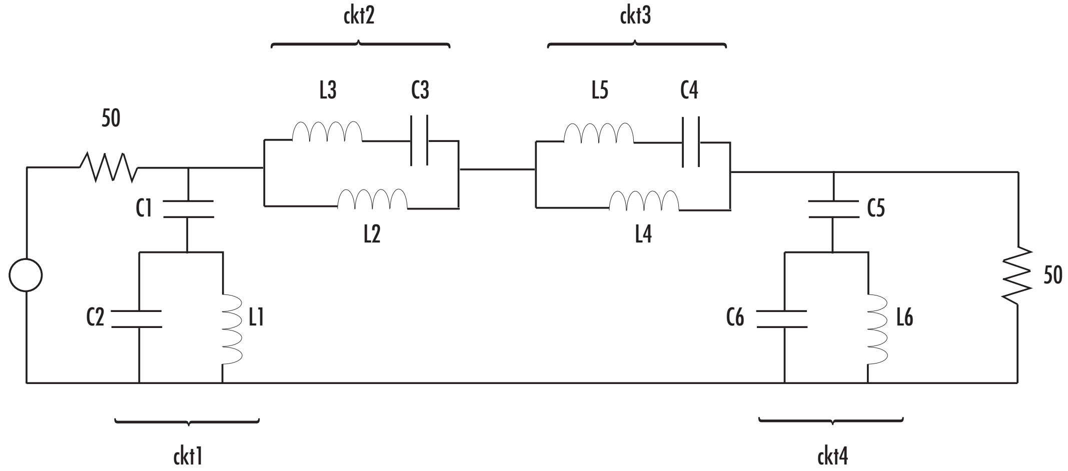
Note The S-parameter values approach -18 dB in the General Passive Network window and -24 dB in the Spectrum window. The additional 6 dB of loss is caused by the source and the load in the `sparam_filter` model.

You have now brought the data from `passive.s2p` into an RF model using a General Passive Network block.

The `.s2p` file may contain data that is measured at frequencies that do not correspond to the simulation frequencies. In this case, you can use the **Interpolation method** parameter to determine how the block determines the data values at the simulation frequencies. For more information, see “Determining the Simulation Frequencies” on page 3-2 and “Interpolating the S-Parameters” on page 3-2.

Importing Circuits from the MATLAB Workspace

The physical blocks in the RF Blockset can only be connected in cascade. However, you can still use RF Blockset blocks to model circuits that contain series and parallel elements. In this example, you model the 50 ohm bandstop filter shown below by creating circuit objects in the MATLAB workspace. You import these circuit objects into a Simulink model using the RF Blockset General Circuit Element block:



1 Define the filter's resistance, capacitance, and inductance values in the MATLAB workspace. At the MATLAB command prompt, type

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

```
L5 = 17.63e-9;
L6 = 25.70e-9;
```

2 Use RF Toolbox functions to create RF circuit objects that model the components labeled ckt1, ckt2, ckt3, and ckt4 in the circuit diagram. At the MATLAB command prompt, type

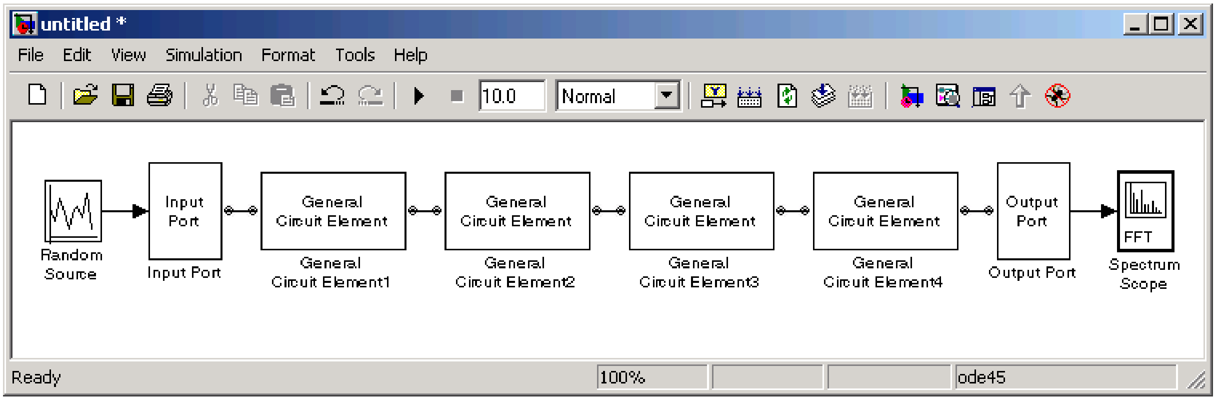
```
ckt1 =
rfckt.series('Ckts',{rfckt.shuntrlc('C',C1),rfckt.shuntrlc('L'
,L1,'C',C2)});
ckt2 =
rfckt.parallel('Ckts',{rfckt.seriesrlc('L',L2),rfckt.seriesrlc
('L',L3,'C',C3)});
ckt3 =
rfckt.parallel('Ckts',{rfckt.seriesrlc('L',L4),rfckt.seriesrlc
('L',L5,'C',C4)});
ckt4 =
rfckt.series('Ckts',{rfckt.shuntrlc('C',C5),rfckt.shuntrlc('L'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

3 Create a new Simulink model.

4 Click-and-drag the following blocks into your model.

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

5 Connect the blocks so your model is similar to the figure below. You need to change the names of the General Circuit Element blocks for them to appear as they do in the figure.



6 Use the Random Source block to create random, complex input values that have a Gaussian distribution. Set the parameters as follows:

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

7 Use the Input Port block to define some of the physical characteristics of the filter. Set the parameters as follows:

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

8 Use the General Circuit Element blocks to import the circuit objects that model the filter components into the Simulink model.

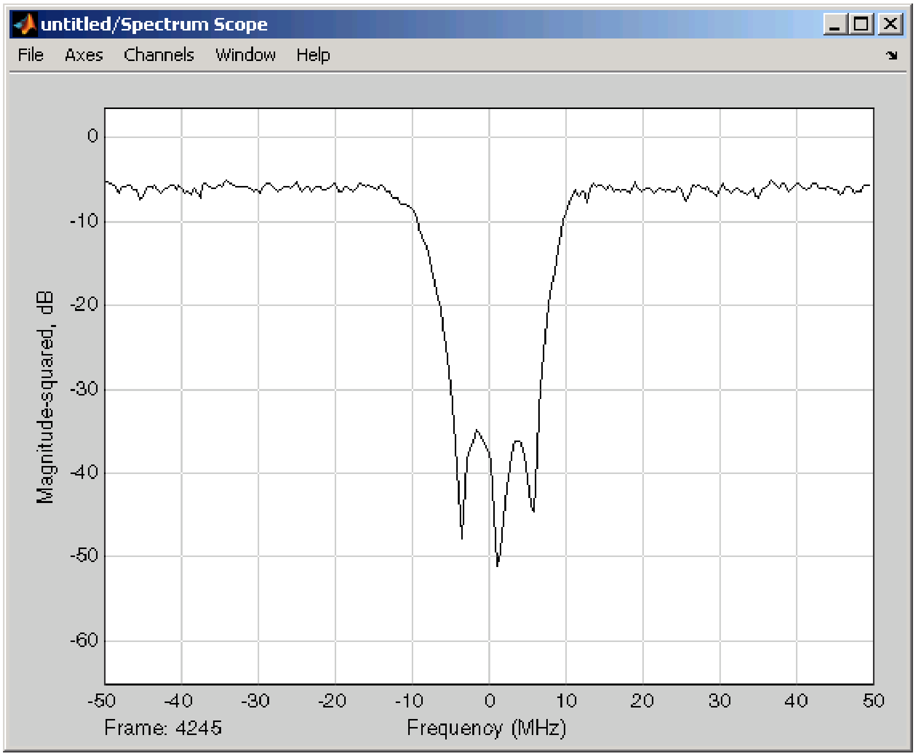
- In the General Circuit Element1 block dialog box, for the **RFCKT object** parameter, enter `ckt1`.
- In the General Circuit Element2 block dialog box, for the **RFCKT object** parameter, enter `ckt2`.
- In the General Circuit Element3 block dialog box, for the **RFCKT object** parameter, enter `ckt3`.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

9 Use the Output Port block to define the load impedance. Set this parameter to `50`.

10 Use the Spectrum Scope block to view the bandstop filter response. Set the parameters as follows:

- On the Scope Properties pane, set the **Number of spectral averages** parameter to `100`.
- On the Axes Properties pane, set the **Frequency range** parameter to `[-Fs/2 ... Fs/2]`, the **Minimum Y-limit** parameter to `-65`, and the **Maximum Y-limit** parameter to `3.6`.

11 Run the model and view the filter response in the Spectrum Scope window.



In this example, you modeled a 50 ohm bandstop filter by creating circuit objects in the MATLAB workspace and importing them into a Simulink model using the General Circuit Element block. For more information about the RF Toolbox functions used in this example, see the `rfckt.series`, `rfckt.parallel`, `rfckt.shuntrlc`, and `rfckt.seriesrlc` function reference pages in the RF Toolbox documentation.

Reference: Geffe, Philip R., “Novel designs for elliptic bandstop filters.” *RF Design*, February 1999.

4

Plotting Network Parameters

Plotting Network Parameters contains these sections.

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.

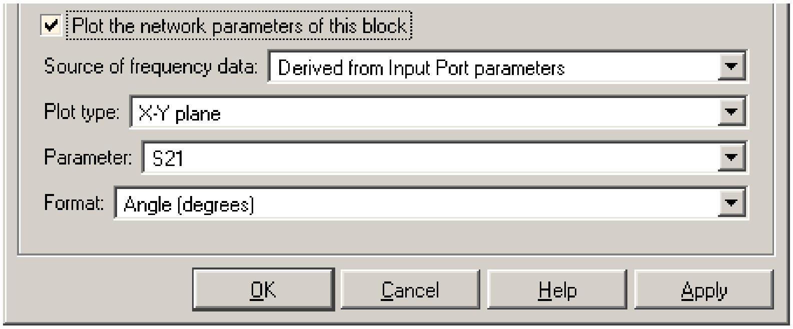
You can request a plot or change a plot whenever the model is not running. While the model is running, any existing plots continue to be displayed, but are not updated.

This section contains these topics.

- "Block Parameters" on page 4-2
- "Example of a Z Smith Chart" on page 4-5
- "Examples of Other Plot Types" on page 4-7

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

Source of frequency data: Derived from Input Port parameters

Plot type: X-Y plane

Parameter: S21

Format: Angle (degrees)

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.

Plot Type	Parameters	Formats
`Composite data`	Generates four separate plots in one figure. The predefined combination of plots differs based on the type of block.	

Plot Type	Parameters	Formats
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
Link budget	S11, S12, S21, S22	Magnitude (decibels) Magnitude (linear) Angle (degrees) Real Imaginary
	VSWRIn, VSWROut	Magnitude (decibels) None

Plot Type	Parameters	Formats
	OIP3 (Output Port block only)	dBm W mW
	NF (Output Port block only)	Magnitude (decibels) None
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

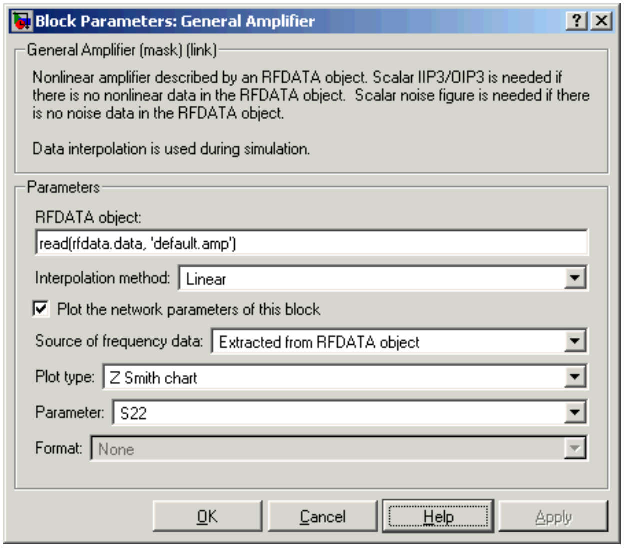
Note To display the legend(s) for a plot, type `legend show` at the command line.

Example of a Z Smith Chart

This example uses the RF Toolbox `read` function to create a data (`rfdata.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 `rfdata.data` object. See the RF Toolbox `rfdata` reference page for information about `rfdata` objects.

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



General Amplifier (mask) (link)

Nonlinear amplifier described by an RFDATA object. Scalar IIP3/OIP3 is needed if there is no nonlinear data in the RFDATA object. Scalar noise figure is needed if there is no noise data in the RFDATA object.

Data interpolation is used during simulation.

Parameters

RFDATA object: read(rfdata.data, 'default.amp')

Interpolation method: Linear

☑ Plot the network parameters of this block

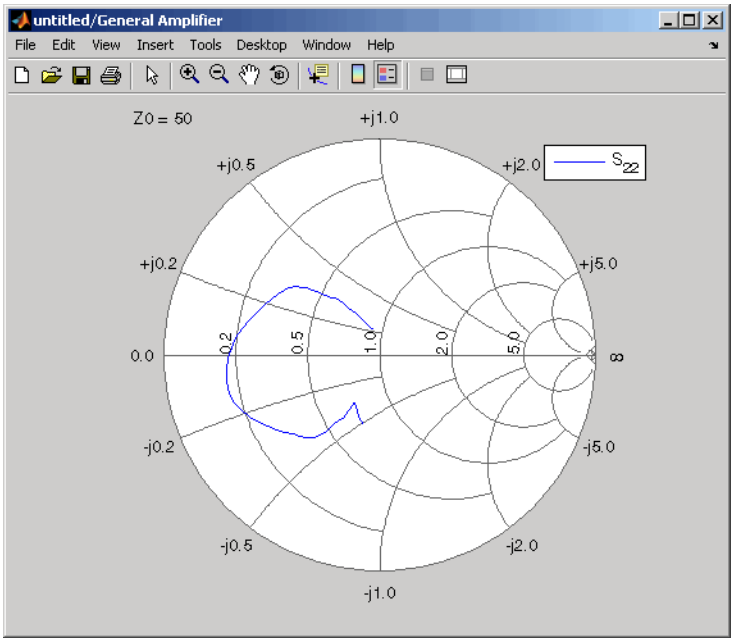
Source of frequency data: Extracted from RFDATA object

Plot type: Z Smith chart

Parameter: S22

Format: None

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

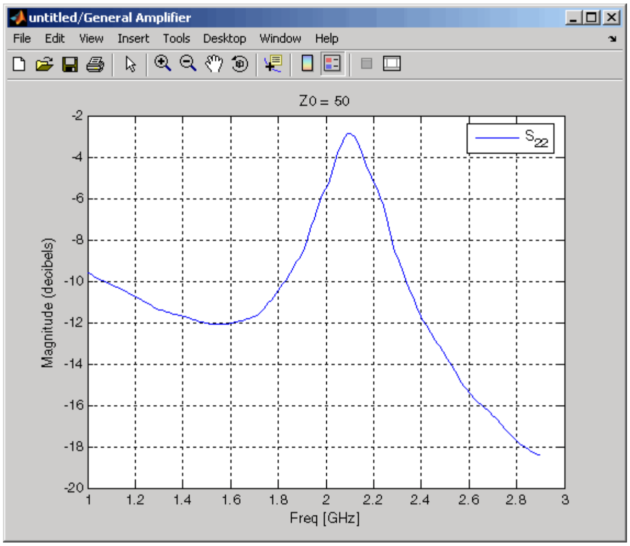


Note To display data tips for a plotted line, check `Data Cursor` on the **Tool** menu, and then click the data cursor on the plotted line to see the frequency and the plotted parameter at that point. See “Data Cursor — Displaying Data Values Interactively” in the MATLAB documentation for more information.

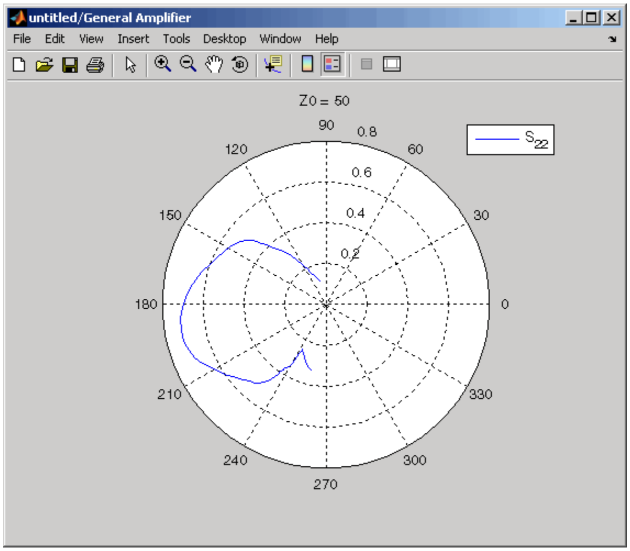
Examples of Other Plot Types

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

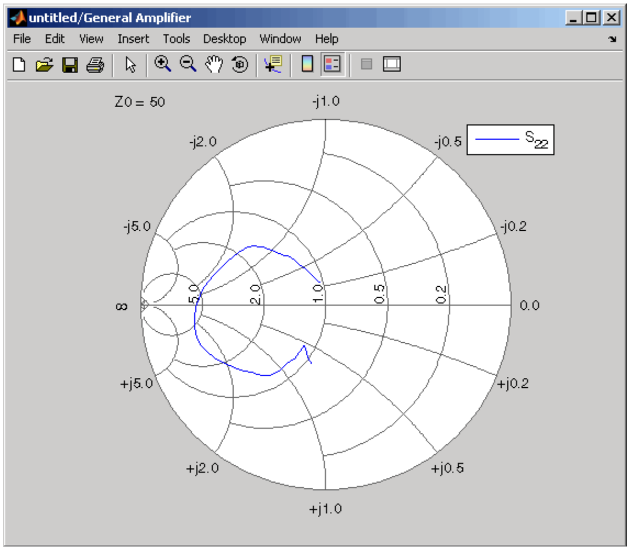
X-Y Plane Plot



Polar-Plane Plot

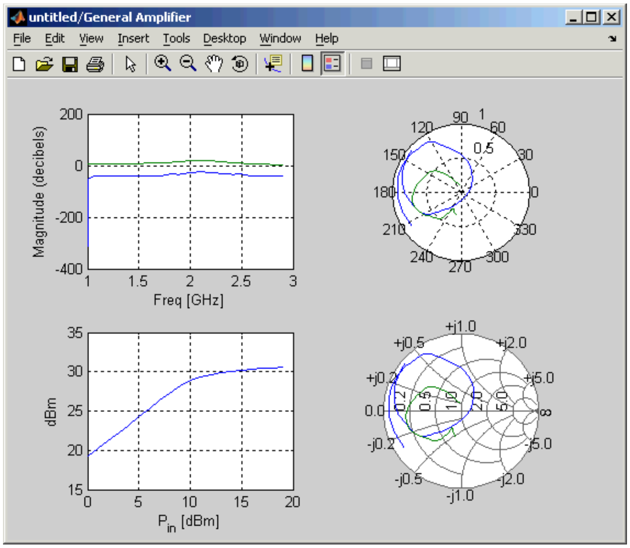

untitled/General Amplifier
File Edit View Insert Tools Desktop Window Help
Z0 = 50
S22
90
0.8
120
60
0.6
150
0.4
30
0.2
180
0
210
330
240
300
270

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.

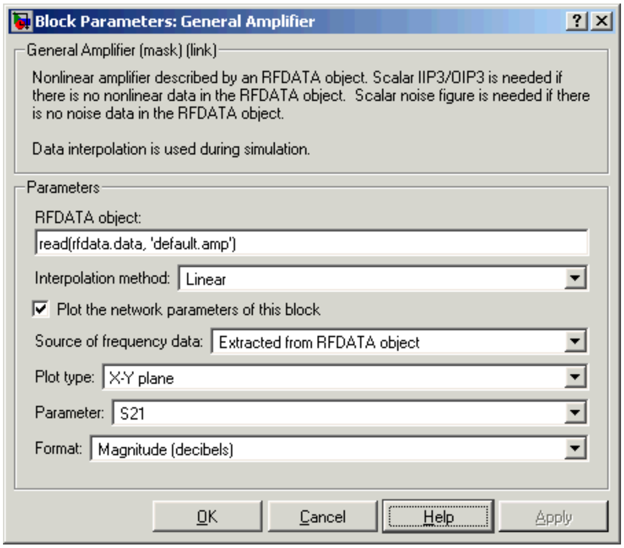
Block Parameter	Result of Change
Frequency range	The plot is redrawn using the new frequency range.
Plot type	The figure is redrawn in the new plot type. Other plot parameters retain their values if those values are valid for the new plot type. Else they revert to their default values.
Parameter	The 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 (`rfdata.data`) object.

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

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



Block Parameters: General Amplifier

General Amplifier (mask) (link)

Nonlinear amplifier described by an RFDATA object. Scalar IIP3/OIP3 is needed if there is no nonlinear data in the RFDATA object. Scalar noise figure is needed if there is no noise data in the RFDATA object.

Data interpolation is used during simulation.

Parameters

RFDATA object: read(rfdata.data, 'default.amp')

Interpolation method: Linear

☑ Plot the network parameters of this block

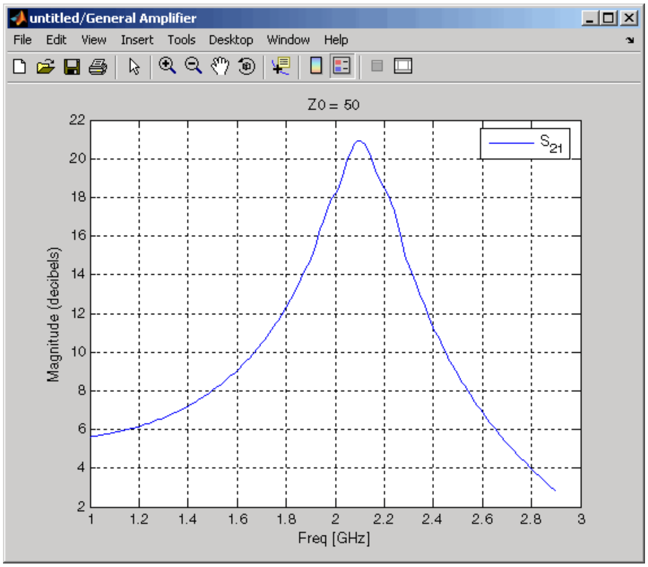
Source of frequency data: Extracted from RFDATA object

Plot type: X-Y plane

Parameter: S21

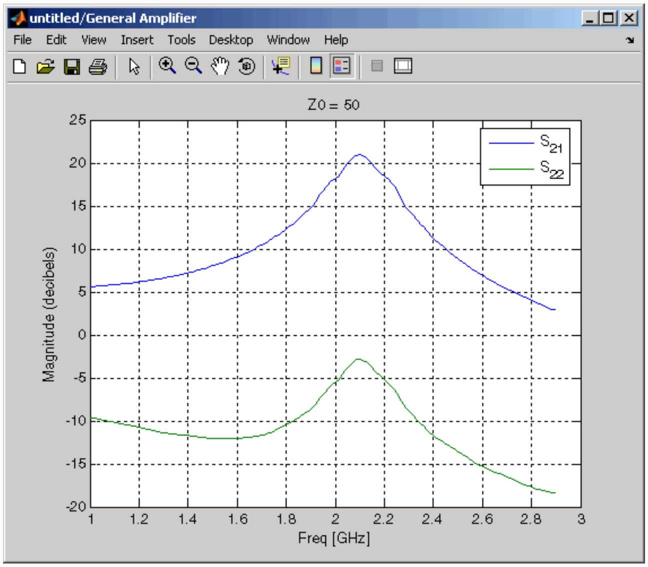
Format: Magnitude (decibels)

The block dialog above produces the plot below.

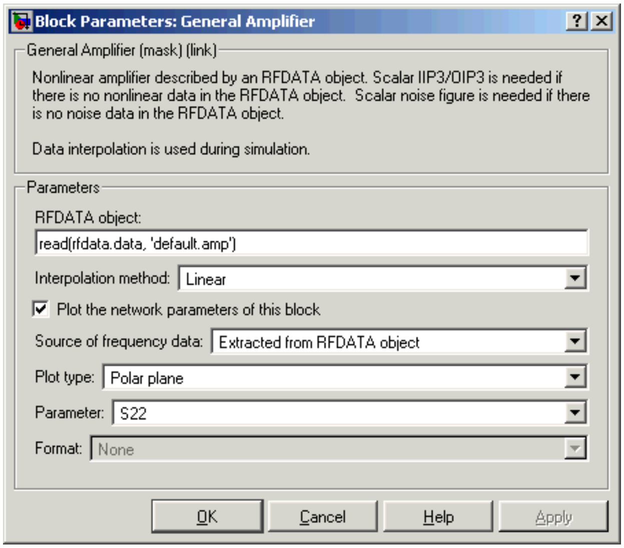


Add the S22 parameter to the plot by changing the value of **Parameter** to `S22`.

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 `S22`, the last parameter selected for the previous plot.



Block Parameters: General Amplifier

General Amplifier (mask) (link)

Nonlinear amplifier described by an RFDATA object. Scalar IIP3/OIP3 is needed if there is no nonlinear data in the RFDATA object. Scalar noise figure is needed if there is no noise data in the RFDATA object.

Data interpolation is used during simulation.

Parameters

RFDATA object:

read(rfdata.data, 'default.amp')

Interpolation method: Linear

☑ Plot the network parameters of this block

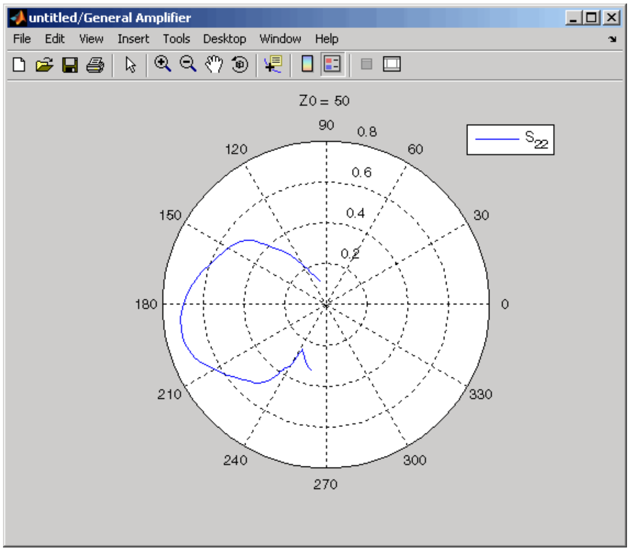
Source of frequency data: Extracted from RFDATA object

Plot type: Polar plane

Parameter: S22

Format: None

This produces the following plot.



5

Block Reference

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.

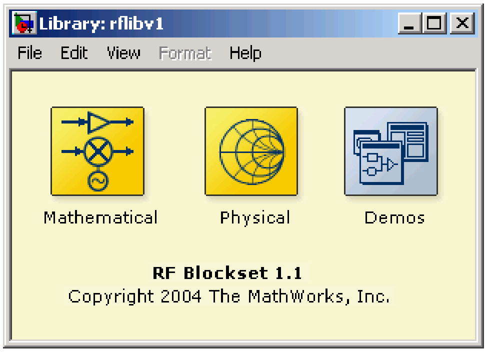
The main library is divided into these sublibraries:

- "Mathematical" on page 5-3 — Contains Simulink-style blocks that describe components in mathematical terms.
- "Physical" on page 5-4 — Contains blocks that model physical components in terms of their geometry and physical characteristics.

You can open the main library by typing

