Agilent Technologies

Passive Circuit DesignGuide

August 2005

Notice

The information contained in this document is subject to change without notice.

Agilent Technologies makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Agilent Technologies shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Warranty

A copy of the specific warranty terms that apply to this software product is available upon request from your Agilent Technologies representative.

Restricted Rights Legend

Use, duplication or disclosure by the U. S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 for DoD agencies, and subparagraphs (c) (1) and (c) (2) of the Commercial Computer Software Restricted Rights clause at FAR 52.227-19 for other agencies.

395 Page Mill Road, Palo Alto, CA 94304 U.S.A.

Acknowledgments

Contents

Chapter 1: Introducing the Passive Circuit DesignGuide

This chapter introduces the Passive Circuit DesignGuide. This manual assumes you have installed the DesignGuide with appropriate licensing codewords.

Getting Started

The Passive Circuit DesignGuide provides SmartComponents and automated-assistants for the design, simulation, optimization and performance analysis of common passive microstrip structures.

The DesignGuide includes SmartComponents for microstrip structures such as lines, couplers, power dividers, filters, and matching networks. All SmartComponents can be modified when selected. Simply select a SmartComponent and redesign or verify its performance.

Automated-assistants include a Design Assistant, Simulation Assistant, Optimization Assistant, and Display Assistant, which enable you to quickly create and verify a design.

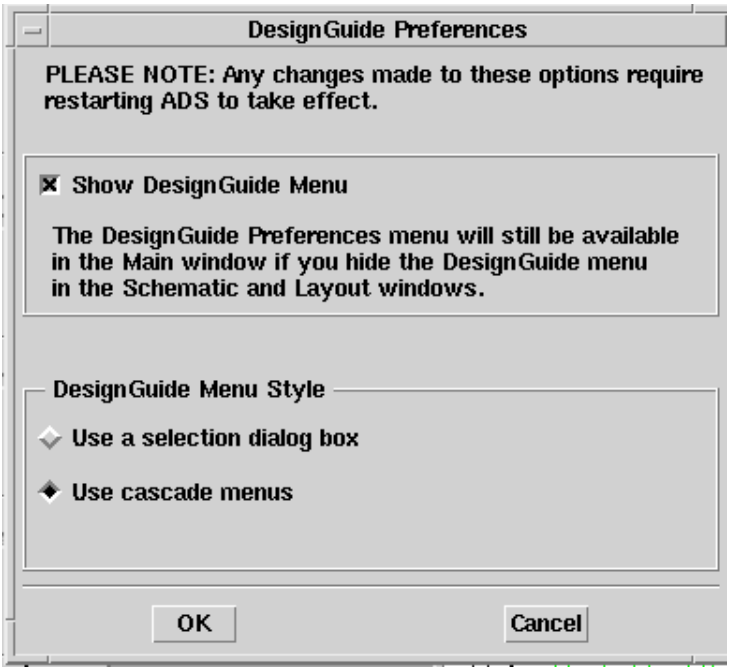
The complexity of Advanced Design System (ADS) is made easily accessible to the designer through the automated assistants. This enables a first-time or casual ADS user to begin benefiting from the capability of ADS quickly. Experienced ADS users will be able to perform tasks faster than ever before. As an example, a microstrip coupled-line filter can be designed, verified and a layout generated in a few minutes saving the designer substantial time.

Display Preferences

DesignGuides can be accessed in the Schematic window through either cascading menus or dialog boxes. You can configure your preferred method in the Main, Schematic, or Layout window. Choosing Preferences brings up a dialog box that enables you to:

Note Use the dialog box menu style on Windows systems because resource issues typically make the operating system unstable.

- Disable all DesignGuide menu commands except Preferences in the Main window and remove the DesignGuide menu in the Schematic and Layout windows.
- Select your preferred interface method (cascading menus vs. dialog boxes).



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

Passive Circuit Design Flow

The Passive Circuit DesignGuide follows standard design procedure:

1. Select a component needed for your design.
2. Provide specifications.
3. Design and analyze the component.
4. If the component performance needs adjustment, optimize the component.

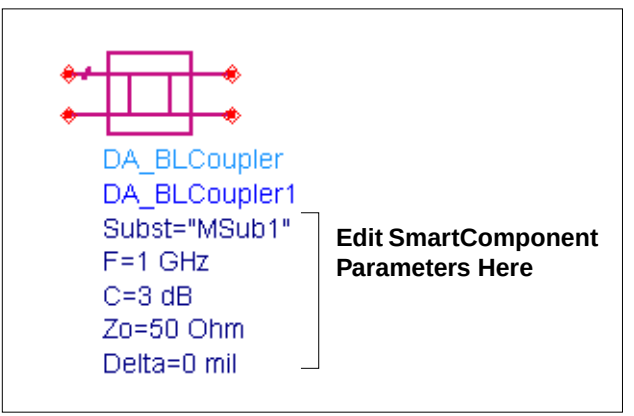
There are two important general concepts: SmartComponents and Automated Assistants.

SmartComponents

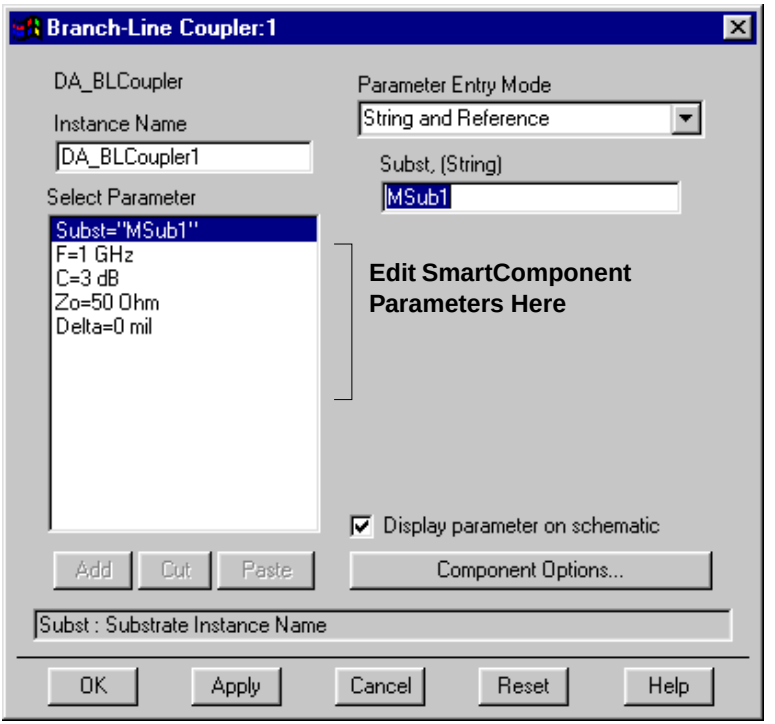
The DesignGuide provides a large number of passive SmartComponents such as couplers, filters, and matching networks. SmartComponents contain specification parameters and a schematic representation of the design.

