Agilent Technologies

# Advanced Design System 2011.01

Feburary 2011
Passive Circuit DesignGuide

**Acknowledgments**
Mentor Graphics is a trademark of Mentor Graphics Corporation in the U.S. and other countries. Mentor products and processes are registered trademarks of Mentor Graphics Corporation. * Calibre is a trademark of Mentor Graphics Corporation in the US and other countries. "Microsoft®, Windows®, MS Windows®, Windows NT®, Windows 2000® and Windows Internet Explorer® are U.S. registered trademarks of Microsoft Corporation. Pentium® is a U.S. registered trademark of Intel Corporation. PostScript® and Acrobat® are trademarks of Adobe Systems Incorporated. UNIX® is a registered trademark of the Open Group. Oracle and Java and registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners. SystemC® is a registered trademark of Open SystemC Initiative, Inc. in the United States and other countries and is used with permission. MATLAB® is a U.S. registered trademark of The Math Works, Inc.. HiSIM2 source code, and all copyrights, trade secrets or other intellectual property rights in and to the source code in its entirety, is owned by Hiroshima University and STARC. FLEXlm is a trademark of Globetrotter Software, Incorporated. Layout Boolean Engine by Klaas Holwerda, v1.7 http://www.xs4all.nl/~kholwerd/bool.html . FreeType Project, Copyright (c) 1996-1999 by David Turner, Robert Wilhelm, and Werner Lemberg. QuestAgent search engine (c) 2000-2002, JObjects. Motif is a trademark of the Open Software Foundation. Netscape is a trademark of Netscape Communications Corporation. Netscape Portable Runtime (NSPR), Copyright (c) 1998-2003 The Mozilla Organization. A copy of the Mozilla Public License is at http://www.mozilla.org/MPL/ . FFTW, The Fastest Fourier Transform in the West, Copyright (c) 1997-1999 Massachusetts Institute of Technology. All rights reserved.

The following third-party libraries are used by the NlogN Momentum solver:

"This program includes Metis 4.0, Copyright © 1998, Regents of the University of Minnesota", http://www.cs.umn.edu/~metis , METIS was written by George Karypis (karypis@cs.umn.edu).

Intel@ Math Kernel Library, http://www.intel.com/software/products/mkl

**Errata** The ADS product may contain references to "HP" or "HPEESOF" such as in file names and directory names. The business entity formerly known as "HP EEsof" is now part of Agilent Technologies and is known as "Agilent EEsof". To avoid broken functionality and to maintain backward compatibility for our customers, we did not change all the names and labels that contain "HP" or "HPEESOF" references.

**Warranty** The material contained in this document is provided "as is", and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Agilent disclaims all warranties, either express or implied, with regard to this documentation and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose.

Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Agilent and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.

**Technology Licenses** The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license. Portions of this product include the SystemC software licensed under Open Source terms, which are available for download at http://systemc.org/ . This software is redistributed by Agilent. The Contributors of the SystemC software provide this software "as is" and offer no warranty of any kind, express or implied, including without limitation warranties or conditions or title and non-infringement, and implied warranties or conditions merchantability and fitness for a particular purpose. Contributors shall not be liable for any damages of any kind including without limitation direct, indirect, special, incidental and consequential damages, such as lost profits. Any provisions that differ from this disclaimer are offered by Agilent only.

**Restricted Rights Legend** U.S. Government Restricted Rights. Software and technical data rights granted to the federal government include only those rights customarily provided to end user customers. Agilent provides this customary commercial license in Software and technical data pursuant to FAR 12.211 (Technical Data) and 12.212 (Computer Software) and, for the Department of Defense, DFARS 252.227-7015 (Technical Data - Commercial Items) and DFARS 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation).

# About Passive Circuit DesignGuide

The *Passive Circuit DesignGuide* documentation provides an introduction to the Passive Circuit DesignGuide. The complexity of the Advanced Design System (ADS) is made easily accessible through the automated capability. A first-time or casual ADS user can begin using the capability of ADS quickly, while experienced ADS users can perform tasks faster than ever before. The Step-by-Step Example describes how a microstrip line and a coupled-line filter can be designed and verified, and a layout generated, in a few minutes.

The Passive Circuit DesignGuide provides *SmartComponents* and automated-assistants for the design and simulation. All SmartComponents can be modified. You simply select a SmartComponent and, with little effort, redesign or verify their performance. The *Using SmartComponents* (dgpas) section answers many common questions relating to DesignGuide use. The section *Using Automated Assistants* (dgpas) introduces Automated Assistants.

## Step-by-Step Example

The step-by-step example takes you through the design, analysis and sensitivity simulation of a microstrip line and a coupled-line filter. After completing this example, you should have a basic understanding of the Utility and be ready to begin using the tool. Follow these steps to begin:

- Setting Up the Design Environment
- Designing and Analyzing a Network

> **Note**
> You should already be familiar with the basic features of Advanced Design System. For help with ADS basic features, refer to the *Schematic Capture and Layout* (usrguide) documentation.

## Setting Up the Design Environment

Before you can use the Passive Circuit DesignGuide, you must set up the design environment by using these steps:

- Setting DesignGuide Preferences
- Opening a Workspace
- Opening a Schematic Window
- Opening the Passive Circuit DesignGuide
- Displaying the SmartComponent Palette

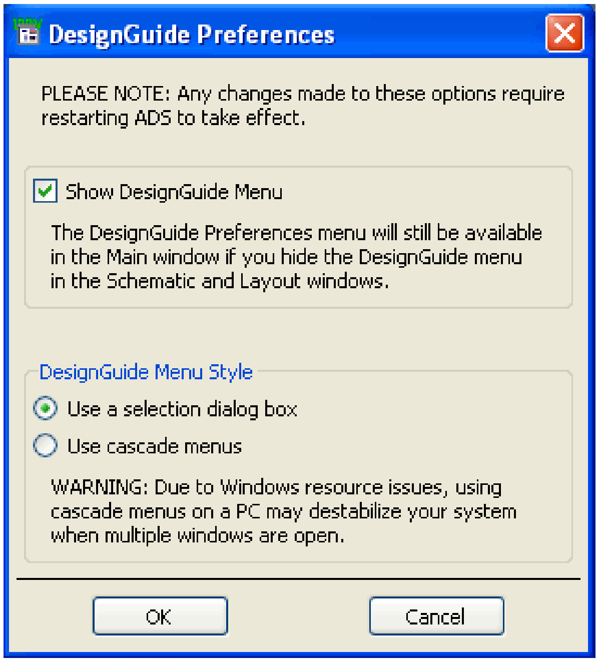
> **Note**
> Before beginning, you must have installed the DesignGuide with appropriate licensing codewords.

### Setting DesignGuide Preferences

All DesignGuides can be accessed through either cascading menus or dialog boxes. You can configure your preferred method in the ADS Main window or from the Schematic window.

To configure access through menus or dialog boxes:

1. From the Main or Schematic window, choose **DesignGuide** > **Preferences** .
2. In the DesignGuide Menu Style group box, choose either **Use a selection dialog box** or **Use cascade menus** .



DesignGuide Preferences

PLEASE NOTE: Any changes made to these options require restarting ADS to take effect.

☑ Show DesignGuide Menu

The DesignGuide Preferences menu will still be available in the Main window if you hide the DesignGuide menu in the Schematic and Layout windows.

DesignGuide Menu Style

● Use a selection dialog box

○ Use cascade menus

WARNING: Due to Windows resource issues, using cascade menus on a PC may destabilize your system when multiple windows are open.

3. Close and restart the program for your preference changes to take effect.

> **Note**
> On PC systems, Windows resource issues might limit the use of cascading menus. When multiple windows are open, your system could become destabilized. Therefore, the dialog box menu style might be best for these situations.

The ADS Main window DesignGuide menu contains these choices:

- **DesignGuide Developer Studio** > **Start DesignGuide Studio** is only available on this menu if you have installed the DesignGuide Developer Studio to open the initial Developer Studio dialog box.
- **DesignGuide Developer Studio** > **Developer Studio Documentation** is only available on this menu if you have installed the DesignGuide Developer Studio to open the DesignGuide Developer Studio documentation.

> **Note**
> Another way to access the DesignGuide Developer Studio documentation is by selecting *Help* > *Topics and Index* > *DesignGuides* > *DesignGuide Developer Studio* from any ADS program window.

- **Add DesignGuide** opens a directory browser in which you can add a DesignGuide to your installation. This is primarily intended for use with DesignGuides that are custom-built through the Developer Studio.
- **List/Remove DesignGuide** opens a list of your installed DesignGuides. Select any that you would like to uninstall and choose the *Remove* button.
- **Preferences** opens a dialog box that enables you to:
    - Disable the DesignGuide menu commands (all except Preferences) in the Main window by unchecking this box. In the Schematic and Layout windows, the

complete DesignGuide menu and all of its commands are removed if this box is unchecked.

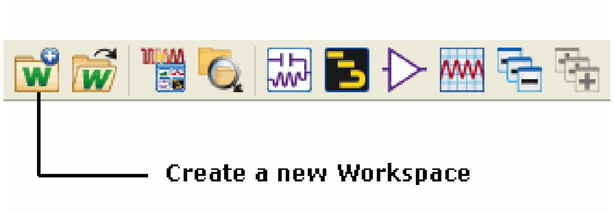
- Select your preferred interface method, either cascading menus or dialog boxes.

## Opening a Workspace

The ADS design environment is set up within a Workspace.

To create a new Workspace:

1. From the ADS Main window, choose **File** > **New** > **Workspace** or click **Create a New Workspace** on the toolbar.



2. In the dialog, define the location of the Workspace and assign a Workspace name.

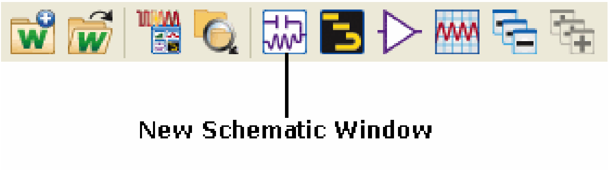
For more details on creating a new space, refer to *Using Workspace* (adstour).

## Opening a Schematic Window

A new schematic is needed to contain the lowpass component for this example.

To open a Schematic window:

1. From the ADS Main window, choose **Window** > **New Schematic** or click **New Schematic Window** on the toolbar. A new Schematic window appears.



> **Note**
> Depending on how your ADS preferences are set, a Schematic window can appear automatically when you create or open a Workspace.

2. In the New Schematic window, provide Library and Cell details to create a cell named *Example* .

For more details on creating a new schematic, refer *Using Designs* (adstour).

## Opening the Passive Circuit DesignGuide

The Passive Circuit DesignGuide is accessed from the **DesignGuide** menu in the Schematic window.

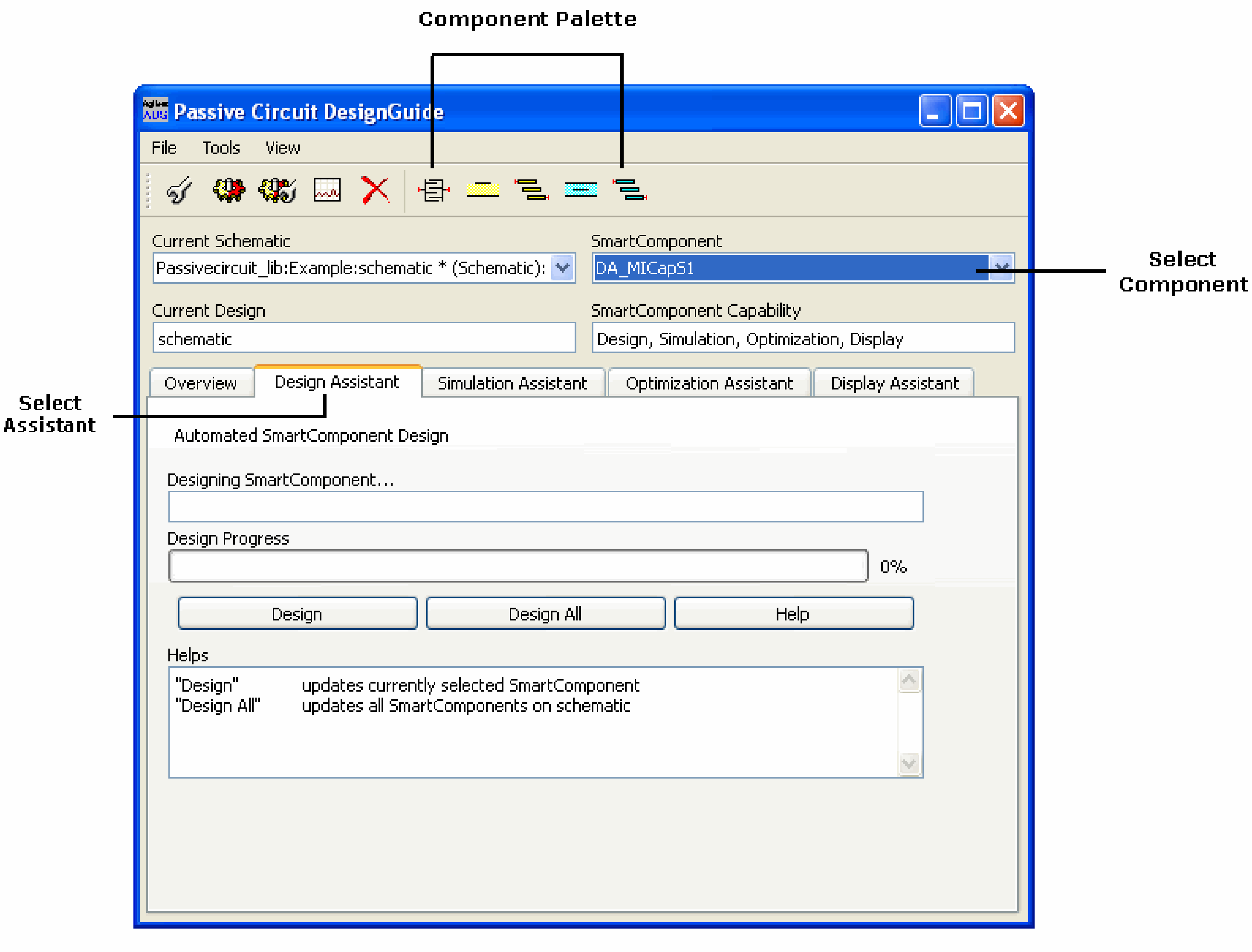
To open the Passive Circuit DesignGuide:

1. In the Schematic window, choose **DesignGuide** > **Passive Circuit** > **Passive Circuit Control Window**. The Control window opens.

## Using the Control Window

All Utility features are available from the Control window. The Control window houses menus, a toolbar, and SmartComponent manipulation controls. The menus and toolbar buttons perform the basic functions of design, delete, and display the SmartComponent palette. Full features are available from each of the tab pages on the window. The window can be placed anywhere on the screen. Explore each tab page by clicking on the tab at the top of each page. Explore the window menus as well to familiarize yourself with the basic Utility capabilities.

The pull-down lists at the top of the control window are designed to help you navigate multiple schematic windows and SmartComponents. You can use the Current Schematic drop-down list box to select any of the currently opened schematic windows. This field is updated any time the Passive Circuit Control Window is selected from the **Tools** menu. From the **SmartComponent** drop-down list box, you can select any of the SmartComponents on the currently selected schematic window.



Current Schematic: Passivecircuit_lib:Example:schematic * (Schematic):

SmartComponent: DA_MICapS1

Current Design: schematic

SmartComponent Capability: Design, Simulation, Optimization, Display

Automated SmartComponent Design

Designing SmartComponent...

Design Progress 0%

Helps

"Design" updates currently selected SmartComponent
"Design All" updates all SmartComponents on schematic

To close the Control window:

- Select **File** > **Exit DesignGuide** from the Control window menubar. (You can also close the window by clicking the **x** at the top of the window.)

Continue the step-by-step example by Designing and Analyzing a Network.

# Designing and Analyzing a Network

In this step-by-step example, you design a microstrip line, and complete the design, analysis, and optimization of a branch-line coupler. A microstrip line can be designed easily given a substrate definition, its characteristic impedance, and length. Using the Utility follows a normal design flow procedure:

- Select components needed for your design from the component palettes (Displaying the SmartComponent Palette) and place the component in your design (Placing Components in the Design).

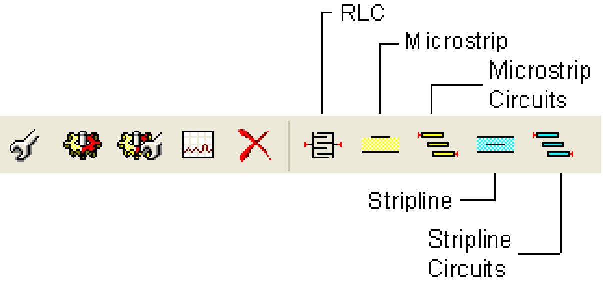
- Provide specifications (Changing Component Parameters).

- Design and analyze the component (Designing the SmartComponent and Optimizing the Branch-Line Coupler).

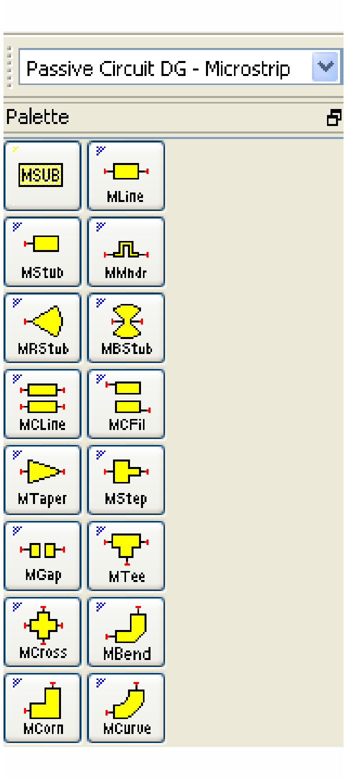
> **Note**
> Before starting this section of the step-by-step example, confirm your setup (Setting Up the Design Environment).

## Displaying the SmartComponent Palette

The DesignGuide contains six SmartComponent palettes that provide quick and easy access to the SmartComponents. The six available component palettes are:



- All contains all of the SmartComponents.
- Lines contains the simple line element SmartComponents.
- RLC contains the distributed resistor, inductor, and capacitor SmartComponents.
- Couplers contains the coupler and power divider SmartComponents.
- Filters contains the distributed filter SmartComponents.
- Match contains the distributed and lumped matching SmartComponents.
  A blue accent in the upper-left corner of a palette button indicates the component is a SmartComponent.



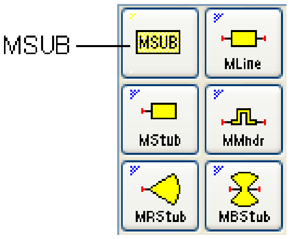
You can display the SmartComponent palettes in one of these ways:

- By clicking *Component Palette* on the Control window toolbar
- By choosing *View* > *Component Palette* from the Control window menu
- By selecting the palette from the Component Palette drop-down list box in the Schematic window toolbar (directly above the palette).

## Placing Components in the Design

To place a component in the design:

1. Display the **Passive Circuit - Microstrip** palette.
2. Click **MSUB** on the component palette to select the component.



3. Click within the schematic window to place the component.
    - You can change the orientation of the SmartComponent *before* placement by selecting from the **Insert > Component > Component Orientation** commands or by selecting **Rotate by -90** repeatedly from the schematic toolbar.
    - The place component mode remains active until you choose **Edit > End Command** from the schematic toolbar.

> **Note**
> When a SmartComponent is placed initially, a temporary component is used to place and specify the parameters for the SmartComponent. This component *does not* contain a subnetwork design. After the DesignGuide has been used to design the SmartComponent, the temporary component is replaced with a permanent component. The SmartComponent is renamed to *DA_ComponentName_DesignName* and an autogenerated design is placed inside the SmartComponent's subnetwork design file. Subsequently, if the SmartComponent parameters are edited, the DesignGuide must be used again to update the subnetwork design file.

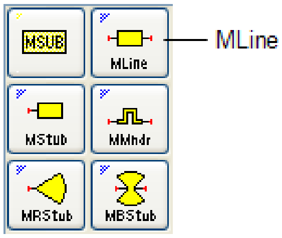
## Changing Component Parameters

To edit the component parameters:

1. In the Schematic window, double-click the **MSUB** component to open the component parameter dialog box.
2. Change the **substrate thickness (H)** to **20 mil** and the **dielectric constant (Er)** to **5** .

> **Note**
> See *Placing and Editing SmartComponents* (dgpas) for details on changing parameters in the design window or component dialog box.

To place the SmartComponent in the design:

1. Click **MLine** on the component palette to select the component.


2. Click within the schematic window to place the component.

To edit the component parameters:

1. In the Schematic window, double-click the **MLine** component to open the component parameter dialog.
2. Change the **center frequency (F)** to **5 GHz** , the **characteristic impedance (Zo)** to **75 Ohm** , and the **electrical length (Lelec)** to **0.25 wavelengths** .

## Designing the SmartComponent

You can design and analyze the SmartComponent from the Control Window.

To start the simulation:

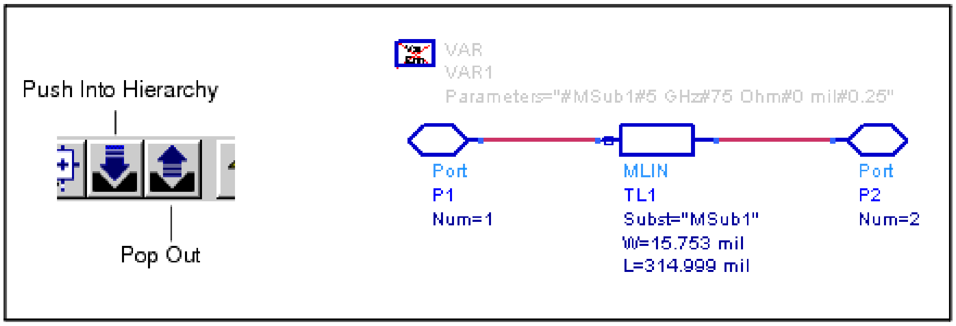
1. Select **DA_MLine1** from the SmartComponent drop-down list.
2. On the *Design Assistant* tab, click **Design** to start a simulation and generate the design for the SmartComponent.

## Examining the SmartComponent Design

You can look at the details of the autogenerated design inside the SmartComponent's subnetwork.

To examine the component's subnetwork:

1. Select the component **DA_MLine1**.
2. Click **Push Into Hierarchy** on the schematic toolbar.
3. After examining the design, click **Pop Out** on the schematic toolbar to close the view.



## Deleting the SmartComponent

- To delete the **DA_MLine1** SmartComponent, choose **Tools** > **Delete SmartComponent** from the Control window. *Do not delete the MSUB component* .

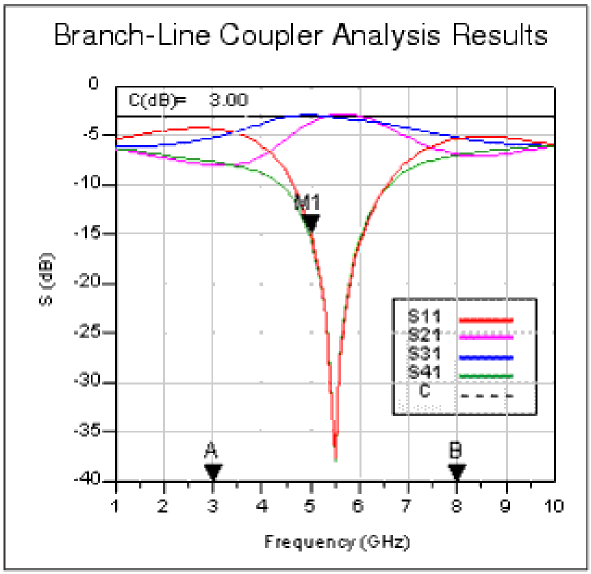
> **Note**
> The *Delete* button on the Control window is different from the *Delete* button on the ADS schematic window toolbar.

## Designing and Analyzing a Branch-Line Coupler

A branch-line coupler can be designed as easily as a microstrip line.

To design and analyze a branch-line coupler:

1. In the *Passive Circuit - Microstrip* palette, select **BLCplr** and place the component in the Schematic window.
2. Click the **BLCoupler** component and change the center frequency (F) to **5 GHz**.
3. Choose the **BLCoupler** component in the *SmartComponent* drop-down list box on the Control Window and then click the **Design Assistant** tab.
4. Click **Design** to generate the design for the SmartComponent.
5. Click the **Simulation Assistant** tab on the Control Window and enter **1 GHz** start frequency, **10 GHz** stop, **20 MHz** step (accept default display specifications).
6. Click **Simulate** to analyze the SmartComponent. The analysis results are shown here.



7. Close the Display window by choosing **File** > **Close Window** from the menu.

## Optimizing the Branch-Line Coupler

The branch-line coupler as designed in the preceding section has a center frequency of 5.5 GHz, which is different from the desired 5 GHz. The difference is due to limitations of the synthesis method used to generate the design. However, the Optimization Assistant can be used to optimize the design so that the center frequency is as specified.

To optimize the design:

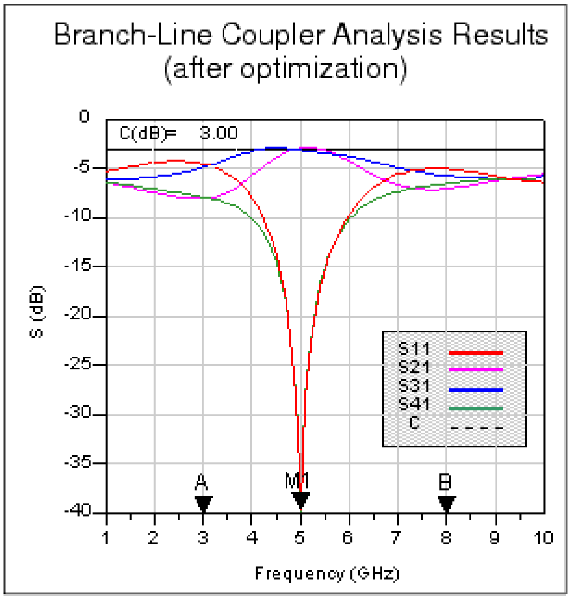
1. Click the **Optimization Assistant** tab on the Control Window and click **Optimize** to optimize the SmartComponent.
2. Click the Simulation Assistant tab on the Control Window.
3. Deselect the *Automatically display results* check box.
4. Click **Simulate** to re-analyze the branch-line coupler.

## Displaying Branch-Line Coupler Analysis Results

If a SmartComponent has been analyzed with the Simulation Assistant, the analysis results can be displayed using the Display Assistant. The results from the branch-line coupler designed and analyzed above can be displayed quickly by following these simple steps.

To display results:

1. Click the **Display Assistant** tab on the Control Window and click the **Display** button to display the existing simulation results.



2. Choose **File** > **Close Window** from the menu to close the Display window. This completes the step-by-step example.

# Using SmartComponents in Passive Circuit DesignGuide

This DesignGuide provides a large number of passive SmartComponents such as couplers, filters, lines and matching networks. SmartComponents are smart sub-network designs that provide the container for specification parameters and a schematic representation of the design when placed into a design. The DesignGuide provides automated design and analysis for these SmartComponents.

## Placing and Editing SmartComponents

SmartComponents can be placed, copied, edited and deleted like other components in the Advanced Design System. The basics of placement, copying, editing and deleting are described briefly in this section.

> **Note**
> For help with ADS basic features, refer to the *Schematic Capture and Layout* (usrguide) documentation.

The DesignGuide contains six SmartComponent palettes that provide quick and easy access to the SmartComponents. The six available component palettes are:

- All contains all of the SmartComponents.
- Lines contains the simple line element SmartComponents.
- RLC contains the distributed resistor, inductor, and capacitor SmartComponents.
- Couplers contains the coupler and power divider SmartComponents.
- Filters contains the distributed filter SmartComponents.
- Match contains the distributed and lumped matching SmartComponents.

The components are placed in the schematic by selecting the SmartComponent from the palette and clicking at the point where you want to place the component in the schematic.

You can display the SmartComponent palettes in one of these ways:

- By clicking *Component Palette* on the Control window toolbar.
- By choosing **View > Component Palette** from the Control window menu.
- By selecting the palette from the Component Palette drop-down list box in the Schematic window toolbar (directly above the palette).

### Placing SmartComponents

To place a SmartComponent in the design:

1. In the Schematic window, select the component from the SmartComponent palette.
2. Click within the design window at the location where you want to place the SmartComponent.
    - You can change the orientation of the SmartComponent *before* placement by selecting from the **Insert > Component > Component Orientation**

commands or by selecting **Rotate by -90** repeatedly from the schematic toolbar.

- The place component mode remains active until you choose **Edit > End Command** from the schematic toolbar.

## Changing Position and Orientation

A SmartComponent is moved by dragging it to any location in the Schematic window.

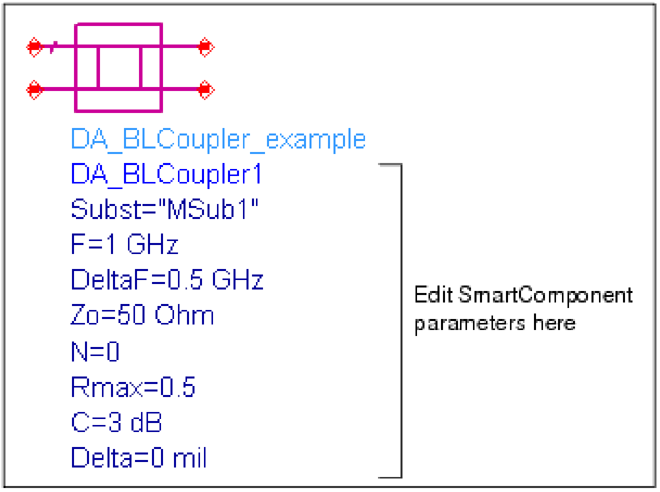
To change the component's orientation:

1. Select **Edit** > **Advanced Rotate** > **Rotate Around Reference** from the Schematic window or select **Rotate Items** from the toolbar.
2. Click the SmartComponent you want to use.
3. Rotate the component. The rotate mode remains active until you select **Edit > End Command** from the toolbar.

## Editing SmartComponents

You can modify the specifications in one of these ways:

- Click the SmartComponent parameters in the schematic window and change them (see The DA_BLCoupler Component.)
- Double-click the SmartComponent to open a dialog box containing all parameters.



DA_BLCoupler_example
DA_BLCoupler1
Subst="MSub1"
F=1 GHz
DeltaF=0.5 GHz
Zo=50 Ohm
N=0
Rmax=0.5
C=3 dB
Delta=0 mil

Edit SmartComponent parameters here

**The DA_BLCoupler Component**

The SmartComponent design (schematic) can be viewed by pushing into the SmartComponent's subnetwork. See *Examining the SmartComponent Design* (dgpas).

A SmartComponent subnetwork is empty until the design is generated (see the note in the section Placing and Editing SmartComponents).

# Copying SmartComponents

SmartComponents can be copied within a design, to another design, or to another Schematic window.

## Copying Within a Schematic

To copy a SmartComponent to the same schematic:

1. Click the SmartComponent to be copied.
2. Select **Edit** > **Copy** and then **Edit** > **Paste** from the schematic window.
3. Click where you want the copy placed.

## Copying Between Schematic Windows

To copy a SmartComponent to another schematic:

1. Click the SmartComponent to be copied.
2. Select **Edit** > **Copy** from the Schematic window.
3. Display the design or schematic window you want to copy the SmartComponent to.
4. Select **Edit** > **Paste** to copy the SmartComponent to the schematic.
5. Click where you want the component placed.

## Copying a SmartComponent as a Unique Design

Initially, all copied SmartComponents refer to the same SmartComponent design. When the *Design Assistant* is used to perform a design operation, the Design Assistant transforms each copied SmartComponent into a unique SmartComponent design. A design operation is accomplished from the Control window.

# Deleting SmartComponents

SmartComponents can be deleted from a design like other components, but completely removing a SmartComponent's files requires the actions described here.

## Deleting from Current Schematic

A SmartComponent can be deleted from a schematic in one of these ways:

- By selecting the component and pressing the **Delete** key.

- By selecting **Delete** from the toolbar.
- By selecting **Edit** > **Delete** from the schematic window.

> **Note**
> This procedure does not remove the SmartComponent files from the workspace directory. To delete files from the workspace directory, see Deleting from Current Workspace.

## Deleting from Current Workspace

To delete a SmartComponent and all associated files from your workspace:

1. In the Schematic window, select the SmartComponent.
2. In the Control window, select **Tools** > **Delete SmartComponent.** or on the toolbar, click **Delete** . This deletes the SmartComponent from the current design and removes all of its files from your workspace. The SmartComponent delete mode remains active until you select **Edit > End Command** from the schematic toolbar.

## Deleting Manually Using File System

You can use your computer's file system to delete a SmartComponent by deleting the appropriate files in the respective library of a Workspace. Delete files that start with *DA_* or *SA_* , contain the SmartComponent title, and end with . *ael, .atf,* or .wrk .

# Using SmartComponents as Standalone Components

After SmartComponents are designed and tested, they can be used as standalone components. The DesignGuide is not needed to use them in new designs unless you wish to modify or analyze them. When using the SmartComponent in a design, however, the power supply pins (Vdd, Vcc, Vp, Vm) must be connected to a DC voltage source whose voltage level corresponds the parameter setting.

## Using an Existing SmartComponent Within the Same Workspace

To use an existing SmartComponent within the same Workspace:

1. Open the Component Library window by selecting **Insert** > **Component** > **Component Library** from the Schematic window or **Display Component Library List** on the toolbar.
2. Select the Library name under **All Libraries** list at the left of the Component Library window. Available components are listed in the Components list at the right of the Component Library window.
3. Select the SmartComponent in the **Components** list.
4. Place the SmartComponent into your schematic by clicking in the Schematic window at the location you wish to place. The insert mode remains active until you click **End Command**.

## Using an Existing SmartComponent in Any Workspace

A library of predesigned reusable SmartComponents can be created by placing the reusable SmartComponents in a Workspace. This Workspace can be included in any Workspace and its SmartComponents can be accessed using the Component Library.

To use an existing SmartComponent in any Workspace:

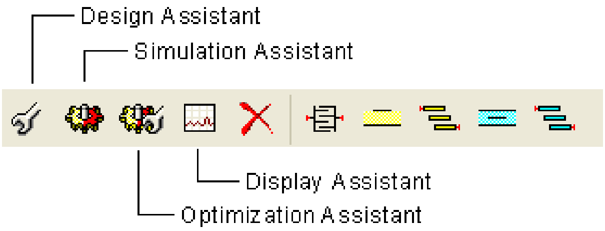
1. Open the Workspace where the SmartComponent needs to be inserted.
2. Open the Library in the Workspace by selecting **File > Open > Library**.
3. Open the Component Library window by selecting **Insert** > **Component** > **Component Library** from the Schematic window or **Display Component Library List** from the toolbar.
4. Select the Library name under **All Libraries** list at the left of the Component Library window. Available components are listed in the Components list at the right of the Component Library window.
5. Select the SmartComponent in the **Components** list.
6. Place the SmartComponent into your schematic by clicking in the Schematic window at the location where you wish to place the component. The insert mode remains active until you click **End Command**.

# Using Automated Assistants in Passive Circuit DesignGuide

This section describes the Automated Assistants available in this Utility.

## Automated Design and Analysis

The Automated Assistants provide quick design, simulation, yield analysis, and performance display for SmartComponents and enable transformation of lumped elements to transmission line elements. Four Automated Assistants are available in this Utility.



- Design Assistant is used to generate and update the design contained within a SmartComponent. The Design Assistant invokes a synthesis engine that generates a design from the given specification. Design Assistant designs and updates a single SmartComponent or all SmartComponents in a design.

- Simulation Assistant is used to analyze the design contained within a SmartComponent. The Simulation Assistant creates a simulation circuit containing the SmartComponent, then performs a simulation. The Simulation Assistant can automatically display the results of the simulation.

- Optimization Assistant is used to optimize the design contained within a SmartComponent. The Optimization Assistant creates an optimization circuit containing the SmartComponent, performs an optimization, and updates the SmartComponent.

- Display Assistant is used to easily and quickly display the performance of a SmartComponent. Display templates have been created for most of the SmartComponents. The display templates are preconfigured templates which provide a comprehensive look at the component's performance.

- Layout Generation. Artwork for all of the passive circuit SmartComponents in this DesignGuide can be generated automatically. The synthesis engine used by the Design Assistant creates a schematic for the SmartComponents that is auto-layout-generation ready. The Generate Layout capability of ADS is used to generate the artwork for the SmartComponents.

Explore each tab page by selecting the associated tab on the control window.

- **Automatic Layout Generation** . Artwork for all of the passive circuit SmartComponents in this DesignGuide can be automatically generated. The synthesis engine used by the Design Assistant creates a schematic for the SmartComponents that is auto-layout-generation ready. The Generate Layout capability of ADS is used to generate the artwork for the SmartComponents. Refer to "Layout Generation" for more information.

# Design Assistant

The Design Assistant is used to generate and update the design contained within a SmartComponent from the given specifications. It will design and update a single SmartComponent or all SmartComponents in a design.