```
rflib
```

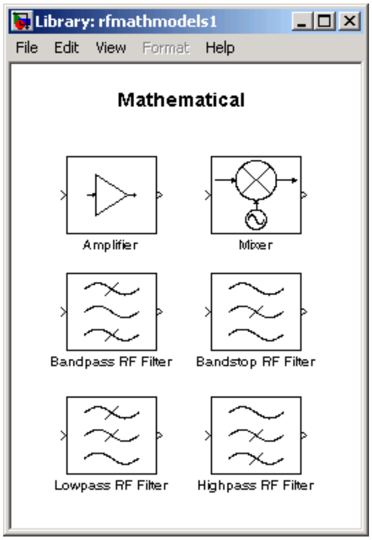
at the MATLAB prompt. Each yellow icon in the window represents a library. Double-click an icon to open the library.



RF Blockset 1.1
Copyright 2004 The MathWorks, Inc.

Mathematical

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



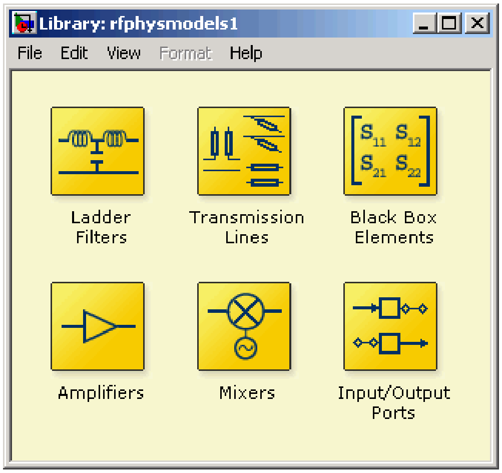
The table below lists the blocks in the Mathematical library. For information about a specific block, see its reference page.

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

Block Name (Continued)	Purpose
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.

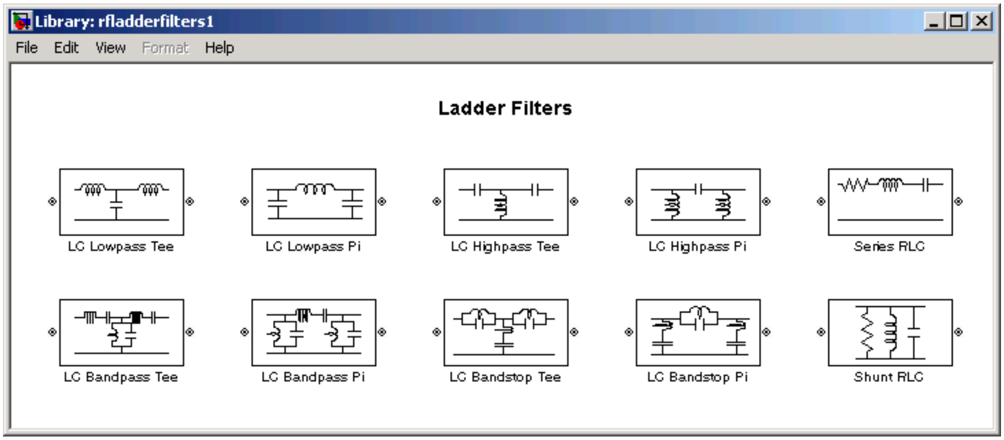


The Physical library contains these sublibraries:

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

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.

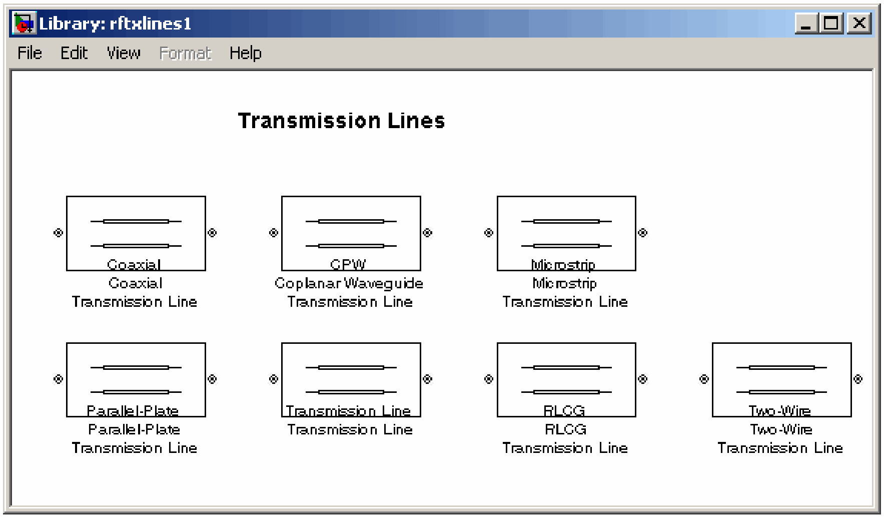


The table below lists the blocks in the Ladder Filters sublibrary. For information about a specific block, see its reference page.

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.

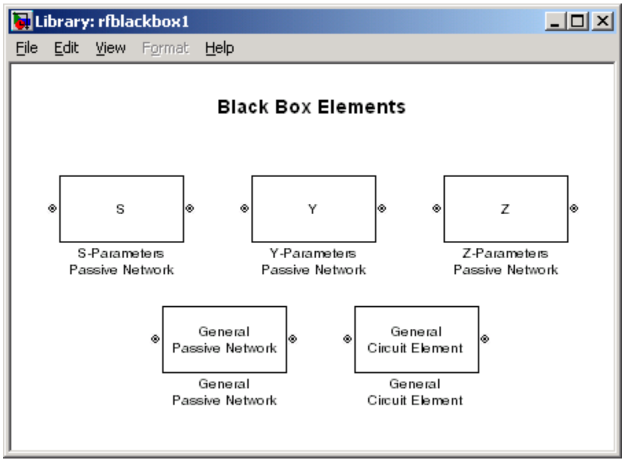

Library: rftxlines1
File Edit View Format Help
Transmission Lines
Coaxial
Coaxial Transmission Line
CPW
Coplanar Waveguide Transmission Line
Microstrip
Microstrip Transmission Line
Parallel-Plate
Parallel-Plate Transmission Line
Transmission Line
Transmission Line
RLCG
RLCG Transmission Line
Two-Wire
Two-Wire Transmission Line

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
Microstrip Transmission Line	Model a microstrip transmission line
Parallel-Plate Transmission Line	Model a parallel-plate transmission line
RLCG Transmission Line	Model RLCG transmission lines
Transmission Line	Model an abstract 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.

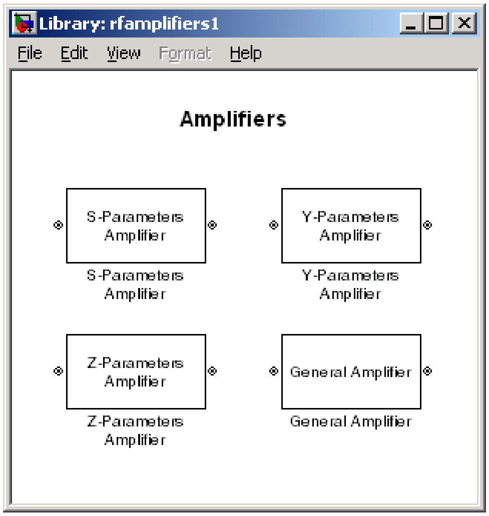


The table below lists the blocks in the Black Box Elements sublibrary. For information about a specific block, see its reference page.

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.

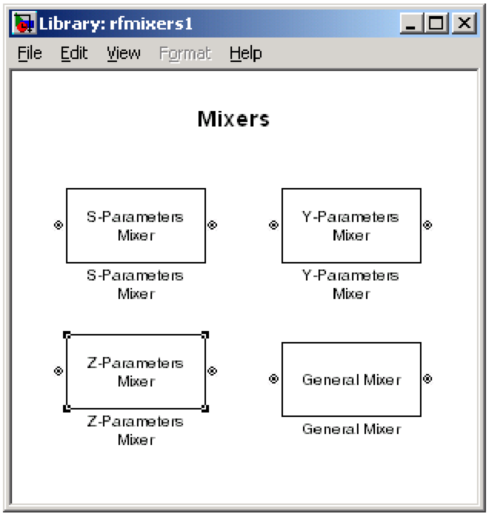


The table below lists the blocks in the Amplifiers sublibrary. For information about a specific block, see its reference page.

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.



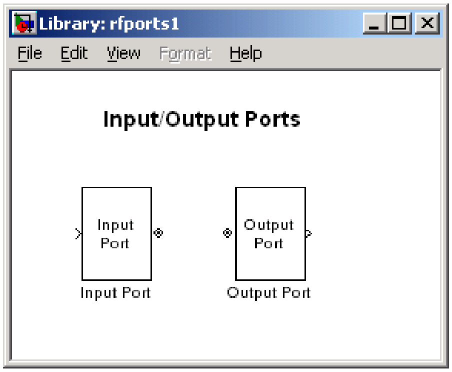
The table below lists the blocks in the Mixers sublibrary. For information about a specific block, see its reference page.

Block Name	Purpose
General Mixer	Model a Mixer described by an `rfdata.data` 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

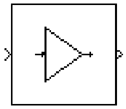
This section contains block reference pages listed alphabetically.

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.



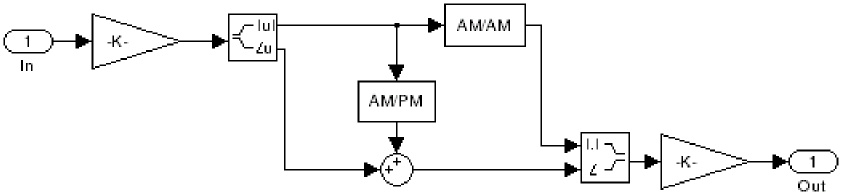
Note This block assumes a nominal impedance of 1 ohm.

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.

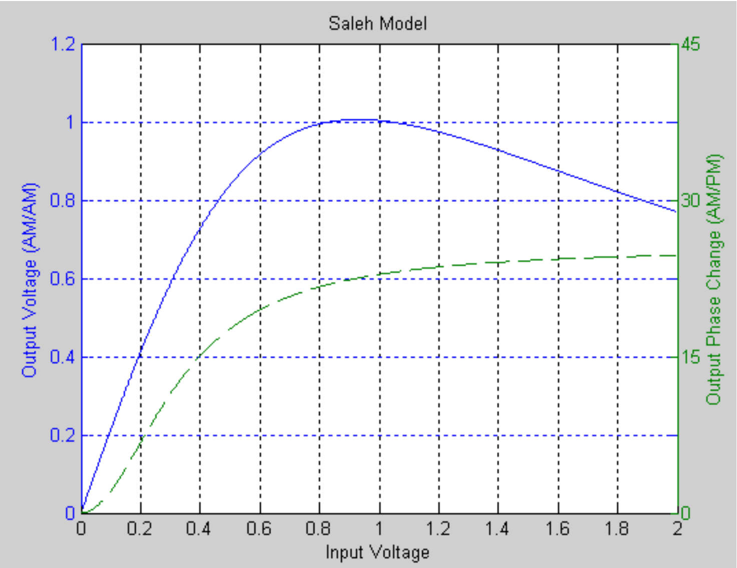
2. Splits the complex signal into its magnitude and angle components.
3. Applies an AM/AM conversion to the magnitude of the signal, according to the selected interpolation method, to produce the magnitude of the output signal.
4. Applies an AM/PM conversion to the phase of the signal, according to the 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.

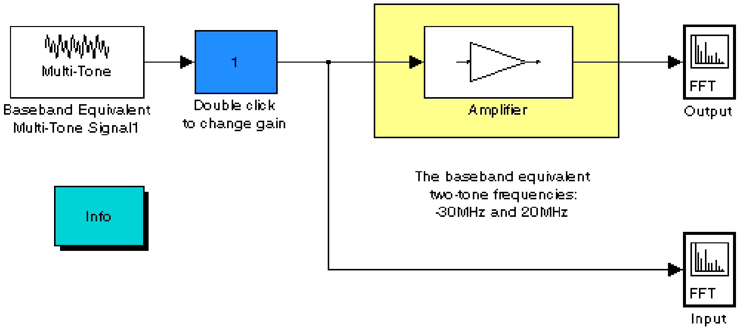
The following figure shows, for the Saleh method, plots of

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

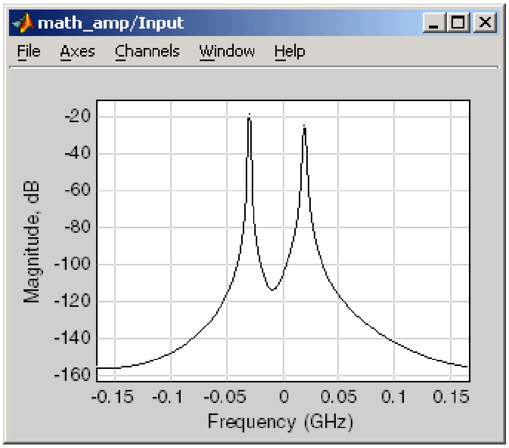


Effects of the Amplifier Block

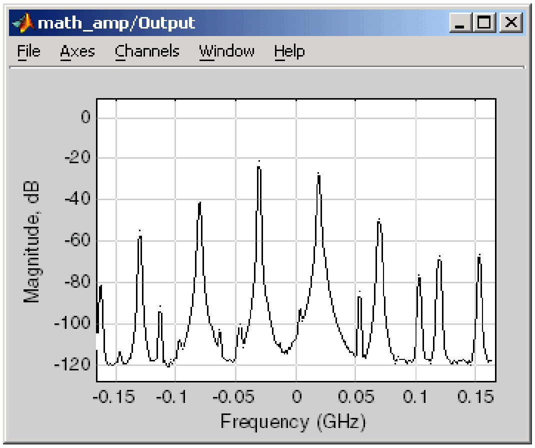
You can see the effect of the Amplifier block in the demo Intermodulation: Mathematical Amplifier.



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 Nonlinear Methods

The following sections discuss parameters specific to the following models:

- "Cubic Polynomial Model." on page 5-20
- "Hyperbolic Tangent Model." on page 5-20
- "Saleh Model" on page 5-21
- "Ghorbani Model" on page 5-21
- "Rapp Model" on page 5-22

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

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

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

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

where u is the magnitude of the scaled signal.

The **AM/PM conversion (degrees per dB)** parameter specifies the phase change. The **Linear gain (dB)** parameter scales the output signal.

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

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

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

where u is the magnitude of the scaled signal.

The **AM/PM conversion (degrees per dB)** parameter specifies the phase change. The **Linear gain (dB)** parameter scales the output signal.

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

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

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

where u is the magnitude of the scaled signal.

The AM/PM parameters, 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

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

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

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

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

Thermal Noise Simulation

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

- `Noise temperature` — specifies the noise in degrees kelvin.
- `Noise factor` — specifies the noise by the following equation

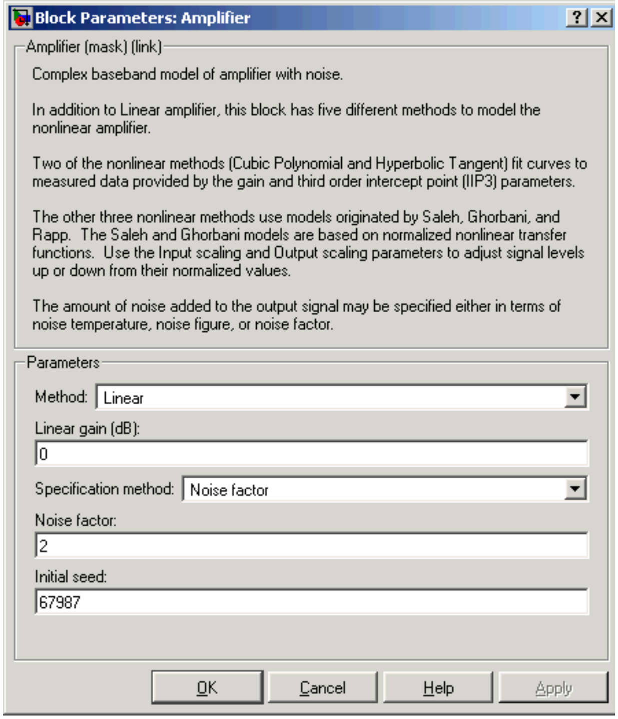
$$\text{Noise factor} = 1 + \frac{\text{Noise temperature}}{290}$$

- `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



Block Parameters: Amplifier

Amplifier (mask) (link)

Complex baseband model of amplifier with noise.

In addition to Linear amplifier, this block has five different methods to model the nonlinear amplifier.

Two of the nonlinear methods (Cubic Polynomial and Hyperbolic Tangent) fit curves to measured data provided by the gain and third order intercept point (IIP3) parameters.

The other three nonlinear methods use models originated by Saleh, Ghorbani, and Rapp. The Saleh and Ghorbani models are based on normalized nonlinear transfer functions. Use the Input scaling and Output scaling parameters to adjust signal levels up or down from their normalized values.

The amount of noise added to the output signal may be specified either in terms of noise temperature, noise figure, or noise factor.

Parameters

Method: Linear

Linear gain (dB): 0

Specification method: Noise factor

Noise factor: 2

Initial seed: 67987

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. For both of these methods, the nominal impedance is 1 ohm. Tunable.

AM/PM conversion (degrees per dB)

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

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

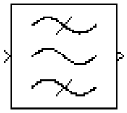
Purpose

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.

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

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

Note This block assumes a nominal impedance of 1 ohm.

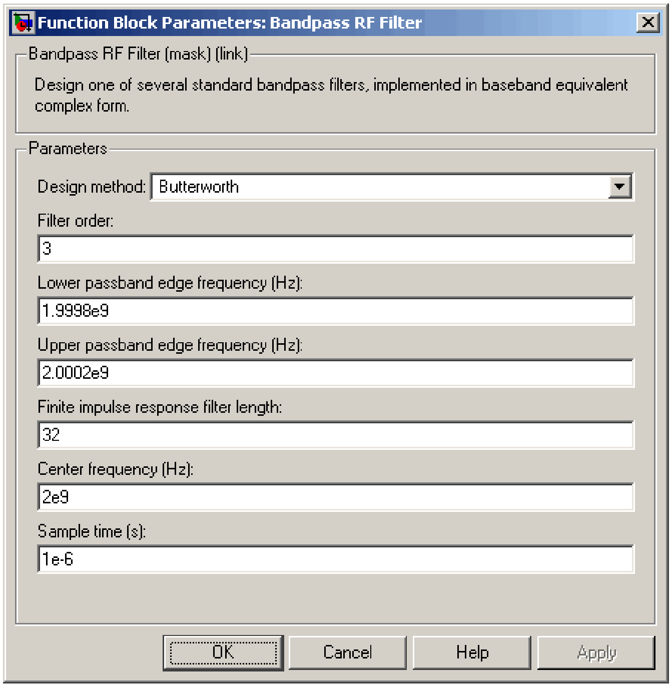
Select the design of the filter from the **Design method** list in the dialog box. For each design method, the block enables you to specify the filter design parameters shown in the following table.

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

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

Note Some RF blocks require the sample time to perform baseband modeling calculations. To ensure the accuracy of these calculations, the Input Port block, as well as the mathematical RF blocks, compare the input sample time 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



Function Block Parameters: Bandpass RF Filter

Bandpass RF Filter (mask) (link)

Design one of several standard bandpass filters, implemented in baseband equivalent complex form.

Parameters

Design method: Butterworth

Filter order: 3

Lower passband edge frequency (Hz): 1.9998e9

Upper passband edge frequency (Hz): 2.0002e9

Finite impulse response filter length: 32

Center frequency (Hz): 2e9

Sample time (s): 1e-6

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.

Finite impulse response filter length

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)

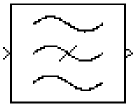
Purpose

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.

Note This block assumes a nominal impedance of 1 ohm.

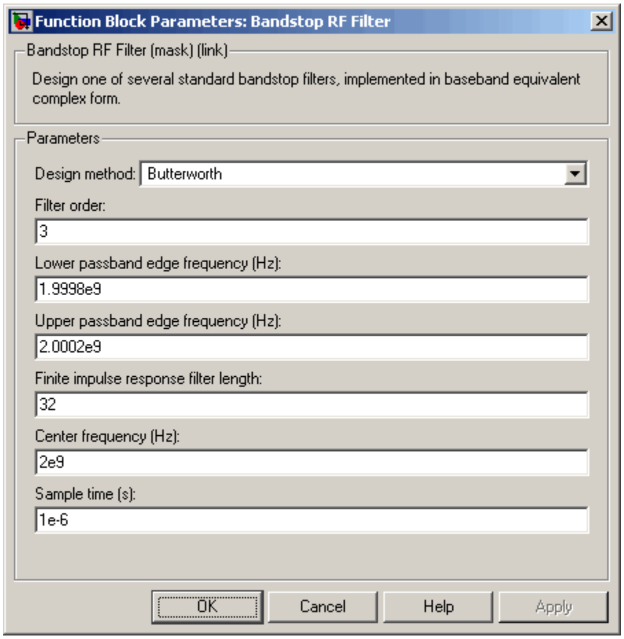
Select the design of the filter from the **Design method** list in the dialog box. For each design method, the block enables you to specify the filter design parameters shown in the following table.

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

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

Note Some RF blocks require the sample time to perform baseband modeling calculations. To ensure the accuracy of these calculations, the Input Port block, as well as the mathematical RF blocks, compare the input sample time 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



Function Block Parameters: Bandstop RF Filter

Bandstop RF Filter (mask) (link)

Design one of several standard bandstop filters, implemented in baseband equivalent complex form.

Parameters

Design method: Butterworth

Filter order: 3

Lower passband edge frequency (Hz): 1.9998e9

Upper passband edge frequency (Hz): 2.0002e9

Finite impulse response filter length: 32

Center frequency (Hz): 2e9

Sample time (s): 1e-6

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.

Finite impulse response filter length

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

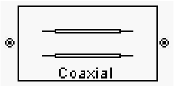
Purpose

Model a coaxial transmission line

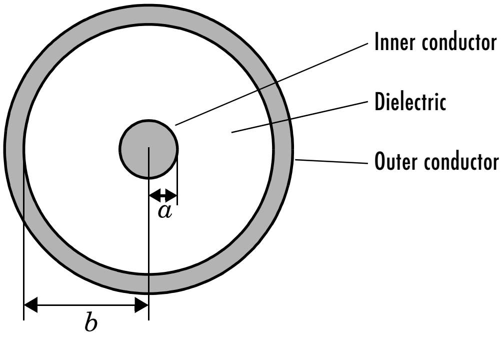
Library

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

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

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

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

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

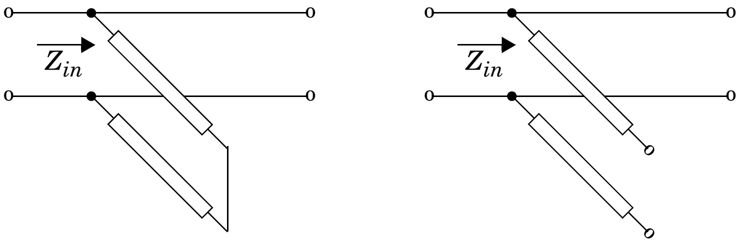
In these equations, σ_{cond} is the conductivity in the conductor and σ_{diel} is the conductivity in the dielectric. μ is the permeability of the dielectric, ε is its permittivity, and skin depth δ is calculated as $1/\sqrt{\pi f\mu\sigma_{\text{cond}}}$. f is a vector of simulation frequencies for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

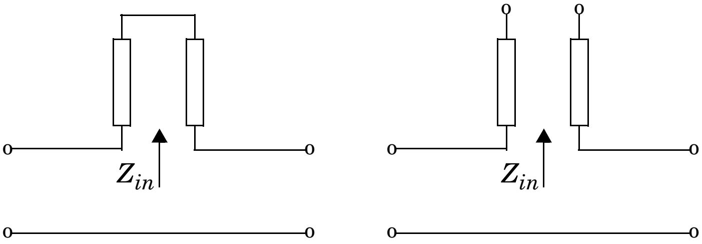
Shunt ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Shunt`, the 2-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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

$$A = 1$$
$$B = 0$$
$$C = 1/Z_{in}$$
$$D = 1$$

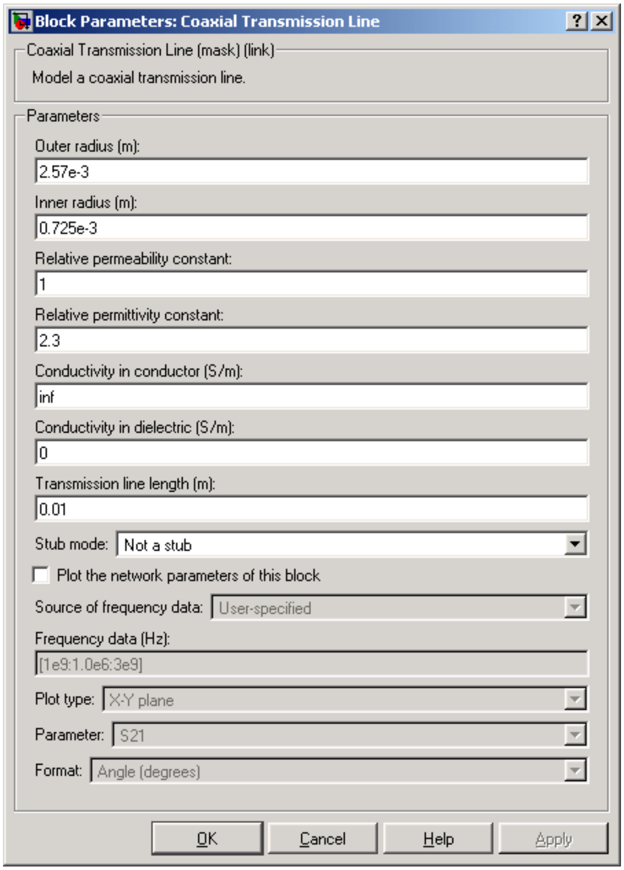
Series ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Series`, the 2-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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



Block Parameters: Coaxial Transmission Line

Coaxial Transmission Line (mask) (link)

Model a coaxial transmission line.

Parameters

Outer radius (m): 2.57e-3

Inner radius (m): 0.725e-3

Relative permeability constant: 1

Relative permittivity constant: 2.3

Conductivity in conductor (S/m): inf

Conductivity in dielectric (S/m): 0

Transmission line length (m): 0.01

Stub mode: Not a stub

□ Plot the 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)

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.

Relative permeability constant

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

Relative permittivity constant

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

Conductivity in conductor (S/m)

Conductivity of the conductor in siemens per meter.

Conductivity in dielectric (S/m)

Conductivity of the dielectric in siemens per meter.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

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

Termination of stub

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

Note For information about plotting, see “Plotting Network Parameters” on page 4-1.

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

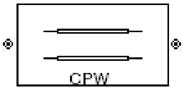
Purpose

Model a coplanar waveguide transmission line

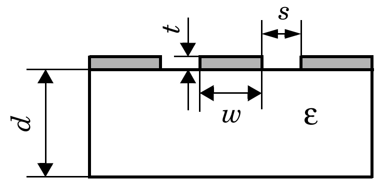
Library

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 (ε).



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 α_a is the attenuation coefficient and β is the wave number. The attenuation coefficient α_a is related to the loss, α, by

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

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

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

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

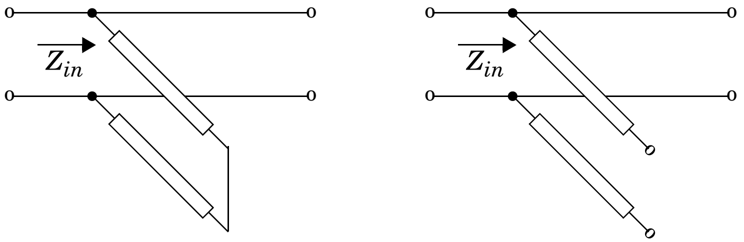
$V_P = c/\sqrt{\varepsilon_{\mathrm{eff}}}$ where $\varepsilon_{\mathrm{eff}}$ is the frequency dependent effective dielectric constant. f is a vector of simulation frequencies for the specified parameters. See the Output Port block for information about determining the simulation frequencies. The phase 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.

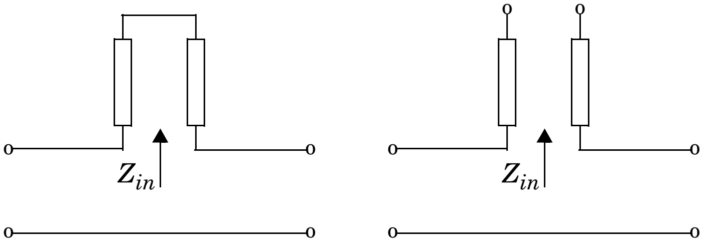
Shunt ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Shunt`, the 2-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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

$$A = 1$$
$$B = 0$$
$$C = 1/Z_{in}$$
$$D = 1$$

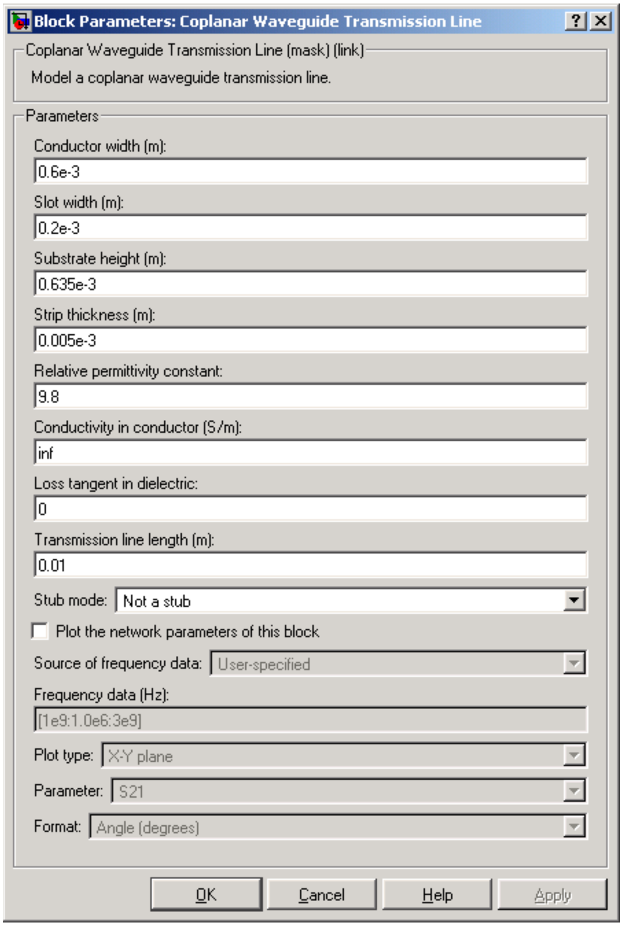
Series ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Series`, the 2-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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



Block Parameters: Coplanar Waveguide Transmission Line

Coplanar Waveguide Transmission Line (mask) (link)

Model a coplanar waveguide transmission line.