SmartComponents are manipulated using several Automated Assistants. These assistants enable you to easily design, simulate, and optimize the SmartComponents.

SmartComponents are smart sub-network designs that can be placed into a schematic. The Branch-Line Coupler SmartComponent is shown here.



The components are placed in the schematic by selecting the desired SmartComponent from the palette and clicking at the point where you want them placed in the schematic. The desired specifications of the SmartComponent are entered by clicking on its parameters and changing them. In addition, a dialog box containing all parameters is available by double-clicking on the SmartComponent (as shown).



The SmartComponent design schematic can be viewed by pushing into the SmartComponent's subnetwork. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. For details on the SmartComponents, refer to Chapter 4, SmartComponent Reference.

Hint Place a branch-line coupler SmartComponent into a schematic by clicking the *BLCplr* palette button and clicking within the Schematic window at the desired placement location. Open the parameter dialog box by double-clicking the branch-line coupler component and edit its parameters.

Creating a design using SmartComponents

1. Choose and place a SmartComponent.
2. Edit the SmartComponent parameters (specifications).

3. Design the SmartComponent using the Design Assistant.
4. Analyze the SmartComponent's performance using the Simulation Assistant.
5. Display the performance of the SmartComponent using the Display Assistant.
6. If necessary, optimize the SmartComponent's performance using the Optimization Assistant.
7. Re-analyze the SmartComponent's performance using the Simulation Assistant.

Automated Assistants

The Passive Circuit DesignGuide provides four Automated Assistants for the simplified design, simulation, optimization, and analysis of SmartComponents. Each Automated Assistant has a tab that is accessed from DesignGuide Control Window.

Design Assistant is used to generate/update a SmartComponent's schematic design. After a SmartComponent is placed and the parameters are specified, you start the Design Assistant to design the component. Subsequently, if the parameters of the SmartComponent are modified, you start the Design Assistant again to update the design. For more information, refer to "Design Assistant" on page 3-1.

Simulation Assistant is used to automatically perform a simulation of a SmartComponent. After a SmartComponent has been designed using the Design Assistant, you start the Simulation Assistant to automatically analyze the component. You can easily examine the simulation results using the Display Assistant. For more information, refer to "Simulation Assistant" on page 3-2.

Optimization Assistant is used to automatically optimize a SmartComponent design so that the desired specifications are achieved. After a SmartComponent has been analyzed using the Simulation Assistant, you can start the Optimization Assistant to automatically optimize the component. After the Optimization Assistant has finished, you can rerun the Simulation Assistant to examine the optimized performance of the SmartComponent. For more information, refer to "Optimization Assistant" on page 3-4.

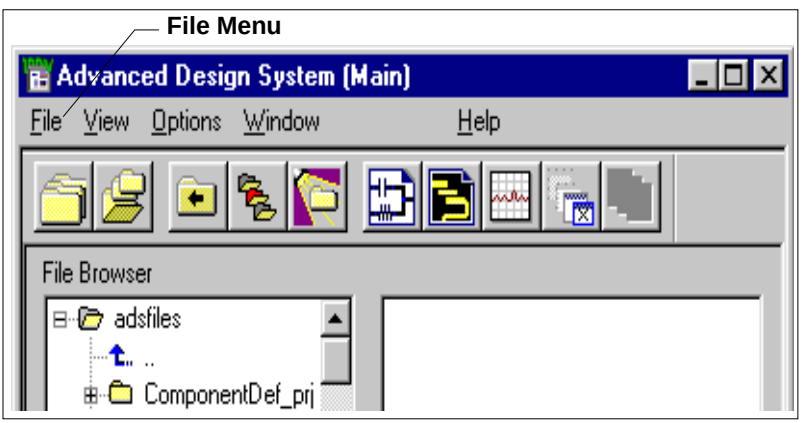
Display Assistant is used to automatically display the analysis results generated using the Simulation Assistant. By starting the Display Assistant, you can quickly display the results generated from the most recent simulation of a SmartComponent. For more detailed see "Display Assistant" on page 3-5.

Accessing the DesignGuide

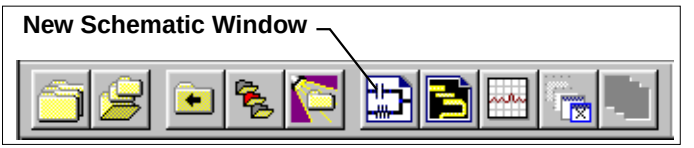
The Passive Circuit DesignGuide is accessed from a Schematic window within ADS.

1. Create or open a project.

   From the ADS main window, choose **File** > **New Project** or **File** > **Open Project**, as shown here. For this example, create a new project called *QuickStart*.



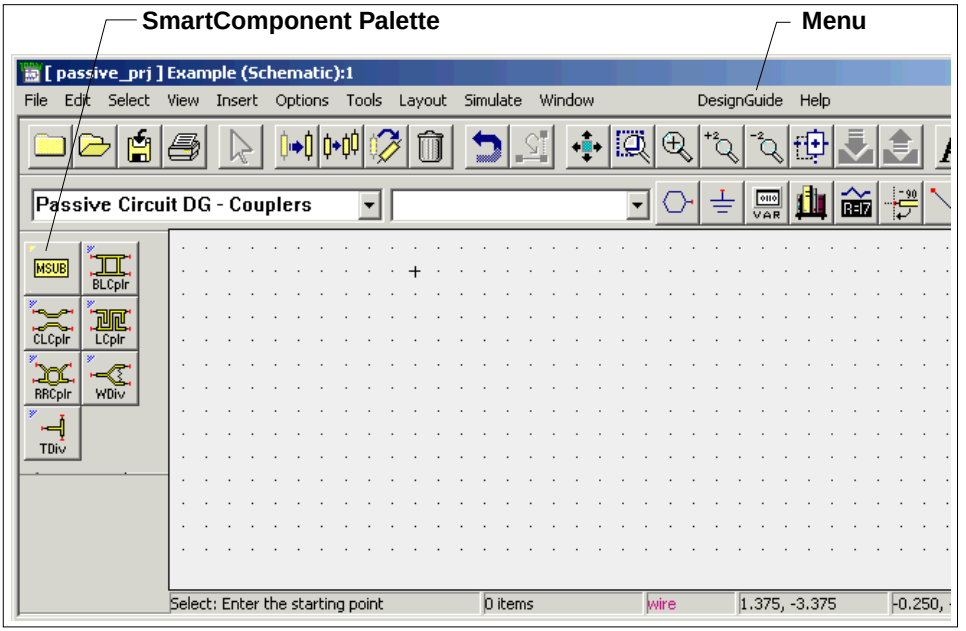
2. To open a Schematic window, choose **Window** > **New Schematic** or click the **New Schematic Window** toolbar button.