The Design Assistant is accessed using the Passive Circuit DesignGuide Control window. From the Control window, full design control is enabled from the Design Assistant tab. Single component design operations can also be accomplished using the Control window menu and toolbar.

## Designing a Single Component

To design a single SmartComponent using the Control window, select the SmartComponent either from the SmartComponent drop-down list box in the upper right corner of the Control window or by clicking on the component in the Schematic window. The design is accomplished using one of the following methods:

- Click the **Design** button on the Design Assistant tab. The design progress is indicated on the tab page.
- Click the **Design** button on the Control window toolbar.
- Choose **Tools** > **Auto-Design** from the Control window menu.

## Designing a Multiple Component

Clicking the **Design All** button on the Design Assistant tab designs all SmartComponents on the current Schematic. Design progress is indicated on the tab page.

**Note**
To avoid screen flicker associated with the design, the Schematic window disappears during the process.

# Simulation Assistant

The Simulation Assistant is used to analyze the design contained within a SmartComponent. The Assistant creates a simulation circuit around the SmartComponent, then automatically performs the appropriate simulation. If set, the Assistant automatically displays the simulation results.

The Simulation Assistant is accessed using the Passive Circuit DesignGuide Control window. From the Control window, full simulation control is enabled from the Simulation

Assistant tab. Also, basic simulation can be accomplished using the Control window menu and toolbar.

For all simulation operations, the selected SmartComponent is designed if necessary, a simulation schematic is created, the simulation is performed, and the results are displayed. The simulation frequency sweep must be specified on the Simulation Assistant tab in the Control window.

> **Note**
> When the Simulation Assistant is used, the simulation schematic is deleted automatically. To retain the schematic that is created, instead of the Simulation Assistant, use the Create Template option described in Using Simulation Templates.

## Simulation Frequency Sweep

The simulation frequency sweep is specified on the Passive Circuit DesignGuide Control window. While performing the simulation from the Control window, select the Simulation Assistant tab and specify the sweep by entering the start frequency, stop frequency, and either frequency step size or number of points. The values entered are stored in the selected SmartComponent (as displayed in the *SmartComponent* drop-down list box) and are recalled each time this SmartComponent is selected.

> **Note**
> If a SmartComponent has been selected from the SmartComponent drop-down list box on the Control window, default frequencies are set for the component.

## Displaying Results Automatically

If you click the *Automatically Display Results* button on the Control window Simulation Assistant tab, the simulation results are displayed automatically after completion of the analysis.

## Simulating a SmartComponent

To simulate a SmartComponent using the Control window, select the SmartComponent either from the SmartComponent drop-down list box in the upper right corner of the Control window or by clicking on the component on the schematic window.

The simulation frequency sweep display option must be specified on the Simulation Assistant tab as previously described.

The simulation is then accomplished using one of the following methods:

- Click **Simulate** on the Simulation Assistant tab.
- Click **Simulate** on the Control window toolbar.
- Choose **Tools** > **Auto-Simulate** from the Control window menu.

## Using Simulation Templates

In some cases, such as when you would like to retain the schematic that is created, you can simulate the SmartComponent manually.

To generate a simulation schematic around the selected SmartComponent:

1. Click the **Create Template** button on the Control window Simulation Assistant tab.
2. You can examine or modify the simulation schematic, then manually start the simulation by choosing **Simulate** > **Simulate** from the Schematic window.
3. When you are finished, click the **Update from Template** button on the Simulation Assistant tab to transfer any changes you have made to the SmartComponent on the simulation schematic to the original SmartComponent and redesign if necessary.

> **Note**
> You can close the simulation schematic by choosing **File** > **Close Design** from the Schematic window menu, although this results in loss of any changes you have made to the SmartComponent.

# Optimization Assistant

The Optimization Assistant is used to optimize the design contained within a SmartComponent. It creates a optimization circuit containing the SmartComponent, then performs an optimization.

The assistant is accessed using the Passive Circuit DesignGuide Control window. From the Control window, full optimization control is enabled from the Optimization Assistant tab. Also, basic optimization can be accomplished using the Control window menu and toolbar.

The Optimization Assistant contains fields that indicate the objective of the optimization operation as well as the physical parameters to be altered during the process.

For all optimization operations, the selected SmartComponent is designed (if necessary), an optimization schematic is created, and the optimization is performed. The optimization results are transferred to the original SmartComponent, and this altered component is redesigned.

For each component, the optimization alters one or more of the physical design dimensions in order to make the component response more closely meet the specified performance.

## Optimizing a SmartComponent

To optimize a SmartComponent using the Control window, follow these steps:

1. Select the desired SmartComponent either from the SmartComponent drop-down list box in the upper right corner of the Control window or by clicking on the component on the schematic window.
2. Optimize the component by either:
    - Pushing the **Optimize** button on the Optimization Assistant tab

- Pushing the **Optimize** button on the Control window toolbar
- Selecting **Tools** > **Auto-Optimize** from the Control window menu

## Optimization Templates

In some cases it can be useful to manually optimize the SmartComponent.

To generate an optimization schematic around the selected SmartComponent, press the **Create Template** button on the Control window Optimization Assistant tab.

You can examine or modify the optimization schematic, then manually start the optimization by selecting **Simulate** > **Simulate** from the Schematic window.

When you are finished, selecting **Simulate** > **Update Optimization Values** causes the optimized values to appear in the *VAR* element in the schematic for your inspection.

Pressing the **Update from Template** button on the Optimization Assistant tab transfers the optimization results to the original SmartComponent and redesign.

> **Note**
> You can manually close the optimization schematic using **File** > **Close Design** from the Schematic window menu, although this will cause optimization results to be lost.

# Display Assistant

The Display Assistant is used to display the performance of a SmartComponent. The display templates are preconfigured display files that provide a comprehensive look at the performance of the component. You can create your own displays or modify the display templates included using the built in features of Advanced Design System. In most situations, the display templates included provide all the information you need.

The Display Assistant is accessed using the Passive Circuit DesignGuide Control window. From the Control window, full display control is enabled from the Display Assistant tab. Basic display selection can also be accomplished using the Control window menu and toolbar.

## Display Templates

The display templates are preconfigured templates that provide a comprehensive look at the performance of the component. Display templates have been created for most of the SmartComponents.

This includes all of the RLC, coupler, filter and matching components. The line components do not have auto-simulation, auto-optimization or auto-display capability because of their simplicity.

You can create your own displays or modify the included display templates using the built in features of Advanced Design System, but in most situations, the display templates included provide all the information you need.

The display templates opened by the Display Assistant have common features that are discussed here. For features unique to the display templates of some SmartComponents, refer to *SmartComponent Reference* (dgfilter).

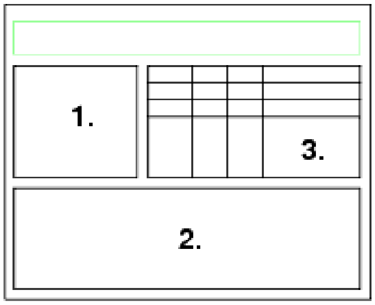
Also, in some cases you can use one of the display templates provided with the DesignGuide for other applications.

To open one of these templates, select the template from the Available Templates field and press the **Open Display Template** button on the Control window Display Assistant tab.

Then you can insert a dataset of your choice using the dataset pull-down list box in the upper left corner of the display. If you find that some parameters in the display template are not defined in the selected dataset, you can make appropriate modifications to the display. These changes can be saved using the commands in the display File menu.

## Basic Layout

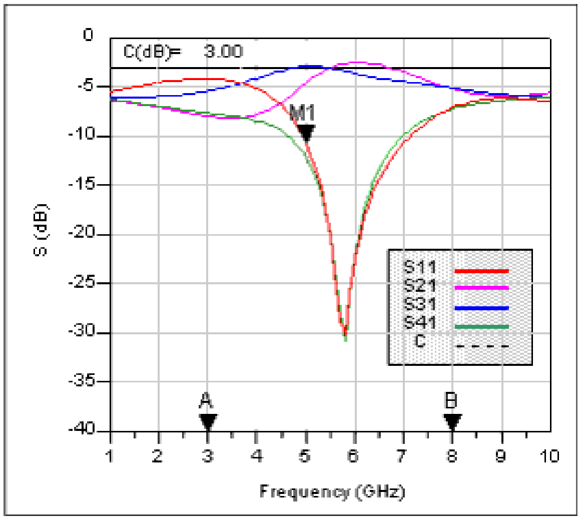
Basic Layout of Display Templates shows the basic layout of the display templates. Area one of the display template contains a graph of the most important parameters of the SmartComponent. Area two contains several graphs that give a comprehensive look at the component's performance. Area three contains a table listing the basic specifications and performance of the component.



**Basic Layout of Display Templates**

## Typical Area 1 Content

Typical Area 1 Graph shows a typical Area 1 graph. The frequency range of the graph is determined by the Simulation Assistant. As you change the frequency range in the Simulation Assistant, this graph updates appropriately.



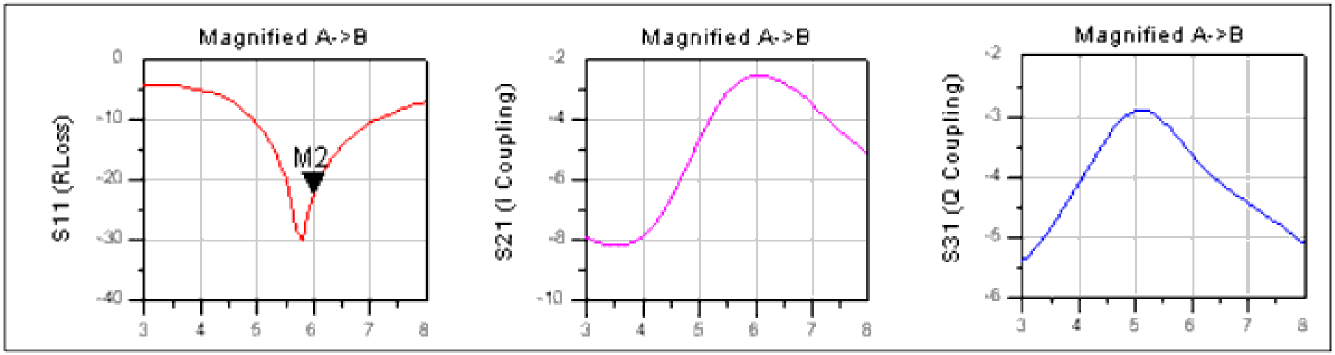
Typical Area 1 Graph

The markers A and B are used to define the frequency range of the graphs in Area 2. This feature is used to zero in on the region of interest and obtain a comprehensive look at the component's performance.

The marker M1 can be moved by dragging it with the mouse. The performance at the frequency given by M1 is shown in the table in Area 3.

## Typical Area 2 Content

Typical graphs from Area 2 are shown in Typical Graphs from Area 2. These graphs provide a quick, comprehensive look at the component's performance. Their frequency range is determined by the location of the "A" and "B" markers found in the main graph.

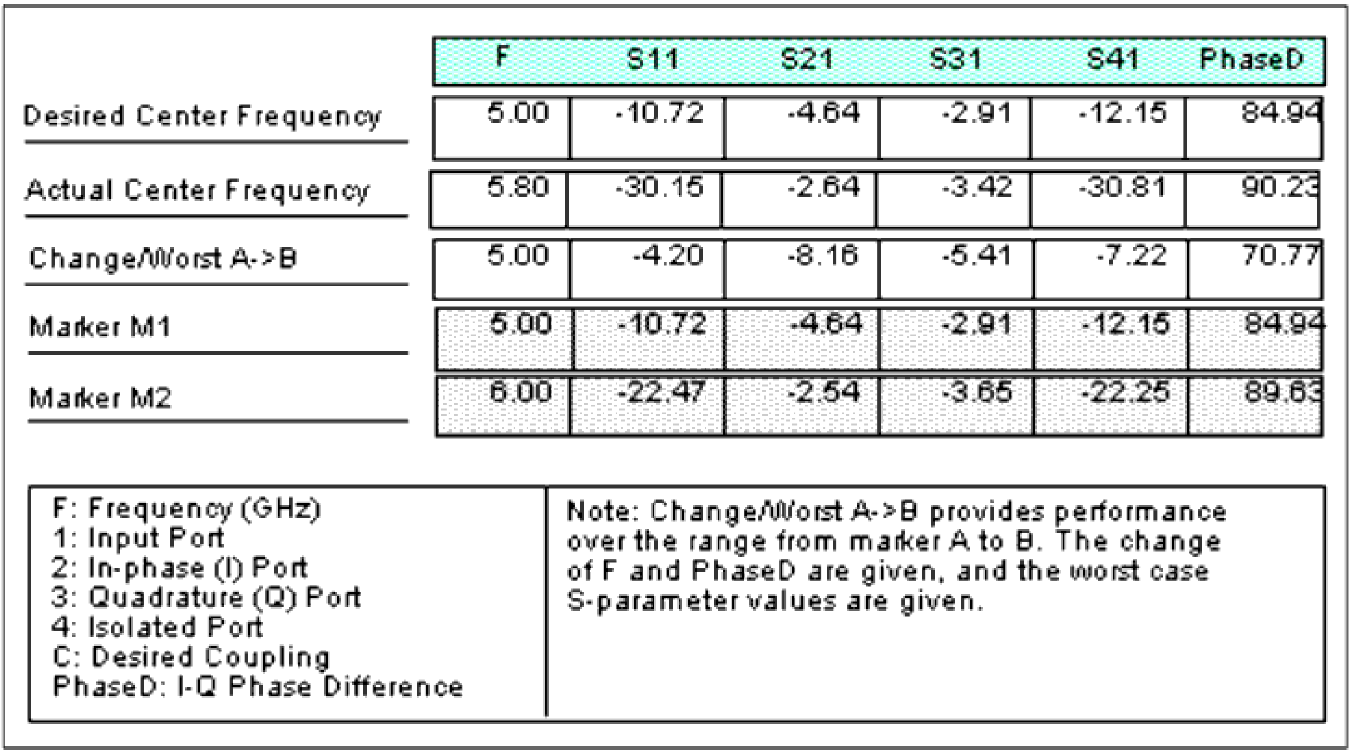


Typical Graphs from Area 2

Any markers such as M2 shown here can be moved by dragging them with the mouse. Performance criteria at the marker frequency is displayed in the table in Area 3.

## Typical Area 3 Content

A typical table from Area 3 is shown in Typical Table from Area 3. The white rows show the desired specifications and important performance criteria for the component. The gray rows give the performance criteria at the user defined marker frequencies. The box below the table provides explanatory information for the table.



| | F | S11 | S21 | S31 | S41 | PhaseD |
|---|---|---|---|---|---|---|
| Desired Center Frequency | 5.00 | -10.72 | -4.64 | -2.91 | -12.15 | 84.94 |
| Actual Center Frequency | 5.80 | -30.15 | -2.64 | -3.42 | -30.81 | 90.23 |
| Change/Worst A->B | 5.00 | -4.20 | -8.16 | -5.41 | -7.22 | 70.77 |
| Marker M1 | 5.00 | -10.72 | -4.64 | -2.91 | -12.15 | 84.94 |
| Marker M2 | 6.00 | -22.47 | -2.54 | -3.65 | -22.25 | 89.63 |

F: Frequency (GHz)
1: Input Port
2: In-phase (I) Port
3: Quadrature (Q) Port
4: Isolated Port
C: Desired Coupling
PhaseD: I-Q Phase Difference

Note: Change/Worst A->B provides performance over the range from marker A to B. The change of F and PhaseD are given, and the worst case S-parameter values are given.

**Typical Table from Area 3**

## Displaying SmartComponent Performance Results

Before using the Display Assistant, a valid dataset from a simulation of the selected SmartComponent must exist in the current workspace data directory. This simulation can be conveniently accomplished using the Simulation Assistant. Refer to Simulation Assistant for details on this step.

To display results from a SmartComponent simulation using the Control window, select the SmartComponent either from the SmartComponent drop-down list box in the upper right corner of the Control window or by clicking on the component on the schematic window. The display is then launched using one of the following methods:

- Push the **Display** button on the Display Assistant tab.
- Push the **Display** button on the Control window toolbar.
- Select **Tools** > **Auto-Display** from the Control window menu.

If no valid dataset exists for the selected SmartComponent, the **Display** button on the Display Assistant tab is insensitive. If the toolbar or menu are used to try to display the results, a message appears indicating that no dataset exists.

# Layout Generation

The Design Assistant creates a schematic for the SmartComponents that is ready for auto-layout generation. Artwork for all of the Passive Circuit DesignGuide SmartComponents can be automatically generated. The ADS Generate Layout capability is used to generate the artwork for the SmartComponents.

> **Note**
> You need an Advanced Design System Layout license to use this feature.

## Creating Layout Artwork

To create artwork for SmartComponents, follow these steps:

1. Choose and place the SmartComponents in the schematic window.
2. Specify the parameters for each SmartComponent.
3. Design the SmartComponents using the Design Assistant.
4. Select **Layout** > **Generate/Update Layout** from the Schematic window.
5. Choose **OK** in the Generate/Update Layout box.

The artwork for each SmartComponent and any other components that have associated artwork is displayed in the Layout window. If the status report checkbox is selected in the Generate/Update Layout box, a layout generation status report opens also.

## Updating Layout Artwork

To edit the properties of a SmartComponent and update the associated artwork, follow these steps:

1. Choose the SmartComponent in the schematic window.
2. Edit the parameters of the SmartComponent.
3. Design the SmartComponent using the Design Assistant.
4. Select **Layout** > **Generate/Update Layout** from the Schematic window.
5. Select **OK** in the Generate/Update Layout box.

The artwork for the SmartComponent is updated and displays in the layout window.

# Passive RLC Components

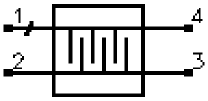
- *MICapP (Microstrip 4-port Interdigital Capacitor)* (dgpas)
- *MICapPG (Microstrip Grounded 2-port Interdigital Capacitor)* (dgpas)
- *MICapS (Microstrip 2-port Interdigital Capacitor)* (dgpas)
- *MICapSG (Microstrip 1-port Interdigital Capacitor)* (dgpas)
- *MREInd (Microstrip Elevated Rectangular Inductor)* (dgpas)
- *MRInd (Microstrip Rectangular Inductor) SmartComponent* (dgpas)
- *MSInd (Microstrip Spiral Inductor)* (dgpas)

- *MTFC (Microstrip Thin Film Capacitor) SmartComponent* (dgpas)
- *TFC (Thin Film Capacitor) SmartComponent* (dgpas)
- *TFR (Thin Film Resistor) SmartComponent* (dgpas)

**Note**
A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# MICapP (Microstrip 4-port Interdigital Capacitor)

**Symbol**



**Parameters**

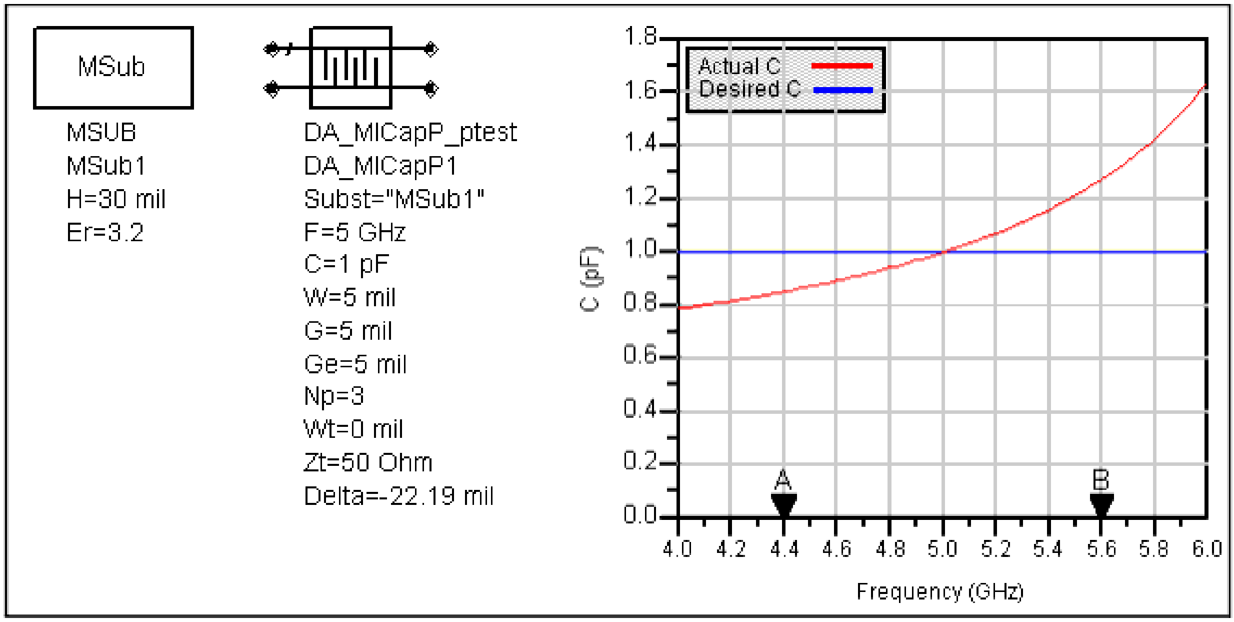
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| C | Capacitance | pF | 1 |
| W | Width of fingers | mil | 5 |
| G | Gap between fingers | mil | 5 |
| Ge | Gap at end of fingers | mil | 5 |
| Np | Number of finger pairs | None | 3 |
| Wt | Width of interconnect (0 if Zt specified) | mil | 0 |
| Zt | Characteristic impedance of interconnect lines; 0 if Wt specified | Ohm | 50 |
| Delta | Length added to fingers for tuning performance | mil | 0 |

**Notes**

1. MICapP designs a capacitance between two adjacent microstrip lines using interdigital fingers. The underlying design uses the MICAP2 component contained in the Tlines-Microstrip palette.
2. The design is accomplished using a simple model that specifies the length required to achieve the capacitance C at the design center frequency given the remaining physical parameters. If the computed length is unreasonable, decreasing or increasing the gap G will increase or decrease the capacitance, respectively and therefore allow altering of the length.
3. Both Wt and Zt specify the properties of the interconnect line. For proper operation, make sure that only one of these parameters is non-zero.
4. For more detailed discussion of the parameters W, G, Ge, Np, and Wt, please refer to the discussion of MICAP2 in the *ADS Microstrip Components* (ccdist) documentation.
5. The Optimization Assistant tunes the length of the fingers to achieve the desired capacitance. Because of the simple design approach used, it is often wise to first roughly tune the design within the Simulation Assistant and subsequently use the optimizer to perform the fine tuning.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

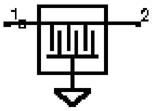
**Example**

A MICapP component was used to design a 1 pF capacitance between two 50 Ohm lines at a center frequency of 5 GHz. Optimization yielded a value of Delta = -22.19 mil.



# MICapPG (Microstrip Grounded 2-port Interdigital Capacitor)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| C | Capacitance | pF | 1 |
| W | Width of fingers | mil | 5 |
| G | Gap between fingers | mil | 5 |
| Ge | Gap at end of fingers | mil | 5 |
| Np | Number of finger pairs | None | 3 |
| Wt | Width of interconnect (0 if Zt specified) | mil | 0 |
| Zt | Characteristic impedance of interconnect lines; 0 if Wt specified | ohm | 50 |
| Delta | Length added to fingers for tuning performance | mil | 0 |

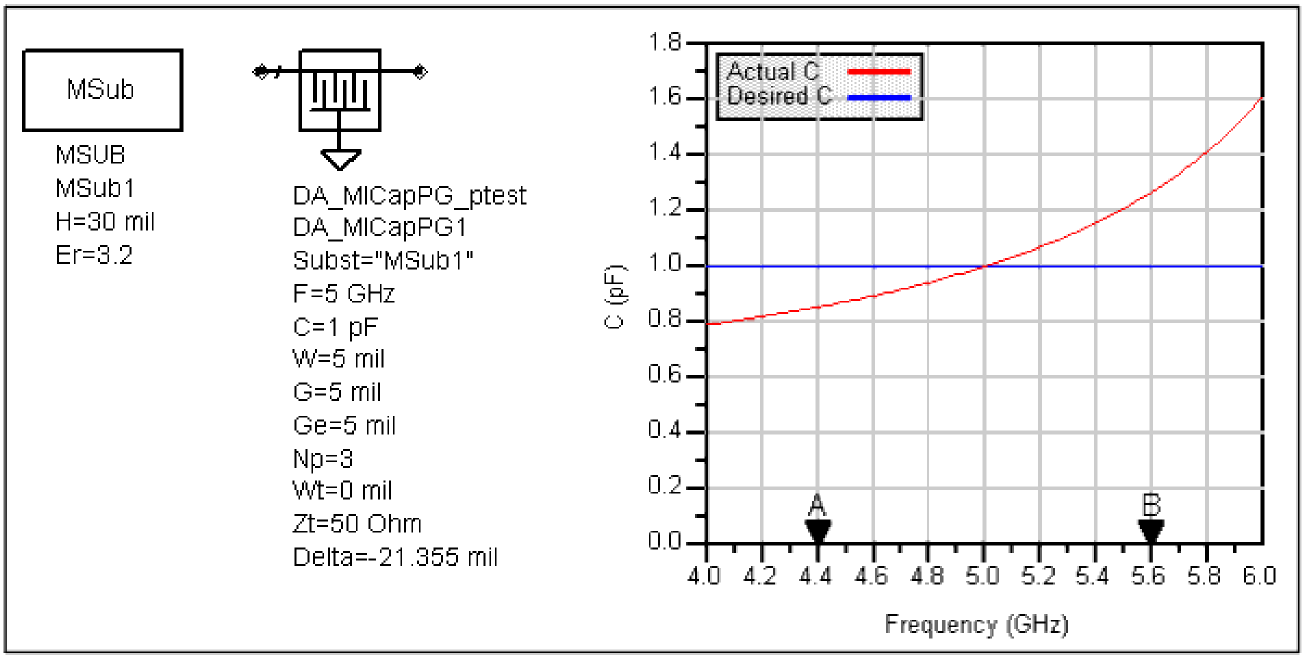
### Notes

1. MICapPG designs a capacitance between a microstrip line and ground using interdigital fingers. The underlying design uses the MICAP4 component contained in the Tlines-Microstrip palette.
2. The design is accomplished using a simple model that specifies the length required to achieve the capacitance C at the design center frequency given the remaining physical parameters. If the computed length is unreasonable, decreasing or increasing the gap G will increase or decrease the capacitance, respectively and therefore allow altering of the length.
3. Both Wt and Zt specify the properties of the interconnect line. For proper operation, make sure that only one of these parameters is non-zero.

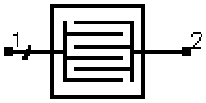
4. For more detailed discussion of the parameters W, G, Ge, Np, and Wt, please refer to the discussion of MICAP4 in the *ADS Microstrip Components* (ccdist) documentation.
5. The Optimization Assistant tunes the length of the fingers to achieve the desired capacitance. Because of the simple design approach used, it is often wise to first roughly tune the design within the Simulation Assistant and subsequently use the optimizer to perform the fine tuning.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

### Example

A MICapPG component was used to design a 1 pF capacitance from a 50 Ohm microstrip line and ground at a center frequency of 5 GHz. Optimization yielded a value of Delta = -21.355 mil.



# MICapS (Microstrip 2-port Interdigital Capacitor)



### Symbol

### Parameters

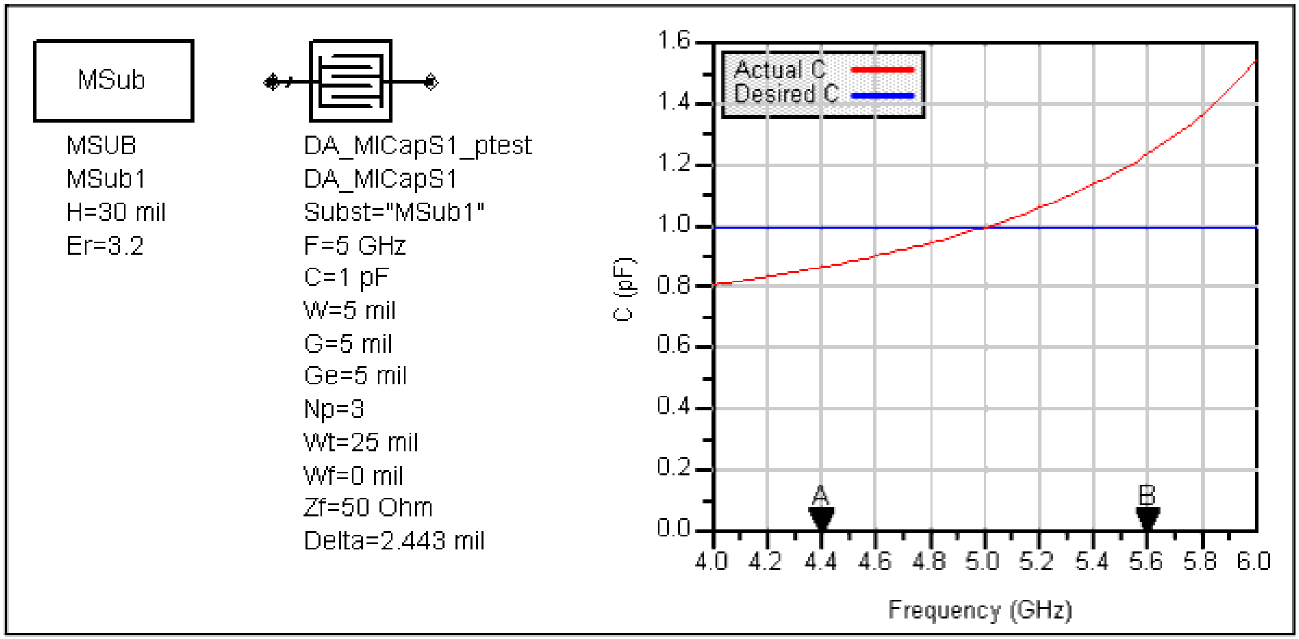
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| C | Capacitance | pF | 1 |
| W | Width of fingers | mil | 5 |
| G | Gap between fingers | mil | 5 |
| Ge | Gap at end of fingers | mil | 5 |
| Np | Number of finger pairs | None | 3 |
| Wt | Width of interconnect (0 if Zt specified) | mil | 25 |
| Wf | Width of feed line (0 if Zf specified) | mil | 0 |
| Zf | Characteristic impedance of feed line; 0 if Wt specified | Ohm | 50 |
| Delta | Length added to fingers for tuning performance | mil | 0 |

### Notes

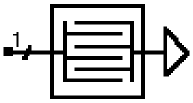
1. MICapS designs a series capacitance within a microstrip line using interdigital fingers. The underlying design uses the MICAP1 component contained in the Tlines-Microstrip palette.
2. The design is accomplished using a simple model that specifies the length required to achieve the capacitance C at the design center frequency given the remaining physical parameters. If the computed length is unreasonable, decreasing or increasing the gap G will increase or decrease the capacitance, respectively and therefore allow altering of the length.
3. Both Wf and Zf specify the properties of the feed line. For proper operation, make sure that only one of these parameters is non-zero.
4. For more detailed discussion of the parameters W, G, Ge, Np, Wt, and Wf, please refer to the discussion of MICAP1 in the *ADS Microstrip Components* (ccdist) documentation.
5. The Optimization Assistant tunes the length of the fingers to achieve the desired capacitance. Because of the simple design approach used, it is often wise to first roughly tune the design within the Simulation Assistant and subsequently use the optimizer to perform the fine tuning.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

### Example

A MICapS component was used to design a 1 pF series capacitance for a 50 Ohm line at a center frequency of 5 GHz. Optimization yielded a value of Delta = 2.443 mil.



# MICapSG (Microstrip 1-port Interdigital Capacitor)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| C | Capacitance | pF | 1 |
| W | Width of fingers | mil | 5 |
| G | Gap between fingers | mil | 5 |
| Ge | Gap at end of fingers | mil | 5 |
| Np | Number of finger pairs | None | 3 |
| Wt | Width of interconnect | mil | 25 |
| Wf | Width of feed line (0 if Zf specified) | mil | 0 |
| Zf | Characteristic impedance of feed line; 0 if Wf specified | Ohm | 50 |
| Delta | Length added to fingers for tuning performance | mil | 0 |

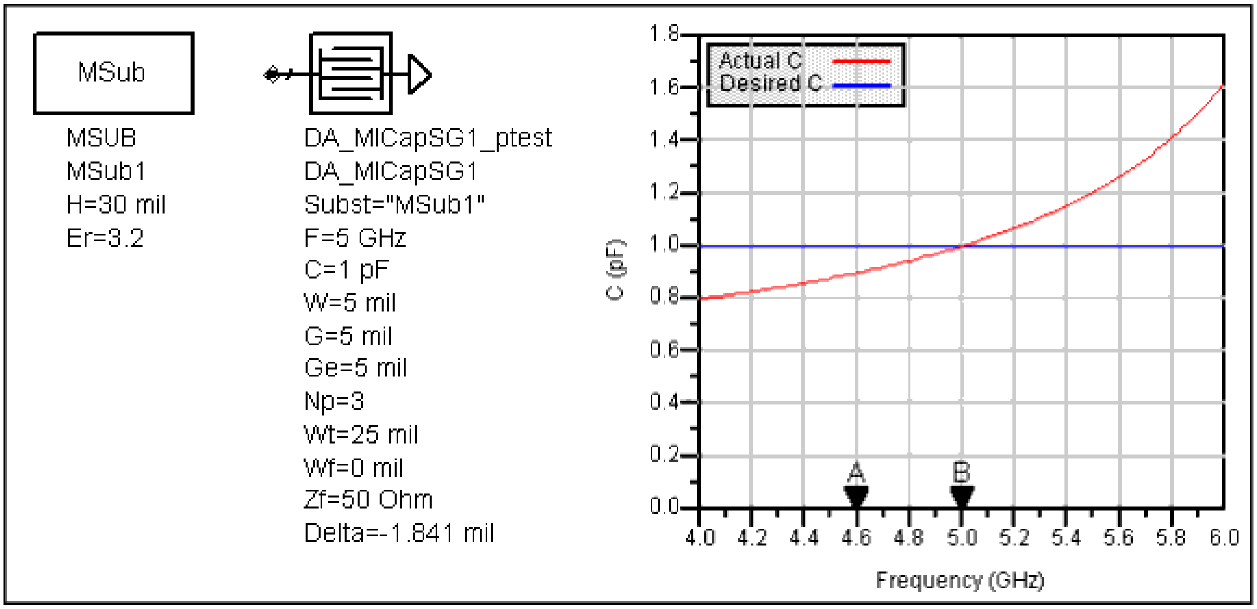
### Notes

1. MICapSG designs a series capacitance between a microstrip line and ground using interdigital fingers. The underlying design uses the MICAP3 component contained in the Tlines-Microstrip palette.
2. The design is accomplished using a simple model that specifies the length required to achieve the capacitance C at the design center frequency given the remaining physical parameters. If the computed length is unreasonable, decreasing or increasing the gap G will increase or decrease the capacitance, respectively and therefore allow altering of the length.
3. Both Wf and Zf specify the properties of the feed line. For proper operation, make sure that only one of these parameters is non-zero.

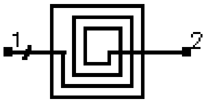
4. For more detailed discussion of the parameters W, G, Ge, Np, Wt, and Wf, please refer to the discussion of MICAP3 in the *ADS Microstrip Components* (ccdist) documentation.
5. The Optimization Assistant tunes the length of the fingers to achieve the desired capacitance. Because of the simple design approach used, it is often wise to first roughly tune the design within the Simulation Assistant and subsequently use the optimizer to perform the fine tuning.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

### Example

A MICapSG component was used to design a 1 pF capacitance for a 50 Ohm line at a center frequency of 5 GHz. Optimization yielded a value of Delta = -1.841 mil.