Parameters

Conductor width (m): 0.6e-3

Slot width (m): 0.2e-3

Substrate height (m): 0.635e-3

Strip thickness (m): 0.005e-3

Relative permittivity constant: 9.8

Conductivity in conductor (S/m): inf

Loss tangent in dielectric: 0

Transmission line length (m): 0.01

Stub mode: Not a stub

□ Plot the 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)

Conductor width (m)

Physical width of the conductor.

Slot width (m)

Physical width of the slot.

Substrate height (m)

Thickness of the dielectric on which the conductor resides.

Strip thickness (m)

Physical thickness of the conductor.

Relative permittivity constant

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

Conductivity in conductor (S/m)

Conductivity of the conductor in siemens per meter.

Loss tangent in dielectric

Loss angle tangent of the dielectric.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

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

Termination of stub

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

Note For information about plotting, see "Plotting Network Parameters" on page 4-1.

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

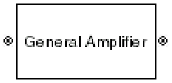
Purpose

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 block contains network Y- or Z-parameters, the block first converts them to S-parameters. See “S-Parameters at Simulation Frequencies” on page 3-2 for more details.

Nonlinearity

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

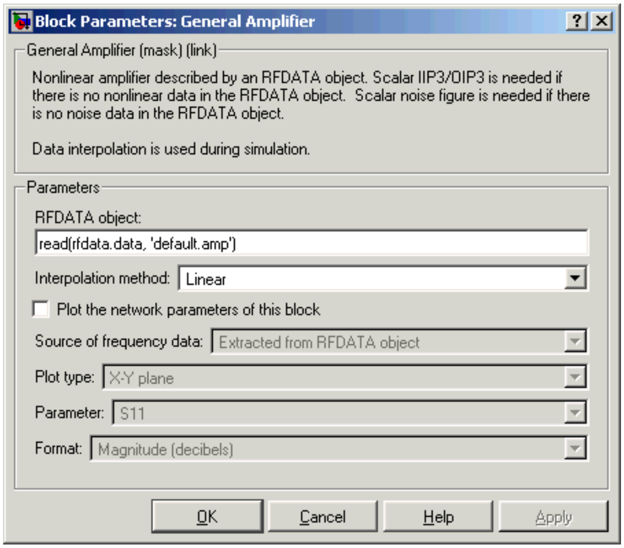
If the `rfdata.data` object contains no power data, the General Amplifier block dialog enables you to enter either the OIP3 or IIP3 as a scalar value for nonlinearity.

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.

Dialog Box



Block Parameters: General Amplifier
General Amplifier (mask) (link)

Nonlinear amplifier described by an RFDATA object. Scalar IIP3/OIP3 is needed if there is no nonlinear data in the RFDATA object. Scalar noise figure is needed if there is no noise data in the RFDATA object.

Data interpolation is used during simulation.

RFDATA object: read(rfdata.data, 'default.amp')

Interpolation method: Linear

□ Plot the network parameters of this block

Source of frequency data: Extracted from RFDATA object

Plot type: X-Y plane

Parameter: S11

Format: Magnitude (decibels)

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

For network data, the method used to interpolate the parameters contained in the `rfdata.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 `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 only if the rfdata.data object contains no power data.

OIP3 (dBm)

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

Noise figure (dB)

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

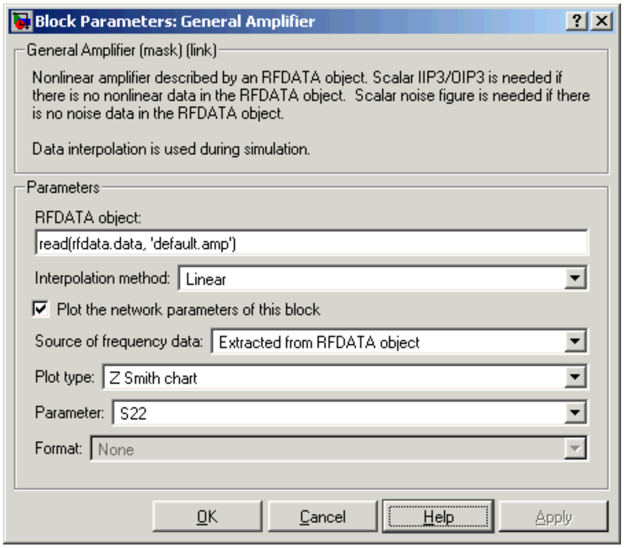
Note For information about plotting the network parameters, see “Plotting Network Parameters” on page 4-1. Use RFTool or the RF Toolbox plotting functions to plot other data.

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.



Block Parameters: General Amplifier

General Amplifier (mask) (link)

Nonlinear amplifier described by an RFDATA object. Scalar IIP3/OIP3 is needed if there is no nonlinear data in the RFDATA object. Scalar noise figure is needed if there is no noise data in the RFDATA object.

Data interpolation is used during simulation.

Parameters

RFDATA object: read(rfdata.data, 'default.amp')

Interpolation method: Linear

☑ Plot the network parameters of this block

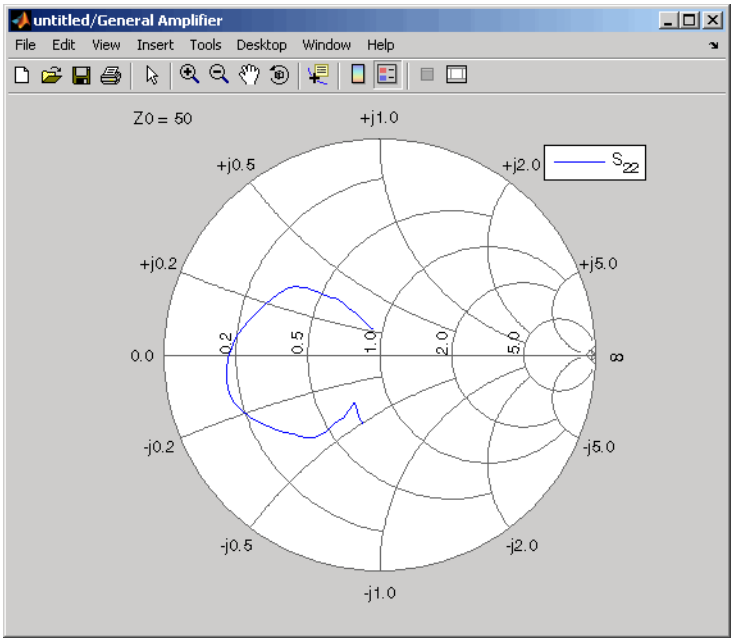
Source of frequency data: Extracted from RFDATA object

Plot type: Z Smith chart

Parameter: S22

Format: None

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

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

`rfdata`, `rfdata.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

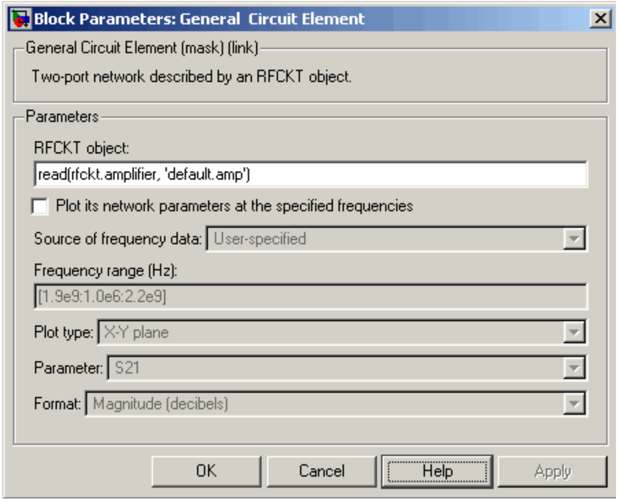
Description

General
Circuit Element

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


Block Parameters: General Circuit Element

General Circuit Element (mask) (link)

Two-port network described by an RFCKT object.

Parameters

RFCKT object: read(rfckt.amplifier, 'default.amp')

□ Plot its network parameters at the specified frequencies

Source of frequency data: User-specified

Frequency range (Hz): [1.9e9:1.0e6:2.2e9]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

RFCKT object

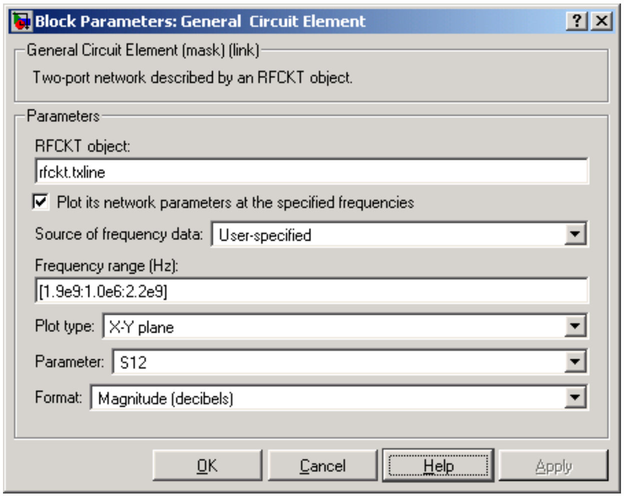
An RF Toolbox circuit (`rfckt`) object. You can specify the object as (1) the handle of a circuit object previously created using the RF Toolbox, (2) an RF Toolbox command such as `rfckt.txline`, `rfckt.coaxial`, or `rfckt.cascade` which creates a default circuit object of the specified type, or (3) 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 4-1.

Examples

Creating a General Circuit Element from an RF Toolbox Object

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



Block Parameters: General Circuit Element

General Circuit Element (mask) (link)

Two-port network described by an RFCKT object.

Parameters

RFCKT object: rfckt.txline

☑ Plot its network parameters at the specified frequencies

Source of frequency data: User-specified

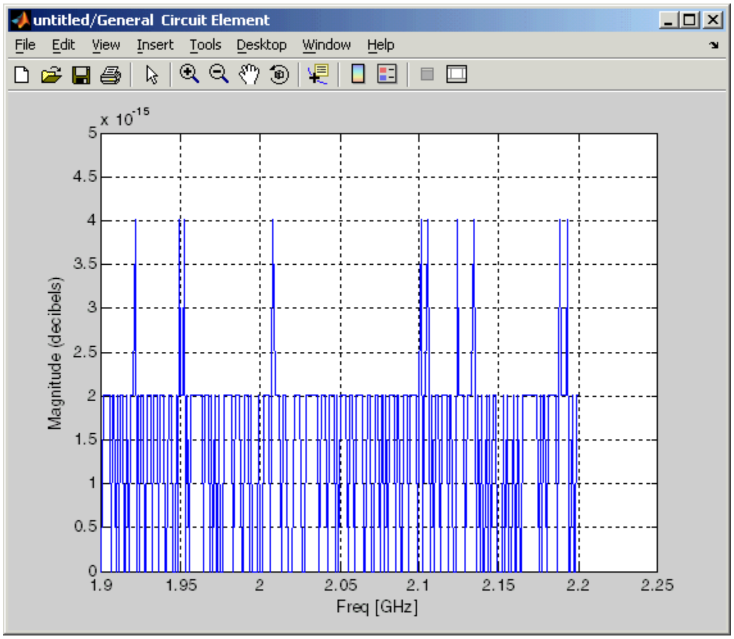
Frequency range (Hz): [1.9e9:1.0e6:2.2e9]

Plot type: X-Y plane

Parameter: S12

Format: Magnitude (decibels)

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.



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 block contains network Y- or Z-parameters, the block first converts them to S-parameters. See "S-Parameters at Simulation Frequencies" on page 3-2 for more details.

Active Noise

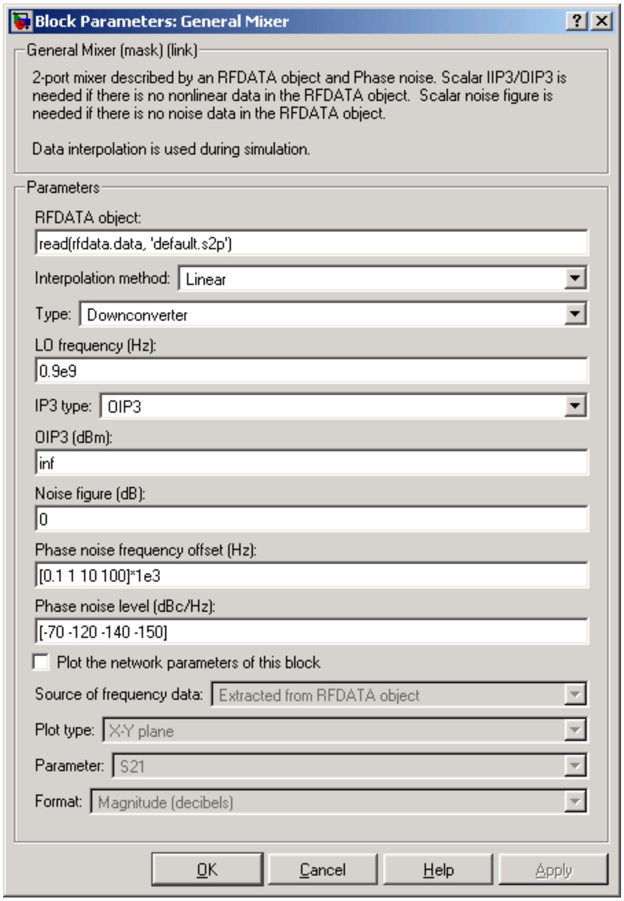
If active spot noise data exists in the `rfdata.data` object, the block uses the data to calculate the noise figure. It first interpolates the noise data for the simulation frequencies, using the specified **Interpolation method**. It then calculates the noise figure using the resulting values.

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



Block Parameters: General Mixer

General Mixer (mask) (link)

2-port mixer described by an RFDATA object and Phase noise. Scalar IIP3/OIP3 is needed if there is no nonlinear data in the RFDATA object. Scalar noise figure is needed if there is no noise data in the RFDATA object.

Data interpolation is used during simulation.

Parameters

RFDATA object: read(rfdata.data, 'default.s2p')

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: Extracted from RFDATA object

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

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.

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 4-1. Use RFTool or the RF Toolbox plotting functions to plot other data.

See Also

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

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

Purpose

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

Library

Black Box Elements sublibrary of the Physical library

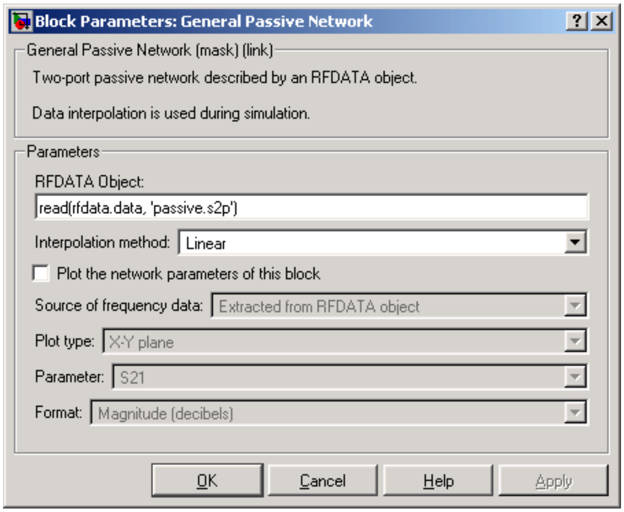
Description

General
Passive Network

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.

If network parameter data and their corresponding frequencies exist as S-parameters in the `rfdata.data` object, the General Passive Network block interpolates the S-parameters to determine the S-parameters at the simulation frequencies. If the block contains network Y- or Z-parameters, the block first converts them to S-parameters. See “S-Parameters at Simulation Frequencies” on page 3-2 for more details.

Dialog Box



Block Parameters: General Passive Network

General Passive Network (mask) (link)

Two-port passive network described by an RFDATA object.

Data interpolation is used during simulation.

Parameters

RFDATA Object: read(rfdata.data, 'passive.s2p')

Interpolation method: Linear

□ Plot the network parameters of this block

Source of frequency data: Extracted from RFDATA object

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

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

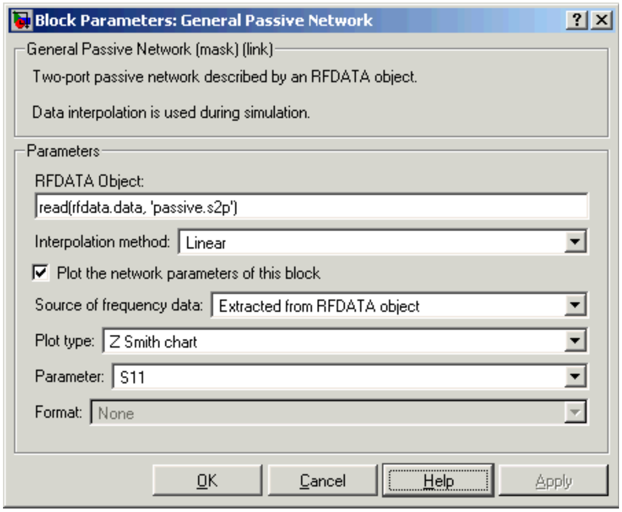
Method used to interpolate the parameters contained in the `rfdata` object. Interpolation can be `Linear` (default), `Spline`, or `Cubic`.

Note For information about plotting, see "Plotting Network Parameters" on page 4-1.

Examples

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



Block Parameters: General Passive Network

General Passive Network (mask) (link)

Two-port passive network described by an RFDATA object.

Data interpolation is used during simulation.

Parameters

RFDATA Object: read(rfdata.data, 'passive.s2p')

Interpolation method: Linear

☑ Plot the network parameters of this block

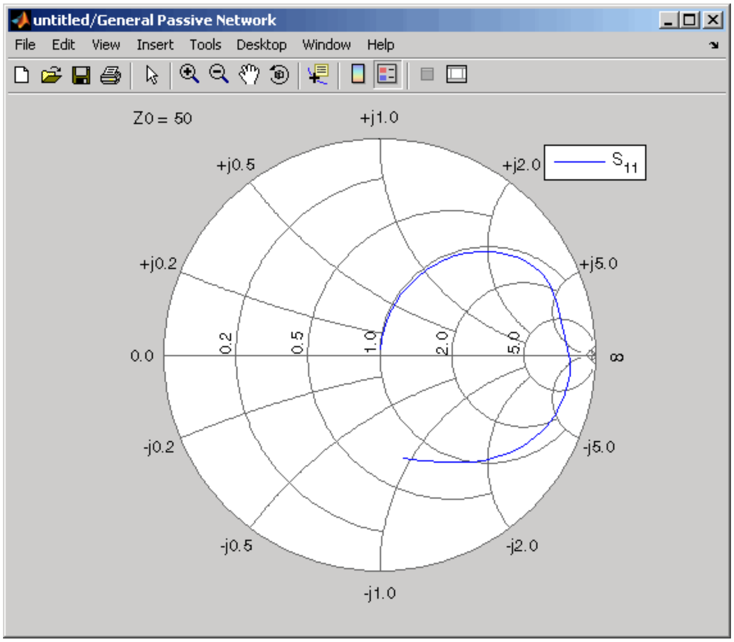
Source of frequency data: Extracted from RFDATA object

Plot type: Z Smith chart

Parameter: S11

Format: None

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


untitled/General Passive Network
File Edit View Insert Tools Desktop Window Help
Z0 = 50
+j1.0
+j0.5
+j2.0
S11
+j0.2
+j5.0
0.0
0.2
0.5
1.0
2.0
5.0
∞
-j0.2
-j5.0
-j0.5
-j2.0
-j1.0

See Also

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

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

`interp1` (MATLAB)

Highpass RF Filter

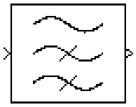
Purpose

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.

Note This block assumes a nominal impedance of 1 ohm.

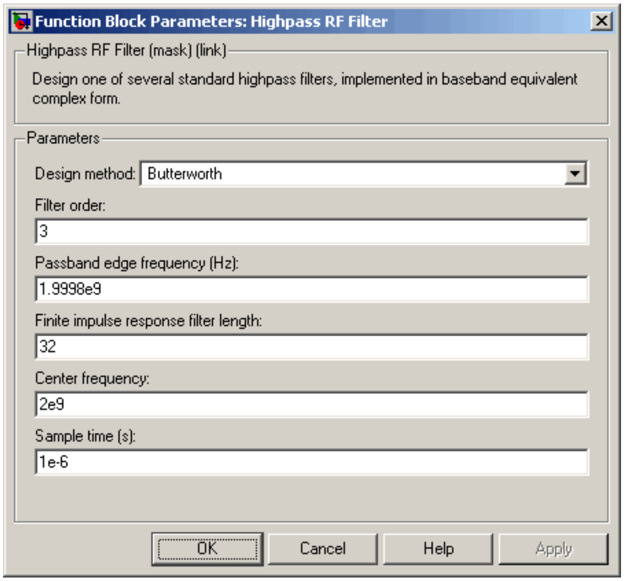
Select the design of the filter from the **Design method** list in the dialog box. For each design method, the block enables you to specify the filter design parameters shown in the following table.

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

The 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.

Dialog Box



Function Block Parameters: Highpass RF Filter

Highpass RF Filter (mask) (link)

Design one of several standard highpass filters, implemented in baseband equivalent complex form.

Parameters

Design method: Butterworth

Filter order: 3

Passband edge frequency (Hz): 1.9998e9

Finite impulse response filter length: 32

Center frequency: 2e9

Sample time (s): 1e-6

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.

Finite impulse response filter length

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`, `besselap` (Signal Processing Toolbox)

Input Port

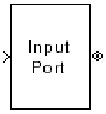
Purpose

Connection block from Simulink environment to RF physical blocks

Library

Input/Output Ports sublibrary of the Physical library

Description

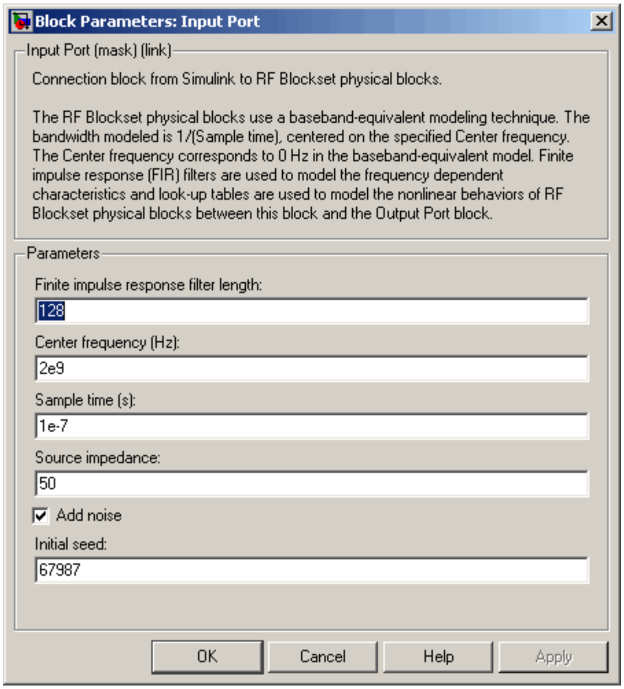


The Input Port block serves as a connecting port from the Simulink, or mathematical, part of the model to an RF physical part of the model. The Input Port block enables you to provide the parameter data needed to calculate the simulation frequencies and the baseband equivalent impulse response for the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

For more information about connecting mathematical and physical parts of a model, see "Creating 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



Input Port (mask) (link)

Connection block from Simulink to RF Blockset physical blocks.

The RF Blockset physical blocks use a baseband-equivalent modeling technique. The bandwidth modeled is 1/(Sample time), centered on the specified Center frequency. The Center frequency corresponds to 0 Hz in the baseband-equivalent model. Finite impulse response (FIR) filters are used to model the frequency dependent characteristics and look-up tables are used to model the nonlinear behaviors of RF Blockset physical blocks between this block and the Output Port block.

Parameters

Finite impulse response filter length: 128

Center frequency (Hz): 2e9

Sample time (s): 1e-7

Source impedance: 50

☑ Add noise

Initial seed: 67987

Finite impulse response filter length

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.

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

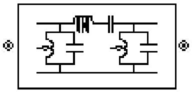
Purpose

Model an LC bandpass pi network

Library

Ladder Filters sublibrary of the Physical library

Description

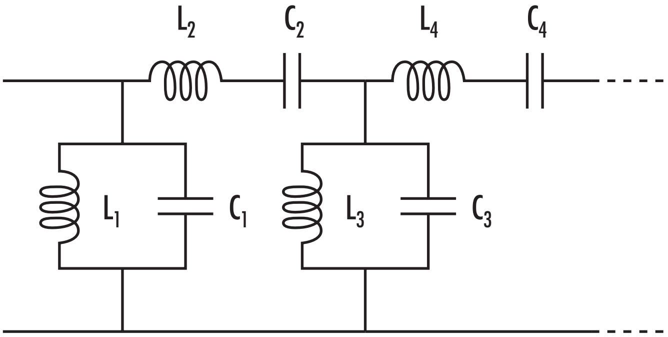


The LC Bandpass Pi block models the LC bandpass pi network described in the block dialog box, in terms of its frequency-dependent S-parameters.

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.

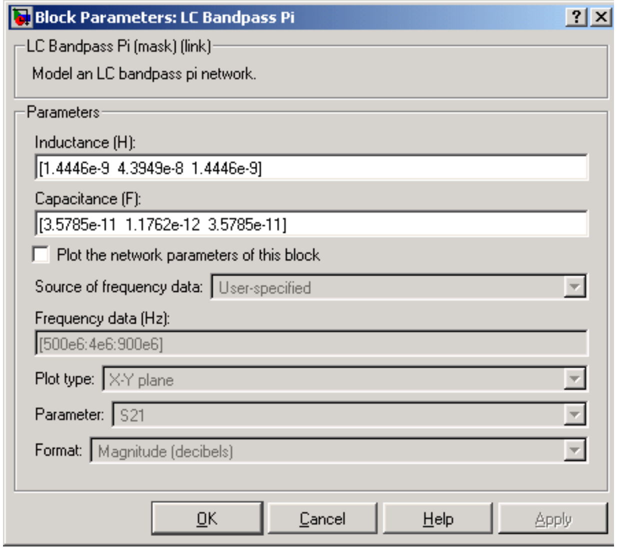
The LC Bandpass Pi block cascades the ABCD-parameters for each series and shunt pair, and then converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

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



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

Dialog Box



Block Parameters: LC Bandpass Pi

LC Bandpass Pi (mask) (link)

Model an LC bandpass pi network.

Parameters

Inductance (H): [1.4446e-9 4.3949e-8 1.4446e-9]

Capacitance (F): [3.5785e-11 1.1762e-12 3.5785e-11]

□ Plot the network parameters of this block

Source of frequency data: User-specified

Frequency data (Hz): [500e6:4e6:900e6]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

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)

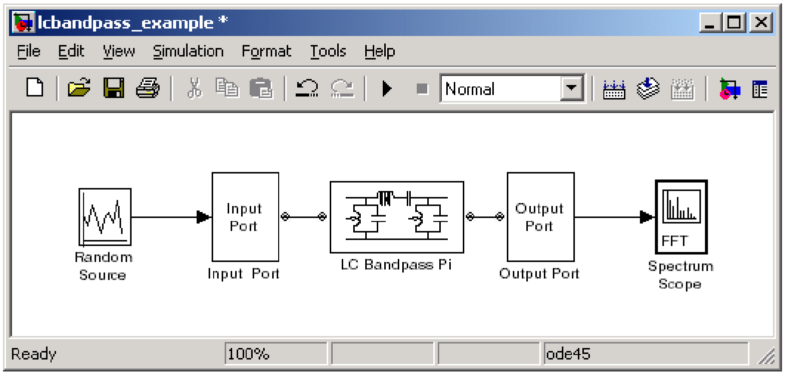
Vector containing the capacitances, in order from source to load, of all capacitors in the network. Its length must be equal to the length of the vector you provide in the **Inductance** parameter. All values must be strictly positive.

Note For information about plotting, see "Plotting Network Parameters" on page 4-1.

Examples

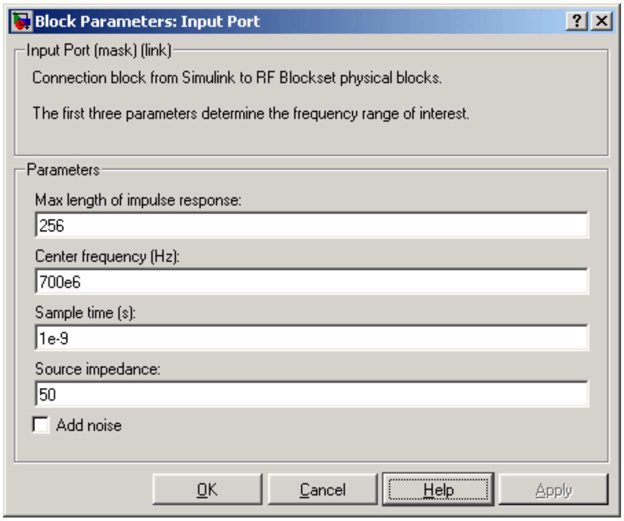
Using a Ladder Filter Block to Filter Gaussian Noise

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-9. This sample time must match the sample time for the physical part of the model, which you provide in the Input Port block diagram.