A new Schematic window appears, as shown here. The DesignGuide features are accessed using the menu, Control Window, and SmartComponent palettes.

To access the Passive Circuit DesignGuide features:

1. From the Schematic window, **DesignGuide** > **Passive Circuit**.
2. To access the Control Window, choose **Passive Circuit Control Window** from the *Passive Circuit* menu.
3. To access the documentation for the DesignGuide, choose either of the following:
   - **DesignGuide** > **Passive Circuit** > **Passive Circuit DesignGuide Documentation** (from ADS Schematic window)
   - **Help** > **Topics and Index** > **DesignGuides** > **Passive Circuit** (from any ADS program window)

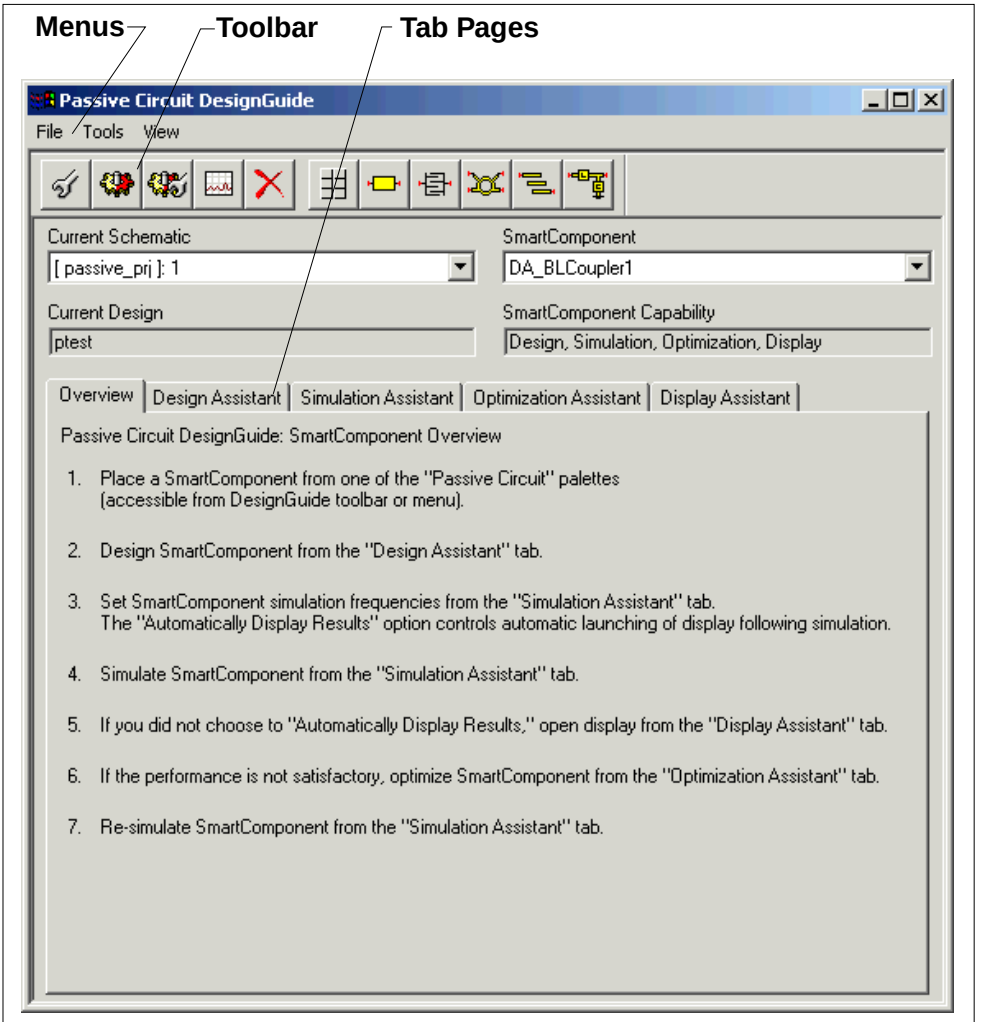


Note Depending on how your ADS preferences are set, a Schematic window may automatically appear when you create or open a project.

The Passive Circuit DesignGuide Control Window

All features are available from the Control Window including menus, a toolbar, and tab pages containing SmartComponent automated assistants.

Passive Circuit DesignGuide Control Window



The menus and toolbar buttons perform the basic functions for each Automated Assistant (Design, Simulate, Optimize, Display) as well as display the

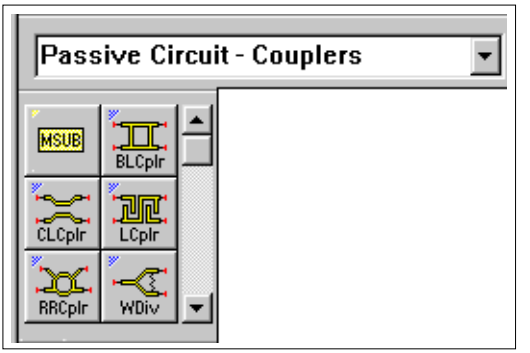
SmartComponent palettes. Full features are available from each of the tab pages on the window. Explore each Automated Assistant tab page by clicking on the tab at the top of each page. Explore the window menus as well to familiarize yourself with the basic DesignGuide capabilities.

The window includes the following features and functions:

- You can place the window anywhere on the screen.
- With the fields at the top of the Control Window, you can navigate multiple Schematic windows and SmartComponents.
  - The *Current Schematic* drop-down list box enables you to select any of the currently opened Schematic windows. This field is also updated when *Passive Circuit Control Window* is selected from the *Passive Circuit* menu.
  - The current design name is also displayed below the *Current Schematic*. The *SmartComponent* drop-down list box enables you to select any of the SmartComponents on the currently selected Schematic window.
  - The *SmartComponent Capability* field informs you of what functions (design, simulate, optimize, and display) the DesignGuide can perform for that particular component.
- To close the Control Window, choose **File** > **Exit DesignGuide** from the Control Window menu bar. The window may also be closed using the window close feature of the operating system (a button marked with an '**x**' at the top of the window).

SmartComponent Palette Access

The SmartComponent palettes are displayed by using the Control Window menus and toolbar. (They can also be chosen from the palette list box in the Schematic window toolbar.) Six palettes are available for accessing the SmartComponents. The *Passive Circuit* palette contains all of the passive SmartComponents. The other five palettes group the components by their functionality. A blue accent in the upper-left corner of a palette button indicates the component is a SmartComponent.



Using the Passive Circuit DesignGuide

This step-by-step example will take you through the design of a microstrip line, and the design, analysis and optimization of a branch-line coupler. After completing these examples, you should have a basic understanding of the DesignGuide.