# MREInd (Microstrip Elevated Rectangular Inductor)



### Symbol

### Parameters

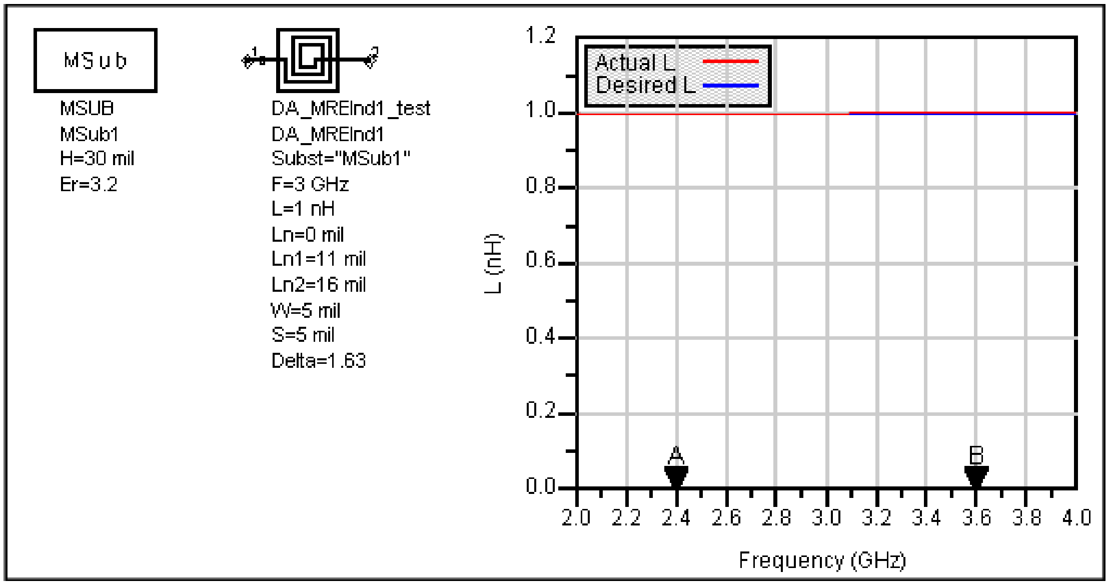
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| L | Inductance | nH | 1 |
| Ln | Length of innermost segment (0 means full length) | mil | 0 |
| Ln1 | Length of second innermost segment | mil | 2.5 |
| Ln2 | Length of second innermost segment | mil | 3.5 |
| W | Conductor width | mil | 1 |
| S | Conductor spacing | mil | 1 |
| Hi | Elevation of inductor above substrate | mil | 12.5 |
| Ti | Thickness of conductors | mil | 0.1 |
| Ri | resistivity (relative to gold) of conductors | None | 1 |
| Sx | spacing limit between support posts (0 to ignore posts) | mil | 0 |
| Cc | coefficient for capacitance of corner support posts | None | 2 |
| Cs | coefficient for capacitance of support posts along segment | None | 1 |
| Wu | width of underpass strip conductor | mil | 0.4 |
| Au | angle of departure from innermost segment | None | 0 |
| UE | extension of underpass beyond inductor | mil | 4 |
| Delta | incremental number of segments for tuning inductance (need not be integer) | None | 0 |

### Notes

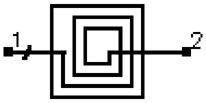
1. MREInd designs an elevated microstrip rectangular inductor. The underlying design uses the MRINDELA component contained in the Tlines-Microstrip palette.
2. The design is accomplished using a simple model that specifies the number of segments in the rectangular inductor required to achieve the inductance L at the design center frequency given the remaining physical parameters.
3. The parameters from Hi through Cs are not actually used in the design process, and therefore final tuning is required to achieve the proper value of the inductance.
4. The tuning parameter Delta represents the number of additional segments to add to the outside of the structure. If it is not an integer value, the outermost segment (L1) will not be full length, with the fractional remainder of Delta specifying the fractional length of this outermost segment. The design process may not produce the parameters that perfectly achieve the specified inductance. Manual tuning of the inductance can be performed by specifying the parameter Delta, which represents the number of turns to add to the design. For example, if the design process determines that the number of required turns is 1.2 and Delta is set to 0.3, the actual number of turns in the underlying inductor design will be 1.2+0.3=1.5. Delta can be positive or negative and, as shown in this example, need not be an integer.
5. The values Ln, Ln1, and Ln2 represent the lengths Ln, Ln-1, and Ln-2 associated with the MRINDELA component. For more detailed discussion of these lengths as well as the parameters from W through UE, please refer to the discussion of MRINDELA in the *ADS Microstrip Components* (ccdist) documentation.
6. Because of the difficulties associated with tuning the inductor using additional discrete segments, no Optimization Assistant is provided. However, tuning can be accomplished quite effectively by manually updating the value of Delta from within the Simulation Assistant. Refer to *Simulation Assistant* (dgpas), as well as the following example for more details.
7. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

### Example

A MREInd component was used to design a 1 nH inductor at a center frequency of 3 GHz. The design used a full-length innermost segment. By tuning the number of segments within the Simulation Assistant, it was determined that a value of Delta = 1.63 would achieve the desired inductance.



# MRInd (Microstrip Rectangular Inductor) SmartComponent



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| L | Inductance | nH | 1 |
| IndType | Inductance type (no bridge or wire bridge) | mil | No Bridge |
| Ln | Length of innermost segment (0 means full length) | mil | 0 |
| Ln1 | Length of second innermost segment | mil | 2.5 |
| Ln2 | Length of second innermost segment | mil | 3.5 |
| W | Conductor width | mil | 1 |
| S | Conductor spacing | mil | 1 |
| Dw | Diameter of bridge round wire (for MRINDWBR) | mil | 0.4 |
| Rb | Resistivity (relative to gold) of bridge wire (for wire bridge) | None | 1 |
| Hw | Height of wire above inductor (for wire bridge) | mil | 15 |
| Aw | Angle of departure from innermost segment (for wire bridge) | None | 0 |
| WE | Extension of bridge beyond inductor (for wire bridge) | mil | 4 |
| Delta | Incremental number of segments for tuning inductance (need not be integer) | None | 0 |

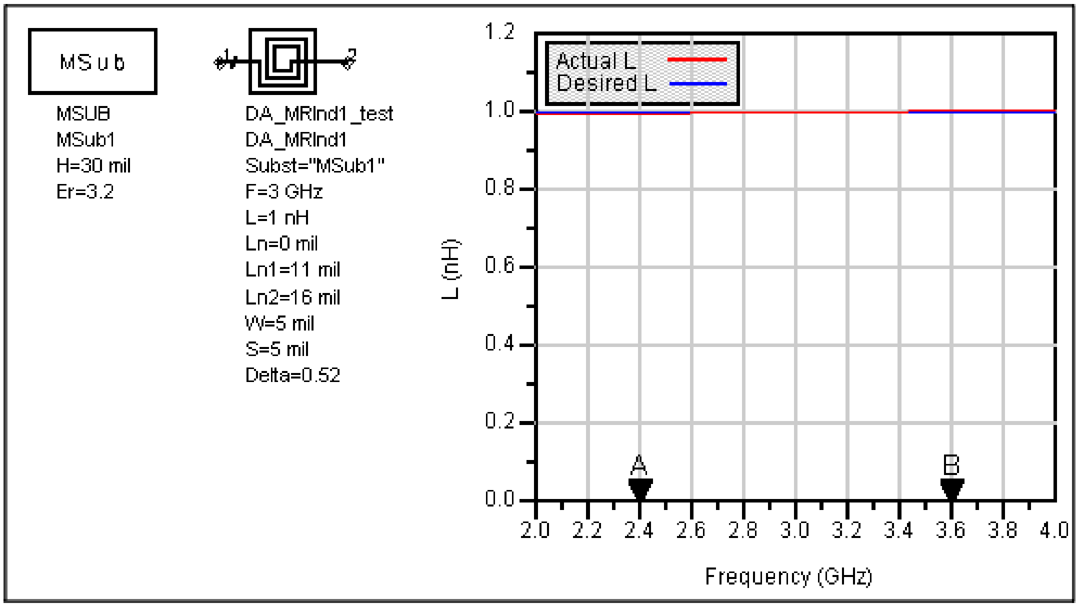
### Notes

1. MRInd designs a microstrip rectangular inductor. The underlying design uses the MRINDNBR and MRINDWBR components contained in the Tlines-Microstrip palette.

2. The design is accomplished using a simple model that specifies the number of segments in the rectangular inductor required to achieve the inductance L at the design center frequency given the remaining physical parameters.
3. The parameters Rb and Hw are not actually used in the design process, and therefore final tuning is required to achieve the proper value of the inductance.
4. The value of IndType specifies the type of inductor that will be designed. If this parameter is set to "No Bridge", then the MRINDNBR component is designed and the parameters from Dw to WE are ignored. If the parameter is set to "Wire Bridge", then the MRINDWBR component is designed and the parameters from Dw to WE are used.
5. The tuning parameter Delta represents the number of additional segments to add to the outside of the structure. If it is not an integer value, the outermost segment (L1) will not be full length, with the fractional remainder of Delta specifying the fractional length of this outermost segment. The design process may not produce the parameters that perfectly achieve the specified inductance. Manual tuning of the inductance can be performed by specifying the parameter Delta, which represents the number of turns to add to the design. For example, if the design process determines that the number of required turns is 1.2 and Delta is set to 0.3, the actual number of turns in the underlying inductor design will be 1.2+0.3=1.5. Delta can be positive or negative and, as shown in this example, need not be an integer.
6. The values Ln, Ln1, and Ln2 represent the lengths Ln, Ln-1, and Ln-2 associated with the MRINDNBR and MRINDWBR components. For more detailed discussion of these lengths as well as the parameters W through WE, please refer to the discussion of these components in the *ADS Microstrip Components* (ccdist) documentation.
7. Because of the difficulties associated with tuning the inductor using additional discrete segments, no Optimization Assistant is provided. However, tuning can be accomplished quite effectively by manually updating the value of Delta from within the Simulation Assistant, refer to *Simulation Assistant* (dgpas), as well as the following example for more details.
8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

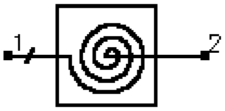
### Example

A MRInd component with no bridge was used to design a 1 nH inductor at a center frequency of 3 GHz. The design used a full-length innermost segment. By tuning the number of segments within the Simulation Assistant, it was determined that a value of Delta = 0.52 would achieve the desired inductance.


MSUB
MSub1
H=30 mil
Er=3.2

DA_MRInd1_test
DA_MRInd1
Subst="MSub1"
F=3 GHz
L=1 nH
Ln=0 mil
Ln1=11 mil
Ln2=16 mil
W=5 mil
S=5 mil
Delta=0.52

# MSInd (Microstrip Spiral Inductor)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| L | Inductance | nH | 10 |
| Ri | Inner radius measured to center of conductor | mil | 50 |
| W | Conductor width | mil | 10 |
| S | Conductor spacing | mil | 10 |
| W1 | Width of strip connected to pin 1 | mil | 10 |
| W2 | Width of strip connected to pin 2 | mil | 10 |
| Delta | Incremental number of turns for tuning inductance (need not be integer) | None | 0 |

### Notes

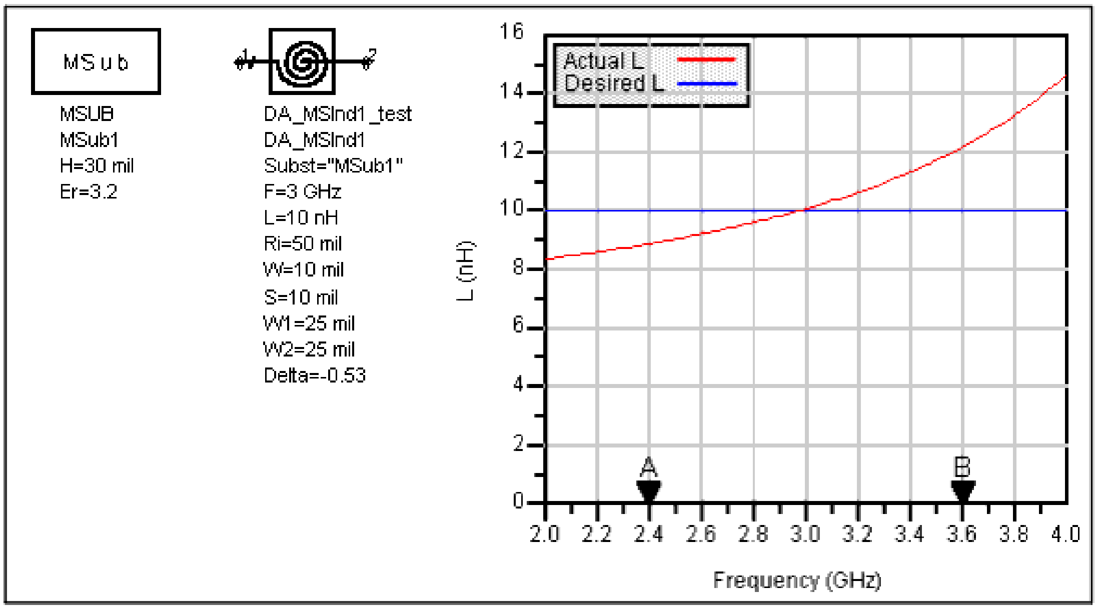
1. MSInd designs a microstrip spiral inductor. The underlying design uses the MSIND component contained in the Tlines-Microstrip palette.
2. The design is accomplished using a simple model that specifies the number of turns in the spiral inductor required to achieve the inductance L at the design center frequency given the remaining physical parameters.
3. The value of Ri specifies the distance from the center of the inductor to the center of the conductor at its innermost point in the spiral. Refer to the discussion of the MSIND component in the *ADS Microstrip Components* (ccdist) documentation for a more detailed discussion of this parameter.
4. The tuning parameter Delta represents the number of additional turns to add to the outside of the structure. Fractional numbers of turns are accommodated (i.e. Delta need not be an integer value). The design process may not produce the parameters that perfectly achieve the specified inductance. Manual tuning of the inductance can

be performed by specifying the parameter Delta, which represents the number of turns to add to the design. For example, if the design process determines that the number of required turns is 1.2 and Delta is set to 0.3, the actual number of turns in the underlying inductor design will be 1.2+0.3=1.5. Delta can be positive or negative and, as shown in this example, need not be an integer.

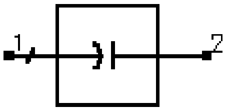
5. The Optimization Assistant tunes the number of turns to achieve the desired inductance.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

### Example

A MSInd component was used to design a 1 nH inductor at a center frequency of 3 GHz. Optimization yielded a value of Delta = -0.53.



## MTFC (Microstrip Thin Film Capacitor) SmartComponent



### Symbol

### Parameters

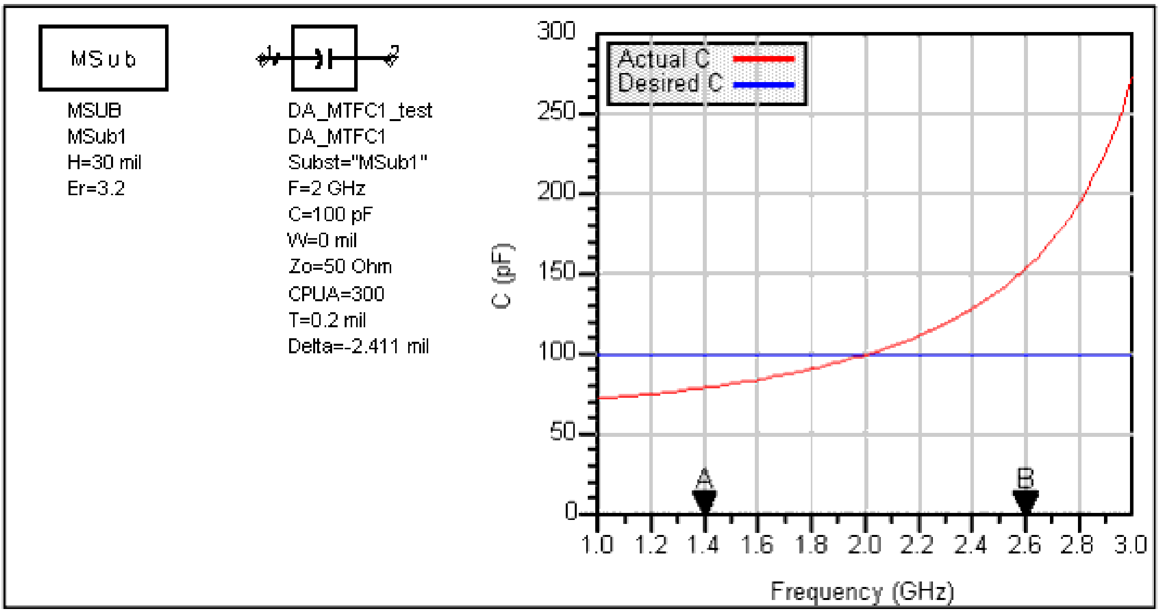
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| C | Capacitance | pF | 1 |
| W | Conductor width; set to 0 if Zo specified | mil | 0 |
| Zo | Characteristic impedance of line for computing W; set to 0 if W specified | Ohm | 50 |
| CPUA | Capacitance per unit area | pf/mm$^2$ | 300 |
| T | Thickness of capacitor dielectric | mil | 0.2 |
| RsT | Sheet resistance of top metal plate | Ohm | 0 |
| RsB | Sheet resistance of bottom metal plate | Ohm | 0 |
| TT | Thickness of top metal plate | mil | 0 |
| TB | Thickness of bottom metal plate | mil | 0 |
| COB | Bottom conductor overlap | mil | 0 |
| COT | Top conductor overlap | mil | 0 |
| DO | Dielectric overlap | mil | 0 |
| Delta | Length added to conductor for tuning capacitance | mil | 0 |

### Notes

1. MTFC designs a microstrip thin film capacitor. The underlying design uses the MTFC component contained in the Tlines-Microstrip palette.
2. The design is accomplished by determining the length required to achieve the desired capacitance using the capacitance per unit area (CPUA) in conjunction with the specified width (W).
3. Since this capacitor is often fed with a microstrip line, either the physical width or the characteristic impedance of a microstrip line on the substrate can be specified. However, only one of the parameters should be non-zero.
4. The parameters from RsT through DO inclusive are not used in the design process but are passed on to the underlying MTFC component and therefore included in any simulations or optimizations.
5. The tuning parameter Delta represents incremental length required to achieve the desired capacitance. It is typically relatively small, as the initial design tends to be accurate.
6. The Optimization Assistant tunes the conductor length to achieve the desired capacitance.
7. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

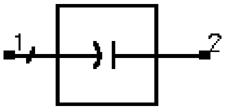
### Example

A MTFC component was used to design a 100 pF capacitor at a center frequency of 2 GHz. The conductor width corresponds to that of a 50 Ohm microstrip line fabricated on MSub1. Optimization yielded a value of Delta = -2.411.



# TFC (Thin Film Capacitor) SmartComponent

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| F | Center frequency | GHz | 1 |
| C | Capacitance | pF | 1 |
| W | Conductor width | mil | 25 |
| T | Thickness of capacitor dielectric | mil | 0.2 |
| Er | Relative dielectric constant | None | 9.6 |
| Rho | Resistivity of conductor (relative to gold) | None | 1 |
| TanD | Dielectric loss tangent | None | 0 |
| CO | Conductor overlap | mil | 0 |
| DO | Dielectric overlap | mil | 0 |
| Delta | Length added to conductor for tuning capacitance | mil | 0 |

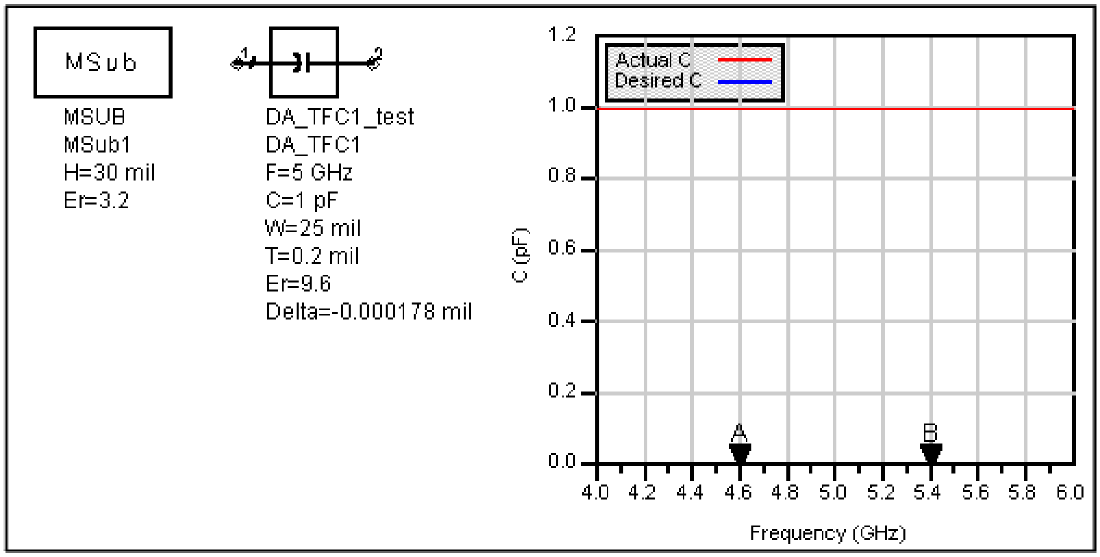
### Notes

1. TFC designs a thin film capacitor. The underlying design uses the TFC component contained in the Tlines-Microstrip palette.
2. The design is accomplished by determining the length required to achieve the desired capacitance using the simple parallel plate capacitor model C = Er*W*L/T. The parameters from Rho through DO inclusive are not used in the design process but are passed on to the underlying TFC component and are therefore included in any simulations or optimizations.
3. The tuning parameter Delta represents incremental length required to achieve the desired capacitance. It is typically relatively small, as the initial design tends to be accurate.
4. The Optimization Assistant tunes the conductor length to achieve the desired capacitance.
5. A SmartComponent subnetwork is empty until the Design Assistant is used to

generate the design. Refer to *Design Assistant* (dgpas).

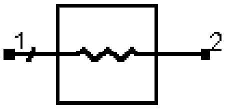
### Example

A TFC component was used to design a 1 pF capacitor at a center frequency of 5 GHz using a dielectric with a 0.2 mil thickness and dielectric constant of 9.6. Optimization yielded a value of Delta = -0.000178 mil.



## TFR (Thin Film Resistor) SmartComponent

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| R | Resistance | Ohm | 50 |
| W | Conductor width (set to 0 if Zo specified) | mil | 0 |
| Zo | Characteristic impedance of line assuming Rs; set to 0 if W specified | Ohm | 50 |
| Rs | Sheet resistivity (Ohms/square) | Ohm | 50 |
| Freq | Frequency for scaling sheet resistivity | Hz | 0 |
| CO | Conductor overlap | mil | 0 |
| Delta | Length added to conductor for tuning resistance | mil | 0 |

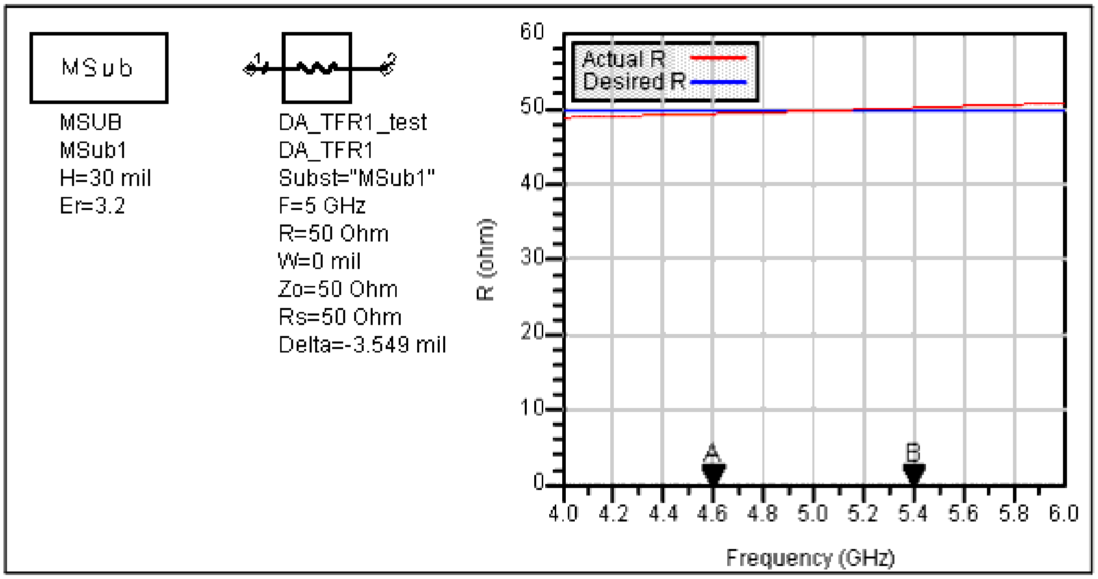
### Notes

1. TFR designs a thin film resistor. The underlying design uses the TFR component contained in the Tlines-Microstrip palette.
2. The design is accomplished by determining the length required to achieve the desired resistance using the sheet resistivity Rs in conjunction with the strip width W. The parameters Freq and CO are not used in the design process but are passed on to the underlying TFR component and are therefore included in any simulations or optimizations.
3. Since this resistor is often fed with a microstrip line, either the physical width or the characteristic impedance of a microstrip line on the substrate can be specified.

4. The tuning parameter Delta represents incremental length required to achieve the desired resistance.
5. The Optimization Assistant tunes the conductor length to achieve the desired resistance.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

### Example

A TFR component was used to design a 50 Ohm resistor at a center frequency of 5 GHz using a conductor with a sheet resistance of 50 Ohm/square. The conductor width was chosen to correspond to that of a 50 Ohm microstrip line on the substrate. Optimization yielded a value of Delta = -3.549 mil.



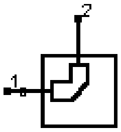
# Passive Microstrip Components

- *MBend (Microstrip Bend Component)* (dgpas)
- *MBStub (Microstrip Butterfly Radial Stub)* (dgpas)
- *MCFil (Microstrip Coupled-Line Filter Element)* (dgpas)
- *MCLine (Microstrip Coupled Line Component)* (dgpas)
- *MCorn (Microstrip Corner Component)* (dgpas)
- *MCross (Microstrip Cross Component)* (dgpas)
- *MCurve (Microstrip Curve Component)* (dgpas)
- *MGap (Microstrip Gap Component)* (dgpas)
- *MLine (Microstrip Line)* (dgpas)
- *MMndr (Microstrip Meander Line)* (dgpas)
- *MRStub (Microstrip Radial Stub) SmartComponent* (dgpas)
- *MStep (Microstrip Step Component)* (dgpas)
- *MStub (Microstrip Stub)* (dgpas)
- *MTaper (Microstrip Taper Component)* (dgpas)
- *MTee (Microstrip Tee Component)* (dgpas)

**Note**
A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# MBend (Microstrip Bend Component)

### Symbol



### Parameters

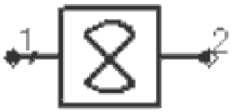
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Zo | Desired characteristic impedance | Ohm | 50 |
| BendType | Type of bend | None | Arbitrary |
| Angle | Angle of bend (for arbitrary angle/miter bend) | None | 90 |
| M | Miter fraction (for arbitrary angle/miter bend) | None | 0.6 |

### Notes

1. MBend designs a microstrip bend given the substrate, desired characteristic impedance, and bend properties. The design realizes the native MBEND, MBEND2, or MBEND3 components.
2. Since the design uses the models inherent to ADS to compute the line width, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
3. BendType can be Arbitrary Angle/Miter (MBEND), 90 Degree/Miter (MBEND2), or 90 Degree/Optimal Miter (MBEND3). The parameters Angle and M are only used for MBEND realizations. Refer to the discussion of these components in the *ADS Microstrip Components* (ccdist) documentation for a more detailed description.
4. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# MBStub (Microstrip Butterfly Radial Stub)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Xin | Input reactance0 | Ohm | 0† |
| Cin | Input capacitance | pF | 0† |
| Lin | Input inductance | nH | 0† |
| W | Width of feed line; set to zero if Z specified | mil | 0 |
| Z | Characteristic impedance of feed line; set to zero if Z specified | Ohm | 50 |
| Angle | Subtended angle of circular sector | None | 60 |
| D | Insertion depth of circular sector in feed line | mil | 3 |
| Delta | Length added to stub for tuning performance | mil | 0 |

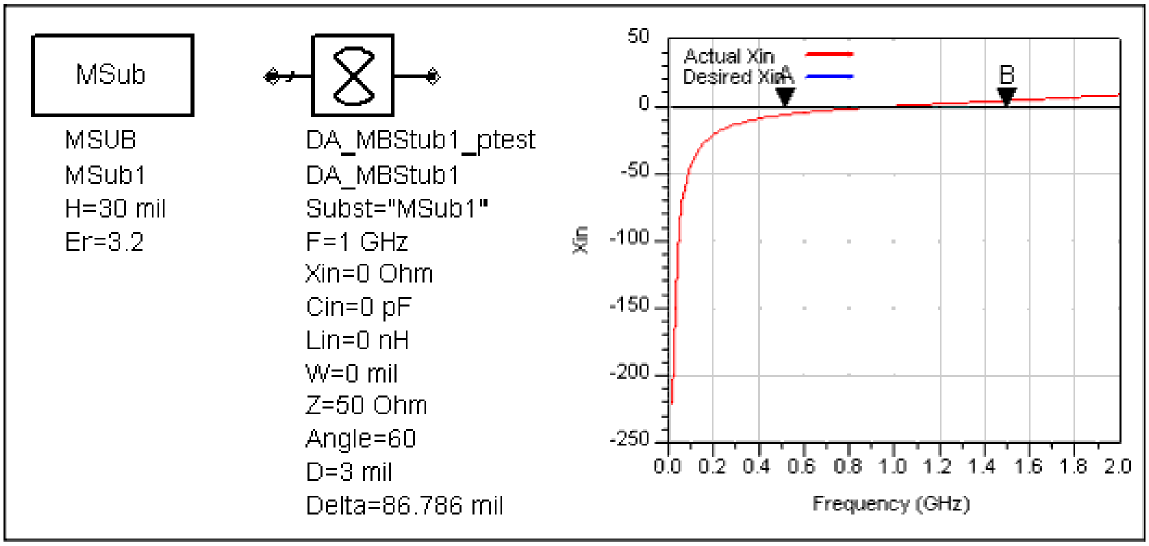
**† Only one of Xin, Cin, and Lin can be non-zero**

### Notes

1. MBStub designs a microstrip butterfly radial stub given the substrate, desired input reactance, and stub dimensions.
2. The stub is designed by dividing the radial lines into several short segments.
3. For proper operation, only one of Xin, Cin, and Lin can be non-zero. If all are zero, the stub is designed to provide an open circuit.
4. Refer to the discussion of the MBSTUB component in the *ADS Microstrip Components* (ccdist) documentation for a more detailed description of the model used for this component.
5. The optimization changes the length of the stubs to achieve the desired input reactance.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

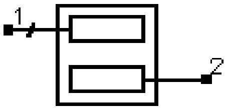
### Example

A MBStub component was used to design an open circuit stub at a center frequency of 1 GHz. Optimization yielded a value of Delta = 88.786 mil.



## MCFil (Microstrip Coupled-Line Filter Element)

### Symbol



### Parameters

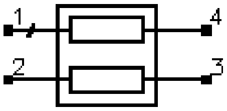
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Zoe | Even-mode characteristic impedance | Ohm | 55 |
| Zoo | Odd-mode characteristic impedance | Ohm | 45 |
| Zo1 | Characteristic impedance of input line at port 1 | Ohm | 50 |
| Zo2 | Characteristic impedance of input line at port 2 | Ohm | 50 |
| Lphys | Physical line length; set to zero if Lelec specified | mil | 0 |
| Lelec | Line length in wavelengths; set to zero if Lphys specified | None | 0.25 |

### Notes

1. MCFil designs a microstrip coupled-line filter component given the substrate, desired even- and odd-mode characteristic impedances, and physical or electrical length.
2. Since the design uses the models inherent to ADS to compute the line width, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
3. For proper operation, either Lphys or Lelec must be zero.
4. Zo1 and Zo2 specify the impedance of the lines attached to this component and are provided to ensure proper pin location in the layout. Refer to the discussion of the MCFIL component in the *ADS Microstrip Components* (ccdist) documentation for a more detailed description of the model used for this component.
5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# MCLine (Microstrip Coupled Line Component)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Zoe | Even-mode characteristic impedance | Ohm | 55 |
| Zoo | Odd-mode characteristic impedance | Ohm | 45 |
| Zo1 | Characteristic impedance of input line at port 1 | Ohm | 50 |
| Zo2 | Characteristic impedance of input line at port 2 | Ohm | 50 |
| Zo3 | Characteristic impedance of input line at port 3 | Ohm | 50 |
| Zo4 | Characteristic impedance of input line at port 4 | Ohm | 50 |
| Lphys | Physical line length; set to zero if Lelec specified | mil | 0 |
| Lelec | Line length in wavelengths; set to zero if Lphys specified | None | 0.25 |

### Notes

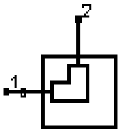
1. MCLine designs a microstrip coupled line component given the substrate, desired even- and odd-mode characteristic impedances, and physical or electrical length.
2. Since the design uses the models inherent to ADS to compute the line width, there is

no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.

3. For proper operation, either Lphys or Lelec must be zero.
4. Zo1 through Zo4 specify the impedance of the lines attached to this component and are provided to ensure proper pin location in the layout. Refer to the discussion of the MCLIN component in the *ADS Microstrip Components* (ccdist) documentation for a more detailed description of the model used for this component.
5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# MCorn (Microstrip Corner Component)

### Symbol



### Parameters

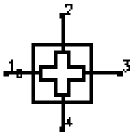
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Zo | Desired characteristic impedance | Ohm | 50 |

### Notes

1. MCorn designs a microstrip corner component given the substrate and characteristic impedance of the input and output lines.
2. Since the design uses the models inherent to ADS to compute the line width, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
3. Refer to the discussion of the MCORN component in the *ADS Microstrip Components* (ccdist) documentation for a more detailed description of the model used for this component.
4. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# MCross (Microstrip Cross Component)

### Symbol



### Parameters

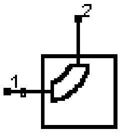
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Z1 | Characteristic impedance of port 1 | Ohm | 50 |
| Z2 | Characteristic impedance of port 2 | Ohm | 50 |
| Z3 | Characteristic impedance of port 3 | Ohm | 50 |
| Z4 | Characteristic impedance of port 4 | Ohm | 50 |

### Notes

1. MCross designs a microstrip cross given the substrate and desired characteristic impedance on each port.
2. Since the design uses the models inherent to ADS to compute the line width, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
3. Refer to the discussion of the MCROSS component in the *ADS Microstrip Components* (ccdist) documentation for a detailed description of this component.
4. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# MCurve (Microstrip Curve Component)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Zo | Desired characteristic impedance | Ohm | 50 |
| Angle | Angle of curve | None | 90 |
| Radius | Radius of curvature (set to zero if Lelec specified) | mil | 100 |
| Lelec | Curve length in wavelengths (set to zero if Radius specified) | None | 0 |
| CurveType | Type of curve | None | Transmission Line |
| Nmode | Number of modes (for Waveguide Model) | None | 2 |

### Notes

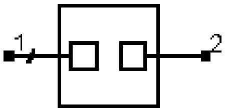
1. MCurve designs a microstrip curve given the substrate, desired characteristic impedance, and curve properties. The design realizes the native MCURVE and MCURVE2 components.
2. Since the design uses the models inherent to ADS to compute the line width, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
3. Either Lelec or Radius must be zero for proper operation.
4. BendType can be Transmission Line Model (MCURVE) or Magnetic Wall Waveguide Model (MCURVE2). The parameter Nmode is used only for MCURVE2. Refer to the to the discussion of these components in the *ADS Microstrip Components* (ccdist)

documentation for a more detailed description.

5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# MGap (Microstrip Gap Component)

### Symbol



### Parameters

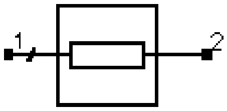
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Zo | Desired characteristic impedance | Ohm | 50 |
| S | Length of gap (spacing) | mil | 50 |

### Notes

1. MGap designs a microstrip gap given the substrate, desired characteristic impedance, and gap width.
2. Since the design uses the models inherent to ADS to compute the line width, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
3. Refer to the discussion of the MGAP component in the *ADS Microstrip Components* (ccdist) documentation for a detailed description of this component.
4. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# MLine (Microstrip Line)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Zo | Desired characteristic impedance | Ohm | 50 |
| Lphys | Physical line length; set to zero if Lelec specified | mil | 0 |
| Lelec | Line length in wavelengths; set to zero if Lphys specified | None | 0.25 |

### Notes

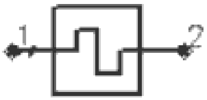
1. MLine designs a microstrip line given the substrate, desired characteristic impedance, and physical or electrical length.
2. Since the design uses the models inherent to ADS to compute the line width and length, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
3. For proper operation, either Lphys or Lelec must be zero.
4. Refer to the discussion of the MLIN component in the *ADS Microstrip Components* (ccdist) documentation for a more detailed description of the model used for this

component.