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



Input Port (mask) (link)

Connection block from Simulink to RF Blockset physical blocks.

The first three parameters determine the frequency range of interest.

Parameters

Max length of impulse response:
256

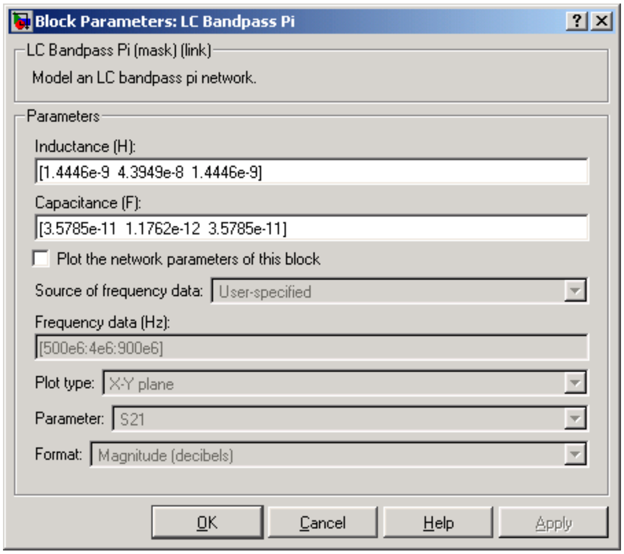
Center frequency (Hz):
700e6

Sample time (s):
1e-9

Source impedance:
50

□ Add noise

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].



Block Parameters: LC Bandpass Pi

LC Bandpass Pi (mask) (link)

Model an LC bandpass pi network.

Parameters

Inductance (H): [1.4446e-9 4.3949e-8 1.4446e-9]

Capacitance (F): [3.5785e-11 1.1762e-12 3.5785e-11]

□ Plot the network parameters of this block

Source of frequency data: User-specified

Frequency data (Hz): [500e6:4e6:900e6]

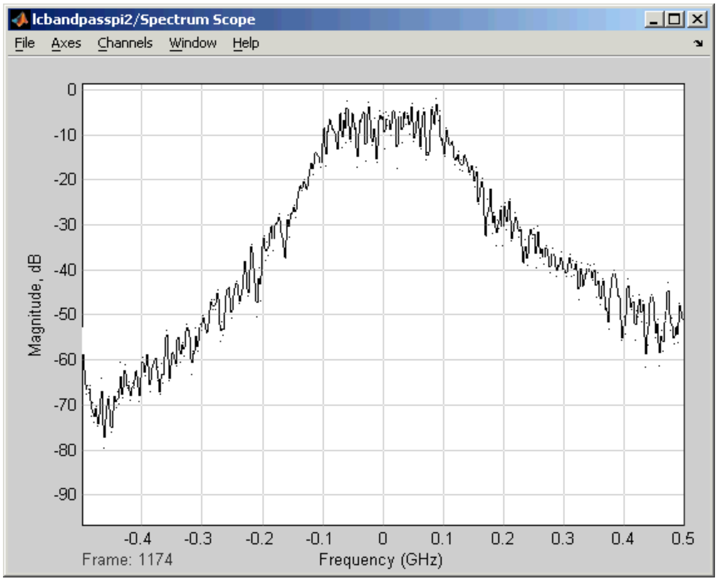
Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

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/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. The **Minimum Y-limit** parameter is -90, and **Maximum Y-limit** is 0.



References

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

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

See Also

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

LC Bandpass Tee

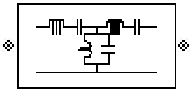
Purpose

Model an LC bandpass tee network

Library

Ladder Filters sublibrary of the Physical library

Description

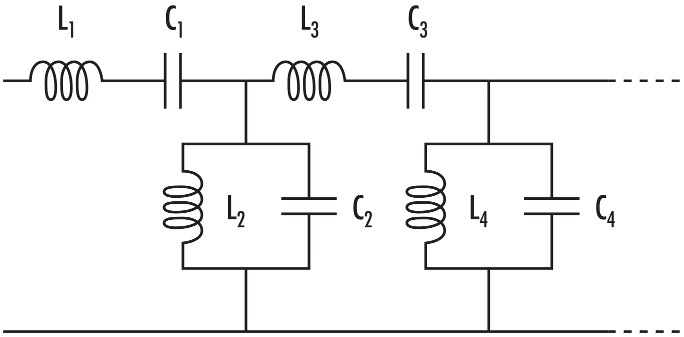


The LC Bandpass Tee block models the LC bandpass tee network described in the block dialog box, in terms of its frequency-dependent S-parameters.

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.

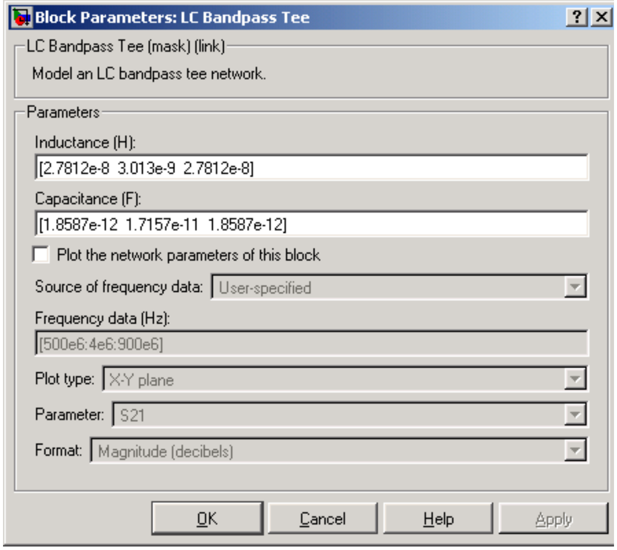
The LC Bandpass Tee block cascades the ABCD-parameters for each series and shunt pair, then converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

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



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

Dialog Box



Block Parameters: LC Bandpass Tee

LC Bandpass Tee (mask) (link)

Model an LC bandpass tee network.

Parameters

Inductance (H):
[2.7812e-8 3.013e-9 2.7812e-8]

Capacitance (F):
[1.8587e-12 1.7157e-11 1.8587e-12]

□ Plot the network parameters of this block

Source of frequency data: User-specified

Frequency data (Hz):
[500e6:4e6:900e6]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

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 4-1.

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, Series RLC, Shunt RLC

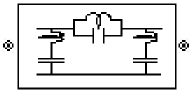
Purpose

Model an LC bandstop pi network

Library

Ladder Filters sublibrary of the Physical library

Description

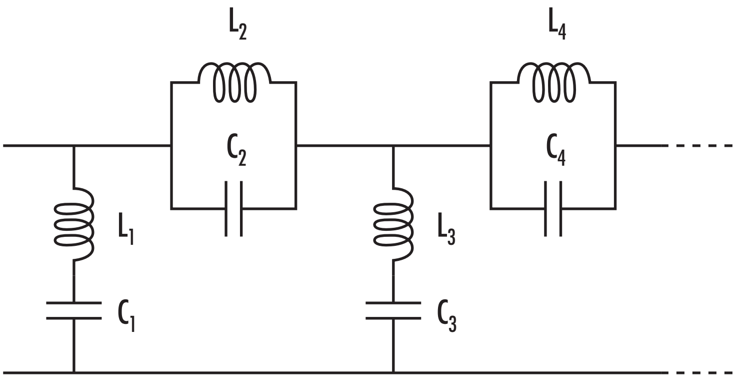


The LC Bandstop Pi block models the LC bandstop pi network described in the block dialog box, in terms of its frequency-dependent S-parameters.

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.

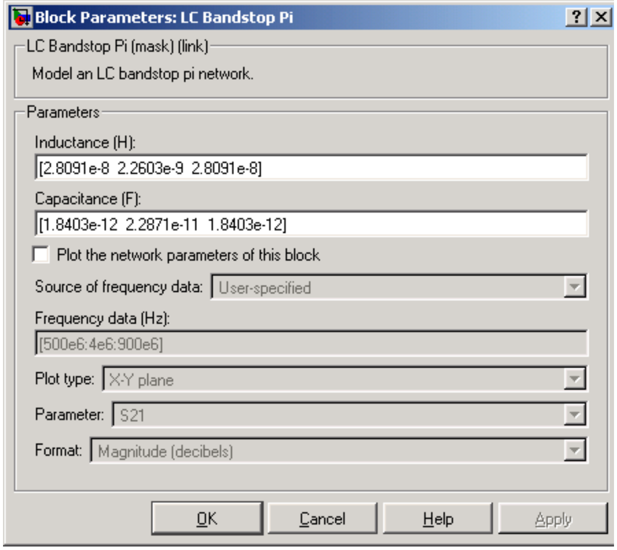
The LC Bandstop Pi block cascades the ABCD-parameters for each series and shunt pair, then converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

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



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

Dialog Box



Block Parameters: LC Bandstop Pi

LC Bandstop Pi (mask) (link)

Model an LC bandstop pi network.

Parameters

Inductance (H): [2.8091e-8 2.2603e-9 2.8091e-8]

Capacitance (F): [1.8403e-12 2.2871e-11 1.8403e-12]

□ Plot the network parameters of this block

Source of frequency data: User-specified

Frequency data (Hz): [500e6:4e6:900e6]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

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 4-1.

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, Series RLC, Shunt RLC

LC Bandstop Tee

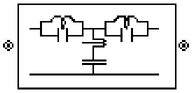
Purpose

Model an LC bandstop tee network

Library

Ladder Filters sublibrary of the Physical library

Description

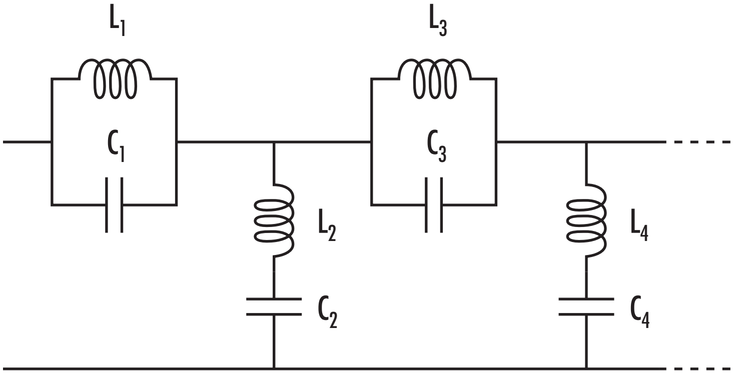


The LC Bandstop Tee block models the LC bandstop tee network described in the block dialog box, in terms of its frequency-dependent S-parameters.

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.

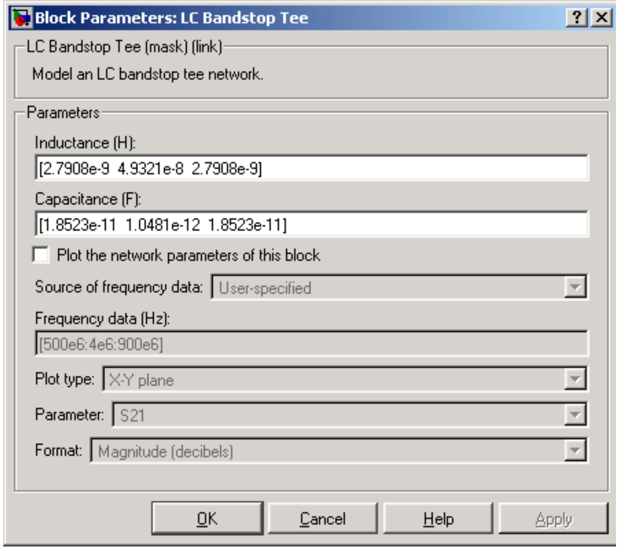
The LC Bandstop Tee block cascades the ABCD-parameters for each series and shunt pair, then converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

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



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

Dialog Box



LC Bandstop Tee (mask) (link)

Model an LC bandstop tee network.

Parameters

Inductance (H): [2.7908e-9 4.9321e-8 2.7908e-9]

Capacitance (F): [1.8523e-11 1.0481e-12 1.8523e-11]

□ Plot the network parameters of this block

Source of frequency data: User-specified

Frequency data (Hz): [500e6:4e6:900e6]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

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 4-1.

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, Series RLC, Shunt RLC

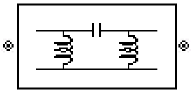
Purpose

Model an LC highpass pi network

Library

Ladder Filters sublibrary of the Physical library

Description

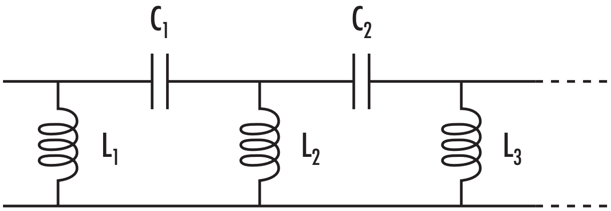


The LC Highpass Pi block models the LC highpass pi network described in the block dialog box, in terms of its frequency-dependent S-parameters.

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.

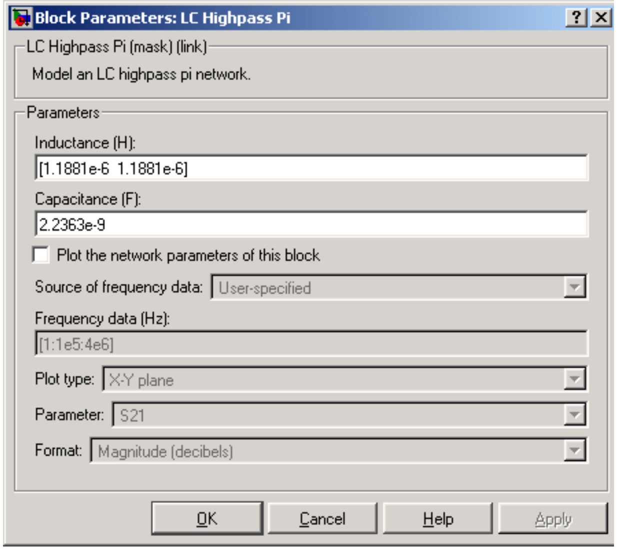
The LC Highpass Pi block cascades the ABCD-parameters for each circuit element, then converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

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



Where [L_1, L_2, L_3, ...] is the value of the `'L'` property, and [C_1, C_2, ...] is the value of the `'C'` property.

Dialog Box



Block Parameters: LC Highpass Pi

LC Highpass Pi (mask) (link)

Model an LC highpass pi network.

Parameters

Inductance (H): [1.1881e-6 1.1881e-6]

Capacitance (F): 2.2363e-9

□ Plot the network parameters of this block

Source of frequency data: User-specified

Frequency data (Hz): [1:1e5:4e6]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

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 4-1.

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, Series RLC, Shunt RLC

LC Highpass Tee

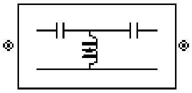
Purpose

Model an LC highpass tee network

Library

Ladder Filters sublibrary of the Physical library

Description

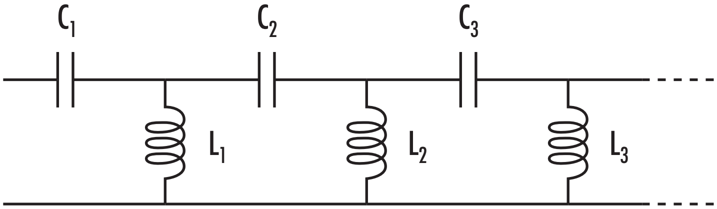


The LC Highpass Tee block models the LC highpass tee network described in the block dialog box, in terms of its frequency-dependent S-parameters.

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.

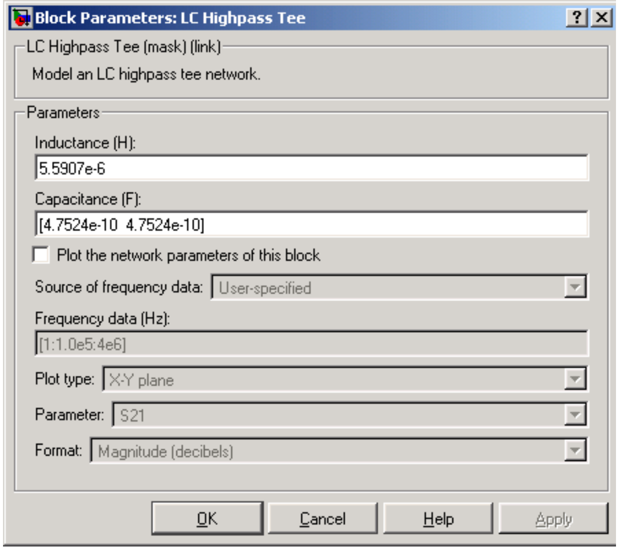
The LC Highpass Tee block cascades the ABCD-parameters for each circuit element, then converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

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



Where [L_1, L_2, L_3, ...] is the value of the `'L'` property, and [C_1, C_2, C_3, ...] is the value of the `'C'` property.

Dialog Box



Block Parameters: LC Highpass Tee

LC Highpass Tee (mask) (link)

Model an LC highpass tee network.

Parameters

Inductance (H): 5.5907e-6

Capacitance (F): [4.7524e-10 4.7524e-10]

☐ Plot the network parameters of this block

Source of frequency data: User-specified

Frequency data (Hz): [1:1.0e5:4e6]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

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 4-1.

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, Series RLC, Shunt RLC

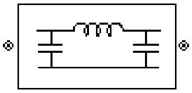
Purpose

Model an LC lowpass pi network

Library

Ladder Filters sublibrary of the Physical library

Description

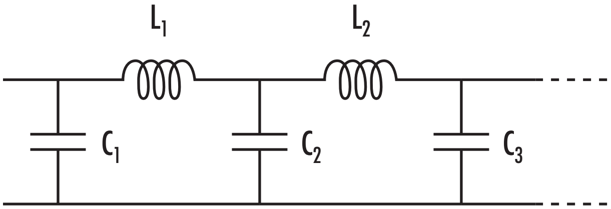


The LC Lowpass Pi block models the LC lowpass pi network described in the block dialog box, in terms of its frequency-dependent S-parameters.

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.

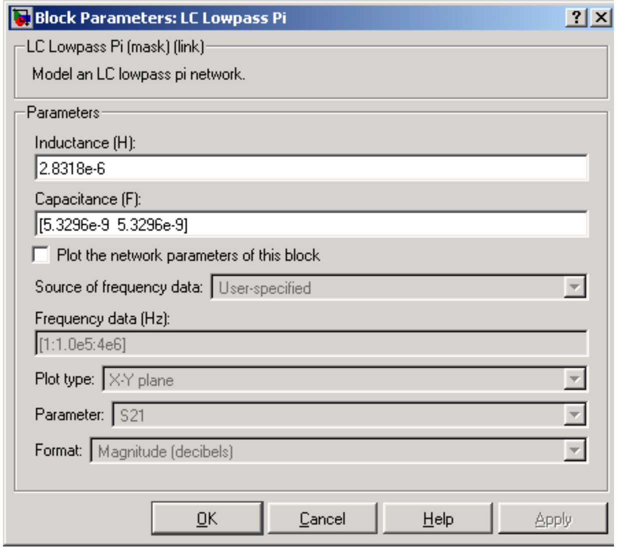
The LC Lowpass Pi block cascades the ABCD-parameters for each circuit element, then converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

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



Where [L_1, L_2, ...] is the value of the `'L'` property, and [C_1, C_2, C_3, ...] is the value of the `'C'` property.

Dialog Box



Block Parameters: LC Lowpass Pi

LC Lowpass Pi (mask) (link)

Model an LC lowpass pi network.

Parameters

Inductance (H): 2.8318e-6

Capacitance (F): [5.3296e-9 5.3296e-9]

□ Plot the network parameters of this block

Source of frequency data: User-specified

Frequency data (Hz): [1:1.0e5:4e6]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

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 4-1.

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, Series RLC, Shunt RLC

LC Lowpass Tee

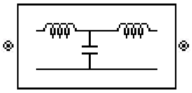
Purpose

Model an LC lowpass tee network

Library

Ladders Filters sublibrary of the Physical library

Description

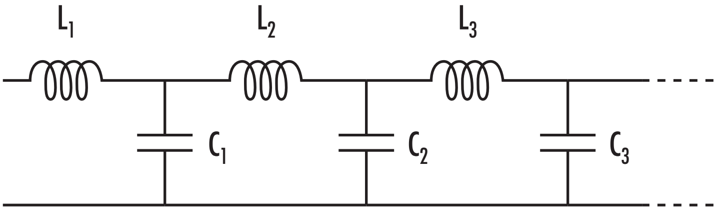


The LC Lowpass Tee block models the LC lowpass tee network described in the block dialog box in terms of its frequency-dependent S-parameters.

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.

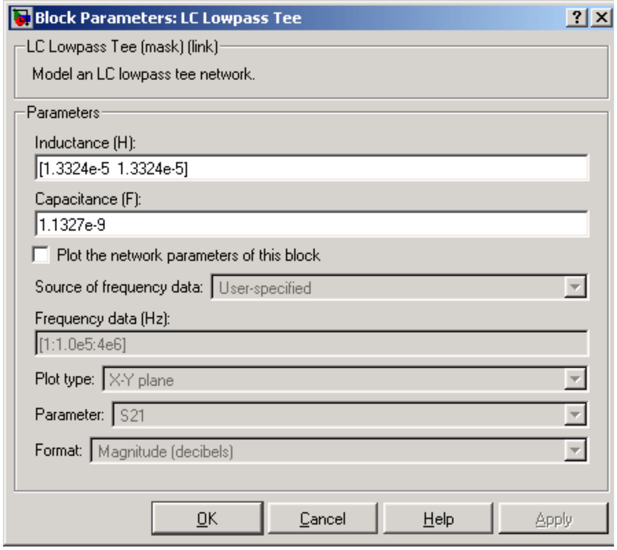
The LC Lowpass Tee block cascades the ABCD-parameters for each circuit element, then converts the cascaded parameters to S-parameters using the RF Toolbox `abcd2s` function.

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



Where [L_1, L_2, L_3, ...] is the value of the `'L'` property, and [C_1, C_2, C_3, ...] is the value of the `'C'` property.

Dialog Box



Block Parameters: LC Lowpass Tee

LC Lowpass Tee (mask) (link)

Model an LC lowpass tee network.

Parameters

Inductance (H): [1.3324e-5 1.3324e-5]

Capacitance (F): 1.1327e-9

□ Plot the network parameters of this block

Source of frequency data: User-specified

Frequency data (Hz): [1:1.0e5:4e6]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

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 4-1.

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, Series RLC, Shunt RLC

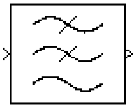
Purpose

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.

Note This block assumes a nominal impedance of 1 ohm.

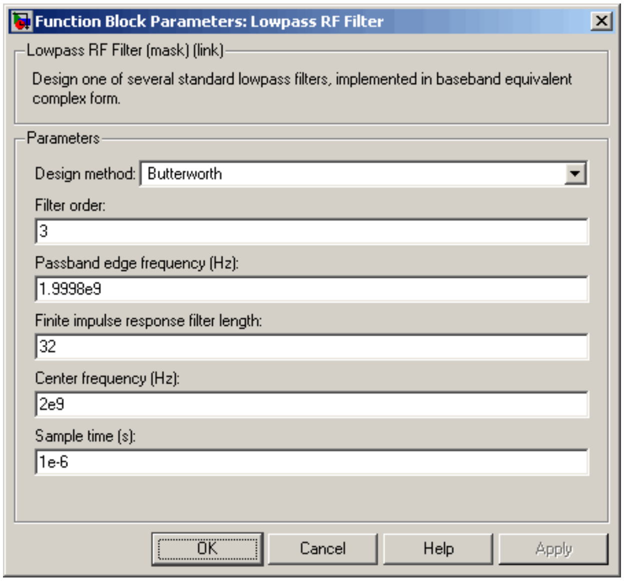
Select the design of the filter from the **Design method** list in the dialog box. For each design method, the block enables you to specify the filter design parameters shown in the following table.

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

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

Note Some RF blocks require the sample time to perform baseband modeling calculations. To ensure the accuracy of these calculations, the Input Port block, as well as the mathematical RF blocks, compare the input sample time 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



Function Block Parameters: Lowpass RF Filter

Lowpass RF Filter (mask) (link)

Design one of several standard lowpass filters, implemented in baseband equivalent complex form.

Parameters

Design method: Butterworth

Filter order: 3

Passband edge frequency (Hz): 1.9998e9

Finite impulse response filter length: 32

Center frequency (Hz): 2e9

Sample time (s): 1e-6

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.

Finite impulse response filter length

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`, `besselap` (Signal Processing Toolbox)

Microstrip Transmission Line

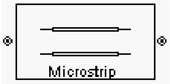
Purpose

Model a microstrip transmission line

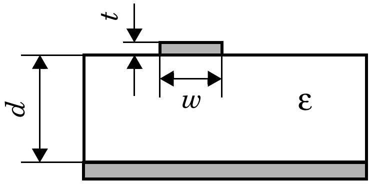
Library

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 (ε).



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 α_a is the attenuation coefficient and β is the wave number. The attenuation coefficient α_a is related to the loss, α, by

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

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

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

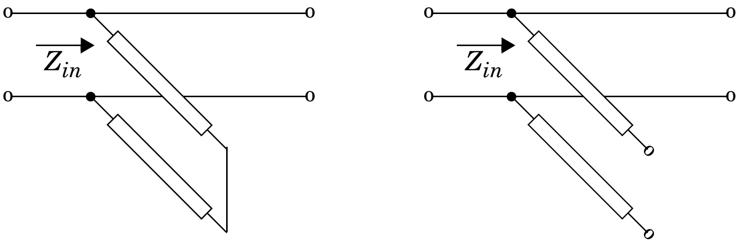
where $V_P = c/\sqrt{\varepsilon_{\mathrm{eff}}}$, and $\varepsilon_{\mathrm{eff}}$ is the frequency dependent effective dielectric constant. f is the vector of simulation frequencies for the specified parameters. See the Output Port block for information about determining the simulation frequencies. 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.

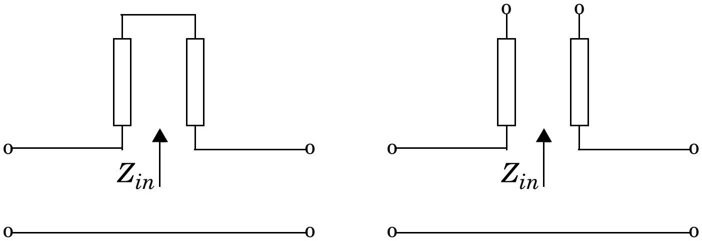
Shunt ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Shunt`, the 2-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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

$$A = 1$$
$$B = 0$$
$$C = 1/Z_{in}$$
$$D = 1$$

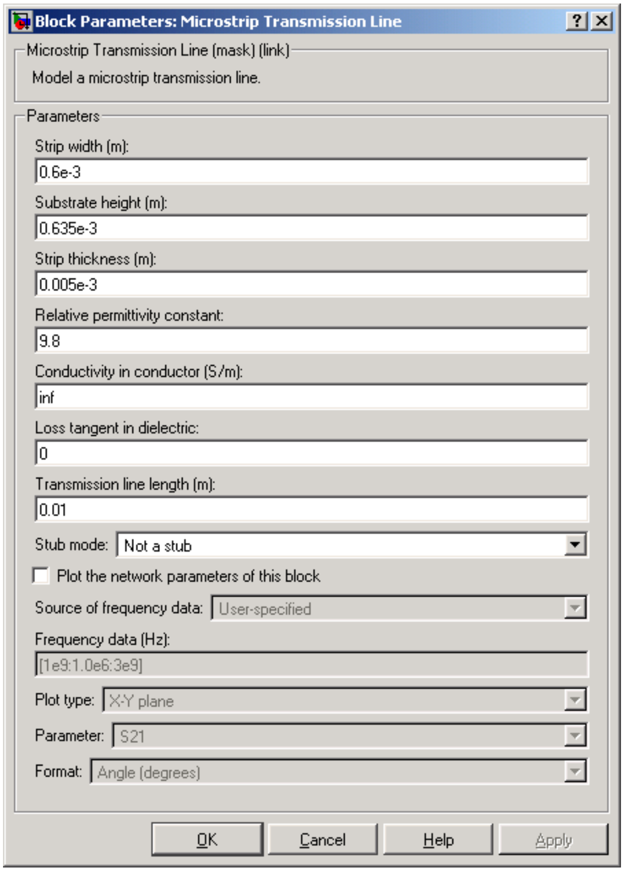
Series ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Series`, the 2-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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

$$A = 1$$
$$B = Z_{in}$$
$$C = 0$$
$$D = 1$$

Dialog Box



Block Parameters: Microstrip Transmission Line

Microstrip Transmission Line (mask) (link)

Model a microstrip transmission line.

Parameters

Strip width (m): 0.6e-3

Substrate height (m): 0.635e-3

Strip thickness (m): 0.005e-3

Relative permittivity constant: 9.8

Conductivity in conductor (S/m): inf

Loss tangent in dielectric: 0

Transmission line length (m): 0.01

Stub mode: Not a stub

□ Plot the 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)

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.

Relative permittivity constant

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