Create a New Design

A new schematic design is needed to contain the microstrip line and branch-line coupler for the following exercises. Follow these simple steps to create a new design named Example.

1. Open a new Schematic window.
2. Choose **File** > **New Design** from the Schematic window to create a Analog/RF Network design named Example.

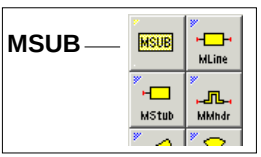
Open the DesignGuide Control Window

1. From the DesignGuide menu on the ADS Schematic window, choose **Passive Circuit**.
2. From the Passive Circuit window, choose **Microstrip Control Window** and click **OK**.

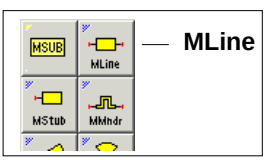
Auto-Design a Microstrip Line Component

A microstrip line can easily be designed given a substrate definition, its characteristic impedance, and length. Follow these simple steps to design a microstrip line.

1. Display the Passive Circuit - Lines palette. Refer to "SmartComponent Palette Access" on page 1-10.
2. Click the **MSUB** palette button, then click within the Schematic window at the desired placement location to place a microstrip substrate definition (MSUB) component.

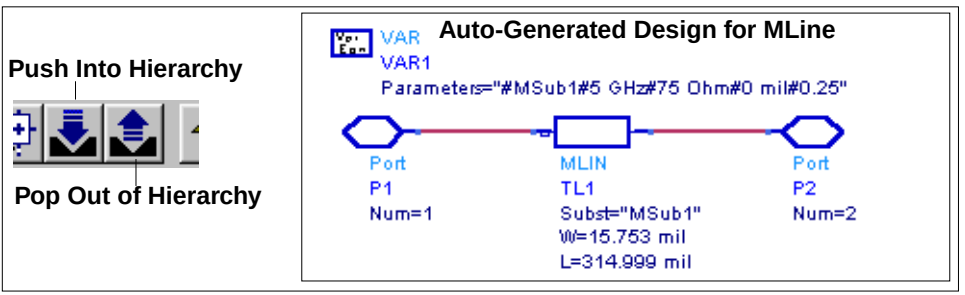


3. Double-click the MSUB component to open the component parameter dialog box and change the substrate thickness (H) to `20 mil` and the dielectric constant (Er) to `5`.
4. In the *Passive Circuit - Lines* palette, click the **MLine** palette button, then click within the Schematic window at the desired placement location to place a microstrip line.



5. Double-click the **MLine** component and change the center frequency (F) to `5 GHz`, the characteristic impedance (Zo) to `75 Ohm`, and the electrical length (Lelec) to `0.25` wavelengths.
6. Choose the **MLine** component either by clicking on it in the Schematic window or selecting it in the *SmartComponent* drop-down list box on the Control Window.
7. On the Control Window, click the **Design Assistant** tab, then click **Design** to generate the design for the SmartComponent.

8. Choose the component **MLine** and click the **Push Into Hierarchy** toolbar button to examine the designed SmartComponent.



9. After examining the design, pop out of the SmartComponent by clicking the **Pop Out of Hierarchy** toolbar button.

10. Choose **Tools** > **Delete SmartComponent** from the DesignGuide Control Window menu to delete the Mline SmartComponent.

Note This is different from the *Delete* button on the ADS Schematic window toolbar.

Design and Analyze a Branch-Line Coupler

A branch-line coupler can be designed as easily as a microstrip line. Follow these simple steps to design and analyze a branch-line coupler.

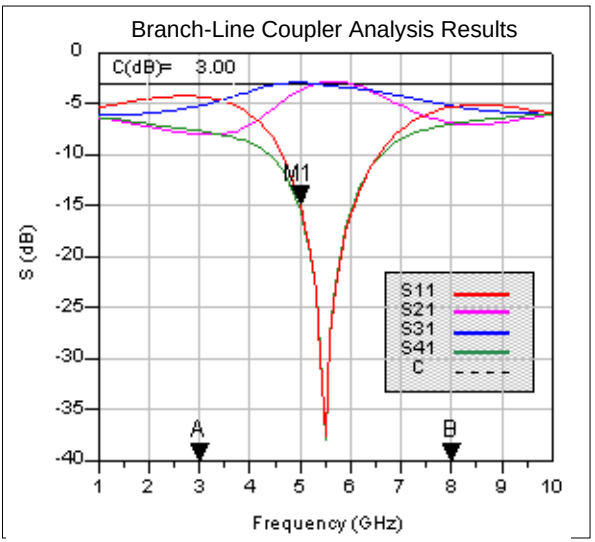
1. In the *Passive Circuit - Couplers* palette, click the **BLCplr** button and then click within the Schematic window at the desired placement location.

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.

Optimize the Branch-Line Coupler

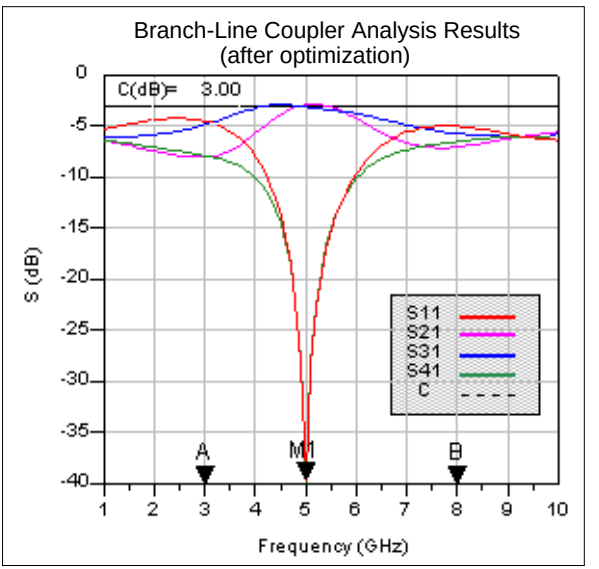
The branch-line coupler as designed in the preceding section has a center frequency of 5.5 GHz, which is different from the desired 5 GHz. The difference is due to limitations of the synthesis method used to generate the design. However, the Optimization Assistant can be used to easily optimize the design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

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.

Display 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 quickly displayed by following these simple steps.

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.

Chapter 2: Using SmartComponents

This chapter describes how to use SmartComponents to design your passive circuit.

Overview

SmartComponents are *smart* sub-network designs that can be placed into a schematic and provide the container for specification parameters and a schematic representation of the design.

This DesignGuide provides a large number of passive SmartComponents such as couplers, filters, lines and matching networks. Several automated-assistants enable you to easily design, simulate (analyze), and optimize the 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 here.

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.