5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# MMndr (Microstrip Meander Line)

**Symbol**



**Parameters**

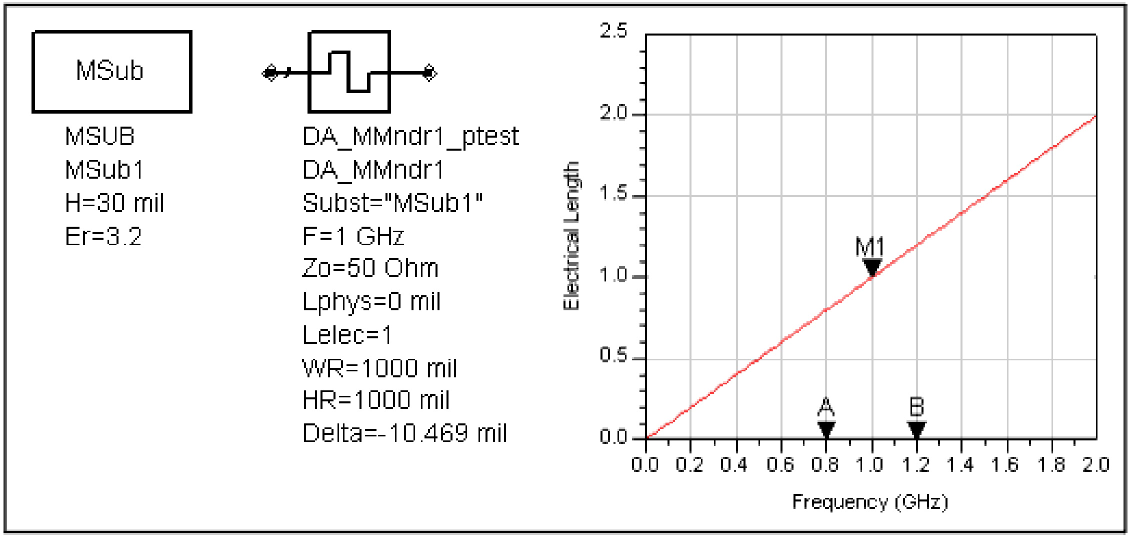
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Zo | Desired characteristic impedance | Ohm | 50 |
| Lphys | Physical line length (set to zero if Lelec specified) | mil | 0 |
| Lelec | Line length in wavelengths (set to zero if Lphys specified) | None | 0.25 |
| WR | Bounding rectangle width | mil | 1000 |
| HR | Bounding rectangle height | mil | 1000 |
| Delta | Length added to vertical lines for tuning performance | mil | 0 |

**Notes**

1. MMndr designs a meandering microstrip line given the substrate, desired characteristic impedance, physical or electrical length, and maximum rectangular dimensions of the line.
2. The line input and output ports are at the center of the rectangle on the side characterized by HR.
3. The final width and height of the bounding box can be smaller than that specified depending on the desired length.
4. For proper operation, either Lphys or Lelec must be zero.
5. Refer to the discussion of the MLIN component in the *ADS Microstrip Components* (ccdist) documentation for a more detailed description of the model used for this component.
6. The optimization minimizes the absolute difference between the transmission phase and that resulting from the specified length. Only the vertical dimension is optimized, and since the corners tend to add excess phase delay the resulting height will be slightly smaller than specified.
7. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

**Example**

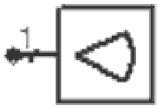
A MMndr component was used to design a 1-wavelength line in a 1-inch by 1-inch square area at a center frequency of 1 GHz. Optimization yielded a value of Delta = -10.469 mil.


MSub

MSUB
MSub1
H=30 mil
Er=3.2

DA_MMndr1_ptest
DA_MMndr1
Subst="MSub1"
F=1 GHz
Zo=50 Ohm
Lphys=0 mil
Lelec=1
WR=1000 mil
HR=1000 mil
Delta=-10.469 mil

# MRStub (Microstrip Radial Stub) SmartComponent



### Symbol

### Parameters

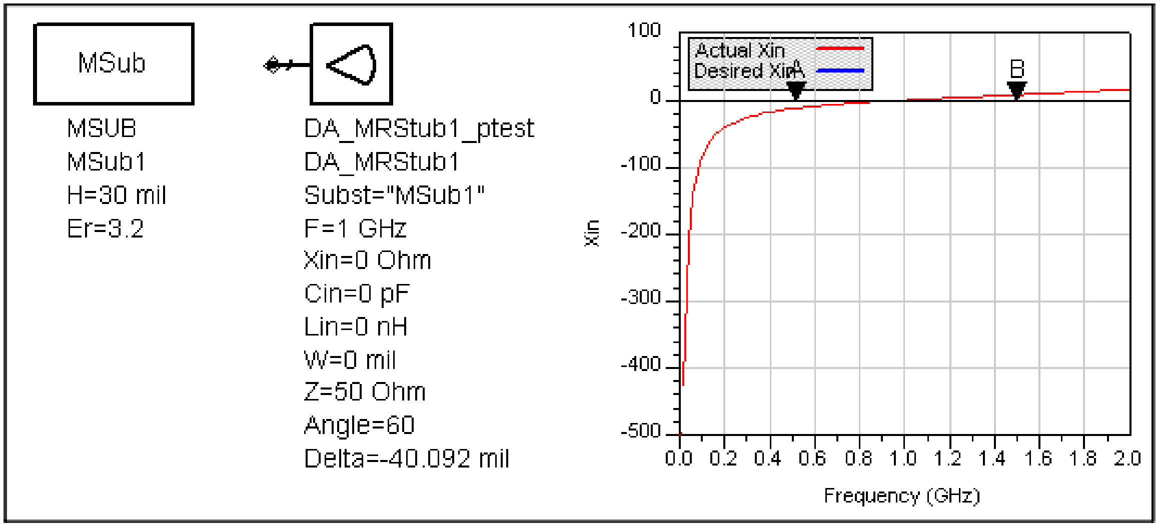
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Xin | Desired input reactance | Ohm | 0 |
| Cin | Desired input capacitance | pF | 0 |
| Lin | Desired input inductance | nH | 0 |
| W | Width of feed line (set to zero if Z specified) | mil | 0 |
| Z | Characteristic impedance of feed line (set to zero if W specified) | Ohm | 50 |
| Angle | Subtended angle of circular sector | None | 60 |
| Delta | Length added to stub for tuning performance | mil | 0 |
| † **Only one of Xin, Cin, and Lin can be non-zero** | | | |

### Notes

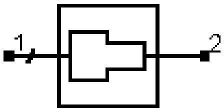
1. MRStub designs a microstrip radial stub given the substrate, desired input reactance, and stub dimensions.
2. The stub is designed by dividing the radial line into several short segments.
3. For proper operation, only one of Xin, Cin, and Lin can be non-zero. If all are zero, the stub is designed to provide an open circuit.
4. Refer to the discussion of the MRSTUB component in the *ADS Microstrip Components* (ccdist) documentation for a more detailed description of the model used for this component.
5. The optimization changes the length of the stubs to achieve the desired input reactance.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

### Example

A MRStub component was used to design an open circuit stub at a center frequency of 1 GHz. Optimization yielded a value of Delta = -40.092 mil.



# MStep (Microstrip Step Component)



### Symbol

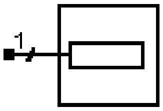
### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Z1 | Characteristic impedance of port 1 | Ohm | 50 |
| Z2 | Characteristic impedance of port 2 | Ohm | 25 |

### Notes

1. MStep designs a microstrip step given the substrate and desired characteristic impedances.
2. Since the design uses the models inherent to ADS to compute the line width, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
3. Refer to the discussion of the MSTEP component in the *ADS Microstrip Components* (ccdist) documentation for a detailed description of this component.
4. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# MStub (Microstrip Stub)



### Symbol

### Parameters

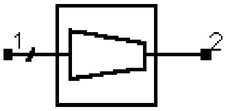
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Zo | Desired characteristic impedance | Ohm | 50 |
| Lphys | Physical line length | mil | 0 |
| Lelec | Line length in wavelengths | None | 0 |
| Xin | Desired input reactance | Ohm | 0 |
| Cin | Desired input capacitance | pF | 0 |
| Lin | Desired input inductance | nH | 0 |
| StubType | Type of stub | None | Open Circuit |

### Notes

1. MStub designs a microstrip open or short circuited stub given the substrate, desired characteristic impedance, and physical or electrical length. The design realizes the native MLOC, MLSC, and MLEF components.
2. Only one of Lphys, Lelec, Xin, Cin, and Lin can be non-zero.
3. Since the design uses the models inherent to ADS to compute the line width and length, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
4. For proper operation, only one of Lphys, Lelec, Xin, Cin, and Lin can be non-zero.
5. StubType can be either Open Circuit (MLOC), End Effect (MLEF), or Short Circuit (MLSC). Refer to the discussion of these components in the *ADS Microstrip Components* (ccdist) documentation for a more detailed description of these different options.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# MTaper (Microstrip Taper Component)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Z1 | Characteristic impedance at port 1 | Ohm | 50 |
| Z2 | Characteristic impedance at port 2 | Ohm | 25 |
| Lphys | Physical line length (set to zero if Lelec specified) | mil | 0 |
| Lelec | Line length in wavelengths (set to zero if Lphys specified) | None | 0.10 |

### Notes

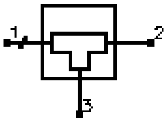
1. MTaper designs a microstrip tapered line given the substrate, desired characteristic impedance, and physical or electrical length.
2. Since the design uses the models inherent to ADS to compute the line width and length, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
3. For proper operation, either Lphys or Lelec must be zero.
4. Z1 and Z2 are used to determine the widths at each end of the taper component.

Refer to the discussion of the MTAPER component in the *ADS Microstrip Components* (ccdist) documentation for a more detailed description of this component.

5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# MTee (Microstrip Tee Component)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Z1 | Characteristic impedance of port 1 | Ohm | 50 |
| Z2 | Characteristic impedance of port 2 | Ohm | 50 |
| Z3 | Characteristic impedance of port 3 | Ohm | 50 |

### Notes

1. MTee designs a microstrip tee given the substrate and desired characteristic impedance at each port.
2. Since the design uses the models inherent to ADS to compute the line width and length, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
3. Z1, Z2, and Z3 are used to determine the widths of each port. Refer to the discussion of the MTEE component in the *ADS Microstrip Components* (ccdist) documentation for a more detailed description of this component.
4. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

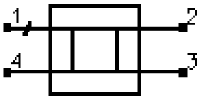
# Passive Microstrip Circuits

- *BLCoupler (Branch-Line Coupler)* (dgpas)
- *CLCoupler (Coupled-Line Coupler)* (dgpas)
- *CLFilter (Coupled-Line Filter)* (dgpas)
- *CMFilter (Comb-Line Filter)* (dgpas)
- *DSMatch (Double-Stub Match)* (dgpas)
- *HPFilter (Hairpin Filter)* (dgpas)
- *IDFilter (Interdigital Filter)* (dgpas)
- *LCoupler (Lange Coupler)* (dgpas)
- *LEMatch (Lumped Component Match)* (dgpas)
- *QWMatch (Quarter-Wave Match)* (dgpas)
- *RAtten (Resistive Attenuator)* (dgpas)
- *RRCoupler (Rat-Race Coupler)* (dgpas)
- *SBFilter (Stub Bandpass Filter)* (dgpas)
- *SIFilter (Stepped Impedance Lowpass Filter)* (dgpas)
- *SLFilter (Stub Lowpass Filter)* (dgpas)
- *SRFilter (Stepped Impedance Resonator Filter)* (dgpas)
- *SSMatch (Single-Stub Match)* (dgpas)
- *TCoupler (Tee Power Divider)* (dgpas)

- *TLMatch (Tapered-Line Match)* (dgpas)
- *WDCoupler (Wilkinson Divider)* (dgpas)
- *ZZFilter (Zig-Zag Coupled-Line Filter)* (dgpas)

**Note**
A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# BLCoupler (Branch-Line Coupler)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| DeltaF | Total frequency bandwidth | GHz | 0.5 |
| Zo | Characteristic impedance | Ohm | 50 |
| ResponseType | Type of frequency response | None | Chebyshev |
| N | Number of coupler sections; set N=0 to compute N | None | 0 |
| Rmax | Maximum voltage reflection coefficient at the input | None | 0.5 |
| C | Coupling coefficient | dB | 3 |
| Delta | Length added to branches for tuning performance | mil | 0 |

### Notes

1. A branch-line coupler outputs from the coupled port (pin 3) a fraction of the power presented at the input (pin 1). The remainder of the power is passed through to the output port (pin 2). At the center frequency the phase difference between the outputs is 90 degrees, with the coupled port representing the quadrature (Q) output and the output port representing the in-phase (I) output. The coupling coefficient specifies the ratio of the input power to the coupled power ($P_1/P_3$). Pin 4 represents the isolated port, and it is typically well isolated from the input port near the center frequency.
2. The coupling coefficient must be positive and greater than 3 dB. Best results are obtained for tight couplings of 6 dB or better (C < 6 dB). Choosing the coupling parameter larger than 6 dB often causes width constraint violations to occur on the MTEE components, resulting in warning messages during design and simulation. A coupling coefficient of 3 dB provides an equal power split between the two outputs.
3. For broadband performance, the coupler can have multiple sections. If the number of sections N is set to zero, the Design Assistant chooses N such that the reflection coefficient is less than Rmax over the bandwidth DeltaF (centered at the design center frequency). The resulting bandwidth can be broader than that specified. Otherwise, rmax and DeltaF are ignored.
4. The ResponseType specifies the distribution of the partial reflection coefficients seen at each section interface - Uniform, Binomial, and Chebyshev distributions are available.
5. The optimization minimizes the input reflection coefficient (S11) at the design center frequency by changing the length of the lines forming the four branches. All branches

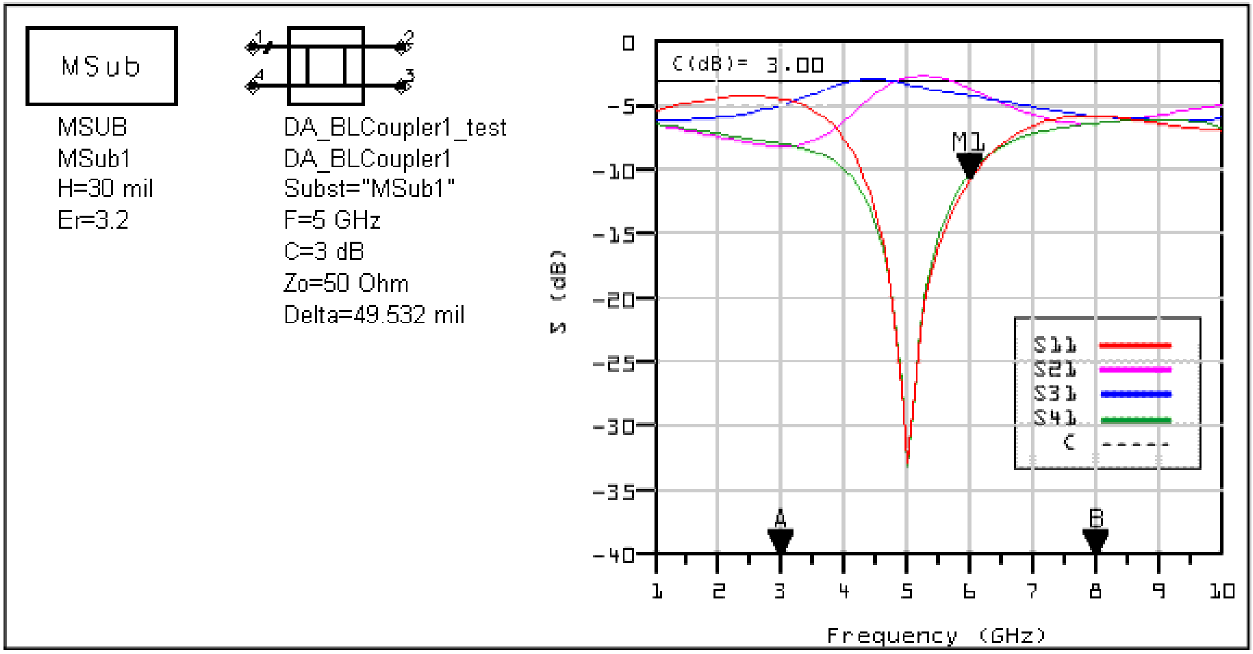
are changed by the same physical length during the optimization.This optimization generally provides very good results but can not guarantee that the specified coupling is attained at the design frequency. More advanced tuning can be performed by changing line width of the branch lines.

6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

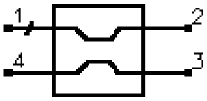
For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 379-383.

### Example

A single-section branch-line coupler was designed for a center frequency of 5 GHz with an equal power split between the I and Q ports. Tuning using the Optimization Assistant yielded a value of Delta = 49.532 mil.



## CLCoupler (Coupled-Line Coupler)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| C | Coupling coefficient | dB | 20 |
| Zo | Characteristic impedance | Ohm | 50 |
| N | Number of coupler sections (N is odd) | None | 1 |
| Delta | Length added to branches for tuning performance | mil | 0 |

### Notes

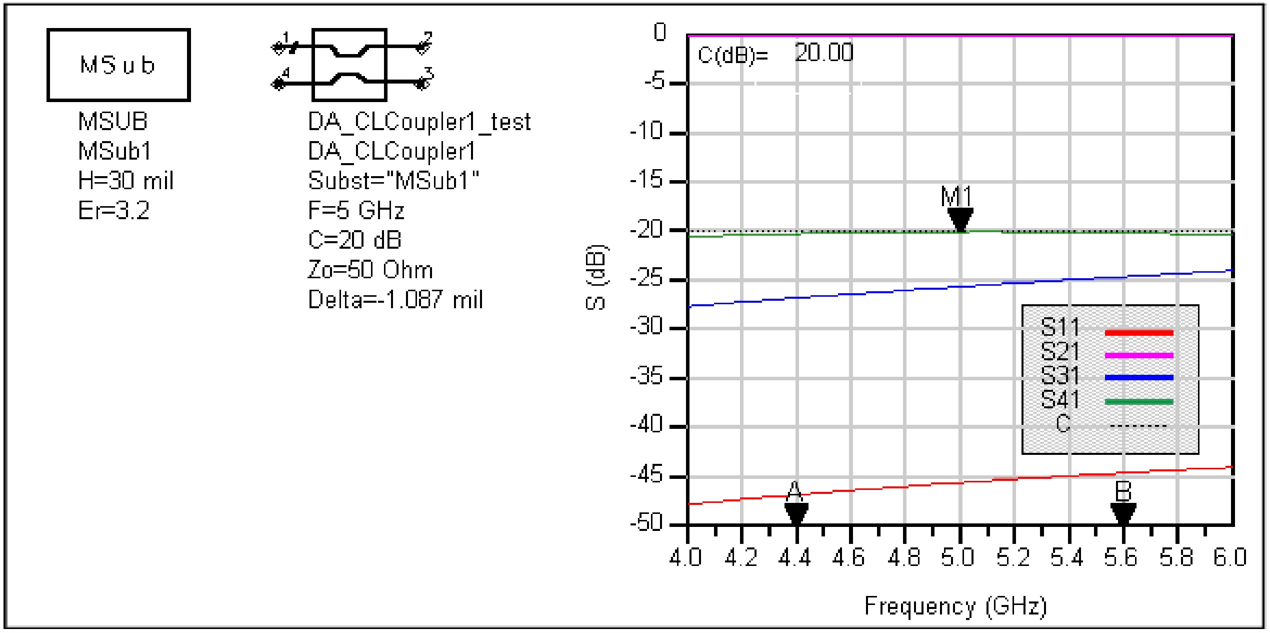
1. A coupled-line coupler outputs from the coupled port (pin 4) a fraction of the power

presented at the input (pin 1). The remainder of the power is passed through to the output port (pin 2). The coupling coefficient specifies the ratio of the input power to the coupled power ($P_1/P_4$). The remaining port is isolated, although the isolation is often similar in value to the coupling coefficient for microstrip realizations.
2. The optimization minimizes the absolute difference between S41 in dB and the specified coupling coefficient at the design center frequency by changing the length of the coupled-line section.
3. The coupling coefficient must be positive and greater than 3 dB. Best results are obtained for weak couplings of roughly 10 dB or more (C >10dB). Choosing the coupling coefficient too small can require a spacing between the coupled lines too small to realize.
4. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 383-394.

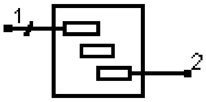
### Example

A coupled-line coupler was designed for a center frequency of 5 GHz with 20 dB of coupling. Tuning using the Optimization Assistant yielded a value of Delta =-1.087 mil.



# CLFilter (Coupled-Line Filter)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| Fs1 | Lower stopband edge frequency | GHz | 1.8 |
| Fp1 | lower passband edge frequency | GHz | 2 |
| Fp2 | upper passband edge frequency | GHz | 2.2 |
| Fs2 | lower stopband edge frequency | GHz | 2.4 |
| Ap | passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| As | stopband edge attenuation | dB | 20 |
| N | number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | type of frequency response (maximally flat or Chebyshev) | None | Maximally Flat |
| Zo | desired input/output impedance | Ohm | 50 |
| CouplingType | Type of input/output coupling | None | Coupled Line Transformer Input |
| Delta | length added to coupled sections for tuning performance | mil | 0 |

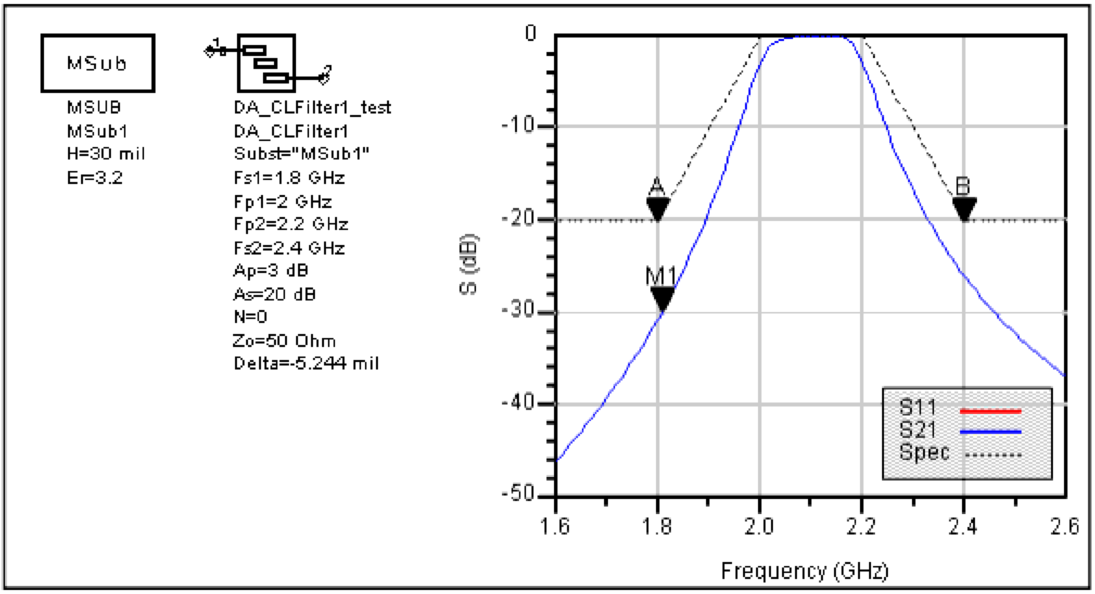
### Notes

1. A coupled-line filter provides a bandpass frequency response between the input and output ports. N coupled-line sections produces an N-1 order filter response. Additional numbers of sections can be used to steepen the transition band roll off or widen the pass bandwidth.
2. Because of the heavy computational burden in determining the line parameters, a brief delay will occur for the design.
3. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.
4. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
5. Using a Coupled Line Transformer Input CouplingType will use and extra coupled line section on the inputs and outputs to feed the device. Choosing Tapped Line Transformer Input will bring the feedline directly into the first resonator.
6. The optimization minimizes the absolute difference between S21 in dB and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequencies. Because only the line lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and spacings of the coupled filter sections.
7. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 477-485.

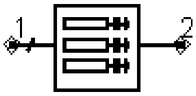
### Example

A coupled-line filter was designed for a maximally flat response with the 3 dB passband edge frequencies at 2 GHz and 2.4 GHz respectively. Coupled Line Transformer Inputs were used. The design required 4 coupled-line sections. Tuning using the Optimization Assistant yielded a value of Delta = -5.244 mil.



# CMFilter (Comb-Line Filter)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | microstrip substrate name | None | Subst1 |
| Fs1 | lower stopband edge frequency | GHz | 1.8 |
| Fp1 | lower passband edge frequency | GHz | 2 |
| Fp2 | upper passband edge frequency | GHz | 2.2 |
| Fs2 | lower stopband edge frequency | GHz | 2.4 |
| Ap | passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| As | stopband edge attenuation | dB | 20 |
| N | number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | type of frequency response (maximally flat or Chebyshev) | None | Maximally Flat |
| Zo | desired input/output impedance | Ohm | 50 |
| Lelec | electrical length of filter resonators (the units of Leclec are in fractions of a wavelength i.e., 0.25=1/4 wavelength) | None | 0.05 |
| ya | normalized interior resonator admittance (0 < ya < 1) | None | 0.7 |
| CouplingType | type of input/output coupling (coupled line or tapped line transformer) | None | Tapped Line Transformer Input |
| Delta | length added to coupled sections for tuning performance | mil | 0 |

### Notes

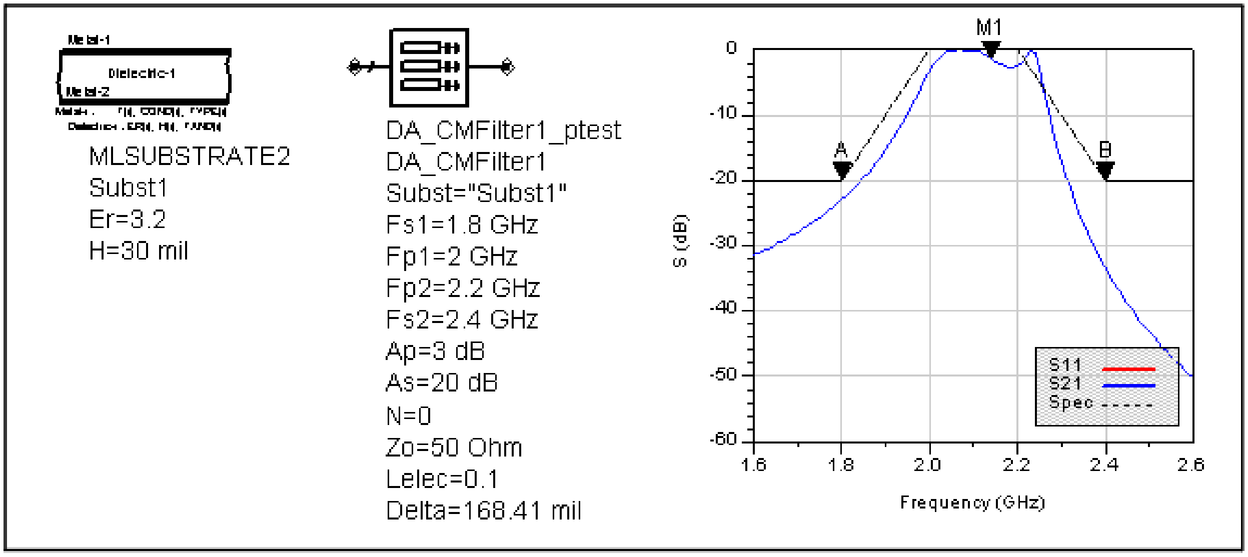
1. A comb-line filter provides a bandpass frequency response between the input and output ports. N coupled-line sections produce an N-1 order filter response. Additional numbers of sections can be used to steepen the transition band roll off or widen the pass bandwidth.
2. Because of the heavy computational burden in determining the line parameters, a brief delay will occur for the design.
3. A two-layer substrate must be used for this topology.

4. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.
5. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
6. Using a CouplingType of "Coupled Line Transformer Input" will use an extra coupled line section on the inputs and outputs to feed the device. Choosing "Tapped Line Transformer Input" will bring the feedline directly into the first resonator.
7. The optimization minimizes the absolute difference between S21 in dB and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequencies. Because only the line lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and spacings of the coupled filter sections.
8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, refer to: Matthaei, Young and Jones, Microwave Filters, Impedance-Matching Networks, and Coupling Structures, Artech House, 1980, pp. 497.

### Example

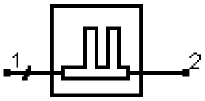
A comb-line filter was designed for a maximally flat response with the 3 dB passband edge frequencies at 2 GHz and 2.2 GHz respectively. The design required 5coupled lines. Tuning using the Optimization Assistant yielded a value of Delta = 168.41 mil.



MLSUBSTRATE2
Subst1
Er=3.2
H=30 mil

DA_CMFilter1_ptest
DA_CMFilter1
Subst="Subst1"
Fs1=1.8 GHz
Fp1=2 GHz
Fp2=2.2 GHz
Fs2=2.4 GHz
Ap=3 dB
As=20 dB
N=0
Zo=50 Ohm
Lelec=0.1
Delta=168.41 mil

# DSMatch (Double-Stub Match)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| Zin | Desired complex input impedance | Ohm | 50 |
| Zload | Complex load impedance to match | Ohm | 100 |
| Zstub1 | Characteristic impedance of stub line 1 (nearest input) | Ohm | 50 |
| Zstub2 | Characteristic impedance of stub line 2 (nearest output) | Ohm | 50 |
| Zline | Characteristic impedance of line between stubs | Ohm | 50 |
| Zfeed1 | Characteristic impedance of line connected to port 1 | Ohm | 50 |
| Zfeed2 | Characteristic impedance of line connected to port 2 | Ohm | 50 |
| Stub1Type | Stub 1 type (open or short circuit) | None | Open Circuit |
| Stub2Type | Stub 2 type (open or short circuit) | None | Open Circuit |
| Delta | Length added to stubs for tuning performance | mil | 0 |

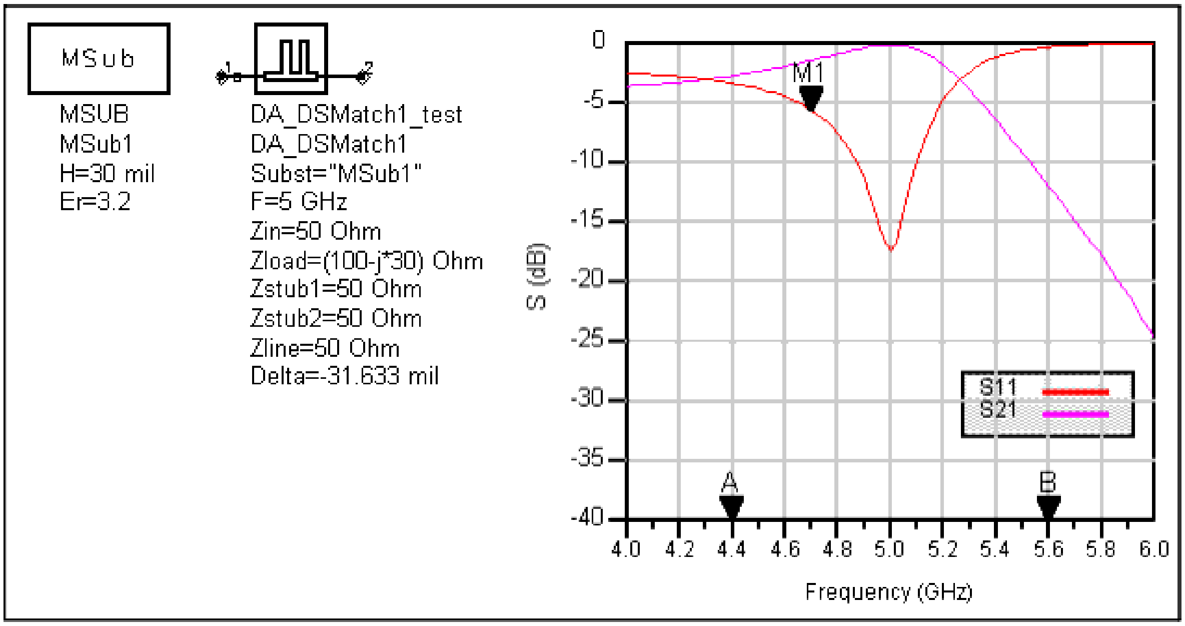
### Notes

1. A double-stub matching network matches a complex load impedance (Zload) to a desired complex input impedance (Zin) using two shunt stubs and a connecting line.
2. An impedance match can be realized using any combination of stub types, although some combinations can be more realizable.
3. Zfeed1 and Zfeed2 are used simply to ensure that the input and output legs of the microstrip tee components are of the proper width.
4. The length of the line between the stubs is nominally chosen to be an eighth of a line wavelength. However, if this will not realize the match, the length of line is computed such that the rotated circle on the Smith chart encloses the load admittance.
5. MTEE component width constraint violations will be avoided generally by choosing similar characteristic impedances for the line, stub, and feed.
6. Since two solutions are possible, the solution that results in the smallest length of stub 1 is chosen. For example, if stub 1 is open circuited, the solution for which stub 1 must realize a capacitive reactance is chosen.
7. The input port termination is set to the conjugate of Zin so that the ideal input reflection coefficient will be zero. The output port termination is set to Zload.
8. The optimization minimizes the value of S11 (referenced to the conjugate of Zin) at the design center frequency by changing the length of the stubs. Both stubs are tuned by the same length.
9. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 266-271.

### Example

A double-stub matching network was designed to match a load impedance of 100 - j30 Ohms to a 50 Ohm line at a center frequency of 5 GHz with open circuited stubs. Tuning using the Optimization Assistant yielded a value of Delta = -31.633 mil.

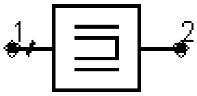


MSUB
MSub1
H=30 mil
Er=3.2

DA_DSMatch1_test
DA_DSMatch1
Subst="MSub1"
F=5 GHz
Zin=50 Ohm
Zload=(100-j*30) Ohm
Zstub1=50 Ohm
Zstub2=50 Ohm
Zline=50 Ohm
Delta=-31.633 mil

# HPFilter (Hairpin Filter)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| Fs1 | Lower stopband edge frequency | GHz | 1.8 |
| Fp1 | Lower passband edge frequency | GHz | 2 |
| Fp2 | Upper passband edge frequency | GHz | 2.2 |
| Fs2 | Lower stopband edge frequency | GHz | 2.4 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| As | Stopband edge attenuation | dB | 20 |
| N | Number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | Type of frequency response (maximally flat or Chebyshev) | None | Maximally Flat |
| Zo | Desired input/output impedance | Ohm | 50 |
| Sphys | Physical spacing between legs within hairpin resonator; set to zero if Selec specified | mil | 0 |
| Selec | Spacing between legs within hairpin resonator in wavelengths; set to zero if Sphys specified | None | 0.25 |
| CouplingType | Type of input/output coupling (coupled line or tapped line transformer) | None | Coupled Line Transmission Input |
| Delta | Length added to coupled sections for tuning performance | mil | 0 |

### Notes

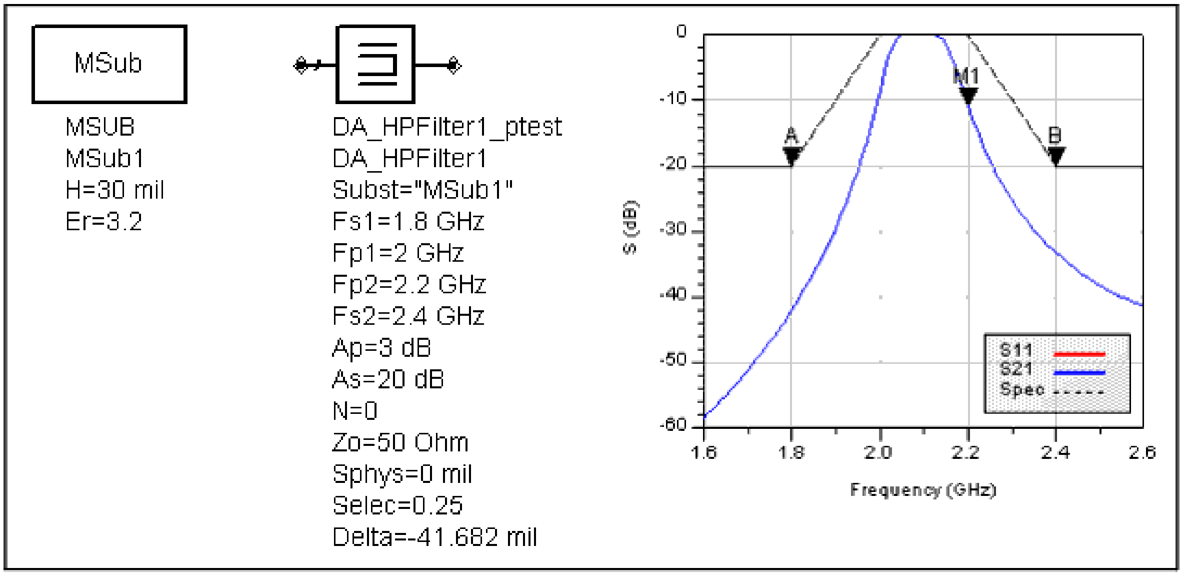
1. A hairpin filter provides a bandpass frequency response between the input and output ports. N coupled-line sections produce an N-1 order filter response. Additional numbers of sections can be used to steepen the transition band roll off or widen the pass bandwidth.
2. Because of the heavy computational burden in determining the line parameters, a brief delay will occur for the design.

3. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.
4. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
5. Using a Coupled Line Transformer Input CouplingType will use and extra coupled line section on the inputs and outputs to feed the device. Choosing Tapped Line Transformer Input will bring the feedline directly into the first resonator.
6. The optimization minimizes the absolute difference between S21 in dB and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequencies. Because only the line lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and spacings of the coupled filter sections.
7. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, refer to: Cristal and Frankel, "Hairpin-line and hybrid hairpin-line/half-wave parallel-coupled-line filters," *IEEE Trans. Microwave Theory and Techniques,* vol MTT-20, pp. 719-728, 1972.