Conductivity in conductor (S/m)

Conductivity of the conductor in siemens per meter.

Loss tangent in dielectric

Loss angle tangent of the dielectric.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

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

Termination of stub

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

Note For information about plotting, see "Plotting Network Parameters" on page 4-1.

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

Mixer

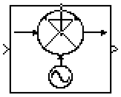
Purpose

Complex baseband model of mixer with phase noise

Library

Mathematical

Description



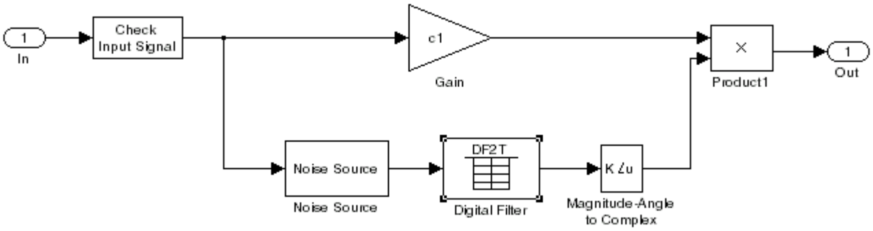
The Mixer block generates a complex baseband model of a mixer, with phase noise whose spectrum is characterized by a $1/f$ slope. The level of the spectrum is specified by the noise power contained in one hertz bandwidth offset from the carrier by a certain frequency.

Note This block assumes a nominal impedance of 1 ohm.

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

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

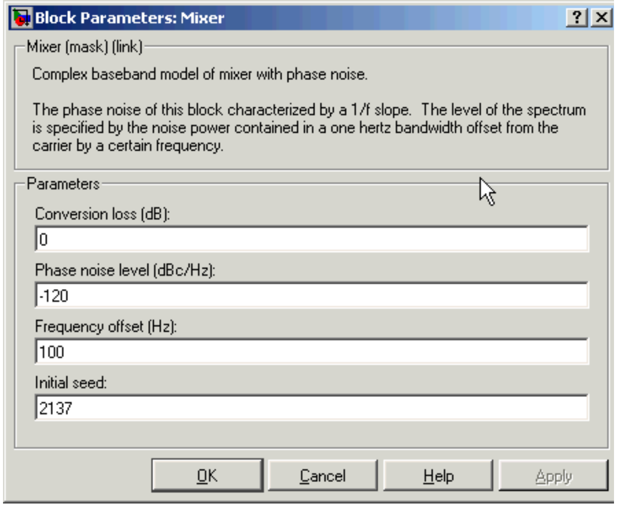
You can view the block's implementation of phase noise by right-clicking on the block and selecting **Look under mask** from the pop-up menu. This displays the following figure:



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.

Dialog Box



Block Parameters: Mixer

Mixer (mask) (link)

Complex baseband model of mixer with phase noise.

The phase noise of this block characterized by a 1/f slope. The level of the spectrum is specified by the noise power contained in a one hertz bandwidth offset from the carrier by a certain frequency.

Parameters

Conversion loss (dB): 0

Phase noise level (dBc/Hz): -120

Frequency offset (Hz): 100

Initial seed: 2137

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

Conversion loss (dB)

Scalar specifying the conversion loss for the mixer. Tunable.

Phase noise level (dBc/Hz)

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

Frequency offset (Hz)

Scalar specifying the frequency offset. Tunable.

Initial seed

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

References

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

See Also

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

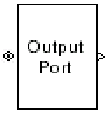
Purpose

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 "Creating 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.

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 finite impulse response filter length N, all of which you specify in the Input Port block dialog box.

The nth element of f, f_n, is given by

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

where

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

3 Calculates the passband transfer function for the frequency range as

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

where V_S and V_L are the source and load voltages and f represents the 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.

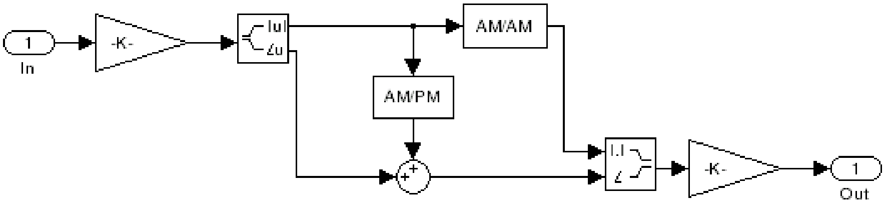
4 Translates the passband transfer function to baseband as $H(f - f_c)$, where f_c is the specified center frequency.

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 finite impulse response filter length. It then truncates the impulse response to a length less than or equal to the 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

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

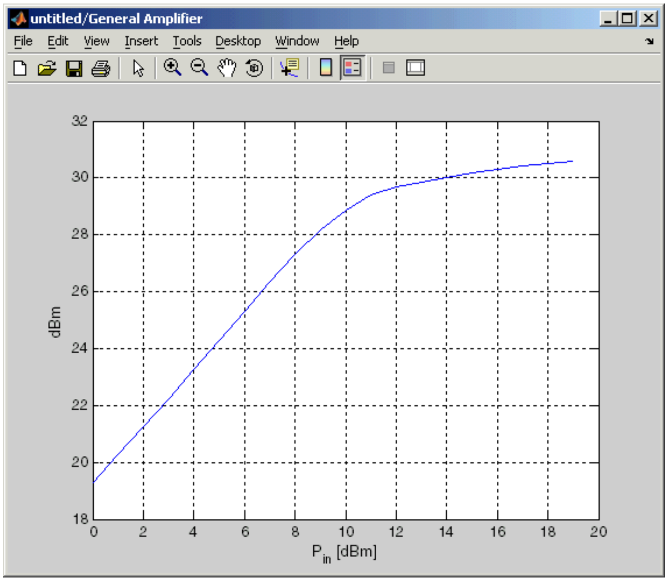
$$PM_{\text{out}} = \text{Phase}$$

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

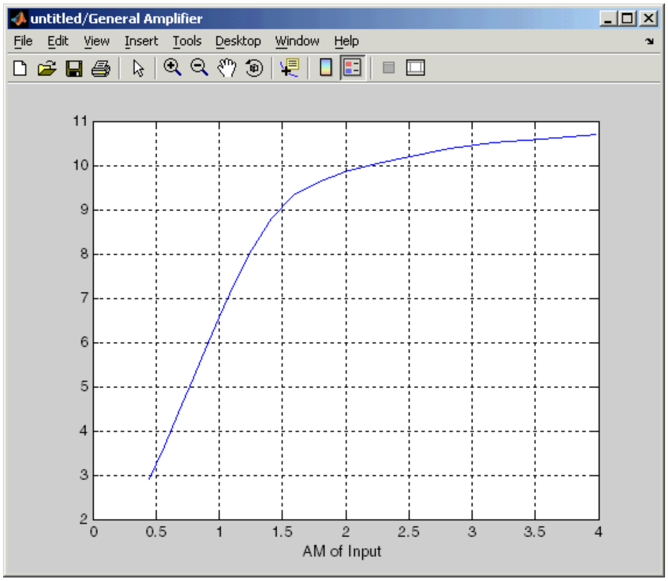
where AM_{in} is the AM of the input voltage, AM_{out} and PM_{out} are the AM and PM of the output voltage, R_s , is the source resistance (50 ohms), R_l is the load resistance (50 ohms), P_{in} is the input power, P_{out} is the output power, and Phase is the phase shift between the input and output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

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

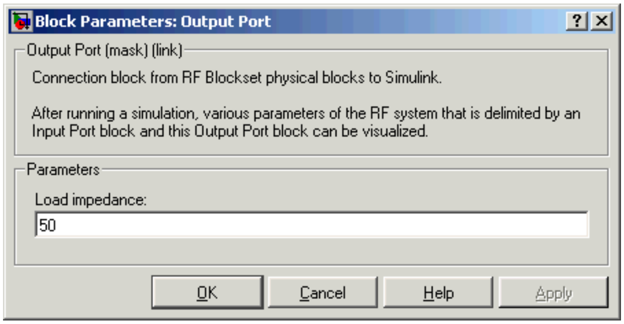
The following figure shows the original power data of an amplifier.



This figure shows the extracted AM/AM nonlinear conversion.



Dialog Box



Block Parameters: Output Port

Output Port (mask) (link)

Connection block from RF Blockset physical blocks to Simulink.

After running a simulation, various parameters of the RF system that is delimited by an Input Port block and this Output Port block can be visualized.

Parameters

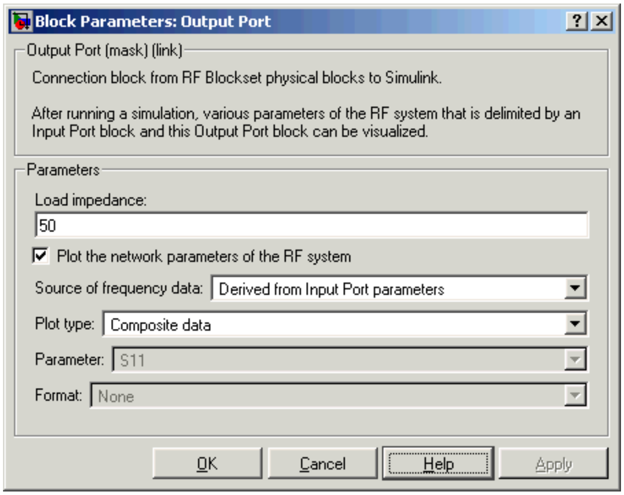
Load impedance: 50

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 4-1 for more information about plotting.



Block Parameters: Output Port

Output Port (mask) (link)

Connection block from RF Blockset physical blocks to Simulink.

After running a simulation, various parameters of the RF system that is delimited by an Input Port block and this Output Port block can be visualized.

Parameters

Load impedance: 50

☑ Plot the network parameters of the RF system

Source of frequency data: Derived from Input Port parameters

Plot type: Composite data

Parameter: S11

Format: None

See Also

Input Port; s2y (RF Toolbox)

Parallel-Plate Transmission Line

Purpose

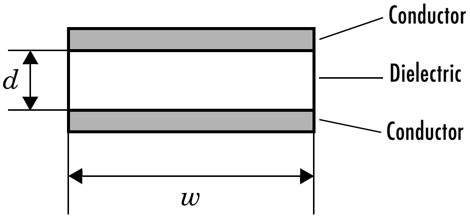
Model parallel-plate transmission line

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

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

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

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

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

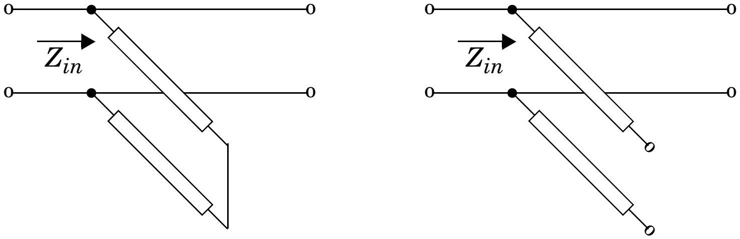
In these equations, σ_{cond} is the conductivity in the conductor and σ_{diel} is the conductivity in the dielectric. μ is the permeability of the dielectric, ε is its permittivity, and skin depth δ is calculated as $1/\sqrt{\pi f\mu\sigma_{\mathrm{cond}}}$. f is the vector of simulation frequencies 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.

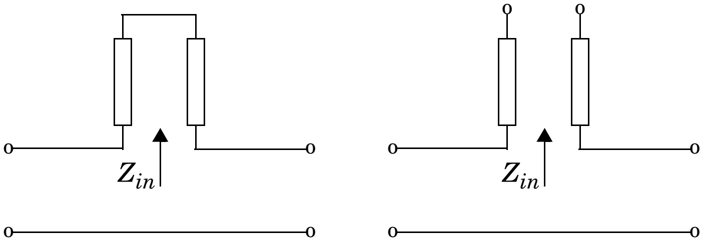
Shunt ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Shunt`, the 2-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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

$$A = 1$$
$$B = 0$$
$$C = 1/Z_{in}$$
$$D = 1$$

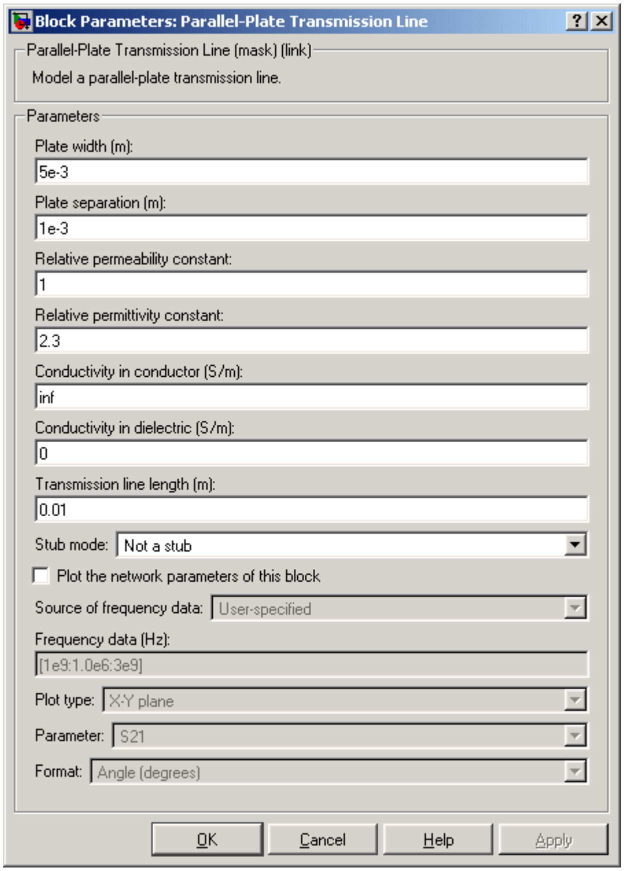
Series ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Series`, the 2-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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

$$A = 1$$
$$B = Z_{in}$$
$$C = 0$$
$$D = 1$$

Dialog Box



Parallel-Plate Transmission Line (mask) (link)

Model a parallel-plate transmission line.

Parameters

Plate width (m): 5e-3

Plate separation (m): 1e-3

Relative permeability constant: 1

Relative permittivity constant: 2.3

Conductivity in conductor (S/m): inf

Conductivity in dielectric (S/m): 0

Transmission line length (m): 0.01

Stub mode: Not a stub

□ Plot the 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)

Plate width (m)

Physical width of the parallel-plate transmission line.

Plate separation (m)

Thickness of the dielectric separating the plates.

Relative permeability constant

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

Relative permittivity constant

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

Conductivity in conductor (S/m)

Conductivity of the conductor in siemens per meter.

Conductivity in dielectric (S/m)

Conductivity of the dielectric in siemens per meter.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

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

Termination of stub

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

Note For information about plotting, see "Plotting Network Parameters" on page 4-1.

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

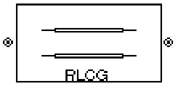
Purpose

Model RLCG transmission lines

Library

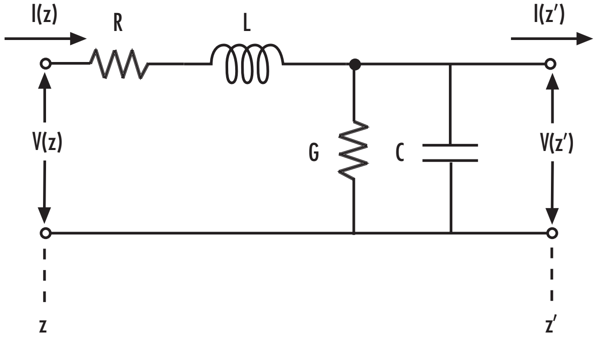
Transmission Lines sublibrary of the Physical library

Description



RLCG
Transmission Line

The RLCG Transmission Line block models the RLCG transmission line described in the block dialog in terms of its frequency-dependent resistance, inductance, capacitance, and conductance. The transmission line, which can be lossy or lossless, is treated as a 2-port linear network.



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

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

Stubless Transmission Line

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

$$S_{11} = 0$$

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

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

$$S_{22} = 0$$

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

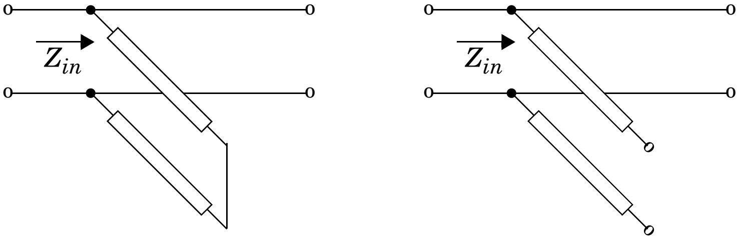
$$k = k_r + jk_i = \sqrt{(R + j2\pi fL)(G + j2\pi fC)}$$

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

Shunt and Series Stubs

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

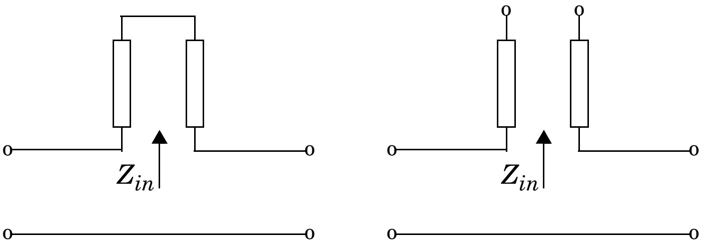
Shunt ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Shunt`, the 2-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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

$$A = 1$$
$$B = 0$$
$$C = 1/Z_{in}$$
$$D = 1$$

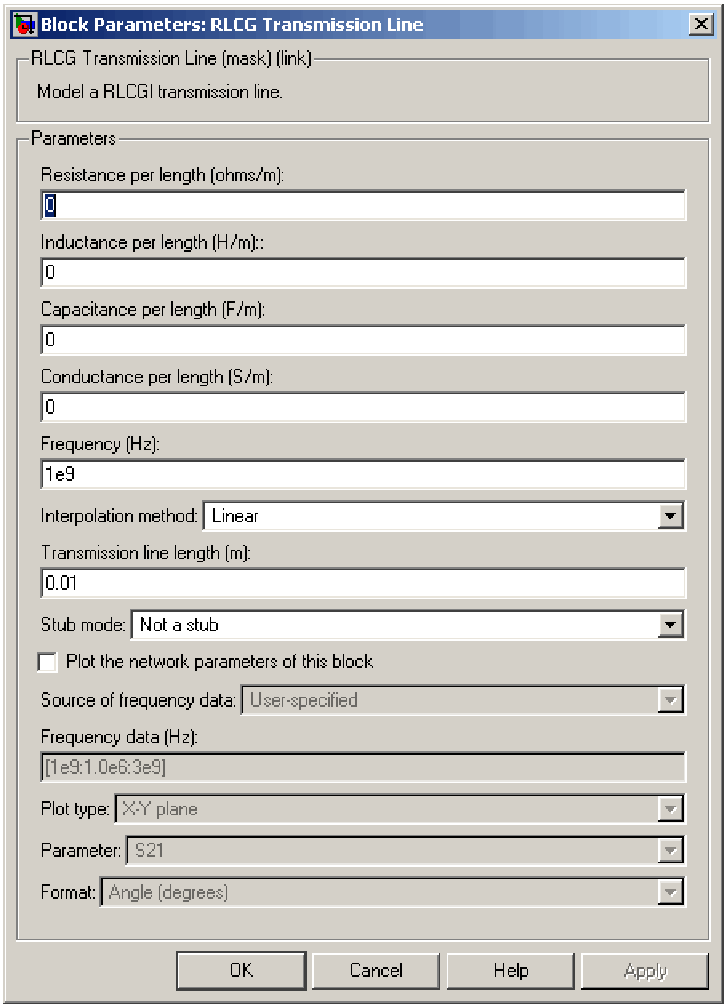
Series ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Series`, the 2-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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

$$A = 1$$
$$B = Z_{in}$$
$$C = 0$$
$$D = 1$$

Dialog Box



Block Parameters: RLCG Transmission Line

RLCG Transmission Line (mask) (link)

Model a RLCGI transmission line.

Parameters

Resistance per length (ohms/m): 0

Inductance per length (H/m):: 0

Capacitance per length (F/m): 0

Conductance per length (S/m): 0

Frequency (Hz): 1e9

Interpolation method: Linear

Transmission line length (m): 0.01

Stub mode: Not a stub

□ Plot the 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)

Resistance per length (ohms/m)

Vector of resistance values in ohms per meter.

Inductance per length (H/m)

Vector of inductance values in henries per meter.

Capacitance per length (F/m)

Vector of capacitance values in farads per meter.

Conductance per length (S/m)

Vector of conductance values in siemens per meter.

Frequency (Hz)

Vector of frequency values at which the resistance, inductance, capacitance, and conductance values are known.

Interpolation method

Specify the interpolation method the block uses to calculate the parameter values at the simulation frequencies. Your choices are `Linear`, `Spline`, or `Cubic`.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

Type of stub. Your choices are `Not a stub`, `Shunt`, or `Series`.

Note For information about plotting, see Chapter 4, “Plotting Network Parameters”.

References

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

See Also

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

S-Parameters Amplifier

Purpose

Model a nonlinear amplifier using its S-parameters

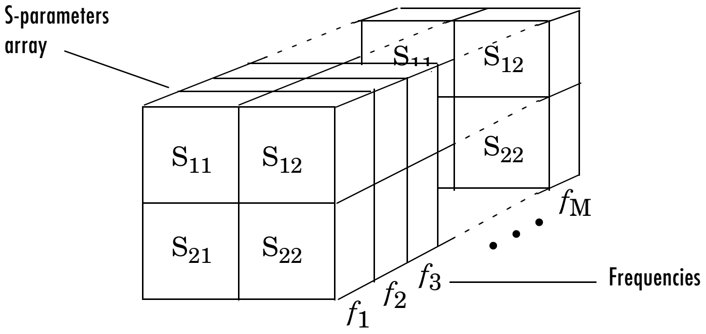
Library

Amplifiers sublibrary of the Physical library

Description

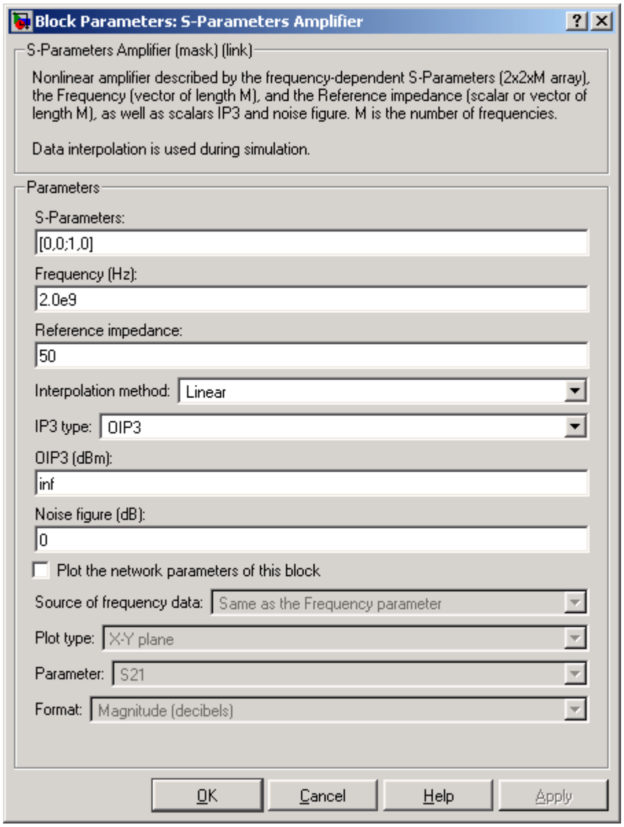
The S-Parameters Amplifier block models the nonlinear amplifier described in the block dialog, in terms of its frequency-dependent S-parameters, their frequencies, and the reference impedance of the S-parameters. The block also takes into account the IP3 value and the noise figure.

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 S-Parameters Amplifier block interpolates the given S-parameters to determine the S-parameters at the simulation frequencies. The simulation frequencies are determined by the Output Port block. See “S-Parameters at Simulation Frequencies” on page 3-2 for more details.

Dialog Box



Block Parameters: S-Parameters Amplifier

S-Parameters Amplifier (mask) (link)

Nonlinear amplifier described by the frequency-dependent S-Parameters (2x2xM array), the Frequency (vector of length M), and the Reference impedance (scalar or vector of length M), as well as scalars IP3 and noise figure. M is the number of frequencies.

Data interpolation is used during simulation.

Parameters

S-Parameters: [0,0;1,0]

Frequency (Hz): 2.0e9

Reference impedance: 50

Interpolation method: Linear

IP3 type: OIP3

OIP3 (dBm): inf

Noise figure (dB): 0

□ 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)

S-Parameters

S-parameters for a nonlinear amplifier in a 2-by-2-by-M array. M is the number of S-parameters.

Frequency (Hz)

Frequencies of the S-parameters as an M-element vector. The order of the frequencies must correspond to the order of the S-parameters in **S-Parameters**. All frequencies must be positive.

Reference impedance

Reference impedance of the S-parameters as a scalar or a vector of length M. The value of this parameter can be real or complex. If you provide a scalar, that value is applied to all frequencies.

Interpolation method

Method used to interpolate the given S-parameters over the range of frequencies. Interpolation can be `Cubic`, `Linear` (default), or `Spline`.

IP3 type

Type of third-order intercept point. The value can be `IIP3` (input intercept point) or `OIP3` (output intercept point).

IIP3 (dBm)

Input power intercept point as a scalar value. This field becomes visible if you select `IIP3` as the IP3 type.

OIP3 (dBm)

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

Noise figure (dB)

Scalar ratio of the available signal-to-noise power ratio at the input to the available signal-to-noise power ratio at the output, $((S_i/N_i)/(S_o/N_o))$.

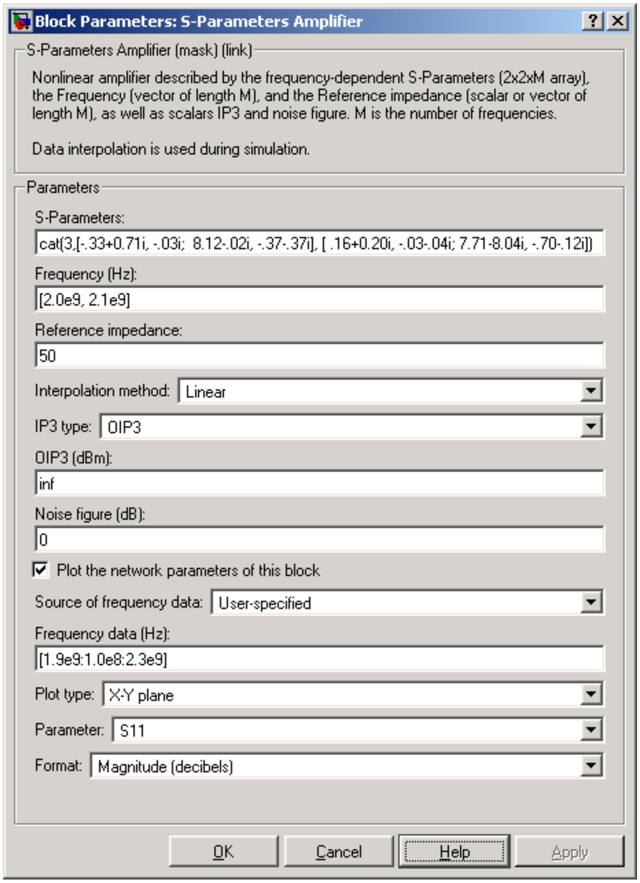
Note For information about plotting, see “Plotting Network Parameters” on page 4-1.

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])
```