There are two methods to display the desired SmartComponent palette:

Open the Passive Circuit DesignGuide Control Window by choosing **DesignGuide** > **Passive Circuit DesignGuide** > **Passive DesignGuide Control Window**. Display the desired SmartComponent palette by clicking one of the **Component Palette** buttons from the Control Window toolbar or by choosing **View** > **Component Palette - <Palette Name>** from the Control Window menu.

Choose the desired SmartComponent palette from the Component Palette drop-down list box in the Schematic window toolbar (directly above the palette).

Placing SmartComponents

To place a SmartComponent:

1. Click on the desired component button in a SmartComponent palette.
2. Click within the Schematic window at the location you want the SmartComponent placed.
3. You may change the orientation of the SmartComponent before placement by choosing from the **Insert** > **Component** > **Component Orientation** commands or by repeatedly clicking **Rotate by -90** from the schematic toolbar.
4. The place component mode will remain active until you choose **End Command** from the Schematic toolbar.

Note When a SmartComponent is initially placed, a temporary component is used to initially place and specify the parameters for the SmartComponent. This component *does not* contain a subnetwork design. After the Design Assistant 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 Design Assistant will need to be used again to update the subnetwork design file.

Copying SmartComponents

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

Copying Within A Design

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

Copying Between Designs or Schematic Windows

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

Note All copied SmartComponents will initially refer to the same SmartComponent design. When the Design Assistant is used to perform a design operation, it will transform each copied SmartComponent into a unique SmartComponent design. A design operation is accomplished by launching the Design Assistant from the DesignGuide Control Window.

Editing SmartComponents

A SmartComponent's position, orientation, and parameters can be edited like any other component in ADS.

Position and Orientation

A SmartComponent is moved by dragging it to any location in the Schematic window. It's orientation is changed by following these steps.

1. Choose **Edit** > **Advanced Rotate/Mirror** > **Rotate** from the Schematic window or click **Rotate Items** from the toolbar.
2. Click on the desired SmartComponent.
3. Rotate the component.
4. The rotate mode will remain active until you select the *End Command* from the toolbar.

Parameters

Parameters are changed by clicking on a SmartComponent parameter in the Schematic window and editing it or by double-clicking a component and editing the parameters in the component dialog box.

Deleting SmartComponents

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

Delete From Current Design

A SmartComponent can be deleted from a design by choosing the component and pressing the **Delete** key, clicking the **Delete** button on the toolbar, or by choosing **Edit** > **Delete** from the Schematic window. However, this does not remove the SmartComponent files from the project directory.

Delete From Current Project

To delete a SmartComponent and all associated files from your project, follow these steps.

1. From the DesignGuide Control Window, click the **Delete SmartComponent** button.
2. Click on the SmartComponent you want deleted. This will delete the SmartComponent from the current design and remove all of its files from your project.
3. The SmartComponent delete mode will remain active until you choose the *End Command* from the Schematic toolbar.

Delete Manually Using File System

You may use your computer's file system to delete a SmartComponent by deleting the appropriate files in the network subdirectory of a project. Delete files that start with *DA_, SA_' and OA_*, contain the SmartComponent title, and end with *.ael* or *.dsn*.

Design, Analysis, Optimization and Layout

The DesignGuide contains several automated assistants that provide automatic design, analysis, and optimization for the SmartComponents. The following assistants are available.

- **Design Assistant.** The Design Assistant is used to generate and update the design contained within a SmartComponent. It invokes a synthesis engine that generates a design from the given specification. It will design and update a single SmartComponent or all SmartComponents in a design. Refer to "Design Assistant" on page 3-1 for more information.
- **Simulation (Analysis) Assistant.** The Simulation Assistant is used to analyze the design contained within a SmartComponent. It creates a simulation circuit containing the SmartComponent, then performs a simulation. It can also automatically display the results of the simulation. Refer to "Simulation Assistant" on page 3-2 for more information.
- **Optimization Assistant.** The Optimization Assistant is used to optimize the design contained within a SmartComponent. It creates an optimization circuit containing the SmartComponent, performs an optimization, and updates the SmartComponent. Refer to "Optimization Assistant" on page 3-4 for more information.
- **Display Assistant.** The Display Assistant is used to 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. Refer to "Display Assistant" on page 3-5 for more information.
- **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" on page 3-9 for more information.

Stand-Alone SmartComponent Usage

Once SmartComponents are designed and tested, they can be used as stand-alone components. The Passive Circuit DesignGuide is not needed to use them in new designs unless you wish to modify or analyze them.

Using an Existing SmartComponent Within the Same Project

1. Open the Component Library window by choosinging **Insert** > **Component** > **Component Library** from the Schematic window or **Display Component Library List** from the toolbar.
2. Choose the project name under **All** > **Sub-networks** in the Libraries list at the left of the Component Library window. Available components will be listed in the Components list at the right of the Component Library window.
3. Choose the desired SmartComponent in the Components list.
4. Place the desired SmartComponent into your design by clicking in the Schematic window at the location you wish it placed. The insert mode will remain active until you click **End Command** on the toolbar.

Using an Existing SmartComponent in Any Project

A library of predesigned reusable SmartComponents can be easily created. This is done by placing the reusable SmartComponents in a project. This project can be included in any project and its SmartComponents will be accessed using the Component Library. Follow these steps.

1. Choose **File** > **Include/Remove Projects** from the main ADS window.
2. Choose the project that contains the desired SmartComponent from the File Browser at the left of the *Include & Remove* window.
3. Click the **Include** button to include the project in the hierarchy.
4. Click **OK**.
5. Open the Component Library window by choosing **Insert** > **Component** > **Component Library** from the Schematic window or **Display Component Library List** from the toolbar.
6. Choose the included project name under **All** > **Sub-networks** in the Libraries list at the left of the Component Library window. Available components will be listed in the Components list at the right of the Component Library window.
7. Choose the desired SmartComponent in the Components list.
8. Place the desired SmartComponent into your design by clicking in the Schematic window at the location you wish it placed. The insert mode will remain active until you select **End Command** from the toolbar.

Chapter 3: Using Automated Assistants

This chapter describes the four Automated Assistants used to design, simulate, optimize, and analyze SmartComponents, followed by instructions for creating layout artwork from the DesignGuide.

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.

Single Component Design

To design a single SmartComponent using the Control Window, 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 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.

Multiple Component Design

Clicking the **Design All** button on the Design Assistant tab designs all SmartComponents on the current Schematic.

Note To avoid screen flicker associated with the design, the Schematic window will disappear during the process.

Design progress is indicated on the tab page.

Simulation Assistant

The Simulation Assistant is used to analyze the design contained within a SmartComponent. It creates a simulation circuit around the SmartComponent, then performs a simulation. If desired it will automatically display 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. Basic simulation can also 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" on page 3-3.