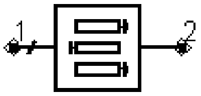
### Example

A hairpin filter was designed for a maximally flat response with the 3 dB passband edge frequencies at 2 GHz and 2.2 GHz respectively. Coupled line transformer inputs were used. The design required 4 coupled sections. Tuning using the Optimization Assistant yielded a value of Delta = -41.682 mil.



# IDFilter (Interdigital Filter)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | Subst1 |
| Fs1 | Lower stopband edge frequency | GHz | 1.8 |
| Fp1 | Lower passband edge frequency | GHz | 2 |
| Fp2 | Upper passband edge frequency | GHz | 2.2 |
| Fs2 | Lower stopband edge frequency | GHz | 2.4 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| As | Stopband edge attenuation | dB | 20 |
| N | Number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | Type of frequency response (Maximally Flat or Chebyshev) | None | Maximally Flat |
| Zo | Desired input/output impedance | Ohm | 50 |
| ya | Normalized interior resonator admittance (0 < ya < 1) | None | 1 |
| CouplingType | Type of input/output coupling (coupled line or tapped line transformer) | None | Coupled Line Transformer Input |
| Delta | Length added to coupled sections for tuning performance | mil | 0 |

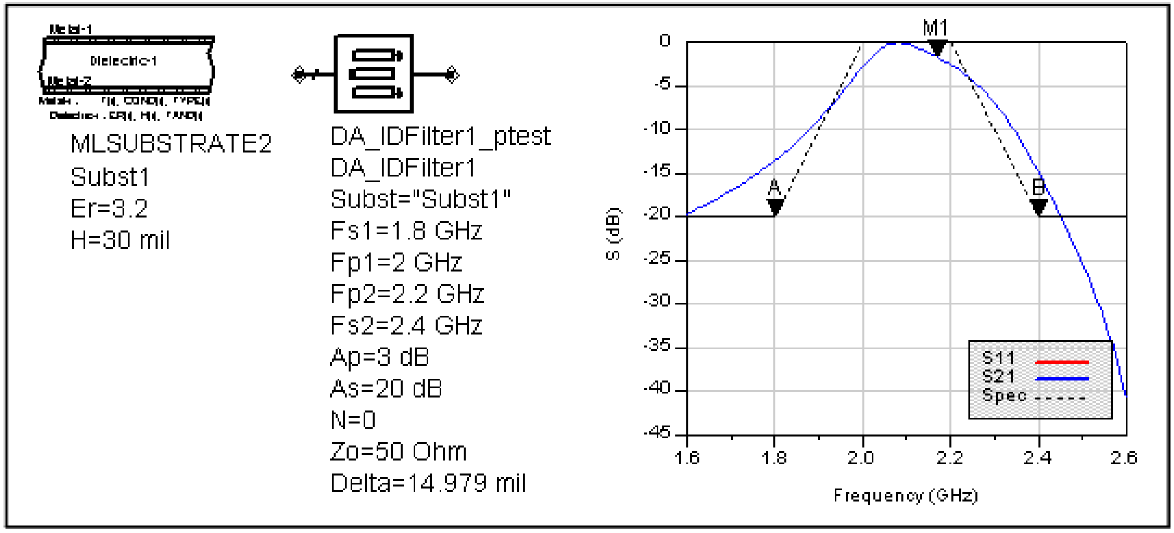
### Notes

1. An interdigital filter provides a bandpass frequency response between the input and output ports. N coupled-line sections produce an N-1 order filter response. Additional numbers of sections can be used to steepen the transition band roll off or widen the pass bandwidth.
2. Because of the heavy computational burden in determining the line parameters, a brief delay will occur for the design.
3. A two-layer substrate must be used for this topology.
4. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.
5. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
6. Using a CouplingType of "Coupled Line Transformer Input" will use an extra coupled line section on the inputs and outputs to feed the device. Choosing "Tapped Line Transformer Input" will bring the feedline directly into the first resonator.
7. The optimization minimizes the absolute difference between S21 in dB and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequencies. Because only the line lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and spacings of the coupled filter sections.
8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, refer to: Matthaei, Young and Jones, Microwave Filters, Impedance-Matching Networks, and Coupling Structures, Artech House, 1980, pp. 614.

### Example

An interdigital filter was designed for a maximally flat response with the 3 dB passband edge frequencies at 2 GHz and 2.2 GHz respectively. The design required 5coupled lines. Tuning using the Optimization Assistant yielded a value of Delta = 14.979 mil.

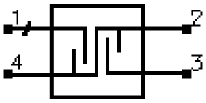


MLSUBSTRATE2
Subst1
Er=3.2
H=30 mil

DA_IDFilter1_ptest
DA_IDFilter1
Subst="Subst1"
Fs1=1.8 GHz
Fp1=2 GHz
Fp2=2.2 GHz
Fs2=2.4 GHz
Ap=3 dB
As=20 dB
N=0
Zo=50 Ohm
Delta=14.979 mil

# LCoupler (Lange Coupler)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| C | Coupling coefficient | dB | 20 |
| N | Number of fingers (4, 6, or 8) | None | 4 |
| Zo | Characteristic impedance | Ohm | 50 |
| Delta | Length added to fingers for tuning performance | mil | 0 |

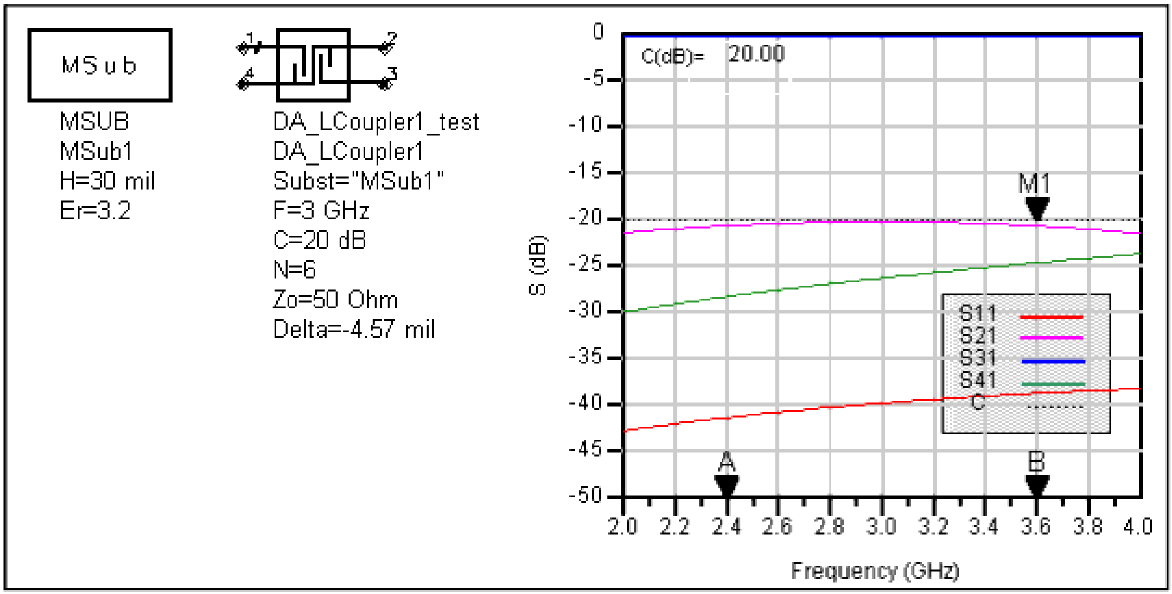
### Notes

1. A Lange coupler outputs from pin 2 a small fraction of the power presented at the input (pin 1). The remainder of the power is passed through pin 3. The coupling coefficient specifies the power ratio $P_1/P_2$. Pin 4 is isolated, and often the isolation is 10 dB better than the coupling coefficient in microstrip realizations.
2. The Lange coupler is best for weak couplings of roughly 10 dB or more (C >10dB). Choosing the coupling coefficient too small can produce an unrealizable design. If the design creates a finger spacing S that is not realizable, increase the value of N.
3. The Design Assistant computes the required even and odd mode impedances to achieve the desired coupling and translates them to finger width and spacing. The length of the fingers is a quarter wavelength at the design frequency.
4. The optimization minimizes the absolute difference between S21 and the specified coupling coefficient at the design center frequency by changing the length of the fingers section.
5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: I. Bahl and P. Bhartia, Microwave Solid State Circuit Design, John Wiley & Sons: New York, 1988, pp. 209-211.

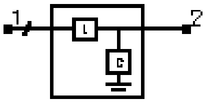
### Example

A Lange coupler was designed for a center frequency of 3 GHz with 20 dB of coupling and 6 fingers. Tuning using the Optimization Assistant yielded a value of Delta = -4.57 mil.



# LEMatch (Lumped Component Match)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| F | Center frequency | GHz | 1 |
| Zin | Desired complex input impedance | Ohm | 50 |
| Zload | Complex load impedance to match | Ohm | 100 |
| NetworkType | Type of network (source to load) | None | Allow Selection |

### Notes

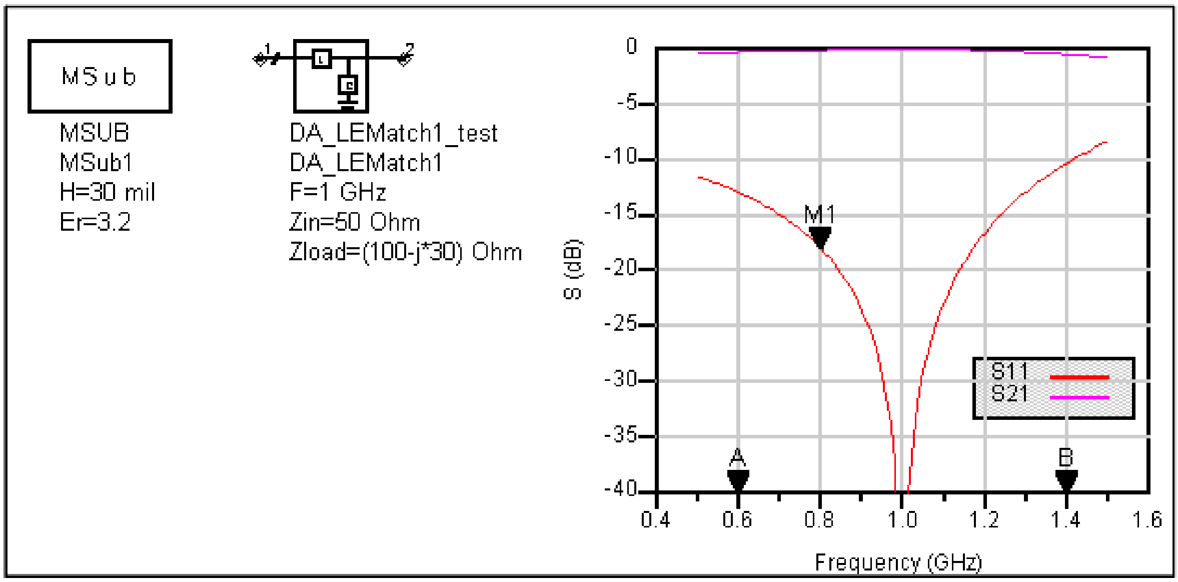
1. A lumped component matching network uses reactive components to match a complex load impedance (Zload) to a desired complex input impedance (Zin).
2. The network type specifies the type of reactive components used (L represents inductance, C represents capacitance) as well as their orientation (series or shunt). The first component specified is that nearest the source, while the second is that nearest the load. If the chosen network type cannot realize the impedance match specified, or if the network type is chosen as "Allow Selection", a dialog box will appear allowing selection from the possible network types that can perform the match.
3. Either two or four distinct networks are possible depending on the load and input impedance specified.
4. The input port termination is set to the conjugate of Zin so that the ideal input reflection coefficient will be zero. The output port termination is set to Zload.
5. No optimization assistant is provided since the design procedure is exact for ideal lumped component models.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to

generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 252-258.

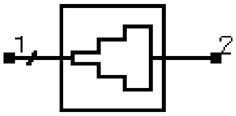
### Example

A lumped component matching network was designed to match a load impedance of 100 - j30 Ohms to an input impedance of 50 Ohms at a center frequency of 1 GHz. A Series L Shunt C configuration was used to realize this match.



## QWMatch (Quarter-Wave Match)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| DeltaF | Total frequency bandwidth | GHz | 0.5 |
| Zo | Desired input impedance | Ohm | 50 |
| Rload | Load impedance to match | Ohm | 100 |
| ResponseType | Type of frequency response | None | Uniform |
| N | Number of quarter-wave sections; set N=0 to compute N | None | 0 |
| Rmax | Maximum voltage reflection coefficient | None | 0.1 |
| Delta | Length added to transformer sections for tuning performance | mil | 0 |

### Notes

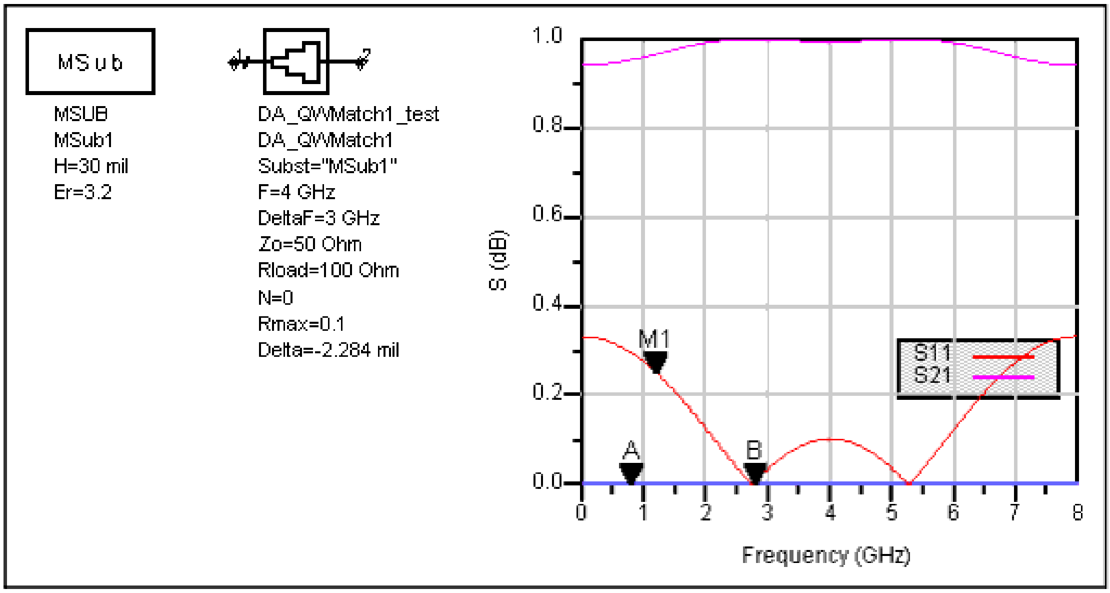
1. A quarter-wave matching network matches a real load impedance (Rload) to a desired real input impedance (Zo) using multiple quarter wavelength sections. A specified frequency response can be realized by proper design of the individual sections.
2. DeltaF is defined as the total bandwidth centered at the design center frequency.

3. If the number of sections N is set to zero, the Design Assistant chooses N such that the reflection coefficient is less that Rmax over the bandwidth DeltaF. The resulting bandwidth may be broader than that specified. Otherwise, Rmax and DeltaF are ignored.
4. The ResponseType specifies the distribution of the partial reflection coefficients seen at each section interface - Uniform, Binomial, and Chebyshev distributions are available. These in turn specify the shape of the reflection coefficient versus frequency.
5. A single-section quarter-wave matching transformer can be designed by setting N = 1.
6. Since the reflection coefficient may not be at its minimum value at the design center frequency, the optimization centers the variation of S11 (referenced to the value of Zo) versus frequency at the center frequency by changing the length of each quarter-wave section. All sections are tuned by the same length during the optimization.
7. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

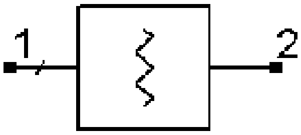
For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 275-288; R. S. Elliott, An Introduction to Guided Waves and Microwave Circuits, Prentice Hall, John Wiley, New Jersey, 1993, pp. 218-224.

### Example

A quarter-wave matching network was designed to match a load impedance of 100 Ohms to a 50 Ohm line at a center frequency of 4 GHz. Specifying a Chebyshev frequency response for which the reflection coefficient remains below 0.1 over a 3 GHz bandwidth dictates 3 quarter-wave sections. Tuning using the Optimization Assistant yielded a value of Delta = -2.284 mil.



## RAtten (Resistive Attenuator)



### Symbol

### Parameters

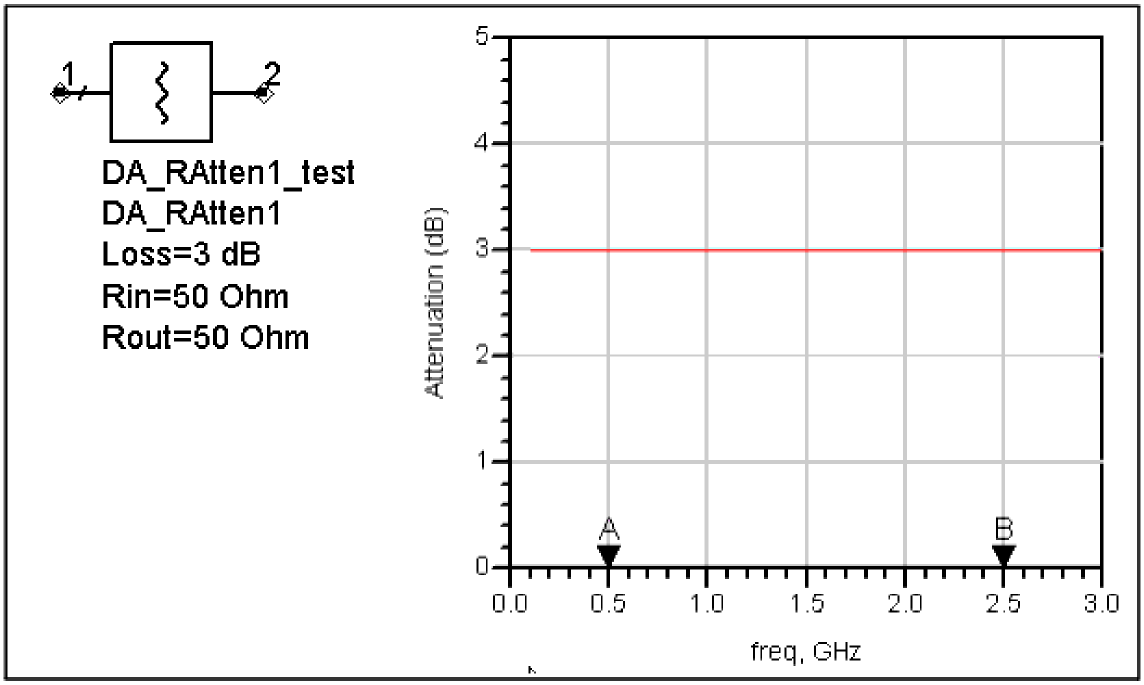
| Name | Description | Unit | Default |
|---|---|---|---|
| Loss | Attenuation | dB | 3 |
| Rin | Input resistance | Ohm | 50 |
| Rout | Output resistance | Ohm | 50 |
| DesignType | Type of attenuator design (pi or tee) | None | Pi |

### Notes

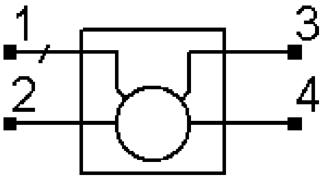
1. A resistive attenuator achieves a specified attenuation while maintaining desired input and output impedance levels.
2. The design specifies the resistance values to achieve the specified level of attenuation for the input and output resistances.
3. The input resistance Rin must be greater than or equal to the output resistance Rout.
4. There is a minimum attenuation that can be achieved for the specified input and output resistances. This attenuation is given by: $MinimumLoss = 20\log\left(\sqrt{\frac{Rin}{Rout}} + \sqrt{\frac{Rin}{Rout}} - 1\right)$
5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

### Example

A 3-dB tee attenuator was designed for a 50 Ohm input and output resistance.



## RRCoupler (Rat-Race Coupler)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| Zo | Characteristic impedance | Ohm | 50 |
| Delta | Length added to ring branches for tuning | mil | 0 |

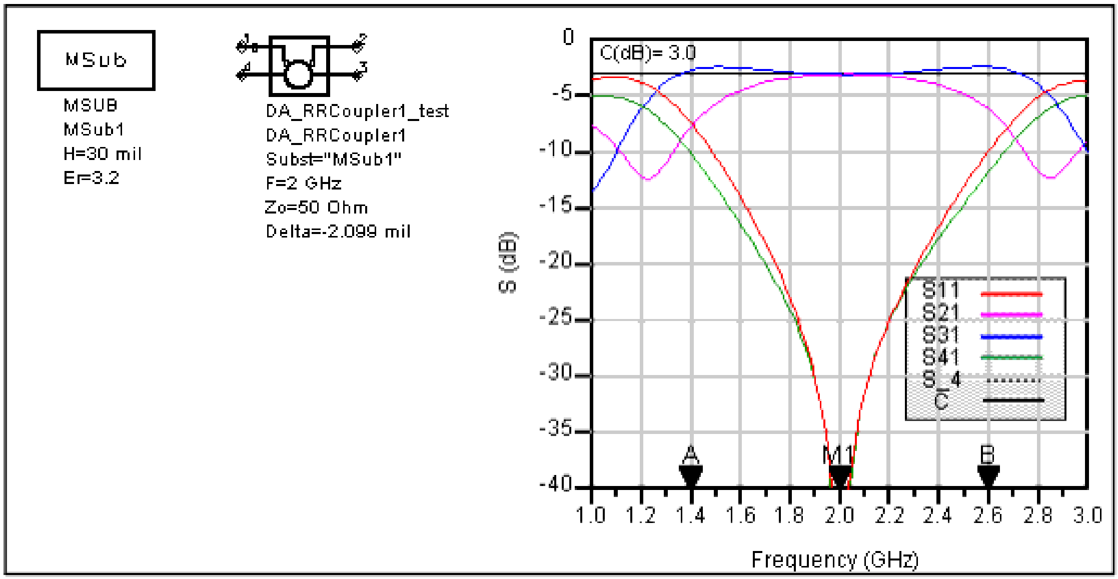
### Notes

1. A rat-race coupler equally divides the power input at port 1 between ports 2 and 3. The signal at the output ports 2 and 3 are in-phase. Port 4 is isolated from port 1. If the signal is driven from port 2, then the power is divided between ports 1 and 4 with port 3 isolated. The signal at ports 1 and 4 are 180 degrees out of phase, and therefore this device is sometimes referred to as a 180-degree hybrid.
2. The design specifies the width and length of the microstrip lines to ensure that the ports are matched to Zo and equal power split is achieved at the design center frequency.
3. The optimization minimizes the value of S11 (referenced to the value of Zo) at the design center frequency by changing the length of the ring.
4. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

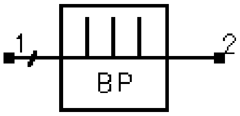
For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 401-407.

### Example

A rat-race coupler was designed for a 50 Ohm system impedance at a center frequency of 2 GHz. Tuning using the Optimization Assistant yielded a value of Delta = -2.099 mil.



# SBFilter (Stub Bandpass Filter)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| Fs1 | Lower stopband edge frequency | GHz | 0.5 |
| Fp1 | Lower passband edge frequency | GHz | 1.3 |
| Fp2 | Upper passband edge frequency | GHz | 2.7 |
| Fs2 | Lower stopband edge frequency | GHz | 3.5 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 0.1 |
| As | Stopband edge attenuation | dB | 20 |
| N | Number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | Type of frequency response (maximally flat or Chebyshev) | None | Chebyshev |
| StubConfig | Configuration of interior shunt stubs | None | Two Parallel Stubs |
| StubType | Shunt Stub Type | None | Short Circuit Quarter Wave |
| Zo | Desired input/output impedance | Ohm | 50 |
| D | Impedance control parameter (0 < D < 1) | None | 1 |
| Finf | Frequency of infinite attenuation (for open circuit stub type) | GHz | 1.0 |
| Delta | Length added to stubs for tuning performance | mil | 0 |

### Notes

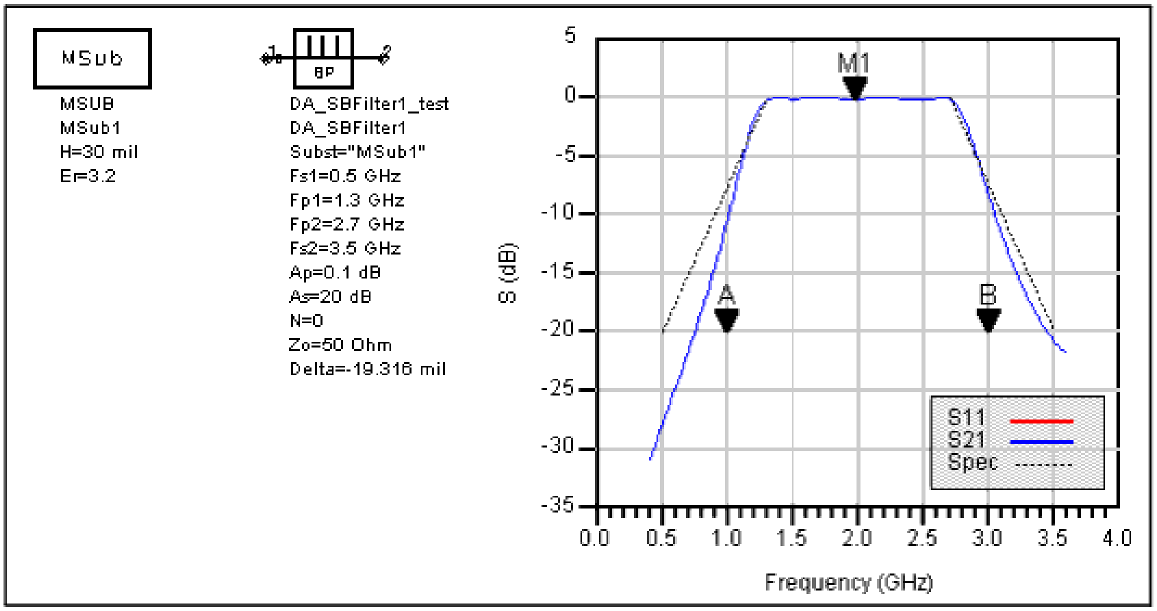
1. A stub bandpass filter provides a bandpass frequency response between the input and output ports. The design uses shunt stubs connected by lengths of transmission line. If the specified passband response is too narrow, large differences in impedance values can result in a non-realizable configuration.
2. This design is typically practical for fractional bandwidths of 0.4 to 0.7 or higher. If the bandwidth is too narrow, the design will generally require large differences in impedances between the stubs and the connecting lines, producing an unrealizable configuration.
3. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.
4. The parameter StubConfig specifies whether the interior stubs (all but those closest to the source and load) are implemented as a single stub or as two stubs in parallel. Choosing a single stub often produces narrower stub line widths, and therefore this parameter can impact the ability to manufacture.
5. The parameter D offers some control over the ratio between the stub impedances at the ends of the filter to those in the interior. In some cases where a MCROSS or MTEE width constraint violation is encountered, decreasing (or increasing) this value in the range 0 < D < 1 can remedy the problem.
6. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
7. The optimization minimizes the absolute difference between S21 and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequencies. All stub lengths are tuned by the same amount. Because only the stub lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and lengths of all lines.
8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: Matthaei, Young and Jones, Microwave

Filters, Impedance-Matching Networks, and Coupling Structures, Artech House, 1980, pp. 595-608.

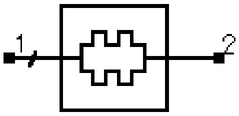
### Example

A stub bandpass filter was designed for a Chebyshev response with a 0.1 dB ripple. The passband edge frequencies are at 1.3GHz and 2.7 GHz respectively. The design uses two parallel stubs in the interior regions with D = 1. The design required 4 stubs. Tuning using the Optimization Assistant yielded a value of Delta = -19.316 mil.



# SIFilter (Stepped Impedance Lowpass Filter)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| Fp | Frequency at passband edge | GHz | 1 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| Fs | Frequency at stopband edge | GHz | 1.2 |
| As | Stopband edge attenuation | dB | 20 |
| N | Number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | Type of frequency response (maximally flat or Chebyshev) | None | Maximally Flat |
| FElement | First filter component (automatic, capacitive, or inductive) | None | Automatic |
| Zo | Desired input/output impedance | Ohm | 50 |
| ZL | Characteristic impedance of low impedance sections | Ohm | 25 |
| ZH | Characteristic impedance of high impedance sections | Ohm | 100 |
| Delta | Length added to filter sections for tuning performance | mil | 0 |

### Notes

1. A stepped impedance filter provides a lowpass frequency response between the input

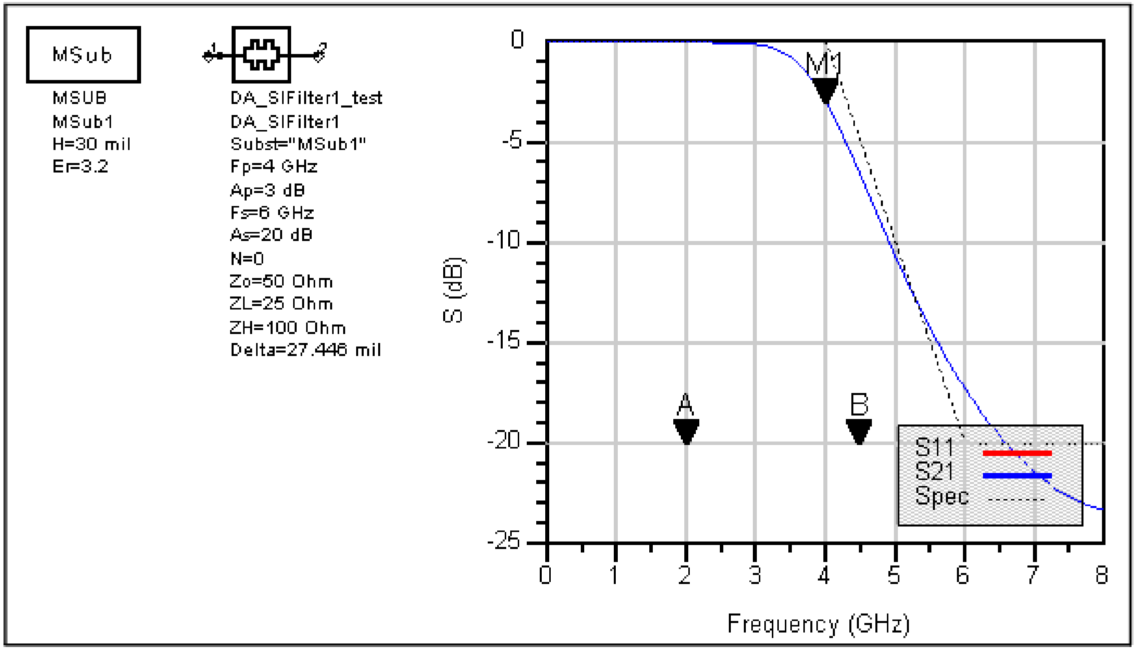
and output ports. The design is realized using alternating wide and narrow microstrip lines.

2. The stepped impedance filter uses wide microstrip lines to approximate shunt capacitors and narrow lines to approximate series inductors in order to provide a lowpass frequency response.
3. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics. In order to obtain an impedance match, Chebyshev designs must use an odd number of components (N).
4. The parameter FElement specifies whether the first stub is inductive or capacitive. If Automatic is chosen, the first component is inductive if Zo/ZL > ZH/Zo and capacitive otherwise.
5. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
6. Because this filter design strategy is approximate, the resulting stopband attenuation can not satisfy the specification. Choosing more sections than that computed by the design can improve the stopband performance.
7. The optimization minimizes the absolute difference between S21 and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequency. All filter sections are tuned by the same amount.
8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: Matthaei, Young and Jones, Microwave Filters, Impedance-Matching Networks, and Coupling Structures, Artech House, 1980, pp. 365-374.

### Example

A stepped impedance lowpass filter was designed for a maximally flat response with a 3 dB attenuation at the passband edge frequency of 4 GHz. Choosing FElement as Automatic results in a capacitive first component. Tuning using the Optimization Assistant yielded a value of Delta = 27.446 mil.

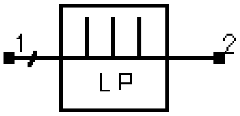


MSub
MSUB
MSub1
H=30 mil
Er=3.2

DA_SIFilter1_test
DA_SIFilter1
Subst="MSub1"
Fp=4 GHz
Ap=3 dB
Fs=6 GHz
As=20 dB
N=0
Zo=50 Ohm
ZL=25 Ohm
ZH=100 Ohm
Delta=27.446 mil

## SLFilter (Stub Lowpass Filter)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| Fp | Frequency at passband edge | GHz | 1 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| Fs | Frequency at stopband edge | GHz | 1.2 |
| As | Stopband edge attenuation | dB | 20 |
| N | Number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | Type of frequency response (maximally flat or Chebyshev) | None | Maximally Flat |
| StubType | Type of stubs (commensurate or variable length) | None | Variable Length Stub |
| FElement | First filter component (automatic, capacitive, or inductive) | None | Automatic |
| Zo | Desired input/output impedance | Ohm | 50 |
| ZS | Characteristic impedance of stubs | Ohm | 50 |
| ZH | Characteristic impedance of connecting sections | Ohm | 90 |
| Delta | Length added to stubs for tuning performance | mil | 0 |

### Notes

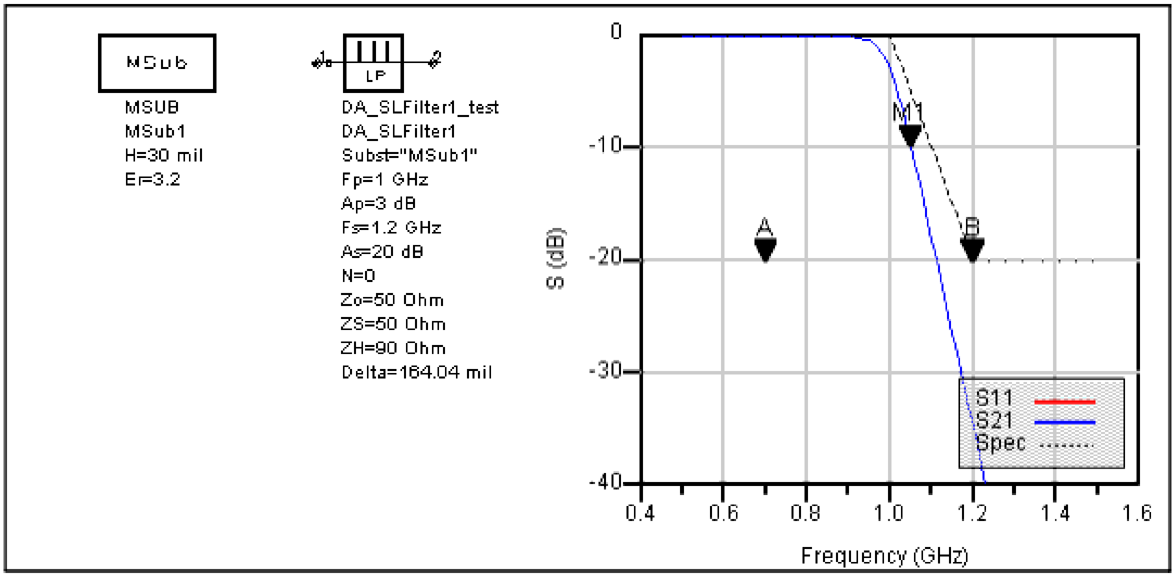
1. A stub lowpass filter provides a lowpass frequency response between the input and output ports. The design is realized using narrow lines to approximate series inductances and shunt open circuited stubs to realize shunt capacitances.
2. The stub lowpass filter uses narrow microstrip lines that approximate series inductors connecting open-circuited stubs that approximate shunt capacitors in order to provide a lowpass frequency response.
3. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics. In order to obtain an impedance match, Chebyshev designs must use an odd number of components (N).
4. If the parameter StubType is set as Commensurate, all stubs will have equal lengths, and the stub line width is computed. The value of ZS is then ignored, and the resulting design can be difficult to realize. If StubType is set as Variable Length Stub, the stub line widths are determined from ZS and the stub lengths are computed from the specifications.
5. The parameter FElement specifies whether the first stub is inductive or capacitive. If Automatic is chosen, the first component is set to be capacitive since this tends to offer improvements in manufacturing.
6. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency. Because this filter design strategy is approximate, the resulting stopband attenuation may not satisfy the specification. Choosing more sections than that computed by the Design Assistant can improve the stopband performance.
7. The optimization minimizes the absolute difference between S21 and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequency by changing the stub lengths. All stubs are tuned by the same amount.

8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, refer to: Matthaei, Young and Jones, Microwave Filters, Impedance-Matching Networks, and Coupling Structures, Artech House, 1980, pp. 375.

### Example

A stub lowpass filter was designed for a maximally flat response with a 3 dB attenuation at the passband edge frequency of 1 GHz. A variable length StubType and automatic FElement resulted in 13 components for the design. Tuning using the Optimization Assistant yielded a value of Delta = 164.04 mil.