Block Parameters: S-Parameters Amplifier

S-Parameters Amplifier (mask) (link)

Nonlinear amplifier described by the frequency-dependent S-Parameters (2x2xM array), the Frequency (vector of length M), and the Reference impedance (scalar or vector of length M), as well as scalars IP3 and noise figure. M is the number of frequencies.

Data interpolation is used during simulation.

Parameters

S-Parameters: cat(3,[-.33+0.71i, -.03i; 8.12-.02i, -.37-.37i], [.16+0.20i, -.03-.04i; 7.71-8.04i, -.70-.12i])

Frequency (Hz): [2.0e9, 2.1e9]

Reference impedance: 50

Interpolation method: Linear

IP3 type: OIP3

OIP3 (dBm): inf

Noise figure (dB): 0

☑ Plot the network parameters of this block

Source of frequency data: User-specified

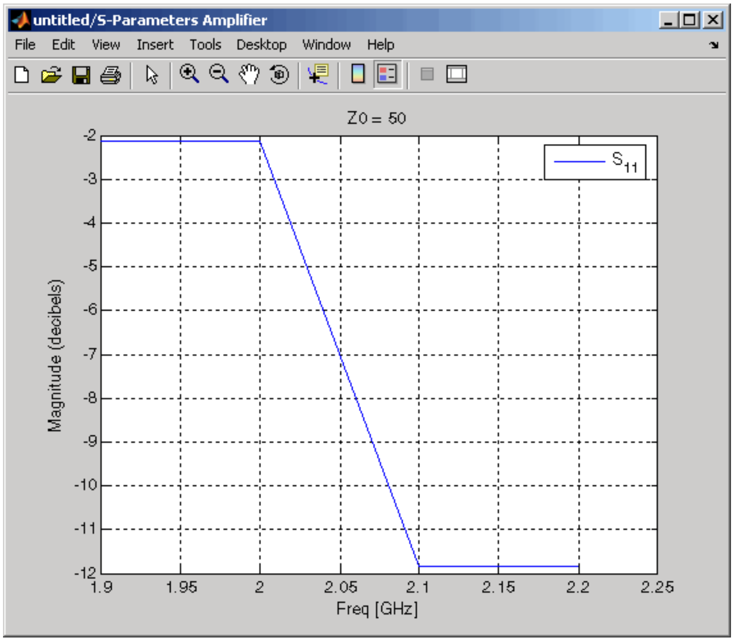
Frequency data (Hz): [1.9e9:1.0e8:2.3e9]

Plot type: X-Y plane

Parameter: S11

Format: Magnitude (decibels)

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

`interp1` (MATLAB)

S-Parameters Mixer

Purpose

Model a mixer using its S-parameters

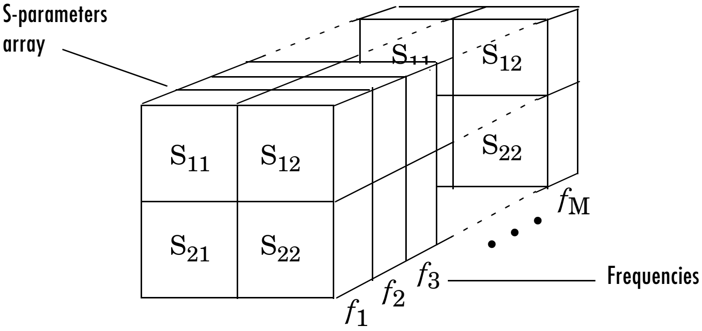
Library

Mixer sublibrary of the Physical library

Description

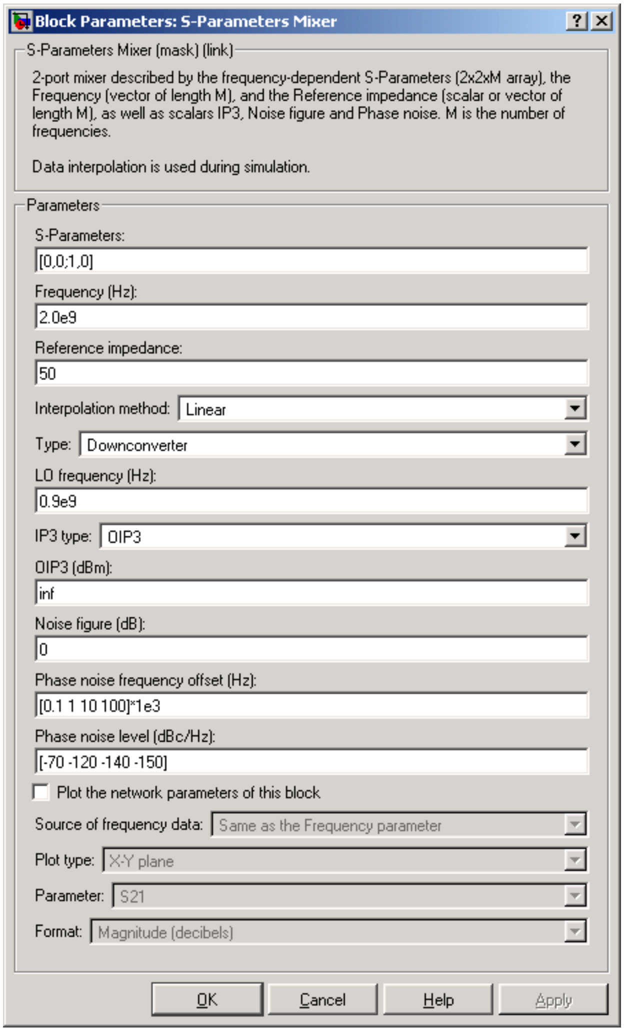
The S-Parameters Mixer block models the nonlinear mixer described in the block dialog, in terms of its frequency-dependent S-parameters, their frequencies, and the reference impedance of the S-parameters. The block also takes phase noise into account.

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 S-Parameters Mixer block interpolates the given S-parameters to determine the S-parameters at the simulation frequencies. The simulation frequencies are determined by the Output Port block. See “S-Parameters at Simulation Frequencies” on page 3-2 for more details.

Dialog Box



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)

S-Parameters

S-parameters for a nonlinear mixer in a 2-by-2-by-M array. M is the number of S-parameters.

Frequency (Hz)

Frequencies of the S-parameters as an M-element vector. The order of the frequencies must correspond to the order of the S-parameters in **S-Parameters**. All frequencies must be positive.

Reference impedance

Reference impedance of the S-parameters as a scalar or a vector of length M. The value of this parameter can be real or complex. If you provide a scalar, that value is applied to all frequencies.

Interpolation method

Method used to interpolate the given S-parameters over the range of frequencies. Interpolation can be `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 4-1.

See Also

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

S-Parameters Passive Network

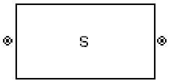
Purpose

Model a passive network using its S-parameters

Library

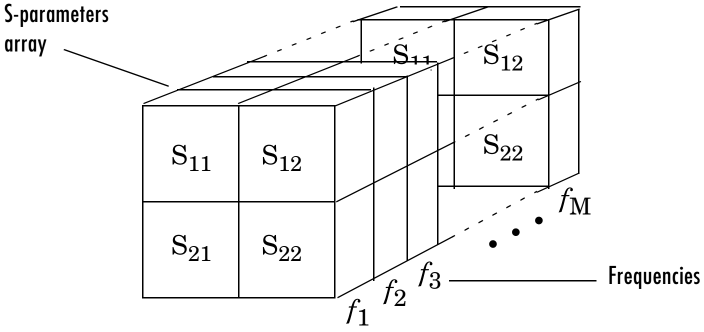
Black Box Elements sublibrary of the Physical library

Description



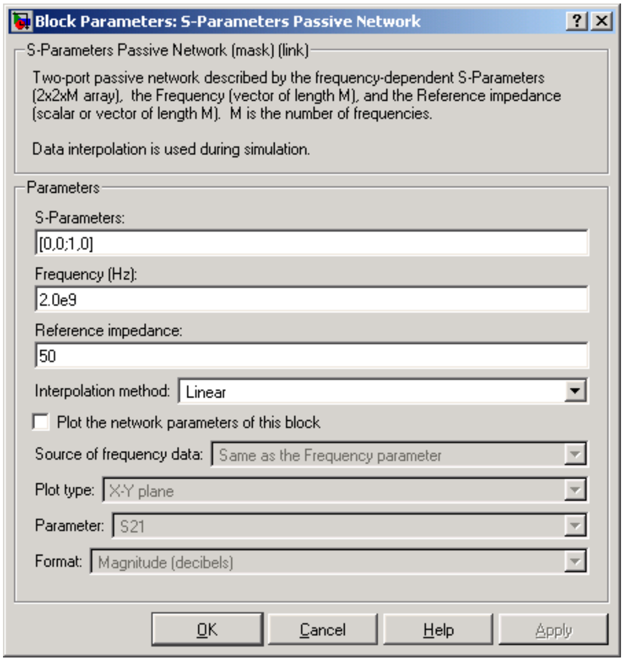
The S-Parameters Passive Network block models the two-port passive network described in the block dialog, in terms of its S-parameters, their frequencies, and the reference impedance of the S-parameters.

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 S-Parameters Passive Network block interpolates the given S-parameters to determine the S-parameters at the simulation frequencies. The simulation frequencies are determined by the Output Port block. See “S-Parameters at Simulation Frequencies” on page 3-2 for more details.

Dialog Box



Block Parameters: S-Parameters Passive Network

S-Parameters Passive Network (mask) (link)

Two-port passive network 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). 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

□ 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)

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 4-1.

Examples

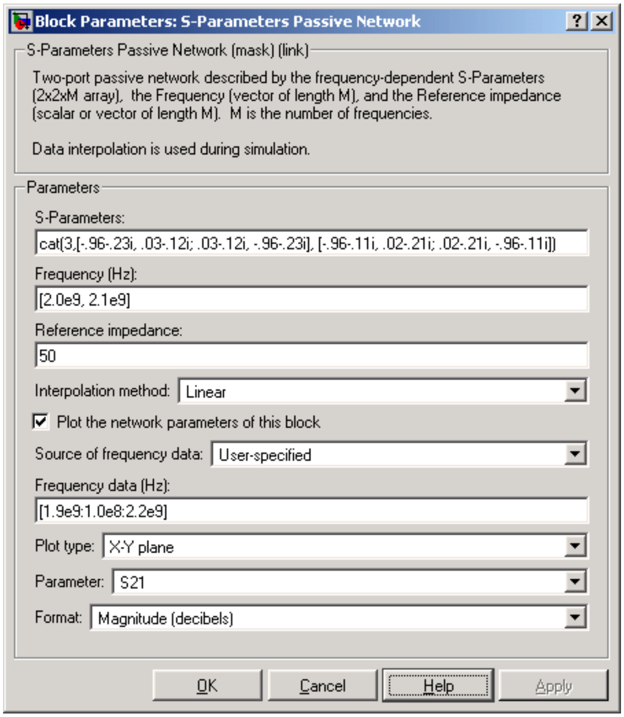
Plotting Parameters with the S-Parameters Passive Network Block

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 (mask) (link)

Two-port passive network 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). M is the number of frequencies.

Data interpolation is used during simulation.

Parameters

S-Parameters: cat(3,[-.96-.23i, .03-.12i; .03-.12i, -.96-.23i], [-.96-.11i, .02-.21i; .02-.21i, -.96-.11i])

Frequency (Hz): [2.0e9, 2.1e9]

Reference impedance: 50

Interpolation method: Linear

☑ Plot the network parameters of this block

Source of frequency data: User-specified

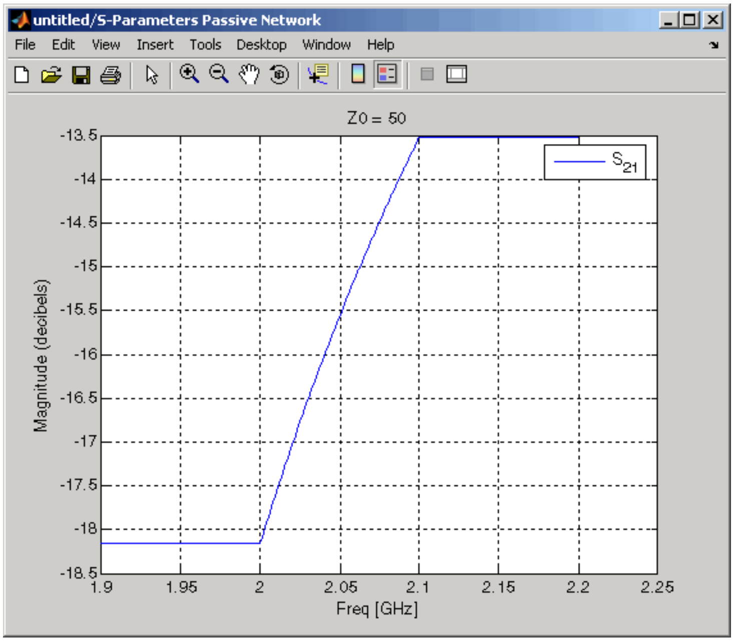
Frequency data (Hz): [1.9e9:1.0e8:2.2e9]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

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.