Simulation Frequency Sweep

The simulation frequency sweep is specified on the Passive Circuit DesignGuide Control Window. If you are performing the simulation from the Control Window, click 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 will be 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 will be set for the component.

Automatically Display Results

If the Automatically Display Results box on the Control Window's Simulation Assistant tab is selected, the simulation results will be automatically displayed upon completion of the analysis.

To Simulate a SmartComponent

To simulate a SmartComponent using the Control Window, 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.

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, it is useful to simulate the SmartComponent manually.

To generate a simulation schematic around the selected SmartComponent, click the **Create Template** button on the Control Window Simulation Assistant tab.

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

When you are finished, clicking the **Update from Template** button on the Simulation Assistant tab will transfer any changes you have made to the SmartComponent on the Simulation schematic to the original SmartComponent and redesign if necessary.

You can also manually close the simulation schematic by choosing **File** > **Close Design** from the Schematic window menu, although this will result is 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. Basic optimization can also 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.

To Optimize 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 may 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** will cause the optimized values to appear in the *VAR* element in the schematic lower left corner for your inspection.

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

You may also 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 easily and quickly display the performance of a SmartComponent.

The Display Assistant is accessed using either 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 included display templates will 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 Chapter 4, SmartComponent Reference.

In some cases it may be useful to use one of the display templates provided with the DesignGuide for other applications.

To gain access to one of these templates, select the desired template from the Available Templates field and press the **Open Display Template** button on the Control Window Display Assistant tab.

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

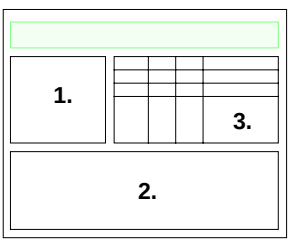
Basic Layout

Following is the basic layout of the display templates.

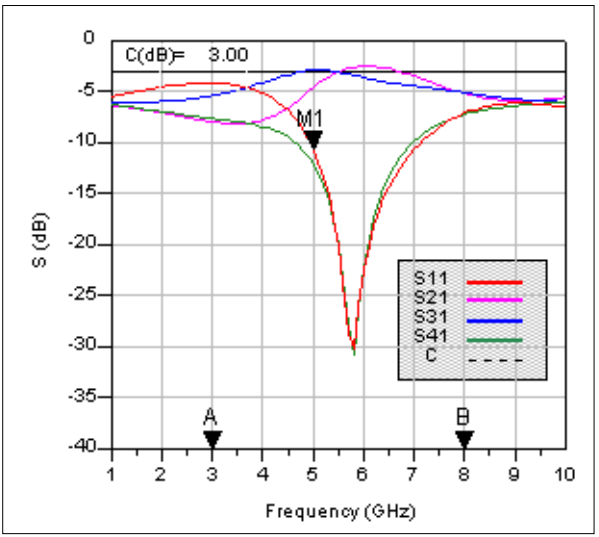
Area 1 of the display template contains a graph of the most important parameters of the SmartComponent.

Area 2 contains several graphs that give a comprehensive look at the component's performance.

Area 3 contains a table listing the basic specifications and performance of the component.



Typical Area 1 Content

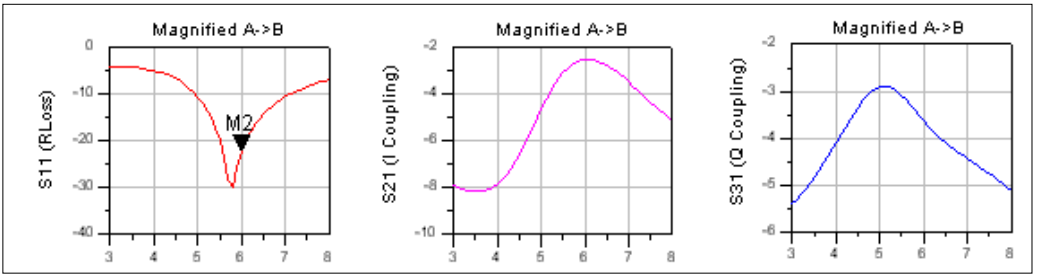


This is a typical Area 1 graph. The frequency range of the graph is determined by the Simulation Assistant. As you change the frequency range in the Simulation Assistant, this graph will update appropriately.

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 will be shown in the table in Area 3.

Typical Area 2 Content



Typical graphs from Area 2 are shown here. 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.

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

Typical Area 3 Content

	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.

A typical table from Area 3 is shown here. 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.

To Display SmartComponent Performance Results

Before using the Display Assistant, a valid dataset from a simulation of the selected SmartComponent must exist in the current project data directory. This simulation can be conveniently accomplished using the Simulation Assistant. Refer to "Simulation Assistant" on page 3-2 for details on this step.

To display results from a SmartComponent simulation using the Control Window, 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. 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 will be insensitive. If the toolbar or menu are used to try to display the results, a message will appear 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 desired SmartComponents in the schematic window.
2. Specify the desired 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 will be displayed in the Layout window. If the status report checkbox is selected in the Generate/Update Layout box, a layout generation status report will also be opened.

Updating Layout Artwork

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

1. Choose the desired SmartComponent in the schematic window.
2. Edit the desired 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 will be updated and displayed in the layout window.

Chapter 4: SmartComponent Reference

This chapter provides detailed information for all passive circuit SmartComponents.

SmartComponent List

Note A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to “Design Assistant” on page 3-1.

Passive Circuit DG - Lines

MBend (Microstrip Bend Component)

MBStub (Microstrip Butterfly Radial Stub)

MCFil (Microstrip Coupled-Line Filter Element)

MCLine (Microstrip Coupled Line Component)

MCorn (Microstrip Corner Component)

MCross (Microstrip Cross Component)

MCurve (Microstrip Curve Component)

MGap (Microstrip Gap Component)

MLine (Microstrip Line)

MMndr (Microstrip Meander Line)

MRStub (Microstrip Radial Stub)

MStep (Microstrip Step Component)

MStub (Microstrip Stub)

MTaper (Microstrip Taper Component)

MTee (Microstrip Tee Component)

Passive Circuit DG - RLC

MICapP (Microstrip 4-port Interdigital Capacitor)

MICapPG (Microstrip Grounded 2-port Interdigital Capacitor)

MICapS (Microstrip 2-port Interdigital Capacitor)

MICapSG (Microstrip 1-port Interdigital Capacitor)

MREInd (Microstrip Elevated Rectangular Inductor)

MRInd (Microstrip Rectangular Inductor)

MSInd (Microstrip Spiral Inductor)

MTFC (Microstrip Thin Film Capacitor)

TFC (Thin Film Capacitor)

TFR (Thin Film Resistor)

Passive Circuit DG - Couplers

BLCoupler (Branch-Line Coupler)

CLCoupler (Coupled-Line Coupler)