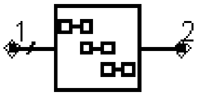


MSUB
MSub1
H=30 mil
Er=3.2

DA_SLFilter1_test
DA_SLFilter1
Subst="MSub1"
Fp=1 GHz
Ap=3 dB
Fs=1.2 GHz
As=20 dB
N=0
Zo=50 Ohm
ZS=50 Ohm
ZH=90 Ohm
Delta=164.04 mil

# SRFilter (Stepped Impedance Resonator Filter)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| Fs1 | Lower stopband edge frequency | GHz | 1.8 |
| Fp1 | Lower passband edge frequency | GHz | 2 |
| Fp2 | Upper passband edge frequency | GHz | 2.2 |
| Fs2 | Lower stopband edge frequency | GHz | 2.4 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| As | Stopband edge attenuation | dB | 20 |
| N | Number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | Type of frequency response (maximally flat or Chebyshev) | None | Maximally Flat |
| Zo | Desired input/output impedance | Ohm | 50 |
| Fsp | Center frequency of first spurious passband | GHz | 4.1 |
| Delta | Length added to filter sections for tuning performance | mil | 0 |

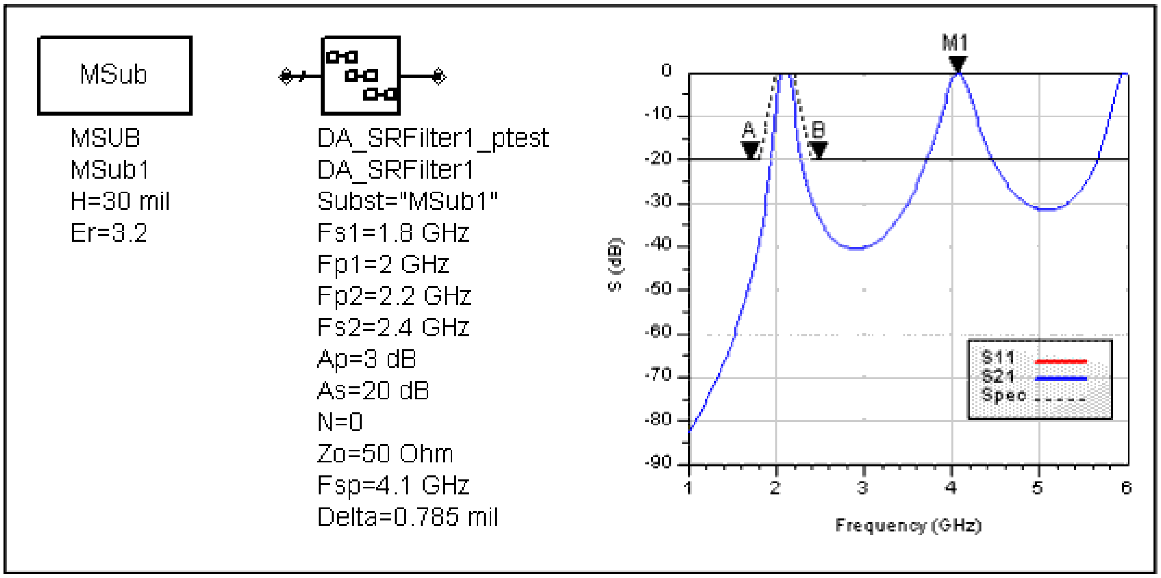
### Notes

1. A stepped impedance resonator filter provides a bandpass frequency response between the input and output ports. N coupled-line sections produce an N−1 order filter response. Additional numbers of sections can be used to steepen the transition band roll off or widen the pass bandwidth. The use on non-uniform impedance for each resonator moves the second pass band center frequency away from the second harmonic of the fundamental frequency.
2. Because of the heavy computational burden in determining the line parameters, a brief delay will occur for the design.
3. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.
4. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
5. The center frequency of the first spurious passband should be on the order of twice the fundamental passband center frequency.
6. The optimization minimizes the absolute difference between S21 in dB and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequencies. Because only the line lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and spacings of the coupled filter sections.
7. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, refer to: Makimoto and Yamashita, "Bandpass filters using parallel coupled stripline stepped impedance resonators," *IEEE Trans. Microwave Theory and Techniques* , vol MTT-28, pp. 1413-1417, 1980.

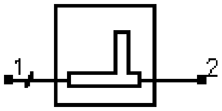
### Example

A stepped impedance resonator filter was designed for a maximally flat response with the 3 dB passband edge frequencies at 2 GHz and 2.2 GHz respectively. The second passband was set to 4.1 GHz. The design required 4 coupled lines. Tuning using the Optimization Assistant yielded a value of Delta = 0.785 mil.



## SSMatch (Single-Stub Match)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| Zin | Desired complex input impedance | Ohm | 50 |
| Zload | Complex load impedance to match | Ohm | 100 |
| Zstub | Characteristic impedance of stub line | Ohm | 50 |
| Zline | Characteristic impedance of line | Ohm | 50 |
| Zfeed | Characteristic impedance of line connected to port 1 or 2 | Ohm | 50 |
| StubType | Stub type (open or short circuit) | None | Open Circuit |
| NetType | Network type (source to load) | None | Automatic |
| Delta | Length added to stub for tuning performance | mil | 0 |

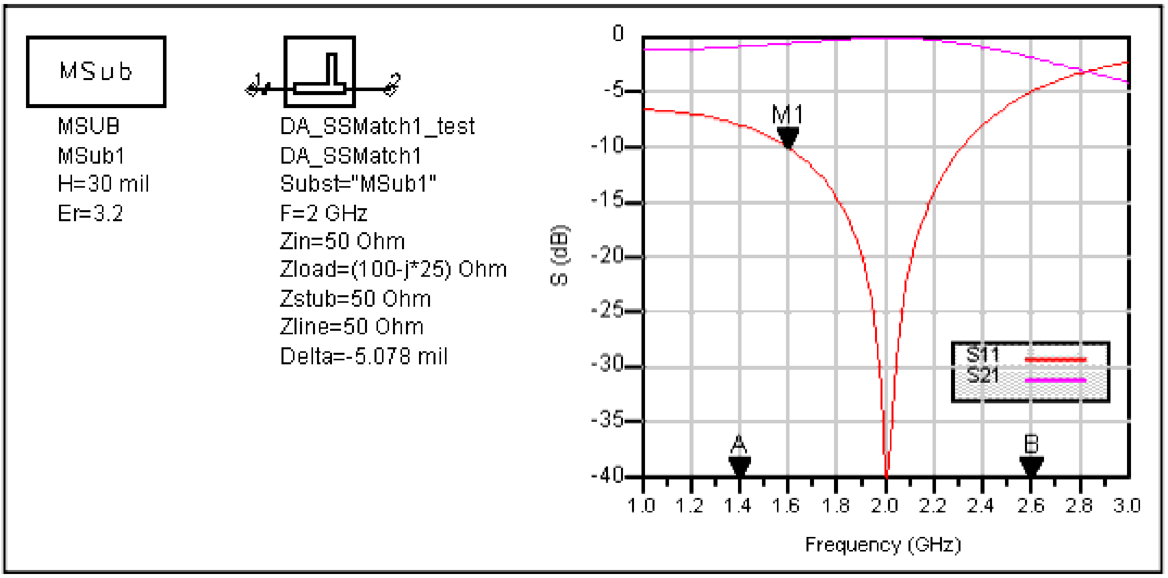
### Notes

1. A single-stub matching network matches a complex load impedance (Zload) to a desired complex input impedance (Zin) using a single shunt stub and length of line.
2. The parameter NetType can be Automatic, stub-line, or line-stub, with the latter two choices representing configurations that have a stub followed by a line or vice-versa as the network is observed from source to load. Many impedance combinations can be realized using both possible types, although some can only be realized using one of the two choices. Choosing Automatic will ensure a realizable choice is given.
3. An impedance match can be realized using either stub type.
4. Zline represents the impedance of the line either next to the source or load (depending on the value of NetType). Zfeed represents the desired impedance on the other side of the stub from the line and is used to ensure that the corresponding leg of the microstrip tee component is of the proper width.
5. MTEE component width constraint violations will be avoided generally by choosing similar characteristic impedances for the line, stub, and feed.
6. The input port termination is set to the conjugate of Zin so that the ideal input reflection coefficient will be zero. The output port termination is set to Zload.
7. The optimization minimizes the value of S11 (referenced to the conjugate of Zin) at the design center frequency by changing the length of the stub.
8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 258-266.

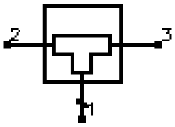
### Example

A single-stub matching network was designed to match a load impedance of 100 - j25 Ohms to a 50 Ohms line at a center frequency of 2GHz. Choosing an open-circuit stub with automatic selection of the NetType resulted in a stub-line configuration. Tuning using the Optimization Assistant yielded a value of Delta =-5.078 mil.



# TCoupler (Tee Power Divider)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| Zo1 | Characteristic impedance of input port 1 | Ohm | 50 |
| Zo2 | Characteristic impedance of output port 2 | Ohm | 50 |
| Zo3 | Characteristic impedance of output port 3 | Ohm | 50 |
| K | Ratio of power out port 2 to power out port 3 | None | 1.0 |
| Delta | Length added to quarter-wave sections for tuning performance | mil | 0 |

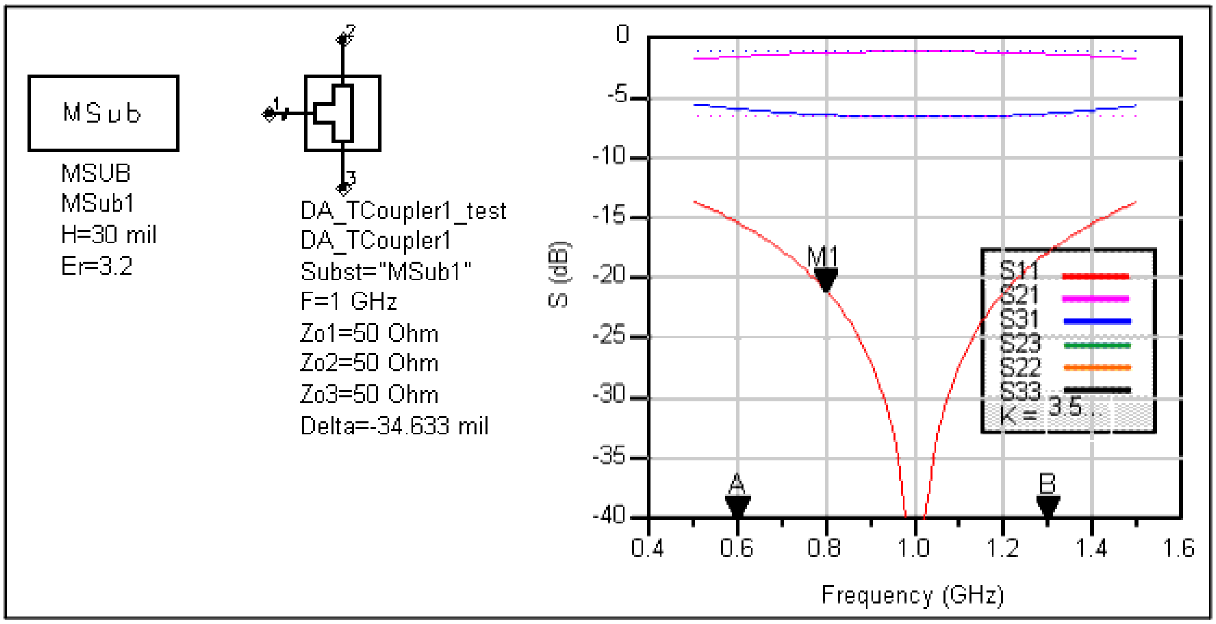
### Notes

1. A tee power divider splits the power at the input (pin 1) between the two outputs (pins 2 and 3). Unequal or equal power splits can be realized. The input port will be matched to its feeding line, although in general the output ports will not be matched.
2. The value of K can be set to realize the desired power split out of ports 2 and 3. However, choosing K larger than 3 to 4 (or smaller than 1/3 to 1/4) can cause the ratio of the widths of the tee branches to violate the range of the MTEE simulation model. While the simulation will still proceed, the results can have some inaccuracies.
3. Quarter-wave matching sections are provided on the output ports to ensure a proper power split is achieved.
4. The optimization minimizes the input reflection coefficient (S11) at the design center frequency by changing the length of the quarter wave transformers on the output legs.
5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 360-361.

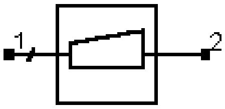
### Example

A tee power divider was designed for a center frequency of 1 GHz with an unequal power split (K=3.5). Tuning using the Optimization Assistant yielded a value of Delta = -34.633 mil.



## TLMatch (Tapered-Line Match)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Design frequency | GHz | 1 |
| Zo | Desired input impedance | Ohm | 50 |
| Rload | Load impedance to match | Ohm | 100 |
| ResponseType | Type of frequency response (exponential, triangular, Klopfenstein) | None | Exponential |
| L | Length of tapered line in wavelengths at frequency F (set L=0 to compute L) | None | 0.0 |
| Rmax | Maximum voltage reflection coefficient | None | 0.1 |
| NSection | Number of linear taper sections per wavelength | None | 20 |

### Notes

1. A tapered-line matching network matches a real load impedance (Rload) to a desired real input impedance (Zo) using a continuously varying line characteristic impedance to realize a specified frequency response. The reflection coefficient remains below the specified maximum value for all frequencies above the design frequency.
2. The continuous impedance taper of this circuit is approximated using a sequence of linear tapers. Increasing the number of sections per wavelength (NSection) will improve the approximation to the defined taper.
3. If the length of the line L is set to zero, the Design Assistant chooses L such that the reflection coefficient is less than Rmax for all frequencies above the design frequency.

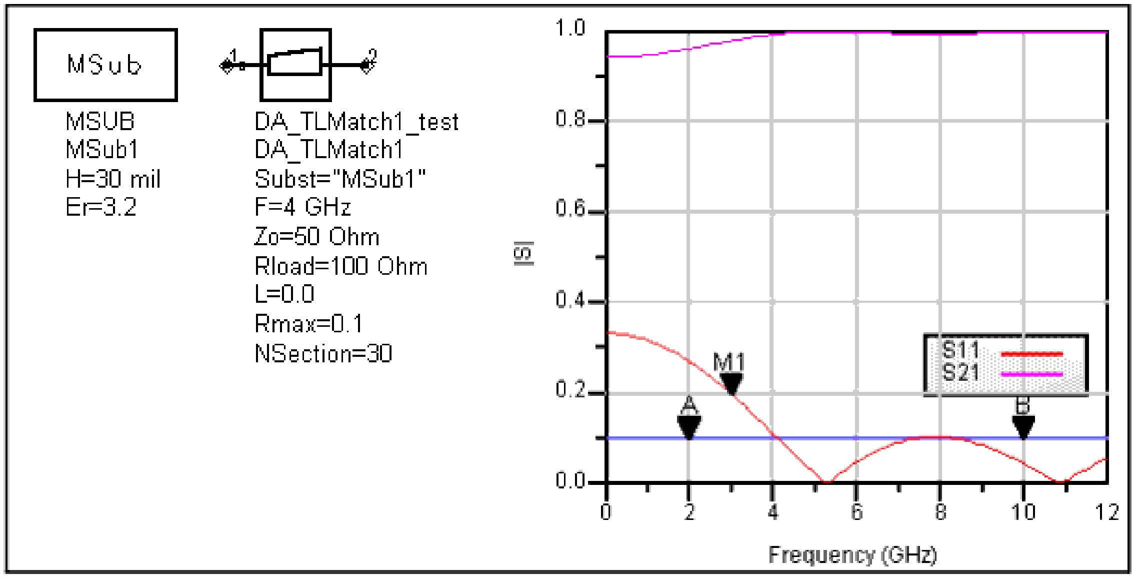
Otherwise, Rmax is ignored.

4. The ResponseType specifies the distribution of the impedance along the length of the line and therefore determines the frequency response of the match.
5. No Optimization Assistant is provided since the design process is exact to within the approximation of the continuous impedance taper using multiple linear taper segments. Increasing the value of NSection will improve the computed response.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

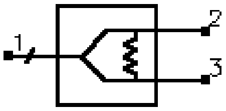
For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 288-295.

### Example

A tapered-line matching network was designed to match a load impedance of 100 Ohms to a 50 Ohms line at a center frequency of 4 GHz. Specifying a Klopfenstein frequency response for which the reflection coefficient remains below 0.1 along with 30 sections per wavelength dictates 18 tapered-line sections to realize the match. The plot has been shown in linear magnitude rather than dB to emphasize the equal-ripple frequency response.



## WDCoupler (Wilkinson Divider)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| F | Center frequency | GHz | 1 |
| DeltaF | Frequency bandwidth | GHz | 0.5 |
| Zo | Characteristic impedance | Ohm | 50 |
| ResponseType | Type of frequency response | None | Uniform |
| N | Number of quarter-wave sections; set N=0 to compute N | None | 0 |
| Rmax | Maximum voltage reflection coefficient on input port | None | 0.1 |
| K | Ratio of power out port 2 to power out port 3 | None | 1.0 |
| Wgap | Width of gap for resistor | mil | 50 |
| Delta | Length added to quarter-wave branches for tuning performance | mil | 0 |

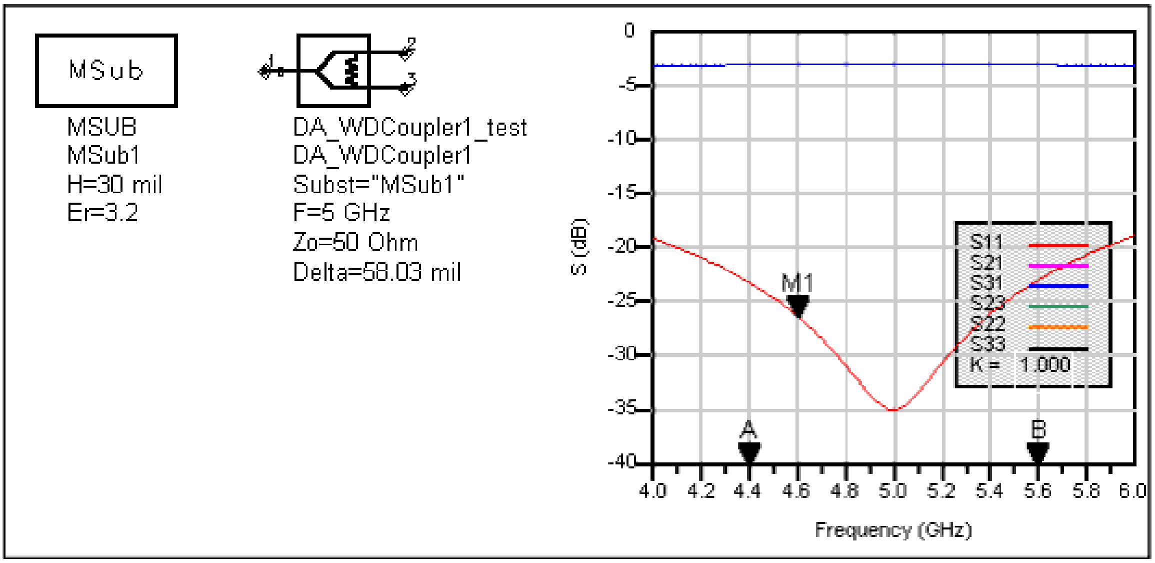
### Notes

1. A Wilkinson power divider splits the power at the input (pin 1) between the two outputs (pins 2 and 3). Unequal or equal power splits can be realized. The signals at the outputs are in phase. All three ports will be matched, and ports 2 and 3 will in general be well isolated from each other.
2. For broadband performance, the divider can have multiple quarter-wave sections. If the number of sections N is set to zero, the Design Assistant chooses N such that such that the reflection coefficient is less than Rmax over the bandwidth DeltaF (centered at the design center frequency). the resulting bandwidth can be broader than that specified. Otherwise, Rmax and DeltaF are ignored.
3. ResponseType specifies the distribution of the partial reflection coefficients seen at each section interface - Uniform, Binomial, and Chebyshev distributions are available. These in turn specify the shape of the reflection coefficient versus frequency.
4. For a single section divider (N=1), the value of K can be set to realize the desired power split out of ports 2 and 3. Be aware that choosing K larger than 3 to 4 (or smaller than 1/3 to 1/4) is likely to cause difficulties in the design.
5. Pozar specifies $K^2$= P3/P2, while the DesignGuide uses $K^2$= P2/P3. Therefore, if you use the equations in Pozar to verify everything, you must substitute 1/K for K. The DesignGuide automatically puts quarter-wave matching sections on ports 2 and 3, so all ports are matched to the characteristic impedance. If you remove these matching segments, the output impedances are those specified by Pozar.
6. The optimization minimizes the input reflection coefficient (S11) at the design center frequency by changing the length of the quarter wave branches forming the divider.
7. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 363-368.

### Example

A single-section Wilkinson power divider (N=1) was designed for a center frequency of 5 GHz with an equal power split (K=1) and a gap width for the resistor of 50 mil. Tuning using the Optimization Assistant yielded a value of Delta = 58.03 mil.

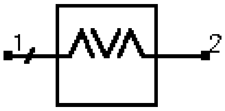


MSub

MSUB
MSub1
H=30 mil
Er=3.2

DA_WDCoupler1_test
DA_WDCoupler1
Subst="MSub1"
F=5 GHz
Zo=50 Ohm
Delta=58.03 mil

# ZZFilter (Zig-Zag Coupled-Line Filter)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Microstrip substrate name | None | MSub1 |
| Fs1 | Lower stopband edge frequency | GHz | 1.8 |
| Fp1 | Lower passband edge frequency | GHz | 2 |
| Fp2 | Upper passband edge frequency | GHz | 2.2 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 2.4 |
| As | Stopband edge attenuation | dB | 3 |
| N | Number of filter sections (or 0 to compute N) | None | 20 |
| ResponseType | Type of frequency response (maximally flat or Chebyshev) | None | 0 |
| Zo | Desired input/output impedance | Ohm | Maximally Flat |
| Delta | Length added to coupled sections for tuning performance | mil | 0 |

### Notes

1. A zig-zag coupled-line filter provides a bandpass frequency response between the input and output ports. The design uses a coupled-line filter topology but includes 90 degree bend between each coupled section so that the layout is more compact.
2. Because of the heavy computational burden in determining the line parameters, a brief delay will occur for the design.
3. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.
4. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
5. The optimization minimizes the absolute difference between S21 and the specified

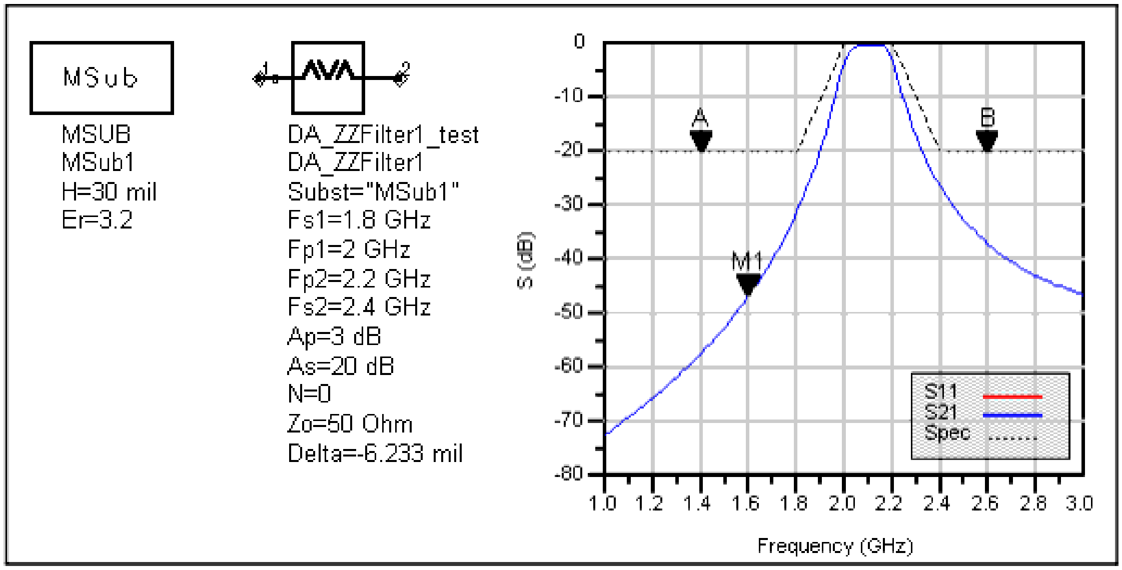
passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequencies. Because only the line lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and spacings of the coupled filter sections.

6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 477-485.

### Example

A zig-zag coupled-line filter was designed for a maximally flat response with the 3 dB passband edge frequencies at 2 GHz and 2.4 GHz respectively. The design required 4 coupled-line sections. Tuning using the Optimization Assistant yielded a value of Delta = -6.233 mil.



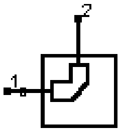
# Passive Stripline Components

- *SBend (Stripline Bend Component)* (dgpas)
- *SCLine (Stripline Coupled Line Component)* (dgpas)
- *SCross (Stripline Cross Component)* (dgpas)
- *SCurve (Stripline Curve Component)* (dgpas)
- *SLine (Stripline Line Component)* (dgpas)
- *SStep (Stripline Step Component)* (dgpas)
- *SStub (Stripline Stub Component)* (dgpas)
- *STee (Stripline Tee Component)* (dgpas)

**Note**
A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# SBend (Stripline Bend Component)

### Symbol



### Parameters

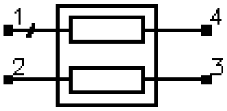
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Design frequency | GHz | 1 |
| Zo | Desired characteristic impedance | Ohm | 50 |
| BendType | Type of bend | None | Unmitered |
| Angle | Angle of bend (for SBEND, SBEND2) | None | 90 |
| M | Miter fraction (for SBEND2) | None | 0.6 |

### Notes

1. SBend designs a stripline bend given the substrate, desired characteristic impedance, and bend properties. The design realizes the native SBEND, SBEND2, or SMITER components.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. Since the design uses the models inherent to ADS to compute the line width, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
4. BendType can be Unmitered (SBEND), Arbitrary Angle/Miter (SBEND2), or 90 Degree/Optimally Mitered (SMITER). SBEND requires the parameter Angle and SBEND2 requires the parameters Angle and M. Refer to the discussion of these components in the *ADS Stripline Components* (ccdist) documentation for a more detailed description.
5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

## SCLine (Stripline Coupled Line Component)

### Symbol



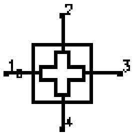
### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Design frequency | GHz | 1 |
| Zoe | Even-mode characteristic impedance | Ohm | 55 |
| Zoo | Odd-mode characteristic impedance | Ohm | 45 |
| Zo1 | Characteristic impedance of input line at port 1 | Ohm | 50 |
| Zo2 | Characteristic impedance of input line at port 2 | Ohm | 50 |
| Zo3 | Characteristic impedance of input line at port 3 | Ohm | 50 |
| Zo4 | Characteristic impedance of input line at port 4 | Ohm | 50 |
| Lphys | Physical line length; set to zero if Lelec specified | mil | 0 |
| Lelec | Line length in wavelengths; set to zero if Lphys specified | None | 0.25 |

### Notes

1. SCLine designs a stripline coupled line component given the substrate, desired even- and odd-mode characteristic impedances, and physical or electrical length.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. Since the design uses the models inherent to ADS to compute the line width, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
4. For proper operation, either Lphys or Lelec must be zero.
5. Zo1 through Zo4 specify the impedance of the lines attached to this component and are provided to ensure proper pin location in the layout. Refer to the discussion of the SCLIN component in the *ADS Stripline Components* (ccdist) documentation for a more detailed description of the model used for this component.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

## SCross (Stripline Cross Component)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Design frequency | GHz | 1 |
| Z1 | Characteristic impedance of port 1 | Ohm | 50 |
| Z2 | Characteristic impedance of port 2 | Ohm | 50 |
| Z3 | Characteristic impedance of port 3 | Ohm | 50 |
| Z4 | Characteristic impedance of port 4 | Ohm | 50 |

### Notes

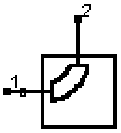
1. SCross designs a stripline cross given the substrate and desired characteristic

impedance on each port.

2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. Since the design uses the models inherent to ADS to compute the line width, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
4. Refer to the discussion of the SCROSS component in the *ADS Stripline Components* (ccdist) documentation for a detailed description of this component.
5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# SCurve (Stripline Curve Component)

**Symbol**



**Parameters**

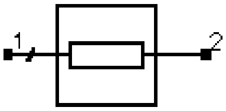
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Design frequency | GHz | 1 |
| Zo | Desired characteristic impedance | Ohm | 50 |
| Angle | Angle of curve | None | 90 |
| Radius | Radius of curvature (set to zero if Lelec specified) | mil | 100 |
| Lelec | Curve length in wavelengths (set to zero if Radius specified) | None | 0 |

**Notes**

1. SCurve designs a stripline curve given the substrate, desired characteristic impedance, and curve properties. The design realizes the native SCURVE component.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. Since the design uses the models inherent to ADS to compute the line width, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
4. Either Lelec or Radius must be zero for proper operation.
5. Refer to the discussion of the SCURVE component in the *ADS Stripline Components* (ccdist) documentation for a more detailed description.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# SLine (Stripline Line Component)

**Symbol**



**Parameters**

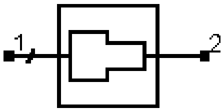
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Design frequency | GHz | 1 |
| Zo | Desired characteristic impedance | Ohm | 50 |
| Lphys | Physical line length; set to zero if Lelec specified | mil | 0 |
| Lelec | Line length in wavelengths; set to zero if Lphys specified | None | 0.25 |

### Notes

1. SLine designs a stripline given the substrate, desired characteristic impedance, and physical or electrical length.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. Since the design uses the models inherent to ADS to compute the line width and length, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
4. For proper operation, either Lphys or Lelec must be zero.
5. Refer to the discussion of the SLIN component in the *ADS Stripline Components* (ccdist) documentation for a more detailed description of the model used for this component.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# SStep (Stripline Step Component)

### Symbol



### Parameters

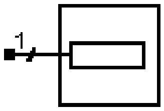
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Design frequency | GHz | 1 |
| Z1 | Characteristic impedance of port 1 | Ohm | 50 |
| Z2 | Characteristic impedance of port 2 | Ohm | 25 |

### Notes

1. SStep designs a stripline step given the substrate and desired characteristic impedances.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. Since the design uses the models inherent to ADS to compute the line width, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
4. Refer to the discussion of the SSTEP component in the *ADS Stripline Components* (ccdist) documentation for a detailed description of this component.
5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# SStub (Stripline Stub Component)

**Symbol**



**Parameters**

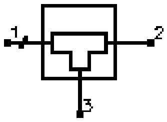
| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Design frequency | GHz | 1 |
| Zo | Desired characteristic impedance | Ohm | 50 |
| Lphys | Physical line length | mil | 0 |
| Lelec | Line length in wavelengths | None | 0 |
| Xin | Desired input reactance | Ohm | 0 |
| Cin | Desired input capacitance | pF | 0 |
| Lin | Desired input inductance | nH | 0 |
| StubType | Type of stub | None | Open Circuit |

**Notes**

1. SStub designs a stripline open or short circuited stub given the substrate, desired characteristic impedance, and physical or electrical length. The design realizes the native SLOC, SLSC, and SLEF components.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. Only one of Lphys, Lelec, Xin, Cin, and Lin can be non-zero.
4. Since the design uses the models inherent to ADS to compute the line width and length, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
5. For proper operation, only one of Lphys, Lelec, Xin, Cin, and Lin can be non-zero.
6. StubType can be either Open Circuit (SLOC), End Effect (SLEF), or Short Circuit (SLSC). Refer to the discussion of these components in the *ADS Stripline Components* (ccdist) documentation for a more detailed description of these different options.
7. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# STee (Stripline Tee Component)

**Symbol**



**Parameters**

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Design frequency | GHz | 1 |
| Z1 | Characteristic impedance of port 1 | Ohm | 50 |
| Z2 | Characteristic impedance of port 2 | Ohm | 50 |
| Z3 | Characteristic impedance of port 3 | Ohm | 50 |

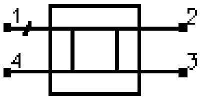
### Notes

1. STee designs a stripline tee given the substrate and desired characteristic impedance at each port.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. Since the design uses the models inherent to ADS to compute the line width and length, there is no need for a dedicated Simulation Assistant, Optimization Assistant, or Display Assistant.
4. Z1, Z2, and Z3 are used to determine the widths of each port. Refer to the discussion of the STEE component in the *ADS Stripline Components* (ccdist) documentation for a more detailed description of this component.
5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

# Passive Stripline Circuits

- *SBLCoupler (Stripline Branch-Line Coupler)* (dgpas)
- *SCLCoupler (Stripline Coupled-Line Coupler)* (dgpas)
- *SCLFilter (Stripline Coupled-Line Filter)* (dgpas)
- *SCMFilter (Stripline Comb-Line Filter)* (dgpas)
- *SDSMatch (Stripline Double-Stub Match)* (dgpas)
- *SHPFilter (Stripline Hairpin Filter)* (dgpas)
- *SIDFilter (Stripline Interdigital Filter)* (dgpas)
- *SQWMatch (Stripline Quarter-Wave Match)* (dgpas)
- *SRRCoupler (Stripline Rat-Race Coupler)* (dgpas)
- *SSBFilter (Stripline Stub Bandpass Filter)* (dgpas)
- *SSIFilter (Stripline Stepped Impedance Lowpass Filter)* (dgpas)
- *SSLFilter (Stripline Stub Lowpass Filter)* (dgpas)
- *SSRFilter (Stripline Stepped Impedance Resonator Filter)* (dgpas)
- *SSSMatch (Stripline Single-Stub Match)* (dgpas)
- *STCoupler (Stripline Tee Power Divider)* (dgpas)
- *SWDCoupler (Stripline Wilkinson Divider)* (dgpas)

# SBLCoupler (Stripline Branch-Line Coupler)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Center frequency | GHz | 1 |
| DeltaF | Total frequency bandwidth | GHz | 0.5 |
| Zo | Characteristic impedance | Ohm | 50 |
| ResponseType | Type of frequency response | None | Chebyshev |
| N | Number of coupler sections; set N=0 to compute N | None | 0 |
| Rmax | Maximum voltage reflection coefficient at the input | None | 0.5 |
| C | Coupling coefficient | dB | 3 |
| Delta | Length added to branches for tuning performance | mil | 0 |

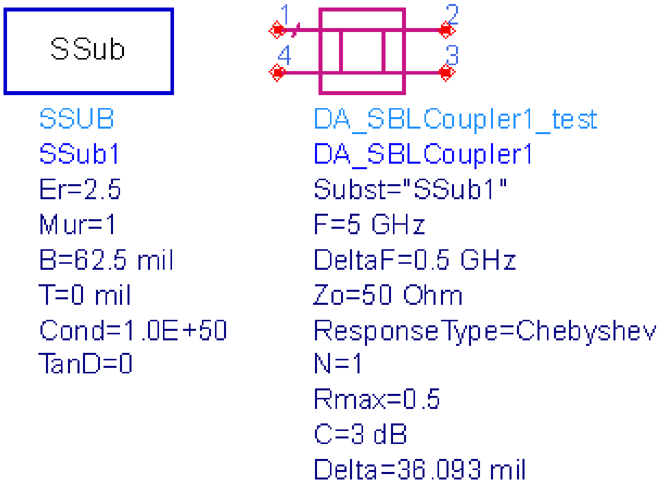
### Notes

1. A branch-line coupler outputs from the coupled port (pin 3) a fraction of the power presented at the input (pin 1). The remainder of the power is passed through to the output port (pin 2). At the center frequency the phase difference between the outputs is 90 degrees, with the coupled port representing the quadrature (Q) output and the output port representing the in-phase (I) output. The coupling coefficient specifies the ratio of the input power to the coupled power ($P_1/P_3$). Pin 4 represents the isolated port, and it is typically well isolated from the input port near the center frequency.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. The coupling coefficient must be positive and greater than 3 dB. Best results are obtained for tight couplings of 6 dB or better (C < 6 dB). Choosing the coupling parameter larger than 6 dB often causes width constraint violations to occur on the STEE components, resulting in warning messages during design and simulation. A coupling coefficient of 3 dB provides an equal power split between the two outputs.
4. For broadband performance, the coupler can have multiple sections. If the number of sections N is set to zero, the Design Assistant chooses N such that the reflection coefficient is less than Rmax over the bandwidth DeltaF (centered at the design center frequency). The resulting bandwidth can be broader than that specified. Otherwise, rmax and DeltaF are ignored.
5. The ResponseType specifies the distribution of the partial reflection coefficients seen at each section interface - Uniform, Binomial, and Chebyshev distributions are available.
6. The optimization minimizes the input reflection coefficient (S11) at the design center frequency by changing the length of the lines forming the four branches. All branches are changed by the same physical length during the optimization.This optimization generally provides very good results but can not guarantee that the specified coupling is attained at the design frequency. More advanced tuning can be performed by changing line width of the branch lines.
7. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 379-383.