See Also

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

`interp1` (MATLAB)

Series RLC

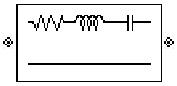
Purpose

Model a series RLC network

Library

Ladders Filters sublibrary of the Physical library

Description



The Series RLC block models the series RLC network described in the block dialog box, in terms of its frequency-dependent S-parameters.

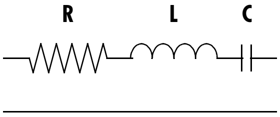
For the given resistance, inductance, and capacitance, the block first calculates the ABCD-parameters for each frequency contained in the vector of simulation frequencies, then converts the ABCD-parameters to S-parameters using the RF Toolbox `abcd2s` function. See the Output Port block for information about simulation frequencies.

For this circuit, A = 1, B = Z, C = 0, and D = 1, where

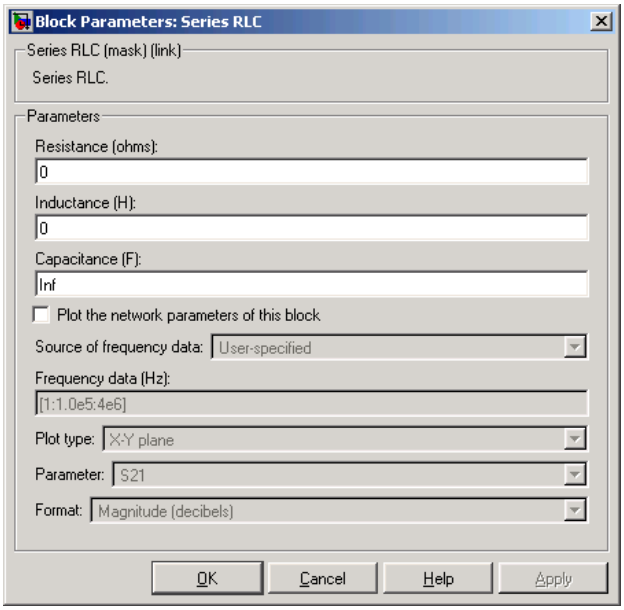
$$Z = \frac{-LC\omega^2 + jRC\omega + 1}{jC\omega}$$

where $\omega = 2\pi f$.

The series RLC object is a two-port network as shown in the circuit diagram below.



Dialog Box



Block Parameters: Series RLC

Series RLC (mask) (link)

Series RLC.

Parameters

Resistance (ohms):

0

Inductance (H):

0

Capacitance (F):

Inf

□ Plot the network parameters of this block

Source of frequency data: User-specified

Frequency data (Hz):

[1:1.0e5:4e6]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

Resistance (ohms)

Scalar value for the resistance. The value must be nonnegative.

Inductance (H)

Scalar value for the inductance. The value must be nonnegative.

Capacitance (F)

Scalar value for the capacitance. The value must be nonnegative.

Note For information about plotting, see “Plotting Network Parameters” on page 4-1.

See Also

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

Shunt RLC

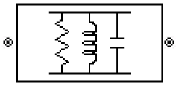
Purpose

Model a shunt RLC network

Library

Ladders Filters sublibrary of the Physical library

Description



The Shunt RLC block models the shunt RLC network described in the block dialog box, in terms of its frequency-dependent S-parameters.

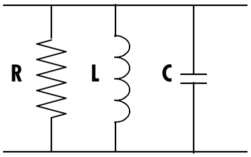
For the given resistance, inductance, and capacitance, the block first calculates the ABCD-parameters for each frequency contained in the vector of simulation frequencies, then converts the ABCD-parameters to S-parameters using the RF Toolbox `abcd2s` function. See the Output Port block for information about simulation frequencies.

For this circuit, A = 1, B = 0, C = Y, and D = 1, where

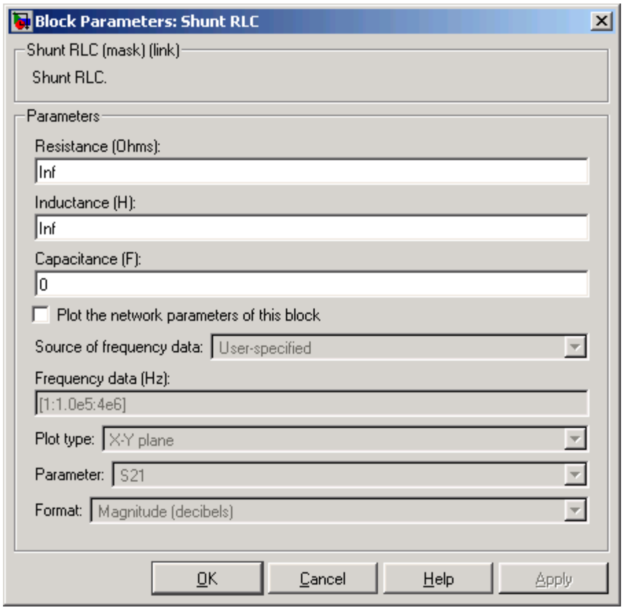
$$Y = \frac{-LC\omega^2 + j(L/R)\omega + 1}{jL\omega}$$

and $\omega = 2\omega f$.

The shunt RLC object is a two-port network as shown in the circuit diagram below.



Dialog Box



Block Parameters: Shunt RLC

Shunt RLC (mask) (link)

Shunt RLC.

Parameters

Resistance (Ohms): Inf

Inductance (H): Inf

Capacitance (F): 0

□ Plot the network parameters of this block

Source of frequency data: User-specified

Frequency data (Hz): [1:1.0e5:4e6]

Plot type: X-Y plane

Parameter: S21

Format: Magnitude (decibels)

Resistance (ohms)

Scalar value for the resistance. The value must be nonnegative.

Inductance (H)

Scalar value for the inductance. The value must be nonnegative.

Capacitance (F)

Scalar value for the capacitance. The value must be nonnegative.

Note For information about plotting, see "Plotting Network Parameters" on page 4-1.

See Also

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

Transmission Line

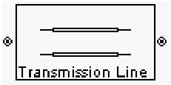
Purpose

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 α_a is the attenuation coefficient and β is the wave number. The attenuation coefficient α_a is related to the loss, α, by

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

and the wave number β is related to the phase velocity, V_P, by

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

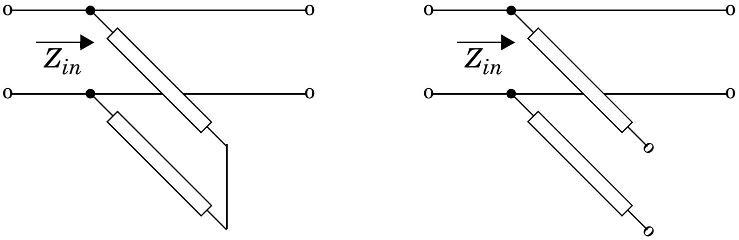
where f is the vector of simulation frequencies for the specified parameters. See the Output Port block for information about determining the simulation frequencies. The phase velocity V_p is also known as the wave propagation velocity.

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.

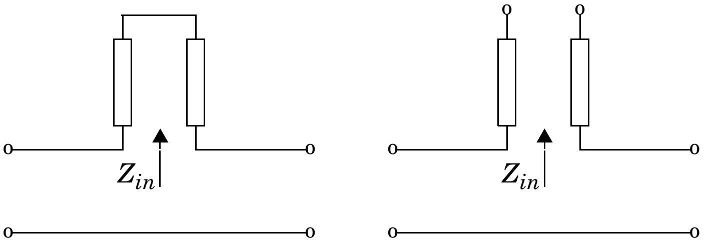
Shunt ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Shunt`, the 2-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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

$$A = 1$$
$$B = 0$$
$$C = 1/Z_{in}$$
$$D = 1$$

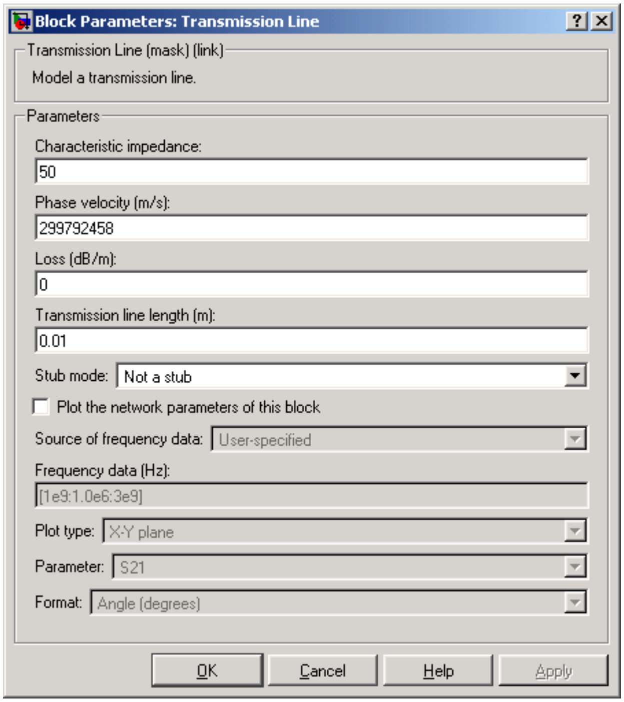
Series ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Series`, the 2-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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

$$A = 1$$
$$B = Z_{in}$$
$$C = 0$$
$$D = 1$$

Dialog Box



Block Parameters: Transmission Line

Transmission Line (mask) (link)

Model a transmission line.

Parameters

Characteristic impedance: 50

Phase velocity (m/s): 299792458

Loss (dB/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)

Characteristic impedance

Characteristic impedance of the transmission line. The value can be complex.

Phase velocity (m/s)

Propagation velocity of a uniform plane wave on the transmission line.

Loss (dB/m)

Reduction in strength of the signal as it travels over the transmission line. Must be positive.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

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

Termination of stub

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

Note For information about plotting, see "Plotting Network Parameters" on page 4-1.

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

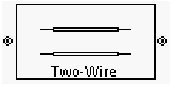
Purpose

Model a two-wire transmission line

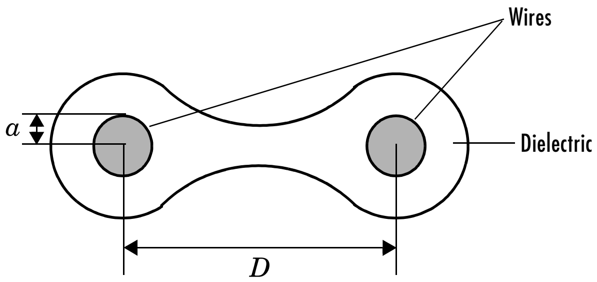
Library

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

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

where

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

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

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

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

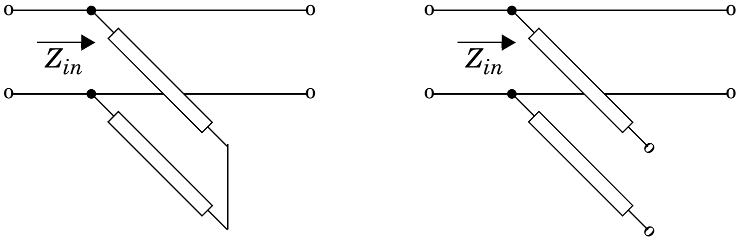
In these equations, σ_{cond} is the conductivity in the conductor and σ_{diel} is the conductivity in the dielectric. μ is the permeability of the dielectric, ε is its permittivity, and skin depth δ is calculated as $1/\sqrt{\pi f \mu \sigma_{\text{cond}}}$. f is the vector of simulation frequencies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

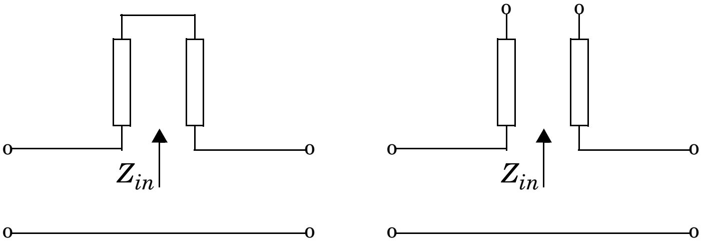
Shunt ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Shunt`, the 2-port network consists of a stub transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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

$$A = 1$$
$$B = 0$$
$$C = 1/Z_{in}$$
$$D = 1$$

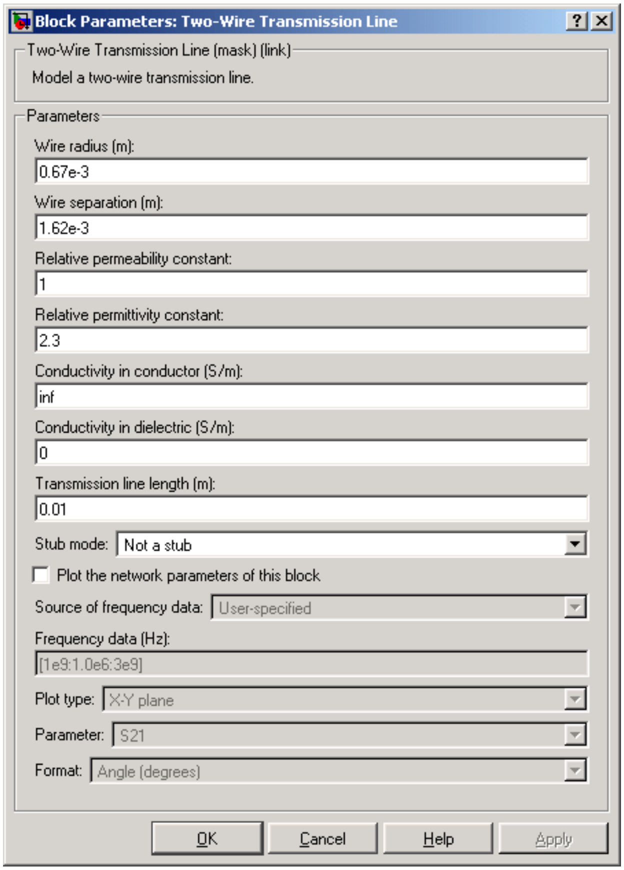
Series ABCD-Parameters. When you set the **Stub mode** parameter in the mask dialog box to `Series`, the 2-port network consists of a series transmission line that you can terminate with either a short circuit or an open circuit as shown here.



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

$$A = 1$$
$$B = Z_{in}$$
$$C = 0$$
$$D = 1$$

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)

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 μ_0.

Relative permittivity constant

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

Conductivity in conductor (S/m)

Conductivity of the conductor in siemens per meter.

Conductivity in dielectric (S/m)

Conductivity of the dielectric in siemens per meter.

Transmission line length (m)

Physical length of the transmission line.

Stub mode

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

Termination of stub

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

Note For information about plotting, see "Plotting Network Parameters" on page 4-1.

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

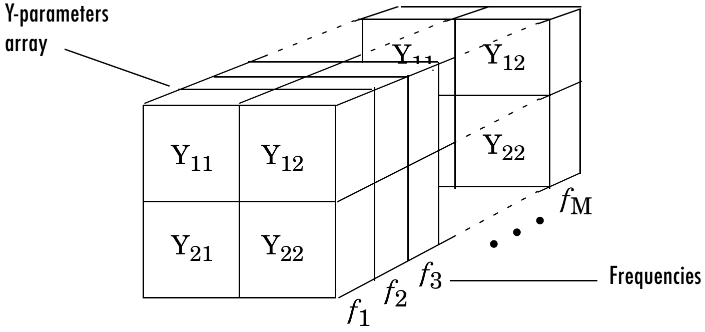
Library

Amplifiers sublibrary of the Physical library

Description

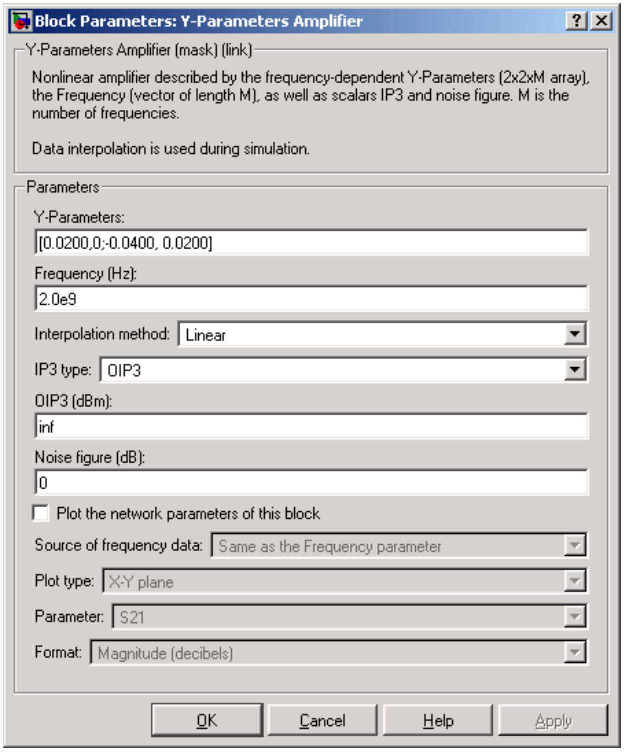
The Y-Parameters Amplifier block models the nonlinear amplifier described in the block dialog, in terms of its frequency-dependent Y-parameters and their frequencies. The block also takes into account the IP3 value and the noise figure.

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 Y-Parameters Amplifier block uses the RF Toolbox `y2s` function to convert the Y-parameters to S-parameters, then interpolates the resulting S-parameters to determine the S-parameters at the simulation frequencies. The simulation frequencies are determined by the Output Port block. See "S-Parameters at Simulation Frequencies" on page 3-2 for more details.

Dialog Box



Block Parameters: Y-Parameters Amplifier

Y-Parameters Amplifier (mask) (link)

Nonlinear amplifier described by the frequency-dependent Y-Parameters (2x2xM array), the Frequency (vector of length M), as well as scalars IP3 and noise figure. M is the number of frequencies.

Data interpolation is used during simulation.

Parameters

Y-Parameters: [0.0200,0;-0.0400, 0.0200]

Frequency (Hz): 2.0e9

Interpolation method: Linear

IP3 type: OIP3

OIP3 (dBm): inf

Noise figure (dB): 0

□ 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)

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, $(S_i/N_i)/(S_o/N_o)$.

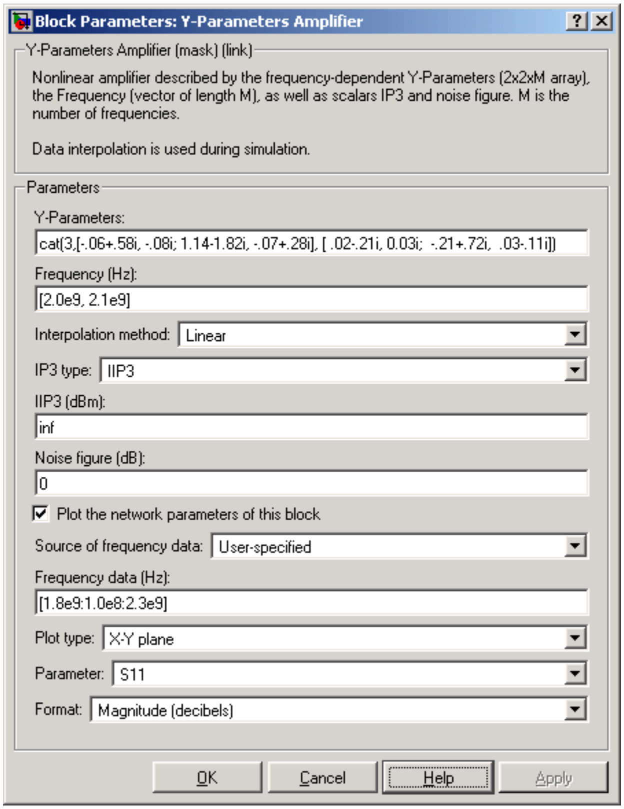
Note For information about plotting, see "Plotting Network Parameters" on page 4-1.

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 (mask) (link)

Nonlinear amplifier described by the frequency-dependent Y-Parameters (2x2xM array), the Frequency (vector of length M), as well as scalars IP3 and noise figure. M is the number of frequencies.

Data interpolation is used during simulation.

Parameters

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

Frequency (Hz): [2.0e9, 2.1e9]

Interpolation method: Linear

IP3 type: IIP3

IIP3 (dBm): inf

Noise figure (dB): 0

☑ Plot the network parameters of this block

Source of frequency data: User-specified

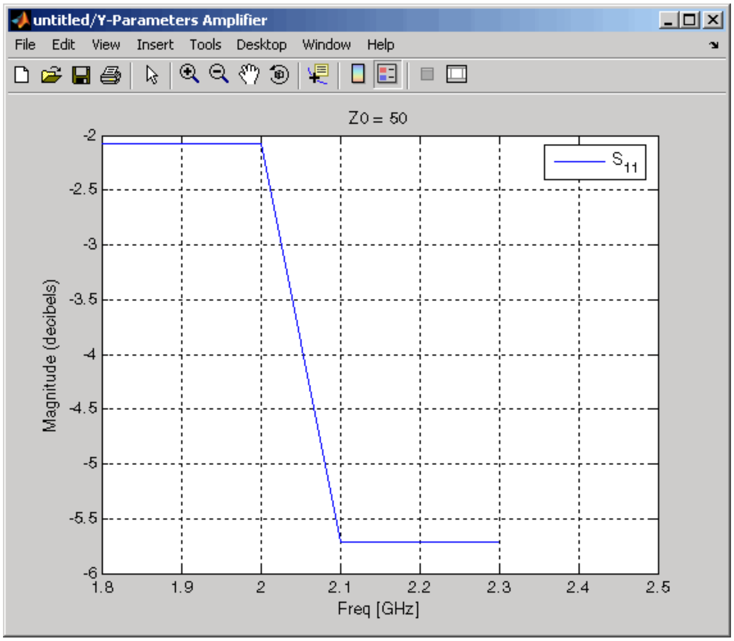
Frequency data (Hz): [1.8e9:1.0e8:2.3e9]

Plot type: X-Y plane

Parameter: S11

Format: Magnitude (decibels)

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

`y2s` (RF Toolbox)

`interp1` (MATLAB)

Y-Parameters Mixer

Purpose

Model a mixer using its Y-parameters

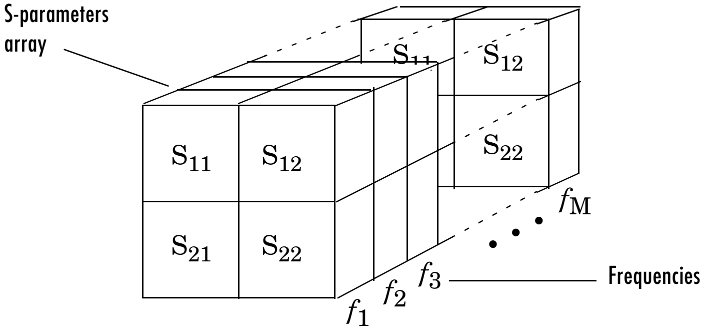
Library

Mixer sublibrary of the Physical library

Description

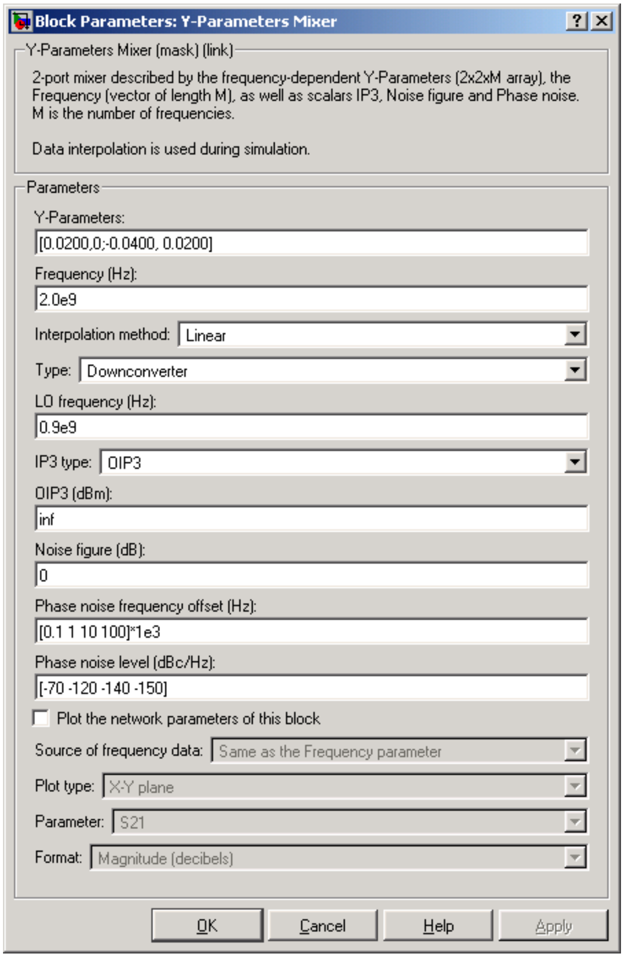
The Y-Parameters Mixer block models the nonlinear mixer described in the block dialog, in terms of its frequency-dependent Y-parameters, their frequencies, and the reference impedance of the Y-parameters. The block also takes phase noise into account.

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 Y-Parameters Mixer block uses the RF Toolbox `y2s` function to convert the Y-parameters to S-parameters, then interpolates the resulting S-parameters to determine the S-parameters at the simulation frequencies. The simulation frequencies are determined by the Output Port block. See “S-Parameters at Simulation Frequencies” on page 3-2 for more details.