LCoupler (Lange Coupler)

RRCoupler (Rat-Race Coupler)

TCoupler (Tee Power Divider)

WDCoupler (Wilkinson Divider)

Passive Circuit DG - Filters

CLFilter (Coupled-Line Filter)

CMFilter (Comb-Line Filter)

HPFilter (Hairpin Filter)

IDFilter (Interdigital Filter)

SBFilter (Stub Bandpass Filter)

SIFilter (Stepped Impedance Lowpass Filter)

SLFilter (Stub Lowpass Filter)

SRFilter (Stepped Impedance Resonator Filter)

ZZFilter (Zig-Zag Coupled-Line Filter)

Passive Circuit DG - Matching

DSMatch (Double-Stub Match)

LEMatch (Lumped Component Match)

QWMatch (Quarter-Wave Match)

RAtten (Resistive Attenuator)

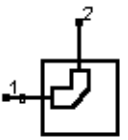
SSMatch (Single-Stub Match)

TLMatch (Tapered-Line Match)

Passive Circuit DG - Lines

MBend (Microstrip Bend Component)

Symbol



Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Zo = desired characteristic impedance, in ohms

BendType = type of bend

Angle = angle of bend (for arbitrary angle/miter bend)

M = miter fraction (for arbitrary angle/miter bend)

Notes

1. MBend designs a microstrip bend given the substrate, desired characteristic impedance, and bend properties. The design will realize 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 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” on page 3-1.

MBStub (Microstrip Butterfly Radial Stub)

Symbol

Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Xin = desired input reactance, in ohms (only one of Xin, Cin, and Lin can be non-zero)

Cin = desired input capacitance, in farads (only one of Xin, Cin, and Lin can be non-zero)

Lin = desired input inductance, in henries (only one of Xin, Cin, and Lin can be non-zero)

W = width of feed line (set to zero if Z specified)

Z = characteristic impedance of feed line (set to zero if W specified)

Angle = subtended angle of circular sector

d = insertion depth of circular sector in feed line

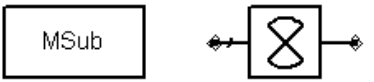
Delta = length added to stub for tuning performance

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 Microstrip Components 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” on page 3-1.

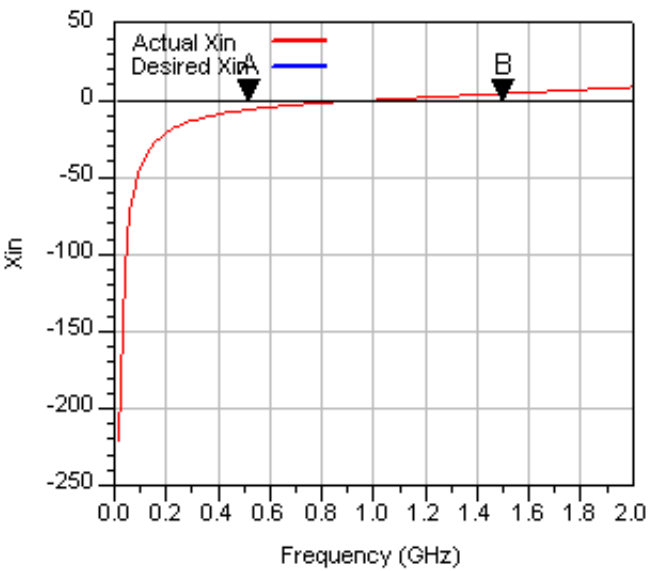
Example

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.



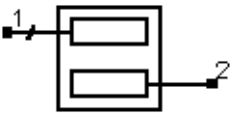
MSUB
MSub1
H=30 mil
Er=3.2

DA_MBStub1_ptest
DA_MBStub1
Subst="MSub1"
F=1 GHz
Xin=0 Ohm
Cin=0 pF
Lin=0 nH
W=0 mil
Z=50 Ohm
Angle=60
D=3 mil
Delta=86.786 mil



MCFil (Microstrip Coupled-Line Filter Element)

Symbol



Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Zoe = desired even-mode characteristic impedance, in ohms

Zoo = desired odd-mode characteristic impedance, in ohms

Zo1 = characteristic impedance of input line at port 1, in ohms

Zo2 = characteristic impedance of input line at port 2, in ohms

Lphys = physical line length (set to zero if Lelec specified)

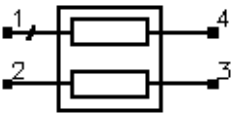
Lelec = line length in wavelengths (set to zero if Lphys specified)

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 Microstrip Components 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” on page 3-1.

MCLine (Microstrip Coupled Line Component)

Symbol



Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Zoe = desired even-mode characteristic impedance, in ohms

Zoo = desired odd-mode characteristic impedance, in ohms

Zo1 = characteristic impedance of input line at port 1, in ohms

Zo2 = characteristic impedance of input line at port 2, in ohms

Zo3 = characteristic impedance of input line at port 3, in ohms

Zo4 = characteristic impedance of input line at port 4, in ohms

Lphys = physical line length (set to zero if Lelec specified)

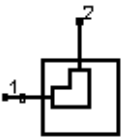
Lelec = line length in wavelengths (set to zero if Lphys specified)

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 MCFIL component in the Microstrip Components 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” on page 3-1.

MCorn (Microstrip Corner Component)

Symbol



Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

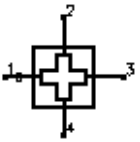
Zo = desired characteristic impedance, in ohms

Notes

1. MCorn designs a microstrip corner component given the substrate and characteristic impedance of the input and output lines.
2. **Note** A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to “Design Assistant” on page 3-1.
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 MCORN component in the Microstrip Components documentation for a more detailed description of the model used for this component.

MCross (Microstrip Cross Component)

Symbol



Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Z1 = characteristic impedance of port 1, in ohms

Z2 = characteristic impedance of port 2, in ohms

Z3 = characteristic impedance of port 3, in ohms

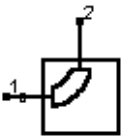
Z4 = characteristic impedance of port 4, in ohms

Notes

1. MCross designs a microstrip cross given the substrate, desired characteristic impedance on each port, and bend properties. The design will realize 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. Refer to the discussion of the MCROSS component in the Microstrip Components 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” on page 3-1.

MCurve (Microstrip Curve Component)

Symbol



Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Zo = desired characteristic impedance, in ohms

Angle = angle of curve

Radius = radius of curvature (set to zero if Lelec specified)

Lelec = curve length in wavelengths (set to zero if Radius specified)

CurveType = type of curve

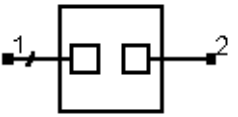
Nmode = number of modes (for Waveguide Model)

Notes

1. MBend designs a microstrip bend given the substrate, desired characteristic impedance, and bend properties. The design will realize 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 Microstrip Components 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” on page 3-1.