### Example

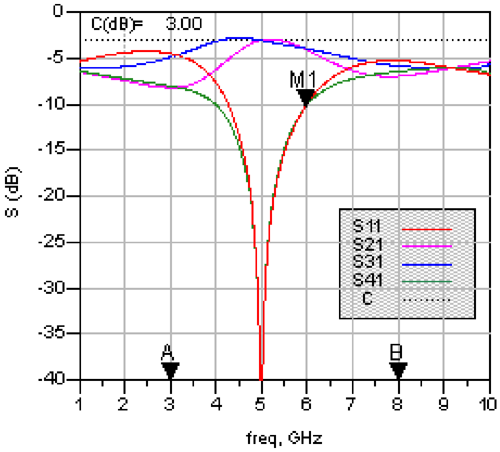
A single-section branch-line coupler was designed for a center frequency of 5 GHz with an equal power split between the I and Q ports. Tuning using the Optimization Assistant yielded a value of Delta = 36.093 mil.



SSub

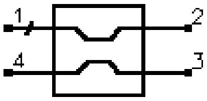
SSUB
SSub1
Er=2.5
Mur=1
B=62.5 mil
T=0 mil
Cond=1.0E+50
TanD=0

DA_SBLCoupler1_test
DA_SBLCoupler1
Subst="SSub1"
F=5 GHz
DeltaF=0.5 GHz
Zo=50 Ohm
ResponseType=Chebyshev
N=1
Rmax=0.5
C=3 dB
Delta=36.093 mil



# SCLCoupler (Stripline Coupled-Line Coupler)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Center frequency | GHz | 1 |
| C | Coupling coefficient | dB | 20 |
| Zo | Characteristic impedance | Ohm | 50 |
| Delta | Length added to branches for tuning performance | mil | 0 |

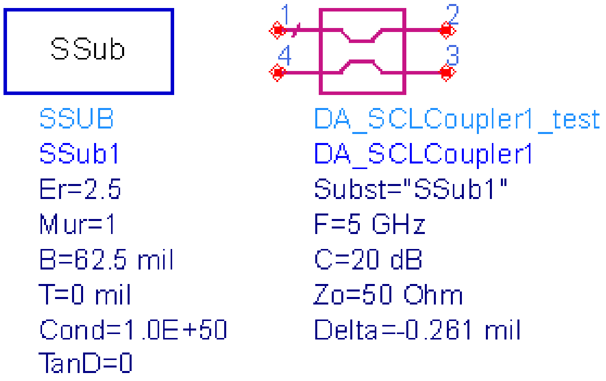
### Notes

1. A coupled-line coupler outputs from the coupled port (pin 4) a fraction of the power presented at the input (pin 1). The remainder of the power is passed through to the output port (pin 2). The coupling coefficient specifies the ratio of the input power to the coupled power ($P_1/P_4$). The remaining port is isolated, although the isolation is often similar in value to the coupling coefficient for microstrip realizations.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. The optimization minimizes the absolute difference between S41 in dB and the specified coupling coefficient at the design center frequency by changing the length of the coupled-line section.
4. The coupling coefficient must be positive and greater than 3 dB. Best results are obtained for weak couplings of roughly 10 dB or more (C > 10dB). Choosing the coupling coefficient too small can require a spacing between the coupled lines too small to realize.
5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 383-394.

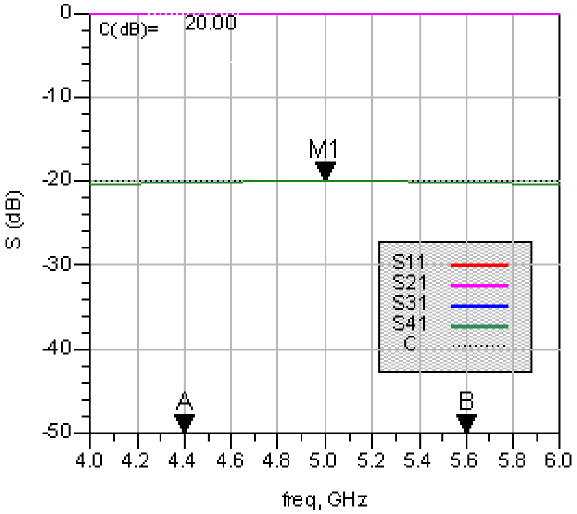
### Example

A coupled-line coupler was designed for a center frequency of 5 GHz with 20 dB of coupling. Tuning using the Optimization Assistant yielded a value of Delta = -0.261 mil.



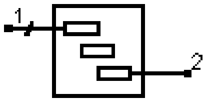
SSUB
SSub1
Er=2.5
Mur=1
B=62.5 mil
T=0 mil
Cond=1.0E+50
TanD=0

DA_SCLCoupler1_test
DA_SCLCoupler1
Subst="SSub1"
F=5 GHz
C=20 dB
Zo=50 Ohm
Delta=-0.261 mil



# SCLFilter (Stripline Coupled-Line Filter)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| Fs1 | Lower stopband edge frequency | GHz | 1.8 |
| Fp1 | lower passband edge frequency | GHz | 2 |
| Fp2 | upper passband edge frequency | GHz | 2.2 |
| Fs2 | lower stopband edge frequency | GHz | 2.4 |
| Ap | passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| As | stopband edge attenuation | dB | 20 |
| N | number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | type of frequency response (maximally flat or Chebyshev) | None | Maximally Flat |
| Zo | desired input/output impedance | Ohm | 50 |
| CouplingType | Type of input/output coupling | None | Coupled Line Transformer Input |
| Delta | length added to coupled sections for tuning performance | mil | 0 |

### Notes

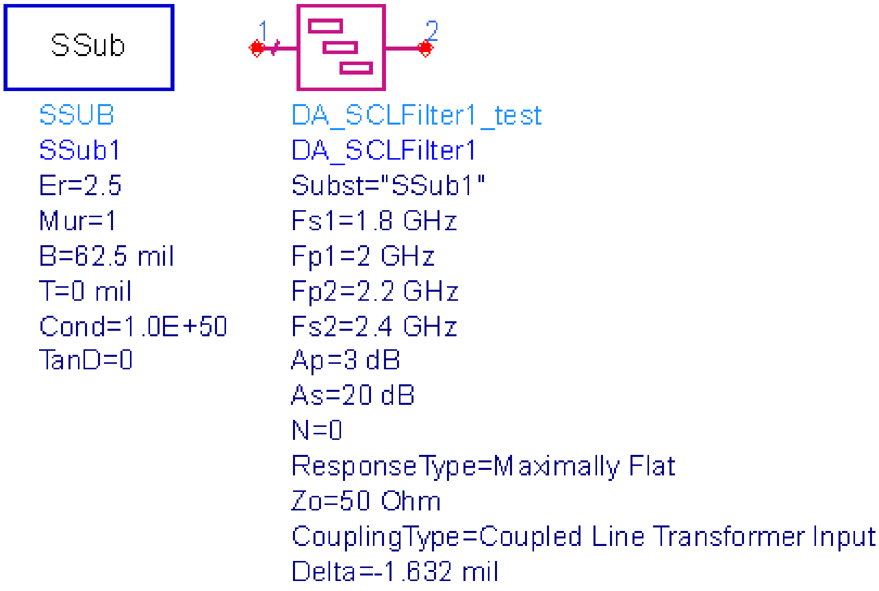
1. A coupled-line filter provides a bandpass frequency response between the input and output ports. N coupled-line sections produces an N-1 order filter response. Additional numbers of sections can be used to steepen the transition band roll off or widen the pass bandwidth.

2. Because of the heavy computational burden in determining the line parameters, a brief delay will occur for the design.
3. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
4. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.
5. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
6. Using a Coupled Line Transformer Input CouplingType will use and extra coupled line section on the inputs and outputs to feed the device. Choosing Tapped Line Transformer Input will bring the feedline directly into the first resonator.
7. The optimization minimizes the absolute difference between S21 in dB and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequencies. Because only the line lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and spacings of the coupled filter sections.
8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 477-485.

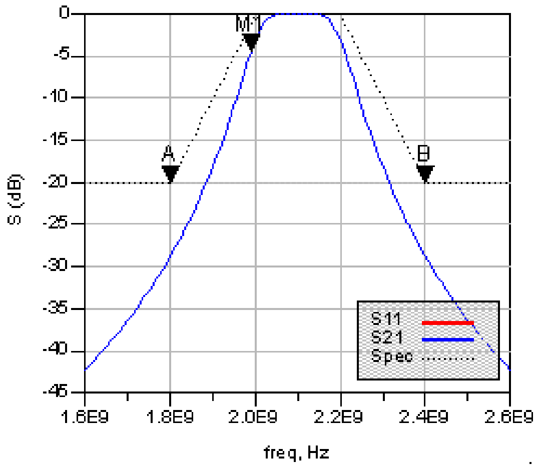
### Example

A coupled-line filter was designed for a maximally flat response with the 3 dB passband edge frequencies at 2 GHz and 2.4 GHz respectively. Coupled Line Transformer Inputs were used. The design required 4 coupled-line sections. Tuning using the Optimization Assistant yielded a value of Delta = -1.632 mil.



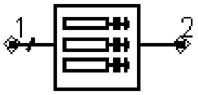
SSUB
SSub1
Er=2.5
Mur=1
B=62.5 mil
T=0 mil
Cond=1.0E+50
TanD=0

DA_SCLFilter1_test
DA_SCLFilter1
Subst="SSub1"
Fs1=1.8 GHz
Fp1=2 GHz
Fp2=2.2 GHz
Fs2=2.4 GHz
Ap=3 dB
As=20 dB
N=0
ResponseType=Maximally Flat
Zo=50 Ohm
CouplingType=Coupled Line Transformer Input
Delta=-1.632 mil



# SCMFilter (Stripline Comb-Line Filter)

## Symbol



## Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | Subst1 |
| Fs1 | Lower stopband edge frequency | GHz | 1.8 |
| Fp1 | Lower passband edge frequency | GHz | 2 |
| Fp2 | Upper passband edge frequency | GHz | 2.2 |
| Fs2 | Lower stopband edge frequency | GHz | 2.4 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| As | Stopband edge attenuation | dB | 20 |
| N | Number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | Type of frequency response (maximally flat or Chebyshev) | None | Maximally Flat |
| Zo | Desired input/output impedance | Ohm | 50 |
| Lelec | Electrical length of filter resonators (the units of Leclec are in fractions of a wavelength i.e., 0.25=1/4 wavelength) | None | 0.05 |
| ya | Normalized interior resonator admittance (0 < ya < 1) | None | 0.7 |
| CouplingType | Type of input/output coupling (coupled line or tapped line transformer) | None | Tapped Line Transformer Input |
| Delta | Length added to coupled sections for tuning performance | mil | 0 |

## Notes

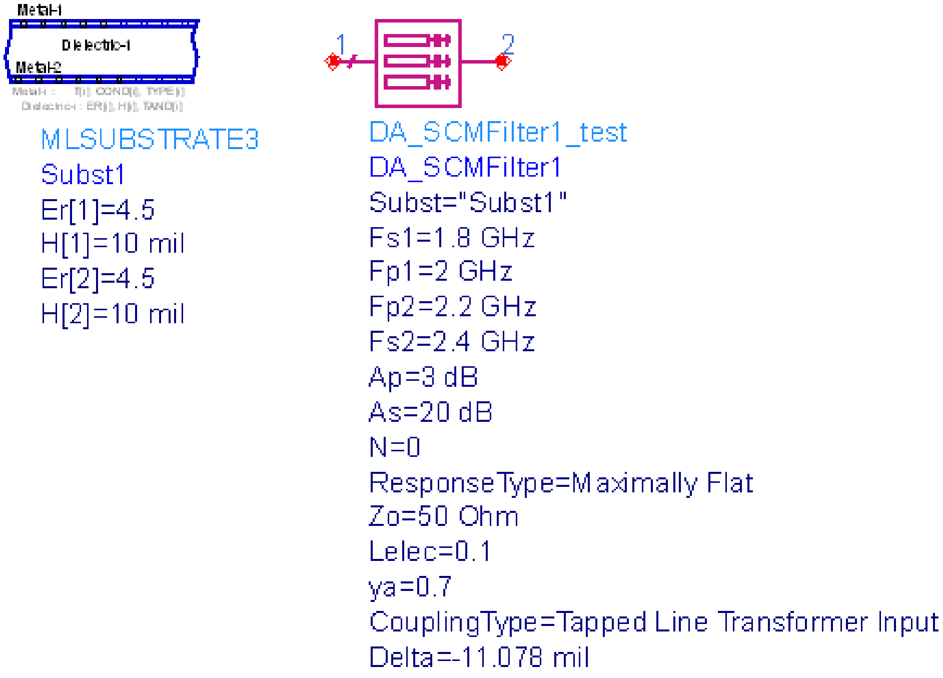
1. A comb-line filter provides a bandpass frequency response between the input and output ports. N coupled-line sections produce an N-1 order filter response. Additional numbers of sections can be used to steepen the transition band roll off or widen the pass bandwidth.
2. Because of the heavy computational burden in determining the line parameters, a brief delay will occur for the design.
3. A three-layer substrate must be used for this topology.
4. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.

5. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
6. Using a CouplingType of "Coupled Line Transformer Input" will use an extra coupled line section on the inputs and outputs to feed the device. Choosing "Tapped Line Transformer Input" will bring the feedline directly into the first resonator.
7. The optimization minimizes the absolute difference between S21 in dB and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequencies. Because only the line lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and spacings of the coupled filter sections.
8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, refer to: Matthaei, Young and Jones, Microwave Filters, Impedance-Matching Networks, and Coupling Structures, Artech House, 1980, pp. 497.

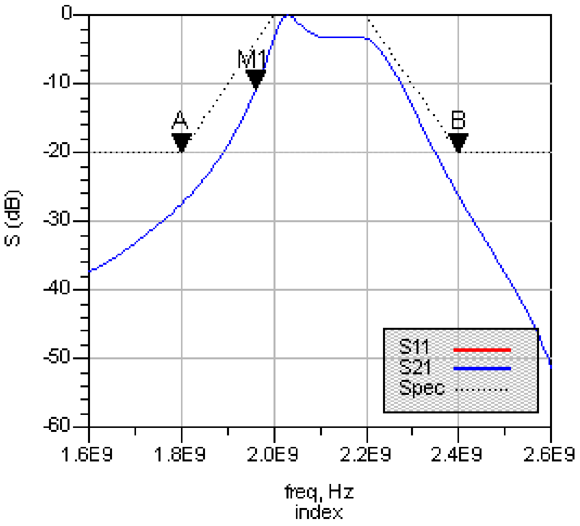
### Example

A comb-line filter was designed for a maximally flat response with the 3 dB passband edge frequencies at 2 GHz and 2.2 GHz respectively. The design required 3 coupled lines. Tuning using the Optimization Assistant yielded a value of Delta = -11.078 mil.


Metal-1
Dielectric-1
Metal-2
Metal-i : T[i], COND[i], TYPE[i]
Dielectric-i : ER[i], H[i], TAND[i]
1
2

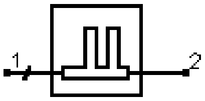
MLSUBSTRATE3
Subst1
Er[1]=4.5
H[1]=10 mil
Er[2]=4.5
H[2]=10 mil

DA_SCMFilter1_test
DA_SCMFilter1
Subst="Subst1"
Fs1=1.8 GHz
Fp1=2 GHz
Fp2=2.2 GHz
Fs2=2.4 GHz
Ap=3 dB
As=20 dB
N=0
ResponseType=Maximally Flat
Zo=50 Ohm
Lelec=0.1
ya=0.7
CouplingType=Tapped Line Transformer Input
Delta=-11.078 mil



# SDSMatch (Stripline Double-Stub Match)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Center frequency | GHz | 1 |
| Zin | Desired complex input impedance | Ohm | 50 |
| Zload | Complex load impedance to match | Ohm | 100 |
| Zstub1 | Characteristic impedance of stub line 1 (nearest input) | Ohm | 50 |
| Zstub2 | Characteristic impedance of stub line 2 (nearest output) | Ohm | 50 |
| Zline | Characteristic impedance of line between stubs | Ohm | 50 |
| Zfeed1 | Characteristic impedance of line connected to port 1 | Ohm | 50 |
| Zfeed2 | Characteristic impedance of line connected to port 2 | Ohm | 50 |
| Stub1Type | Stub 1 type (open or short circuit) | None | Open Circuit |
| Stub2Type | Stub 2 type (open or short circuit) | None | Open Circuit |
| Delta | Length added to stubs for tuning performance | mil | 0 |

### Notes

1. A double-stub matching network matches a complex load impedance (Zload) to a desired complex input impedance (Zin) using two shunt stubs and a connecting line.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. An impedance match can be realized using any combination of stub types, although some combinations can be more realizable.
4. Zfeed1 and Zfeed2 are used simply to ensure that the input and output legs of the microstrip tee components are of the proper width.
5. The length of the line between the stubs is nominally chosen to be an eighth of a line wavelength. However, if this will not realize the match, the length of line is computed

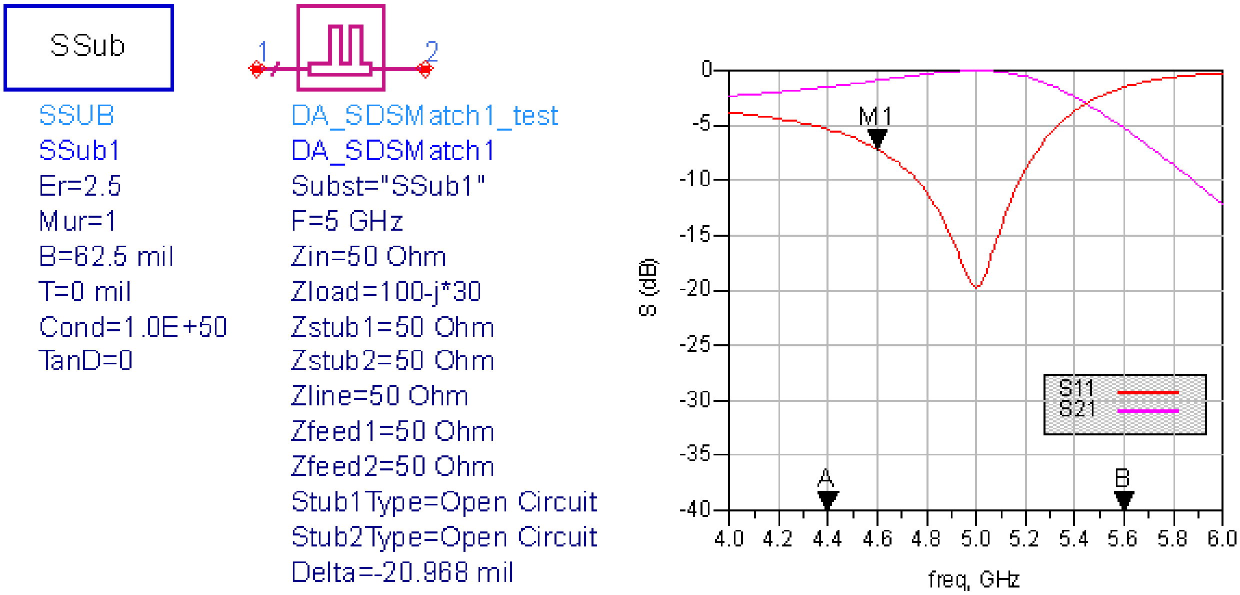
such that the rotated circle on the Smith chart encloses the load admittance.

6. STEE component width constraint violations will be avoided generally by choosing similar characteristic impedances for the line, stub, and feed.
7. Since two solutions are possible, the solution that results in the smallest length of stub 1 is chosen. For example, if stub 1 is open circuited, the solution for which stub 1 must realize a capacitive reactance is chosen.
8. The input port termination is set to the conjugate of Zin so that the ideal input reflection coefficient will be zero. The output port termination is set to Zload.
9. The optimization minimizes the value of S11 (referenced to the conjugate of Zin) at the design center frequency by changing the length of the stubs. Both stubs are tuned by the same length.
10. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 266-271.

### Example

A double-stub matching network was designed to match a load impedance of 100 - j30 Ohms to a 50 Ohm line at a center frequency of 5 GHz with open circuited stubs. Tuning using the Optimization Assistant yielded a value of Delta =-20.968 mil.

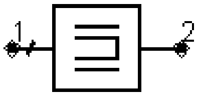


SSUB
SSub1
Er=2.5
Mur=1
B=62.5 mil
T=0 mil
Cond=1.0E+50
TanD=0

DA_SDSMatch1_test
DA_SDSMatch1
Subst="SSub1"
F=5 GHz
Zin=50 Ohm
Zload=100-j*30
Zstub1=50 Ohm
Zstub2=50 Ohm
Zline=50 Ohm
Zfeed1=50 Ohm
Zfeed2=50 Ohm
Stub1Type=Open Circuit
Stub2Type=Open Circuit
Delta=-20.968 mil

# SHPFilter (Stripline Hairpin Filter)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| Fs1 | Lower stopband edge frequency | GHz | 1.8 |
| Fp1 | Lower passband edge frequency | GHz | 2 |
| Fp2 | Upper passband edge frequency | GHz | 2.2 |
| Fs2 | Lower stopband edge frequency | GHz | 2.4 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| As | Stopband edge attenuation | dB | 20 |
| N | Number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | Type of frequency response (maximally flat or Chebyshev) | None | Maximally Flat |
| Zo | Desired input/output impedance | Ohm | 50 |
| Sphys | Physical spacing between legs within hairpin resonator; set to zero if Selec specified | mil | 0 |
| Selec | Spacing between legs within hairpin resonator in wavelengths; set to zero if Sphys specified | None | 0.25 |
| CouplingType | Type of input/output coupling (coupled line or tapped line transformer) | None | Coupled Line Transmission Input |
| Delta | Length added to coupled sections for tuning performance | mil | 0 |

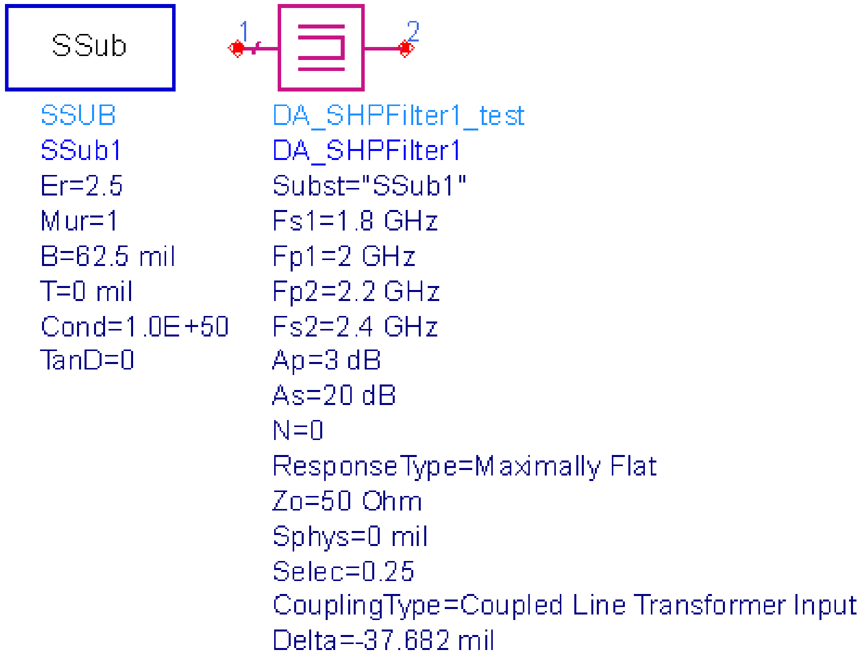
### Notes

1. A hairpin filter provides a bandpass frequency response between the input and output ports. N coupled-line sections produce an N-1 order filter response. Additional numbers of sections can be used to steepen the transition band roll off or widen the pass bandwidth.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. Because of the heavy computational burden in determining the line parameters, a brief delay will occur for the design.
4. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.
5. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
6. Using a Coupled Line Transformer Input CouplingType will use and extra coupled line section on the inputs and outputs to feed the device. Choosing Tapped Line Transformer Input will bring the feedline directly into the first resonator.
7. The optimization minimizes the absolute difference between S21 in dB and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequencies. Because only the line lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and spacings of the coupled filter sections.
8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, refer to: Cristal and Frankel, "Hairpin-line and hybrid hairpin-line/half-wave parallel-coupled-line filters," *IEEE Trans. Microwave Theory and Techniques,* vol MTT-20, pp. 719-728, 1972.

### Example

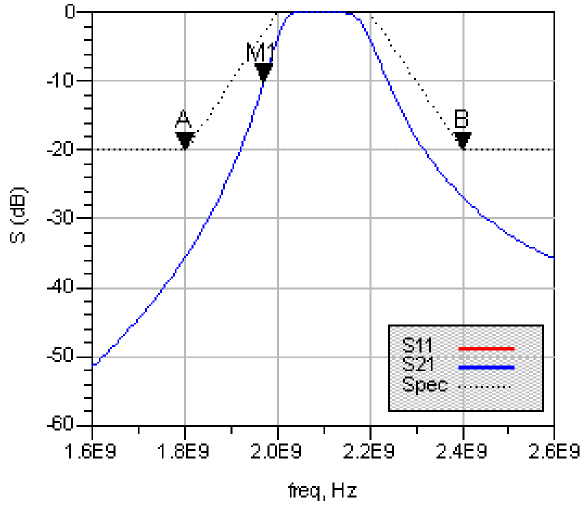
A hairpin filter was designed for a maximally flat response with the 3 dB passband edge frequencies at 2 GHz and 2.2 GHz respectively. Coupled line transformer inputs were used. The design required 4 coupled sections. Tuning using the Optimization Assistant yielded a value of Delta = -37.682 mil.



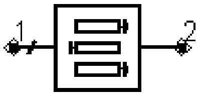
SSub

SSUB
SSub1
Er=2.5
Mur=1
B=62.5 mil
T=0 mil
Cond=1.0E+50
TanD=0

DA_SHPFilter1_test
DA_SHPFilter1
Subst="SSub1"
Fs1=1.8 GHz
Fp1=2 GHz
Fp2=2.2 GHz
Fs2=2.4 GHz
Ap=3 dB
As=20 dB
N=0
ResponseType=Maximally Flat
Zo=50 Ohm
Sphys=0 mil
Selec=0.25
CouplingType=Coupled Line Transformer Input
Delta=-37.682 mil



# SIDFilter (Stripline Interdigital Filter)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | Subst1 |
| Fs1 | Lower stopband edge frequency | GHz | 1.8 |
| Fp1 | Lower passband edge frequency | GHz | 2 |
| Fp2 | Upper passband edge frequency | GHz | 2.2 |
| Fs2 | Lower stopband edge frequency | GHz | 2.4 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| As | Stopband edge attenuation | dB | 20 |
| N | Number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | Type of frequency response (Maximally Flat or Chebyshev) | None | Maximally Flat |
| Zo | Desired input/output impedance | Ohm | 50 |
| ya | Normalized interior resonator admittance (0 < ya < 1) | None | 1 |
| CouplingType | Type of input/output coupling (coupled line or tapped line transformer) | None | Coupled Line Transformer Input |
| Delta | Length added to coupled sections for tuning performance | mil | 0 |

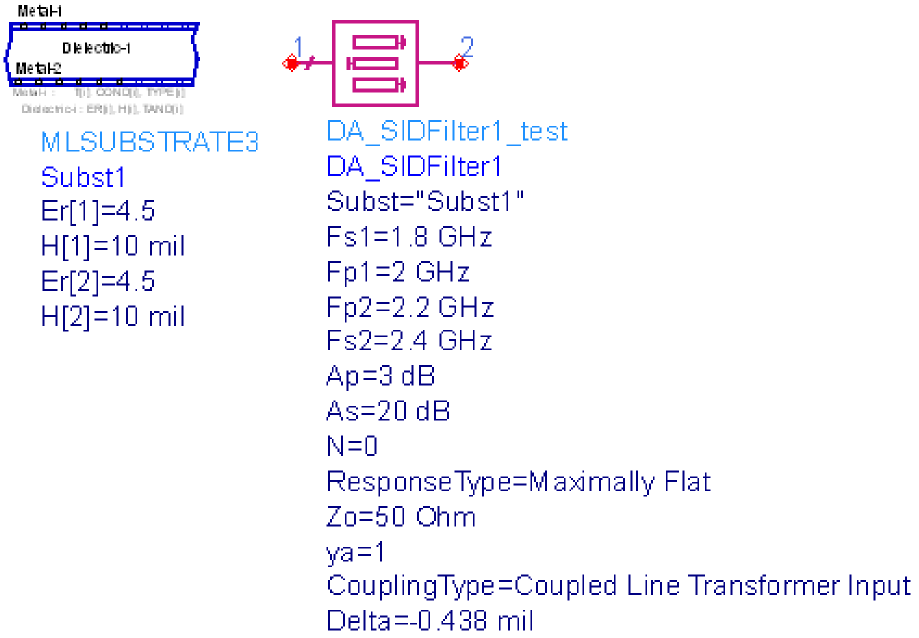
### Notes

1. An interdigital filter provides a bandpass frequency response between the input and output ports. N coupled-line sections produce an N-1 order filter response. Additional numbers of sections can be used to steepen the transition band roll off or widen the pass bandwidth.
2. Because of the heavy computational burden in determining the line parameters, a brief delay will occur for the design.
3. A three-layer substrate must be used for this topology.
4. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.
5. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
6. Using a CouplingType of "Coupled Line Transformer Input" will use an extra coupled line section on the inputs and outputs to feed the device. Choosing "Tapped Line Transformer Input" will bring the feedline directly into the first resonator.
7. The optimization minimizes the absolute difference between S21 in dB and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequencies. Because only the line lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and spacings of the coupled filter sections.
8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, refer to: Matthaei, Young and Jones, Microwave Filters, Impedance-Matching Networks, and Coupling Structures, Artech House, 1980, pp. 614.

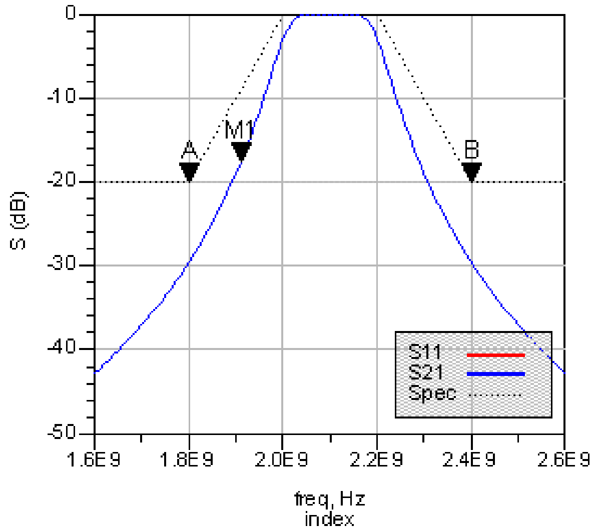
### Example

An interdigital filter was designed for a maximally flat response with the 3 dB passband edge frequencies at 2 GHz and 2.2 GHz respectively. The design required 5 coupled lines. Tuning using the Optimization Assistant yielded a value of Delta = -0.438 mil.


Metal-1
Dielectric-1
Metal-2
Metal-i : T[i], COND[i], TYPE[i]
Dielectric-i : ER[i], H[i], TAND[i]
1
2

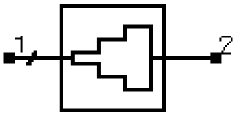
MLSUBSTRATE3
Subst1
Er[1]=4.5
H[1]=10 mil
Er[2]=4.5
H[2]=10 mil

DA_SIDFilter1_test
DA_SIDFilter1
Subst="Subst1"
Fs1=1.8 GHz
Fp1=2 GHz
Fp2=2.2 GHz
Fs2=2.4 GHz
Ap=3 dB
As=20 dB
N=0
ResponseType=Maximally Flat
Zo=50 Ohm
ya=1
CouplingType=Coupled Line Transformer Input
Delta=-0.438 mil



# SQWMatch (Stripline Quarter-Wave Match)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Center frequency | GHz | 1 |
| DeltaF | Total frequency bandwidth | GHz | 0.5 |
| Zo | Desired input impedance | Ohm | 50 |
| Rload | Load impedance to match | Ohm | 100 |
| ResponseType | Type of frequency response | None | Uniform |
| N | Number of quarter-wave sections; set N=0 to compute N | None | 0 |
| Rmax | Maximum voltage reflection coefficient | None | 0.1 |
| Delta | Length added to transformer sections for tuning performance | mil | 0 |

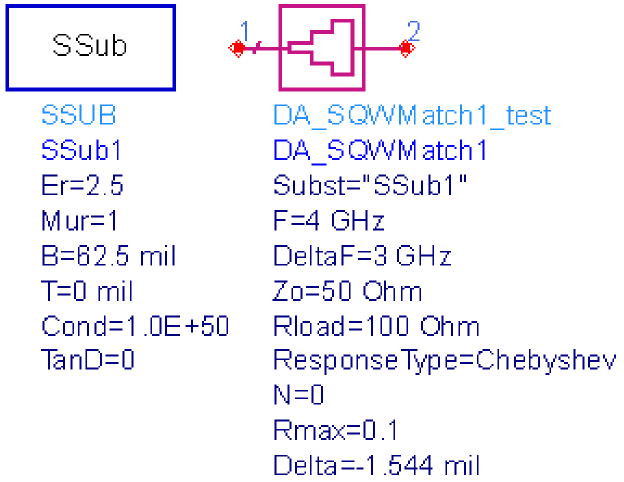
### Notes

1. A quarter-wave matching network matches a real load impedance (Rload) to a desired real input impedance (Zo) using multiple quarter wavelength sections. A specified frequency response can be realized by proper design of the individual sections.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. DeltaF is defined as the total bandwidth centered at the design center frequency.
4. If the number of sections N is set to zero, the Design Assistant chooses N such that the reflection coefficient is less that Rmax over the bandwidth DeltaF. The resulting bandwidth may be broader than that specified. Otherwise, Rmax and DeltaF are ignored.
5. The ResponseType specifies the distribution of the partial reflection coefficients seen at each section interface - Uniform, Binomial, and Chebyshev distributions are available. These in turn specify the shape of the reflection coefficient versus frequency.
6. A single-section quarter-wave matching transformer can be designed by setting N = 1.
7. Since the reflection coefficient may not be at its minimum value at the design center frequency, the optimization centers the variation of S11 (referenced to the value of Zo) versus frequency at the center frequency by changing the length of each quarter-wave section. All sections are tuned by the same length during the optimization.
8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 275-288; R. S. Elliott, An Introduction to Guided Waves and Microwave Circuits, Prentice Hall, John Wiley, New Jersey, 1993, pp. 218-224.

### Example

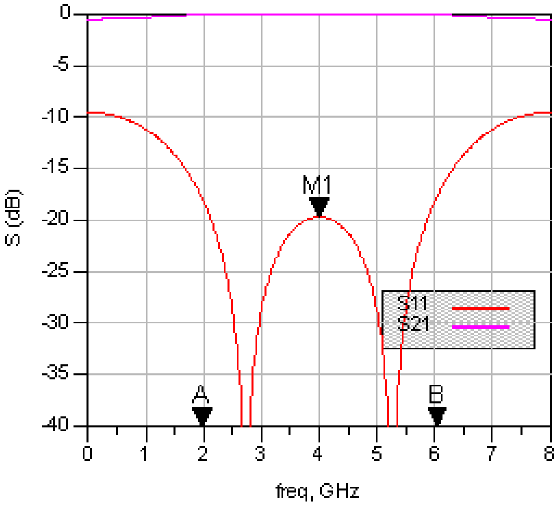
A quarter-wave matching network was designed to match a load impedance of 100 Ohms to a 50 Ohm line at a center frequency of 4 GHz. Specifying a Chebyshev frequency response for which the reflection coefficient remains below 0.1 over a 3 GHz bandwidth dictates 3 quarter-wave sections. Tuning using the Optimization Assistant yielded a value of Delta =-1.544 mil.