Dialog Box



Block Parameters: Y-Parameters Mixer

Y-Parameters Mixer (mask) (link)

2-port mixer described by the frequency-dependent Y-Parameters (2x2xM array), the Frequency (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

Y-Parameters: [0.0200,0;-0.0400, 0.0200]

Frequency (Hz): 2.0e9

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)

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 4-1.

See Also

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

Y-Parameters Passive Network

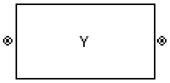
Purpose

Model a passive network using its Y-parameters

Library

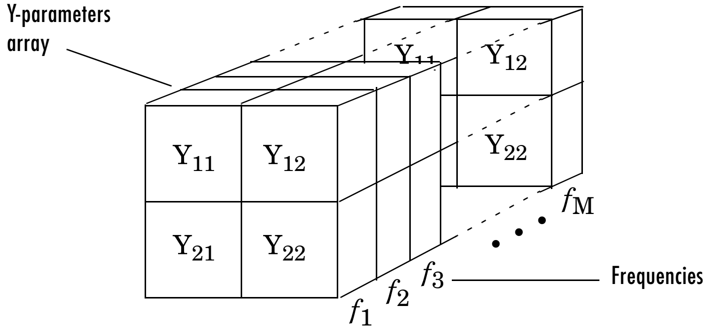
Black Box Elements sublibrary of the Physical library

Description



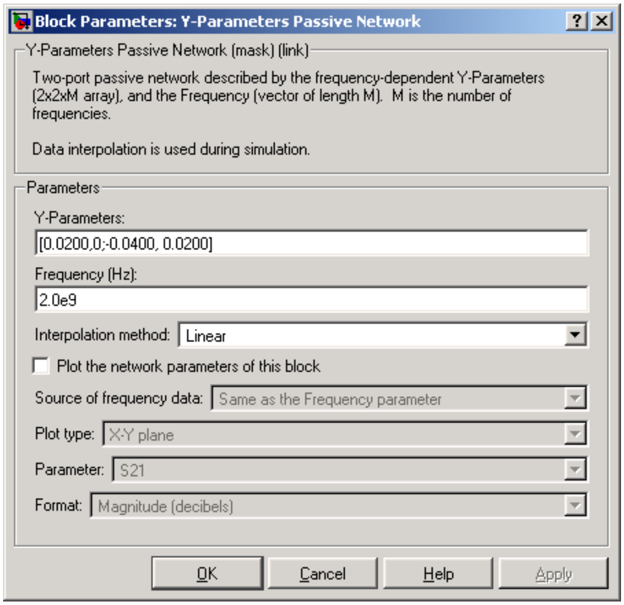
The Y-Parameters Passive Network block models the two-port passive network described in the block dialog, in terms of its Y-parameters and their associated frequencies.

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 Y-Parameters Passive Network block uses the RF Toolbox `y2s` function to convert the Y-parameters to S-parameters, then interpolates the resulting S-parameters to determine the S-parameters at the simulation frequencies. The simulation frequencies are determined by the Output Port block. See "S-Parameters at Simulation Frequencies" on page 3-2 for more details.

Dialog Box



Block Parameters: Y-Parameters Passive Network

Y-Parameters Passive Network (mask) (link)

Two-port passive network described by the frequency-dependent Y-Parameters (2x2xM array), and the Frequency (vector of length M). M is the number of frequencies.

Data interpolation is used during simulation.

Parameters

Y-Parameters: [0.0200,0;-0.0400, 0.0200]

Frequency (Hz): 2.0e9

Interpolation method: Linear

□ 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)

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`.

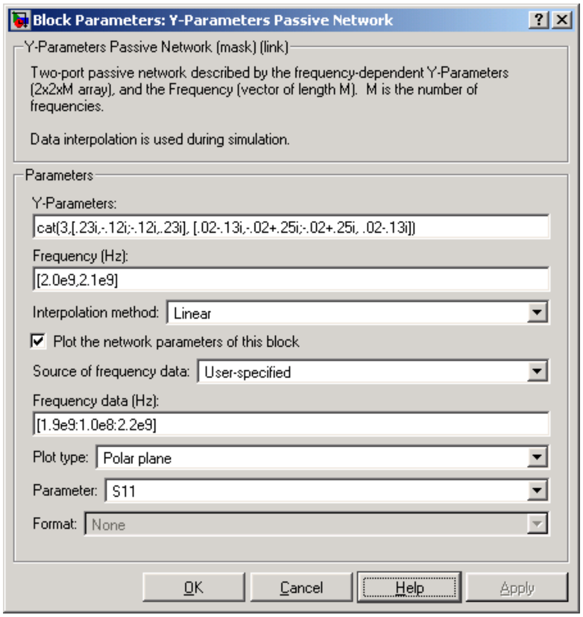
Note For information about plotting, see “Plotting Network Parameters” on page 4-1.

Examples

Plotting Parameters with the Y-Parameters Passive Network Block

The following example specifies Y-parameters [.23i, -.12i; -.12i, .23i] and [.02-.13i, -.02+.25i; -.02+.25i, .02-.13i] at frequencies 2.0 GHz and 2.1 GHz respectively. It uses the MATLAB `cat` function to create the 2-by-2-by-2 Y-parameters array.

```
cat(3,[.23i,-.12i;-.12i,.23i],...
       [.02-.13i,-.02+.25i;-.02+.25i, .02-.13i])
```



Block Parameters: Y-Parameters Passive Network

Y-Parameters Passive Network (mask) (link)

Two-port passive network described by the frequency-dependent Y-Parameters (2x2xM array), and the Frequency (vector of length M). M is the number of frequencies.

Data interpolation is used during simulation.

Parameters

Y-Parameters: cat(3,[.23i,-.12i;-.12i,.23i], [.02-.13i,-.02+.25i;-.02+.25i, .02-.13i])

Frequency (Hz): [2.0e9,2.1e9]

Interpolation method: Linear

☑ Plot the network parameters of this block

Source of frequency data: User-specified

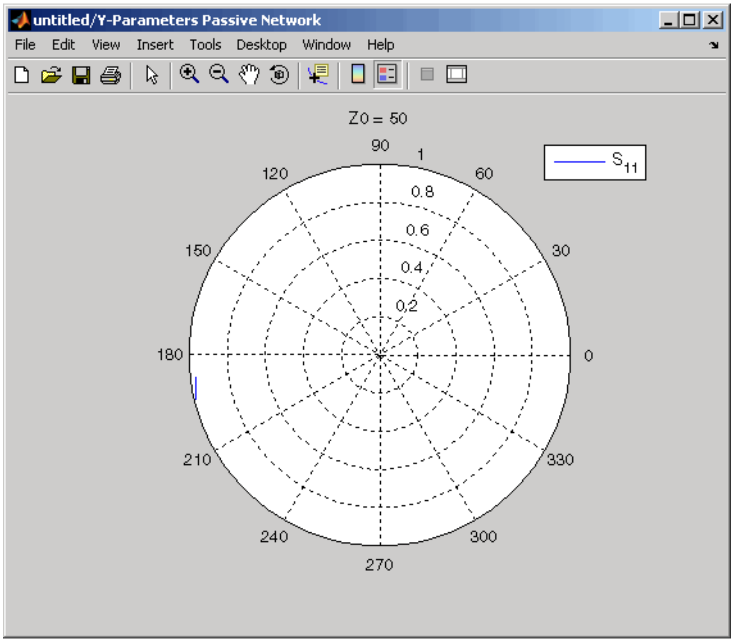
Frequency data (Hz): [1.9e9:1.0e8:2.2e9]

Plot type: Polar plane

Parameter: S11

Format: None

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.


untitled/Y-Parameters Passive Network
File Edit View Insert Tools Desktop Window Help
Z0 = 50
90
1
60
120
0.8
0.6
30
150
0.4
0.2
180
0
210
330
240
300
270
S_{11}

See Also

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

`y2s` (RF Toolbox)

`interp1` (MATLAB)

Z-Parameters Amplifier

Purpose

Model a nonlinear amplifier using its Z-parameters

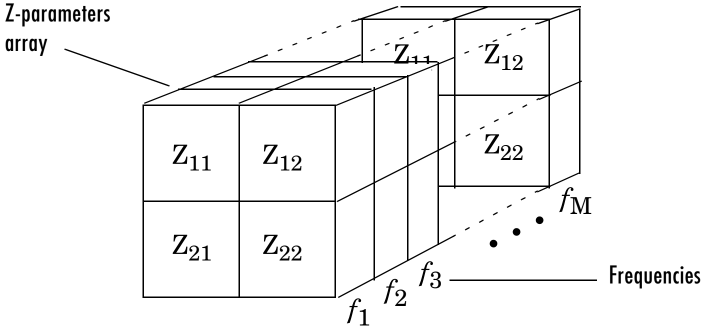
Library

Amplifiers sublibrary of the Physical library

Description

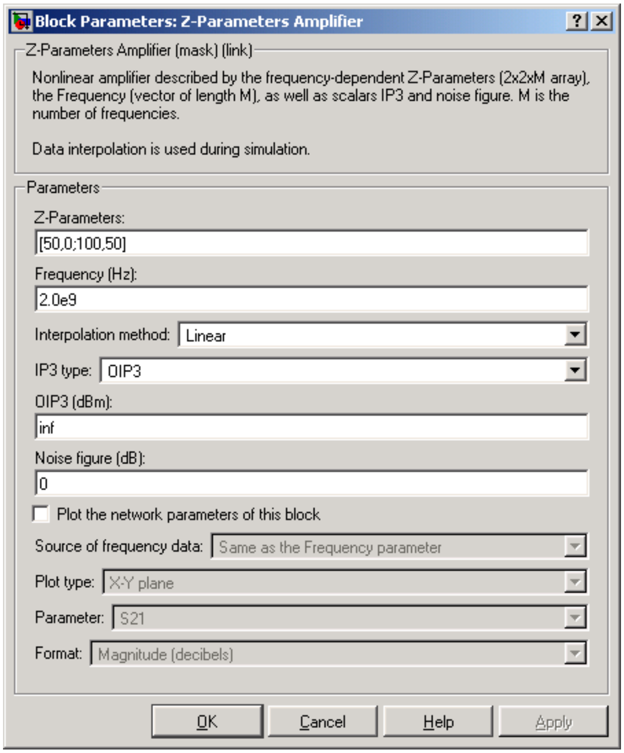
The Z-Parameters Amplifier block models the nonlinear amplifier described in the block dialog, in terms of its frequency-dependent Z-parameters and their frequencies. The block also takes into account the IP3 value and the noise figure.

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 Z-Parameters Amplifier block uses the RF Toolbox `y2s` function to convert the Z-parameters to S-parameters, then interpolates the resulting S-parameters to determine the S-parameters at the simulation frequencies. The simulation frequencies are determined by the Output Port block. See “S-Parameters at Simulation Frequencies” on page 3-2 for more details.

Dialog Box



Block Parameters: Z-Parameters Amplifier

Z-Parameters Amplifier (mask) (link)

Nonlinear amplifier described by the frequency-dependent Z-Parameters (2x2xM array), the Frequency (vector of length M), as well as scalars IP3 and noise figure. M is the number of frequencies.

Data interpolation is used during simulation.

Parameters

Z-Parameters: [50,0;100,50]

Frequency (Hz): 2.0e9

Interpolation method: Linear

IP3 type: OIP3

OIP3 (dBm): inf

Noise figure (dB): 0

□ 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)

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`.

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, $(S_i/N_i)/(S_o/N_o)$.

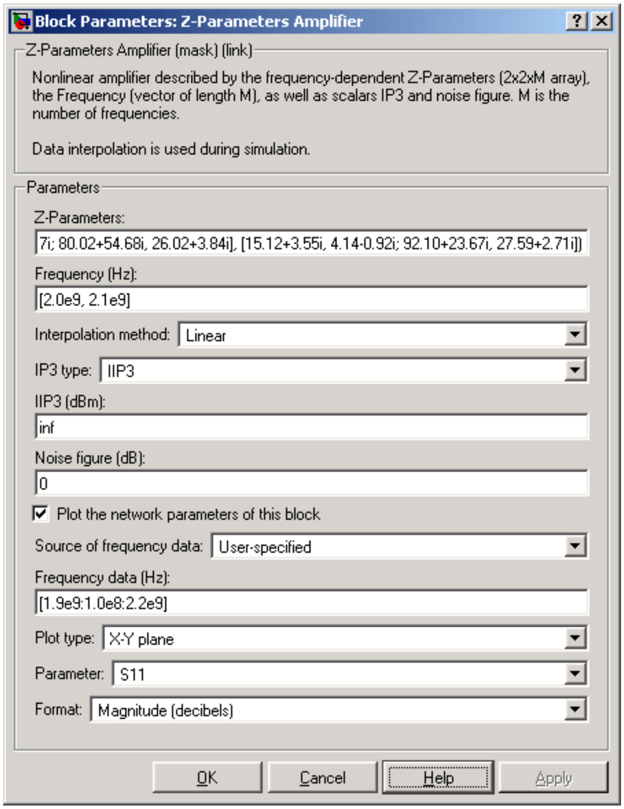
Note For information about plotting, see "Plotting Network Parameters" on page 4-1.

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])
```



Block Parameters: Z-Parameters Amplifier

Z-Parameters Amplifier (mask) (link)

Nonlinear amplifier described by the frequency-dependent Z-Parameters (2x2xM array), the Frequency (vector of length M), as well as scalars IP3 and noise figure. M is the number of frequencies.

Data interpolation is used during simulation.

Parameters

Z-Parameters: 7i; 80.02+54.68i, 26.02+3.84i], [15.12+3.55i, 4.14-0.92i; 92.10+23.67i, 27.59+2.71i])

Frequency (Hz): [2.0e9, 2.1e9]

Interpolation method: Linear

IP3 type: IIP3

IIP3 (dBm): inf

Noise figure (dB): 0

☑ Plot the network parameters of this block

Source of frequency data: User-specified

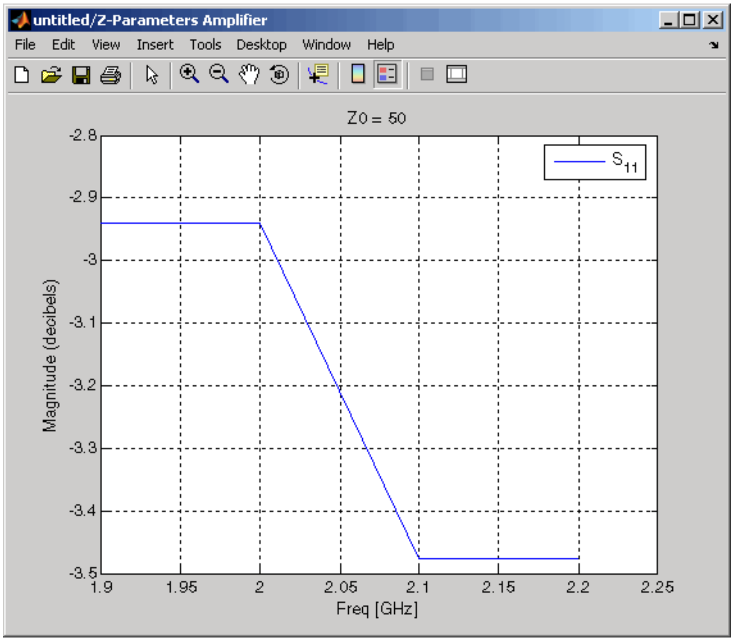
Frequency data (Hz): [1.9e9:1.0e8:2.2e9]

Plot type: X-Y plane

Parameter: S11

Format: Magnitude (decibels)

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.



See Also

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

`z2s` (RF Toolbox)

`interp1` (MATLAB)

Z-Parameters Mixer

Purpose

Model a mixer using its Z-parameters

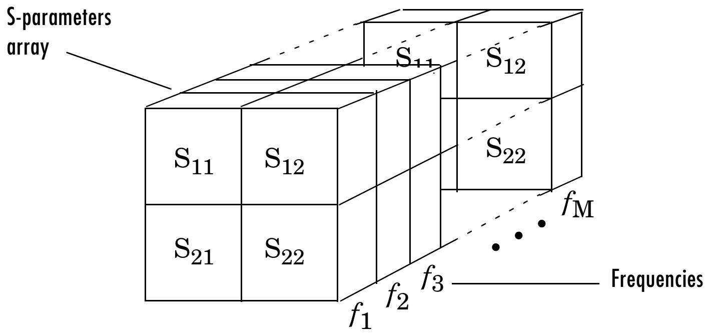
Library

Mixer sublibrary of the Physical library

Description

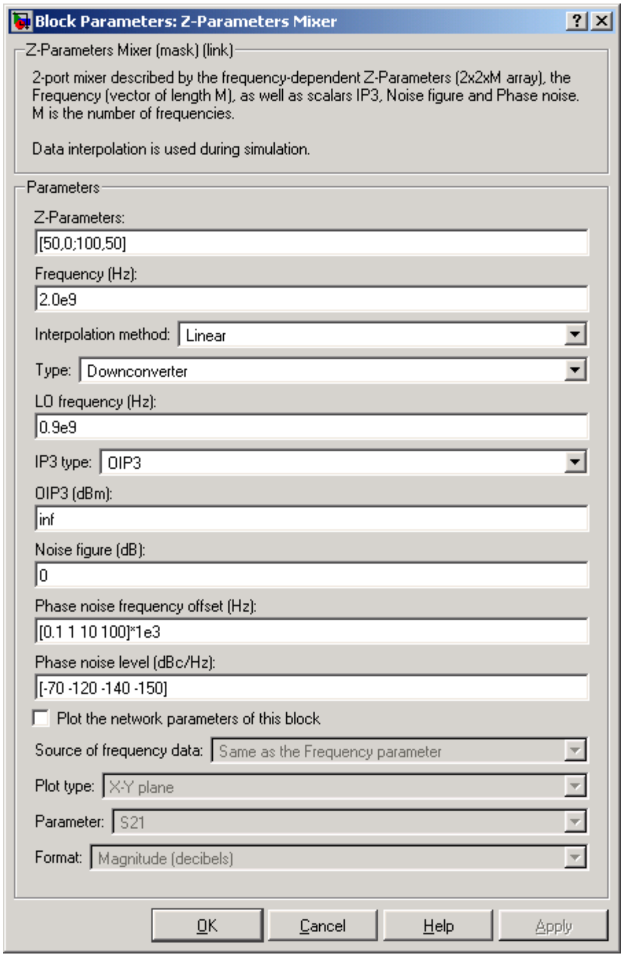
The Z-Parameters Mixer block models the nonlinear mixer described in the block dialog, in terms of its frequency-dependent Z-parameters, their frequencies, and the reference impedance of the Z-parameters. The block also takes phase noise into account.

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 Z-Parameters Mixer block uses the RF Toolbox `y2s` function to convert the Z-parameters to S-parameters, then interpolates the resulting S-parameters to determine the S-parameters at the simulation frequencies. The simulation frequencies are determined by the Output Port block. See “S-Parameters at Simulation Frequencies” on page 3-2 for more details.

Dialog Box



Block Parameters: Z-Parameters Mixer

Z-Parameters Mixer (mask) (link)

2-port mixer described by the frequency-dependent Z-Parameters (2x2xM array), the Frequency (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

Z-Parameters: [50,0;100,50]

Frequency (Hz): 2.0e9

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)

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.

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 4-1.

See Also

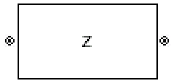
General Mixer, Output Port, S-Parameters Mixer, Y-Parameters Mixer

Z-Parameters Passive Network

Purpose Model a passive network using its Z-parameters

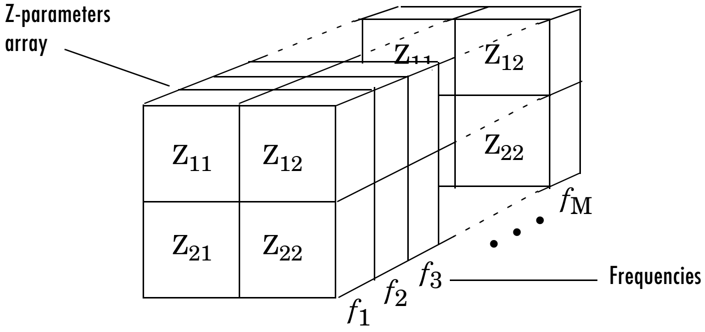
Library Black Box Elements sublibrary of the Physical library

Description



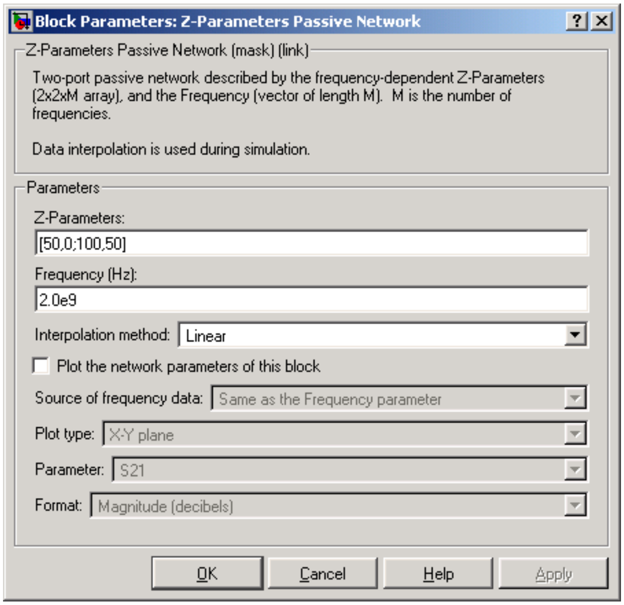
The Z-Parameters Passive Network block models the two-port passive network described in the block dialog, in terms of its Z-parameters and their associated frequencies.

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 uses the RF Toolbox y2s function to convert the Z-parameters to S-parameters, then interpolates the resulting S-parameters to determine the S-parameters at the simulation frequencies. The simulation frequencies are determined by the Output Port block. See "S-Parameters at Simulation Frequencies" on page 3-2 for more details.

Dialog Box



Block Parameters: Z-Parameters Passive Network

Z-Parameters Passive Network (mask) (link)

Two-port passive network described by the frequency-dependent Z-Parameters (2x2xM array), and the Frequency (vector of length M). M is the number of frequencies.

Data interpolation is used during simulation.

Parameters

Z-Parameters: [50,0;100,50]

Frequency (Hz): 2.0e9

Interpolation method: Linear

□ 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)

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`.

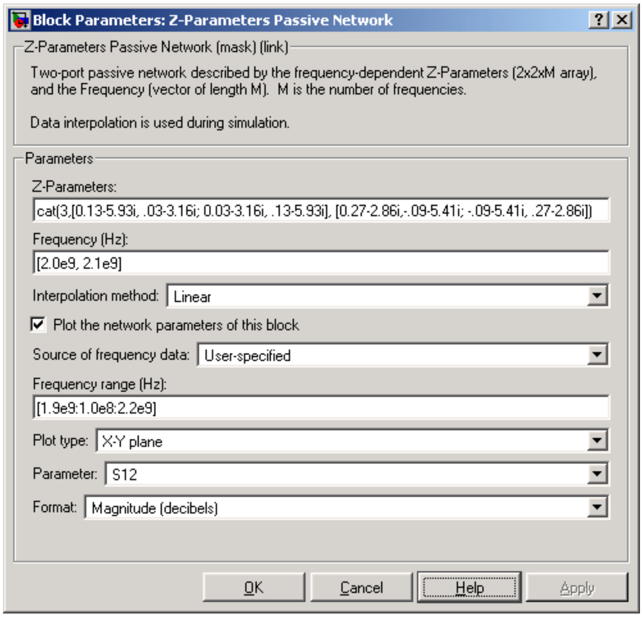
Note For information about plotting, see “Plotting Network Parameters” on page 4-1.

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])
```



Block Parameters: Z-Parameters Passive Network

Z-Parameters Passive Network (mask) (link)

Two-port passive network described by the frequency-dependent Z-Parameters (2x2xM array), and the Frequency (vector of length M). M is the number of frequencies.

Data interpolation is used during simulation.

Parameters

Z-Parameters: 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])

Frequency (Hz): [2.0e9, 2.1e9]

Interpolation method: Linear

☑ Plot the network parameters of this block

Source of frequency data: User-specified

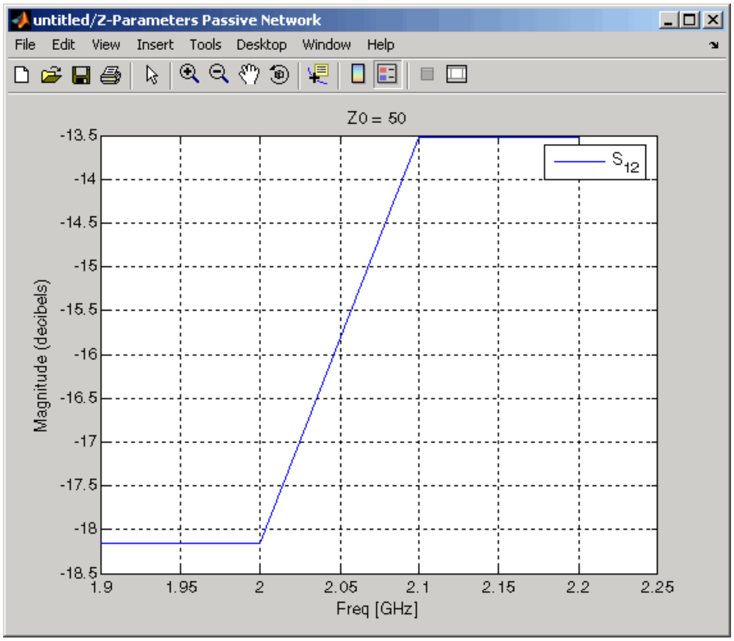
Frequency range (Hz): [1.9e9:1.0e8:2.2e9]

Plot type: X-Y plane

Parameter: S12

Format: Magnitude (decibels)

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.



See Also

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

`z2s` (RF Toolbox)

`interp1` (MATLAB)

Index