SSub

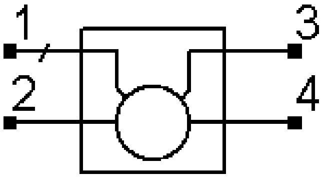
SSUB
SSub1
Er=2.5
Mur=1
B=62.5 mil
T=0 mil
Cond=1.0E+50
TanD=0

DA_SQWMatch1_test
DA_SQWMatch1
Subst="SSub1"
F=4 GHz
DeltaF=3 GHz
Zo=50 Ohm
Rload=100 Ohm
ResponseType=Chebyshev
N=0
Rmax=0.1
Delta=-1.544 mil



# SRRCoupler (Stripline Rat-Race Coupler)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Center frequency | GHz | 1 |
| Zo | Characteristic impedance | Ohm | 50 |
| Delta | Length added to ring branches for tuning | mil | 0 |

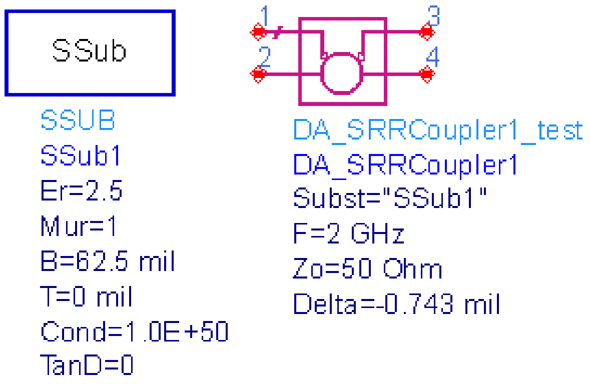
### Notes

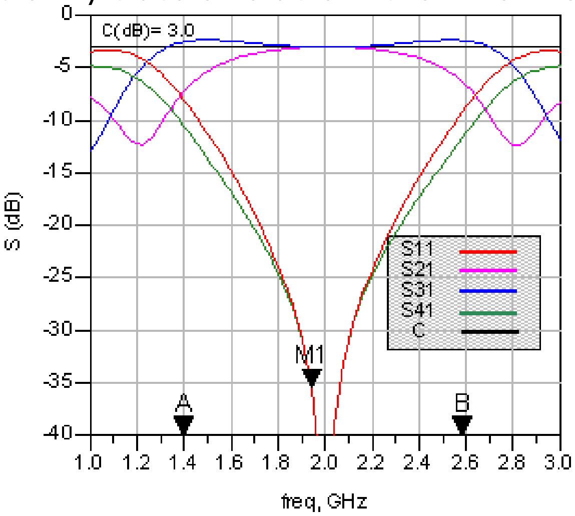
1. A rat-race coupler equally divides the power input at port 1 between ports 2 and 3. The signal at the output ports 2 and 3 are in-phase. Port 4 is isolated from port 1. If the signal is driven from port 2, then the power is divided between ports 1 and 4 with port 3 isolated. The signal at ports 1 and 4 are 180 degrees out of phase, and therefore this device is sometimes referred to as a 180-degree hybrid.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. The design specifies the width and length of the microstrip lines to ensure that the ports are matched to Zo and equal power split is achieved at the design center frequency.
4. The optimization minimizes the value of S11 (referenced to the value of Zo) at the design center frequency by changing the length of the ring.
5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 401-407.

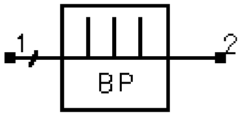
### Example

A rat-race coupler was designed for a 50 Ohm system impedance at a center frequency of 2 GHz. Tuning using the Optimization Assistant yielded a value of Delta = -0.743 mil.





# SSBFilter (Stripline Stub Bandpass Filter)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| Fs1 | Lower stopband edge frequency | GHz | 0.5 |
| Fp1 | Lower passband edge frequency | GHz | 1.3 |
| Fp2 | Upper passband edge frequency | GHz | 2.7 |
| Fs2 | Lower stopband edge frequency | GHz | 3.5 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 0.1 |
| As | Stopband edge attenuation | dB | 20 |
| N | Number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | Type of frequency response (maximally flat or Chebyshev) | None | Chebyshev |
| StubConfig | Configuration of interior shunt stubs | None | Two Parallel Stubs |
| StubType | Shunt Stub Type | None | Short Circuit Quarter Wave |
| Zo | Desired input/output impedance | Ohm | 50 |
| D | Impedance control parameter (0 < D < 1) | None | 1 |
| Finf | Frequency of infinite attenuation (for open circuit stub type) | GHz | 1.0 |
| Delta | Length added to stubs for tuning performance | mil | 0 |

### Notes

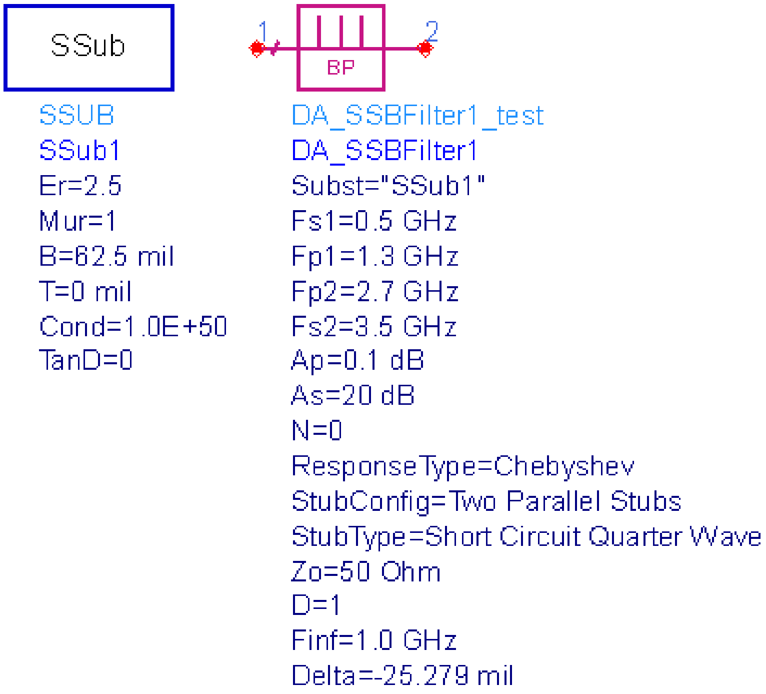
1. A stub bandpass filter provides a bandpass frequency response between the input and output ports. The design uses shunt stubs connected by lengths of transmission line. If the specified passband response is too narrow, large differences in impedance values can result in a non-realizable configuration.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.

3. This design is typically practical for fractional bandwidths of 0.4 to 0.7 or higher. If the bandwidth is too narrow, the design will generally require large differences in impedances between the stubs and the connecting lines, producing an unrealizable configuration.
4. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.
5. The parameter StubConfig specifies whether the interior stubs (all but those closest to the source and load) are implemented as a single stub or as two stubs in parallel. Choosing a single stub often produces narrower stub line widths, and therefore this parameter can impact the ability to manufacture.
6. The parameter D offers some control over the ratio between the stub impedances at the ends of the filter to those in the interior. In some cases where a SCROSS or STEE width constraint violation is encountered, decreasing (or increasing) this value in the range 0 < D < 1 can remedy the problem.
7. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
8. The optimization minimizes the absolute difference between S21 and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequencies. All stub lengths are tuned by the same amount. Because only the stub lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and lengths of all lines.
9. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: Matthaei, Young and Jones, Microwave Filters, Impedance-Matching Networks, and Coupling Structures, Artech House, 1980, pp. 595-608.

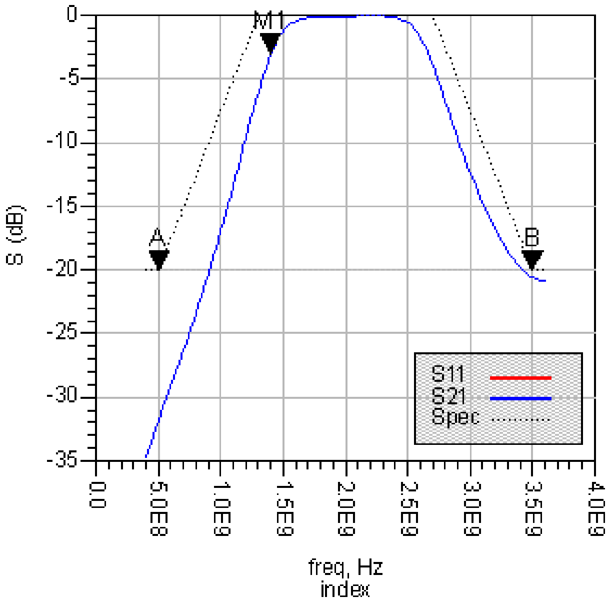
### Example

A stub bandpass filter was designed for a Chebyshev response with a 0.1 dB ripple. The passband edge frequencies are at 1.3GHz and 2.7 GHz respectively. The design uses two parallel stubs in the interior regions with D = 1. The design required 4 stubs. Tuning using the Optimization Assistant yielded a value of Delta = -25.279 mil.

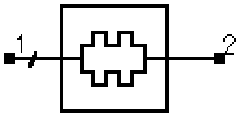


SSUB
SSub1
Er=2.5
Mur=1
B=62.5 mil
T=0 mil
Cond=1.0E+50
TanD=0

DA_SSBFilter1_test
DA_SSBFilter1
Subst="SSub1"
Fs1=0.5 GHz
Fp1=1.3 GHz
Fp2=2.7 GHz
Fs2=3.5 GHz
Ap=0.1 dB
As=20 dB
N=0
ResponseType=Chebyshev
StubConfig=Two Parallel Stubs
StubType=Short Circuit Quarter Wave
Zo=50 Ohm
D=1
Finf=1.0 GHz
Delta=-25.279 mil



# SSIFilter (Stripline Stepped Impedance Lowpass Filter)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| Fp | Frequency at passband edge | GHz | 1 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| Fs | Frequency at stopband edge | GHz | 1.2 |
| As | Stopband edge attenuation | dB | 20 |
| N | Number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | Type of frequency response (maximally flat or Chebyshev) | None | Maximally Flat |
| FElement | First filter component (automatic, capacitive, or inductive) | None | Automatic |
| Zo | Desired input/output impedance | Ohm | 50 |
| ZL | Characteristic impedance of low impedance sections | Ohm | 25 |
| ZH | Characteristic impedance of high impedance sections | Ohm | 100 |
| Delta | Length added to filter sections for tuning performance | mil | 0 |

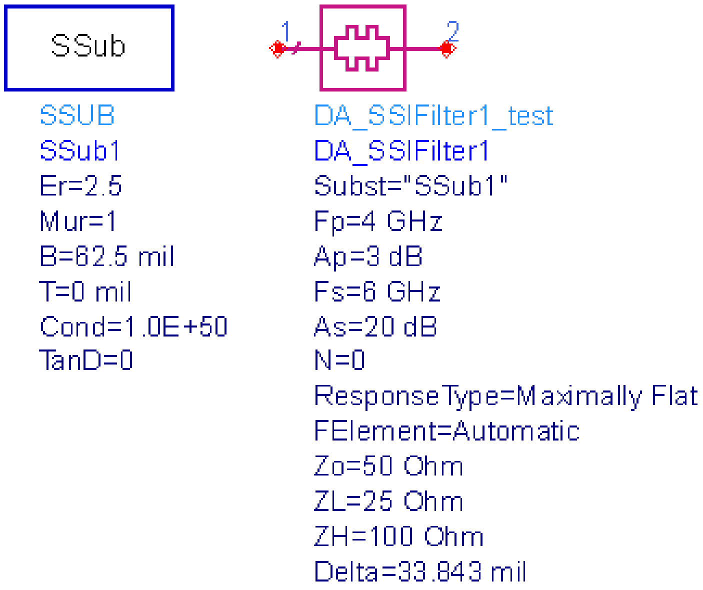
### Notes

1. A stepped impedance filter provides a lowpass frequency response between the input and output ports. The design is realized using alternating wide and narrow microstrip lines.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. The stepped impedance filter uses wide microstrip lines to approximate shunt capacitors and narrow lines to approximate series inductors in order to provide a lowpass frequency response.
4. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics. In order to obtain an impedance match, Chebyshev designs must use an odd number of components (N).
5. The parameter FElement specifies whether the first stub is inductive or capacitive. If Automatic is chosen, the first component is inductive if Zo/ZL > ZH/Zo and capacitive otherwise.
6. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
7. Because this filter design strategy is approximate, the resulting stopband attenuation can not satisfy the specification. Choosing more sections than that computed by the design can improve the stopband performance.
8. The optimization minimizes the absolute difference between S21 and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequency. All filter sections are tuned by the same amount.
9. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: Matthaei, Young and Jones, Microwave Filters, Impedance-Matching Networks, and Coupling Structures, Artech House, 1980, pp. 365-374.

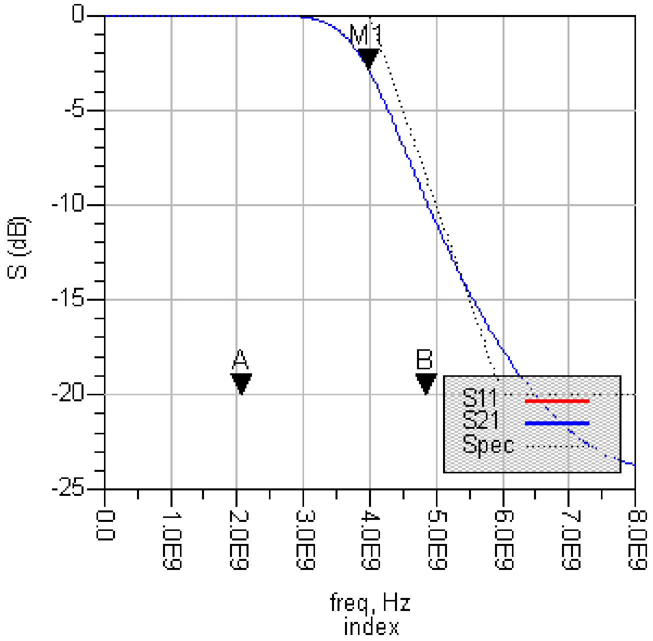
### Example

A stepped impedance lowpass filter was designed for a maximally flat response with a 3 dB attenuation at the passband edge frequency of 4 GHz. Choosing FElement as Automatic results in a capacitive first component. Tuning using the Optimization Assistant yielded a value of Delta = 38.843 mil.

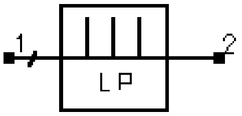


SSUB
SSub1
Er=2.5
Mur=1
B=62.5 mil
T=0 mil
Cond=1.0E+50
TanD=0

DA_SSIFilter1_test
DA_SSIFilter1
Subst="SSub1"
Fp=4 GHz
Ap=3 dB
Fs=6 GHz
As=20 dB
N=0
ResponseType=Maximally Flat
FElement=Automatic
Zo=50 Ohm
ZL=25 Ohm
ZH=100 Ohm
Delta=33.843 mil



# SSLFilter (Stripline Stub Lowpass Filter)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| Fp | Frequency at passband edge | GHz | 1 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| Fs | Frequency at stopband edge | GHz | 1.2 |
| As | Stopband edge attenuation | dB | 20 |
| N | Number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | Type of frequency response (maximally flat or Chebyshev) | None | Maximally Flat |
| StubType | Type of stubs (commensurate or variable length) | None | Variable Length Stub |
| FElement | First filter component (automatic, capacitive, or inductive) | None | Automatic |
| Zo | Desired input/output impedance | Ohm | 50 |
| ZS | Characteristic impedance of stubs | Ohm | 50 |
| ZH | Characteristic impedance of connecting sections | Ohm | 90 |
| Delta | Length added to stubs for tuning performance | mil | 0 |

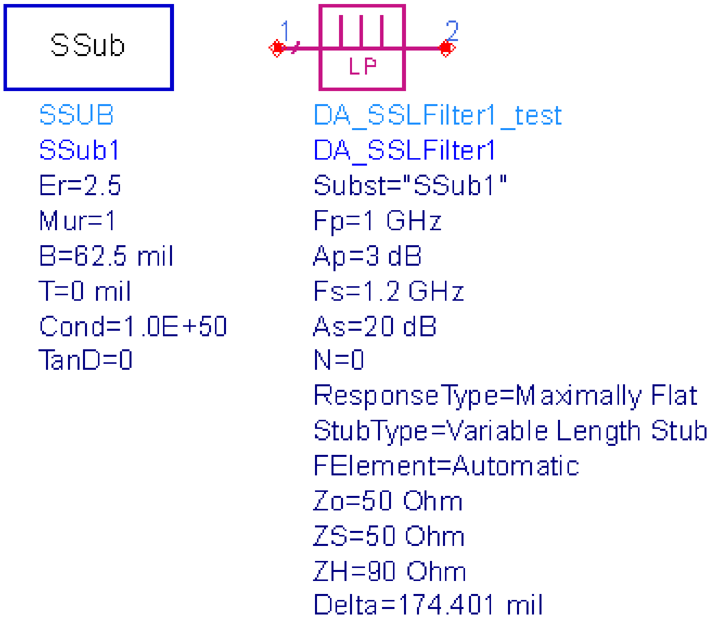
### Notes

1. A stub lowpass filter provides a lowpass frequency response between the input and output ports. The design is realized using narrow lines to approximate series inductances and shunt open circuited stubs to realize shunt capacitances.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. The stub lowpass filter uses narrow microstrip lines that approximate series inductors connecting open-circuited stubs that approximate shunt capacitors in order to provide a lowpass frequency response.
4. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics. In order to obtain an impedance match, Chebyshev designs must use an odd number of components (N).
5. If the parameter StubType is set as Commensurate, all stubs will have equal lengths, and the stub line width is computed. The value of ZS is then ignored, and the resulting design can be difficult to realize. If StubType is set as Variable Length Stub, the stub line widths are determined from ZS and the stub lengths are computed from the specifications.
6. The parameter FElement specifies whether the first stub is inductive or capacitive. If Automatic is chosen, the first component is set to be capacitive since this tends to offer improvements in manufacturing.
7. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency. Because this filter design strategy is approximate, the resulting stopband attenuation may not satisfy the specification. Choosing more sections than that computed by the Design Assistant can improve the stopband performance.
8. The optimization minimizes the absolute difference between S21 and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequency by changing the stub lengths. All stubs are tuned by the same amount.
9. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, refer to: Matthaei, Young and Jones, Microwave Filters, Impedance-Matching Networks, and Coupling Structures, Artech House, 1980, pp. 375.

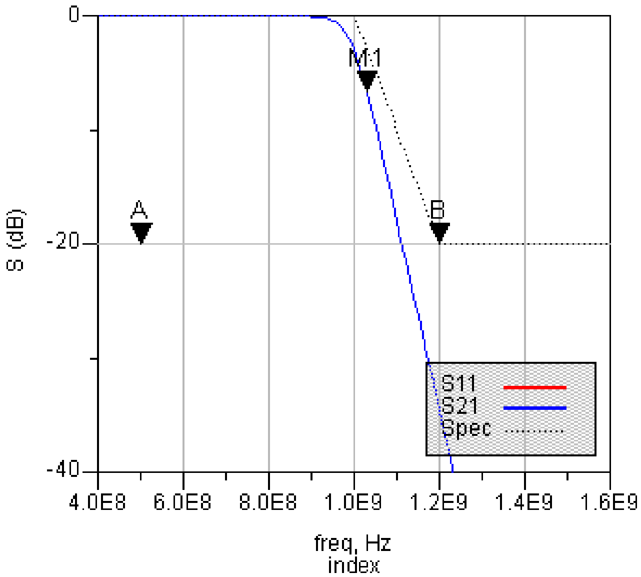
### Example

A stub lowpass filter was designed for a maximally flat response with a 3 dB attenuation at the passband edge frequency of 1 GHz. A variable length StubType and automatic FElement resulted in 13 components for the design. Tuning using the Optimization Assistant yielded a value of Delta = 174.401 mil.

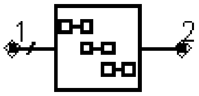


SSUB
SSub1
Er=2.5
Mur=1
B=62.5 mil
T=0 mil
Cond=1.0E+50
TanD=0

DA_SSLFilter1_test
DA_SSLFilter1
Subst="SSub1"
Fp=1 GHz
Ap=3 dB
Fs=1.2 GHz
As=20 dB
N=0
ResponseType=Maximally Flat
StubType=Variable Length Stub
FElement=Automatic
Zo=50 Ohm
ZS=50 Ohm
ZH=90 Ohm
Delta=174.401 mil



# SSRFilter (Stripline Stepped Impedance Resonator Filter)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| Fs1 | Lower stopband edge frequency | GHz | 1.8 |
| Fp1 | Lower passband edge frequency | GHz | 2 |
| Fp2 | Upper passband edge frequency | GHz | 2.2 |
| Fs2 | Lower stopband edge frequency | GHz | 2.4 |
| Ap | Passband edge attenuation (or ripple for Chebyshev) | dB | 3 |
| As | Stopband edge attenuation | dB | 20 |
| N | Number of filter sections (or 0 to compute N) | None | 0 |
| ResponseType | Type of frequency response (maximally flat or Chebyshev) | None | Maximally Flat |
| Zo | Desired input/output impedance | Ohm | 50 |
| Fsp | Center frequency of first spurious passband | GHz | 4.1 |
| Delta | Length added to filter sections for tuning performance | mil | 0 |

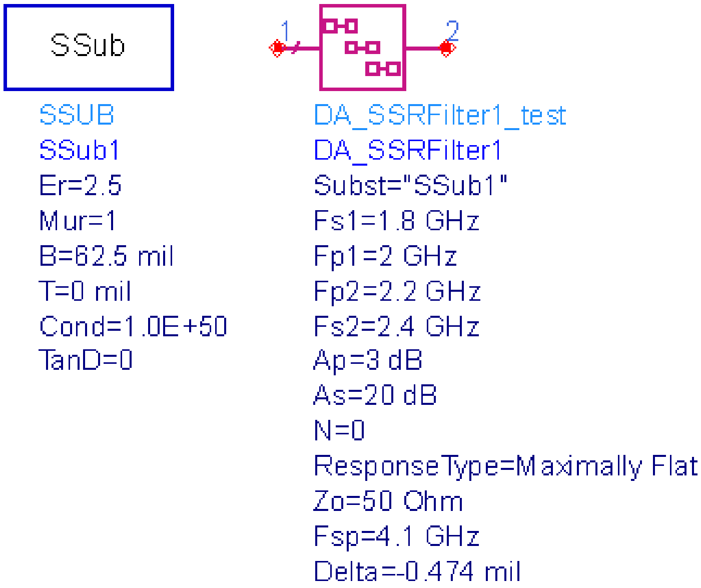
### Notes

1. A stepped impedance resonator filter provides a bandpass frequency response between the input and output ports. N coupled-line sections produce an N−1 order filter response. Additional numbers of sections can be used to steepen the transition band roll off or widen the pass bandwidth. The use on non-uniform impedance for each resonator moves the second pass band center frequency away from the second harmonic of the fundamental frequency.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. Because of the heavy computational burden in determining the line parameters, a brief delay will occur for the design.
4. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.
5. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
6. The center frequency of the first spurious passband should be on the order of twice the fundamental passband center frequency.
7. The optimization minimizes the absolute difference between S21 in dB and the specified passband edge attenuation (which equals the ripple for Chebyshev response) at the passband edge frequencies. Because only the line lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and spacings of the coupled filter sections.
8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, refer to: Makimoto and Yamashita, "Bandpass filters using parallel coupled stripline stepped impedance resonators," *IEEE Trans. Microwave Theory and Techniques* , vol MTT-28, pp. 1413-1417, 1980.

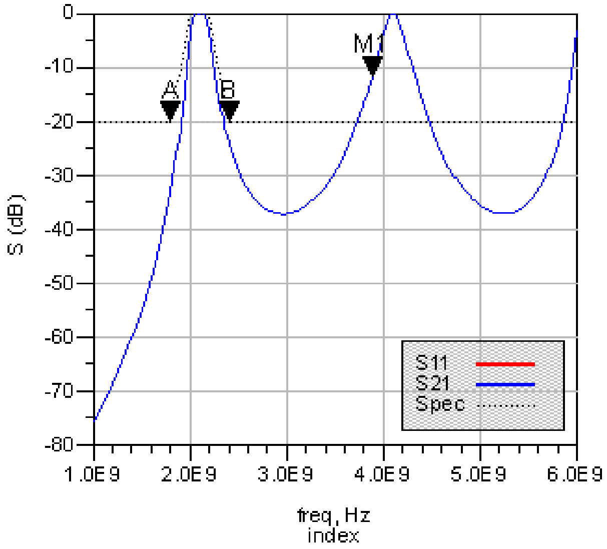
### Example

A stepped impedance resonator filter was designed for a maximally flat response with the 3 dB passband edge frequencies at 2 GHz and 2.2 GHz respectively. The second passband was set to 4.1 GHz. The design required 4 coupled lines. Tuning using the Optimization Assistant yielded a value of Delta = -0.474 mil.

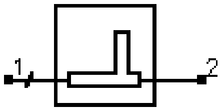


SSUB
SSub1
Er=2.5
Mur=1
B=62.5 mil
T=0 mil
Cond=1.0E+50
TanD=0

DA_SSRFilter1_test
DA_SSRFilter1
Subst="SSub1"
Fs1=1.8 GHz
Fp1=2 GHz
Fp2=2.2 GHz
Fs2=2.4 GHz
Ap=3 dB
As=20 dB
N=0
ResponseType=Maximally Flat
Zo=50 Ohm
Fsp=4.1 GHz
Delta=-0.474 mil



## SSSMatch (Stripline Single-Stub Match)



### Symbol

### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Stripline substrate name | None | SSub1 |
| F | Center frequency | GHz | 1 |
| Zin | Desired complex input impedance | Ohm | 50 |
| Zload | Complex load impedance to match | Ohm | 100 |
| Zstub | Characteristic impedance of stub line | Ohm | 50 |
| Zline | Characteristic impedance of line | Ohm | 50 |
| Zfeed | Characteristic impedance of line connected to port 1 or 2 | Ohm | 50 |
| StubType | Stub type (open or short circuit) | None | Open Circuit |
| NetType | Network type (source to load) | None | Automatic |
| Delta | Length added to stub for tuning performance | mil | 0 |

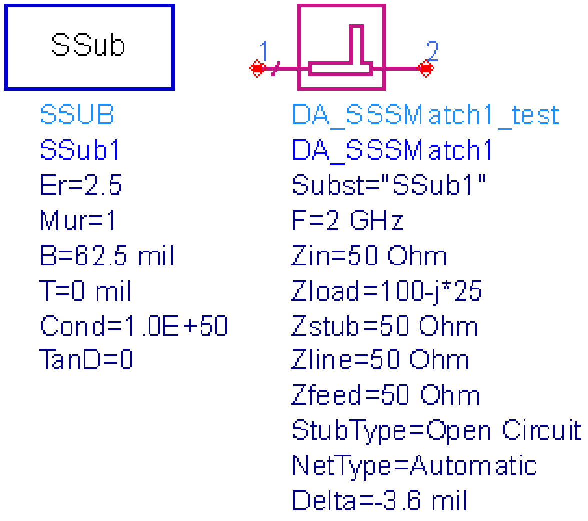
### Notes

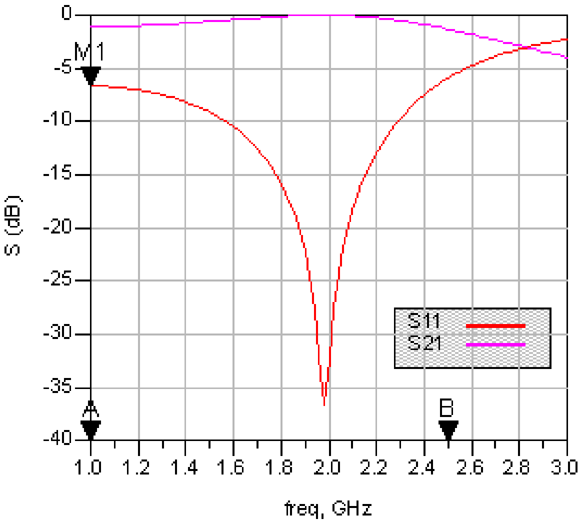
1. A single-stub matching network matches a complex load impedance (Zload) to a desired complex input impedance (Zin) using a single shunt stub and length of line.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. The parameter NetType can be Automatic, stub-line, or line-stub, with the latter two choices representing configurations that have a stub followed by a line or vice-versa as the network is observed from source to load. Many impedance combinations can be realized using both possible types, although some can only be realized using one of the two choices. Choosing Automatic will ensure a realizable choice is given.
4. An impedance match can be realized using either stub type.
5. Zline represents the impedance of the line either next to the source or load (depending on the value of NetType). Zfeed represents the desired impedance on the other side of the stub from the line and is used to ensure that the corresponding leg of the microstrip tee component is of the proper width.
6. STEE component width constraint violations will be avoided generally by choosing similar characteristic impedances for the line, stub, and feed.
7. The input port termination is set to the conjugate of Zin so that the ideal input reflection coefficient will be zero. The output port termination is set to Zload.
8. The optimization minimizes the value of S11 (referenced to the conjugate of Zin) at the design center frequency by changing the length of the stub.
9. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 258-266.

### Example

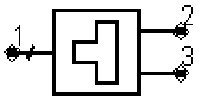
A single-stub matching network was designed to match a load impedance of 100 - j25 Ohms to a 50 Ohm line at a center frequency of 2GHz. Choosing an open-circuit stub with automatic selection of the NetType resulted in a stub-line configuration. Tuning using the Optimization Assistant yielded a value of Delta = -3.6 mil.





# STCoupler (Stripline Tee Power Divider)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Strip substrate name | None | SSub1 |
| F | Center frequency | GHz | 1 |
| Zo1 | Characteristic impedance of input port 1 | Ohm | 50 |
| Zo2 | Characteristic impedance of output port 2 | Ohm | 50 |
| Zo3 | Characteristic impedance of output port 3 | Ohm | 50 |
| K | Ratio of power out port 2 to power out port 3 | None | 1.0 |
| Delta | Length added to quarter-wave sections for tuning performance | mil | 0 |

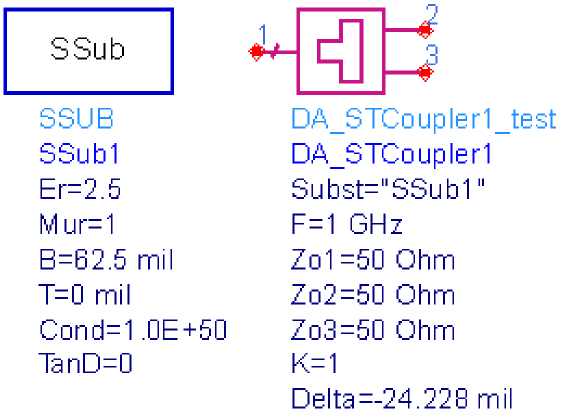
### Notes

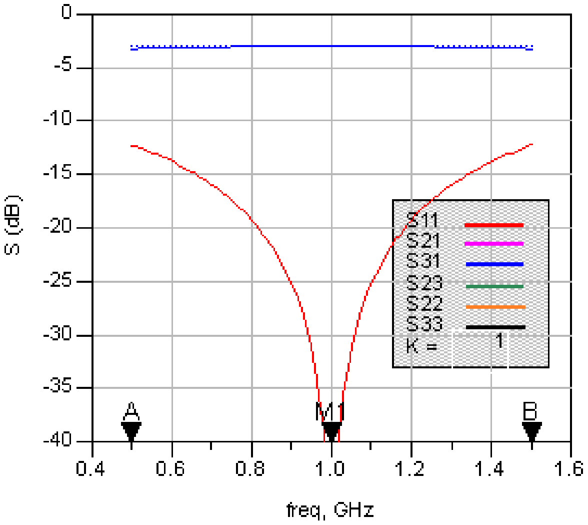
1. A tee power divider splits the power at the input (pin 1) between the two outputs (pins 2 and 3). Unequal or equal power splits can be realized. The input port will be matched to its feeding line, although in general the output ports will not be matched.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. The value of K can be set to realize the desired power split out of ports 2 and 3. However, choosing K larger than 3 to 4 (or smaller than 1/3 to 1/4) can cause the ratio of the widths of the tee branches to violate the range of the STEE simulation model. While the simulation will still proceed, the results can have some inaccuracies.
4. Quarter-wave matching sections are provided on the output ports to ensure a proper power split is achieved.
5. The optimization minimizes the input reflection coefficient (S11) at the design center frequency by changing the length of the quarter wave transformers on the output legs.
6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 360-361.

### Example

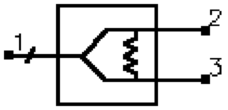
A tee power divider was designed for a center frequency of 1 GHz with an equal power split (K = 1). Tuning using the Optimization Assistant yielded a value of Delta = -24.228 mil.





## SWDCoupler (Stripline Wilkinson Divider)

### Symbol



### Parameters

| Name | Description | Unit | Default |
|---|---|---|---|
| Subst | Strip substrate name | None | SSub1 |
| F | Center frequency | GHz | 1 |
| DeltaF | Frequency bandwidth | GHz | 0.5 |
| Zo | Characteristic impedance | Ohm | 50 |
| ResponseType | Type of frequency response | None | Uniform |
| N | Number of quarter-wave sections; set N=0 to compute N | None | 0 |
| Rmax | Maximum voltage reflection coefficient on input port | None | 0.1 |
| K | Ratio of power out port 2 to power out port 3 | None | 1.0 |
| Wgap | Width of gap for resistor | mil | 50 |
| Delta | Length added to quarter-wave branches for tuning performance | mil | 0 |

### Notes

1. A Wilkinson power divider splits the power at the input (pin 1) between the two outputs (pins 2 and 3). Unequal or equal power splits can be realized. The signals at the outputs are in phase. All three ports will be matched, and ports 2 and 3 will in general be well isolated from each other.
2. The substrate may be either SSUB or SSUBO, although the schematic simulation accuracy for SSUBO depends on how the underlying ADS stripline models interpret the SSUBO definitions.
3. For broadband performance, the divider can have multiple quarter-wave sections. If

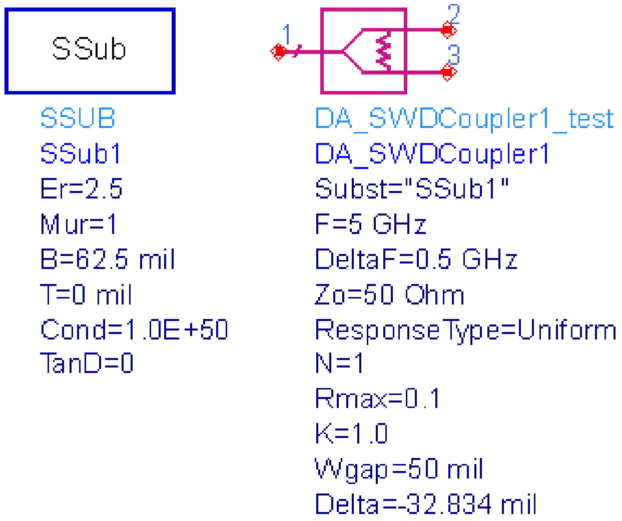
the number of sections N is set to zero, the Design Assistant chooses N such that such that the reflection coefficient is less than Rmax over the bandwidth DeltaF (centered at the design center frequency). the resulting bandwidth can be broader than that specified. Otherwise, Rmax and DeltaF are ignored.

4. ResponseType specifies the distribution of the partial reflection coefficients seen at each section interface - Uniform, Binomial, and Chebyshev distributions are available. These in turn specify the shape of the reflection coefficient versus frequency.
5. For a single section divider (N=1), the value of K can be set to realize the desired power split out of ports 2 and 3. Be aware that choosing K larger than 3 to 4 (or smaller than 1/3 to 1/4) is likely to cause difficulties in the design.
6. Pozar specifies $K^2$= P3/P2, while the DesignGuide uses $K^2$= P2/P3. Therefore, if you use the equations in Pozar to verify everything, you must substitute 1/K for K. The DesignGuide automatically puts quarter-wave matching sections on ports 2 and 3, so all ports are matched to the characteristic impedance. If you remove these matching segments, the output impedances are those specified by Pozar.
7. The optimization minimizes the input reflection coefficient (S11) at the design center frequency by changing the length of the quarter wave branches forming the divider.
8. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to *Design Assistant* (dgpas).

For a more detailed discussion of this device, see: D. M. Pozar, Microwave Engineering, 2nd Edition, John Wiley & Sons: New York, 1998, pp. 363-368.

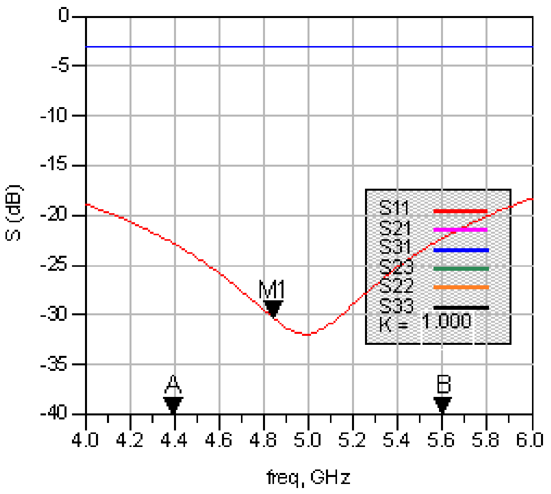
### Example

A single-section Wilkinson power divider (N=1) was designed for a center frequency of 5 GHz with an equal power split (K=1) and a gap width for the resistor of 50 mil. Tuning using the Optimization Assistant yielded a value of Delta = -32.834 mil.



SSUB
SSub1
Er=2.5
Mur=1
B=62.5 mil
T=0 mil
Cond=1.0E+50
TanD=0

DA_SWDCoupler1_test
DA_SWDCoupler1
Subst="SSub1"
F=5 GHz
DeltaF=0.5 GHz
Zo=50 Ohm
ResponseType=Uniform
N=1
Rmax=0.1
K=1.0
Wgap=50 mil
Delta=-32.834 mil


S (dB)
freq, GHz
M1
A
B
S11
S21
S31
S23
S22
S33
K = 1.000