MGap (Microstrip Gap Component)

Symbol



Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Zo = desired characteristic impedance, in ohms

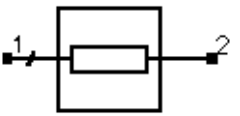
S = length of gap (spacing)

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 Microstrip Components 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” on page 3-1.

MLine (Microstrip Line)

Symbol



Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Zo = desired characteristic impedance, in ohms

Lphys = physical line length (set to zero if Lelec specified)

Lelec = line length in wavelengths (set to zero if Lphys specified)

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 Microstrip Components 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” on page 3-1.

MMndr (Microstrip Meander Line)

Symbol

Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Zo = desired characteristic impedance, in ohms

Lphys = physical line length (set to zero if Lelec specified)

Lelec = line length in wavelengths (set to zero if Lphys specified)

WR = bounding rectangle width, in meters

HR = bounding rectangle height, in meters

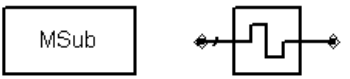
Delta = length added to vertical lines for tuning performance

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 will be at the center of the rectangle on the side characterized by HR.
3. The final width and height of the bounding box may 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 Microstrip Components 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” on page 3-1.

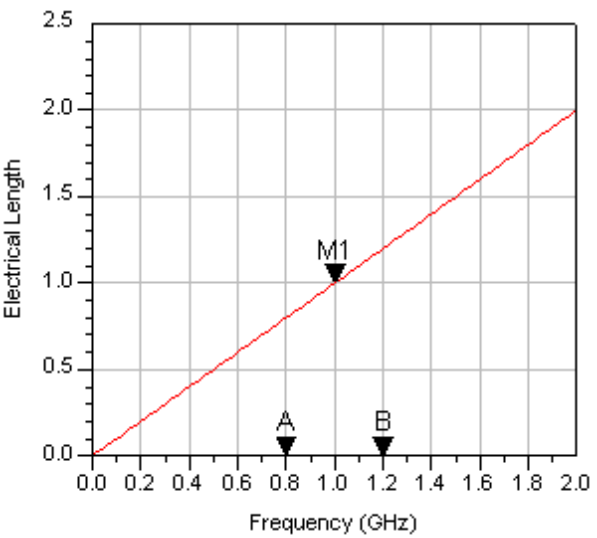
Example

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



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)

Symbol

Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Xin = desired input reactance, in ohms (only one of Xin, Cin, and Lin can be non-zero)

Cin = desired input capacitance, in farads (only one of Xin, Cin, and Lin can be non-zero)

Lin = desired input inductance, in henries (only one of Xin, Cin, and Lin can be non-zero)

W = width of feed line (set to zero if Z specified)

Z = characteristic impedance of feed line (set to zero if W specified)

Angle = subtended angle of circular sector

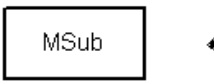
Delta = length added to stub for tuning performance

Notes

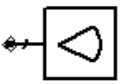
1. MRStub designs a microstrip radial stub given the substrate, desired input reactance, and stub dimensions.
2. The stub is designed by dividing the radial line into several short segments.
3. For proper operation, only one of Xin, Cin, and Lin can be non-zero. If all are zero, the stub is designed to provide an open circuit.
4. Refer to the discussion of the MRSTUB component in the Microstrip Components 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” on page 3-1.

Example

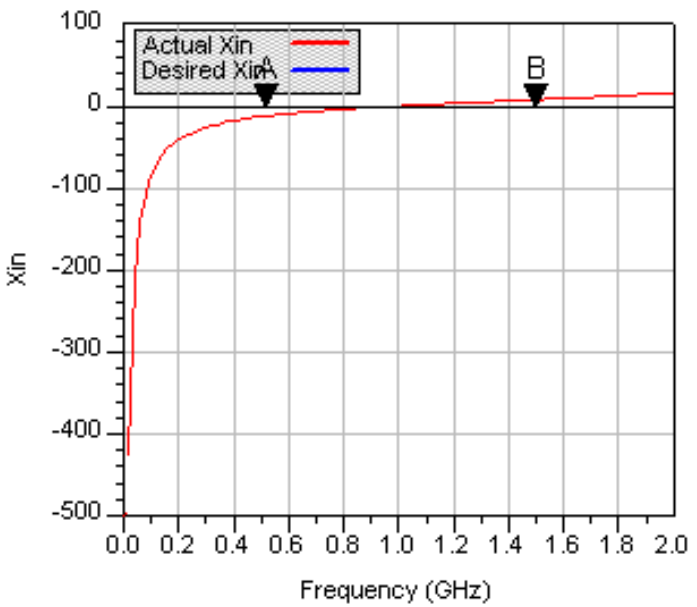
A MRStub component was used to design an open circuit stub at a center frequency of 1 GHz. Optimization yielded a value of Delta = -40.092 mil.



MSUB
MSub1
H=30 mil
Er=3.2

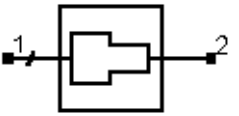


DA_MRStub1_ptest
DA_MRStub1
Subst="MSub1"
F=1 GHz
Xin=0 Ohm
Cin=0 pF
Lin=0 nH
W=0 mil
Z=50 Ohm
Angle=60
Delta=-40.092 mil



MStep (Microstrip Step Component)

Symbol



Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Z1 = characteristic impedance of port 1, in ohms

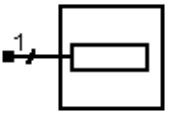
Z2 = characteristic impedance of port 2, in ohms

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 Microstrip Components 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” on page 3-1.

MStub (Microstrip Stub)

Symbol



Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Zo = desired characteristic impedance, in ohms

Lphys = physical line length

Lelec = line length in wavelengths

Xin = desired input reactance, in ohms

Cin = desired input capacitance, in farads

Lin = desired input inductance, in henries

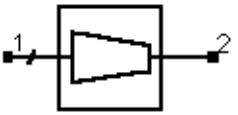
StubType = type of stub

Notes

1. MStub designs a microstrip open or short circuited stub given the substrate, desired characteristic impedance, and physical or electrical length. The design will realize 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 Microstrip Components 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” on page 3-1.

MTaper (Microstrip Taper Component)

Symbol



Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Z1 = characteristic impedance at port 1, in ohms

Z2 = characteristic impedance at port 2, in ohms

Lphys = physical line length (set to zero if Lelec specified)

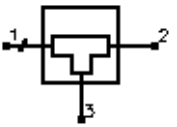
Lelec = line length in wavelengths (set to zero if Lphys specified)

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 Microstrip Components 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” on page 3-1.

MTee (Microstrip Tee Component)

Symbol



Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Z1 = characteristic impedance of port 1, in ohms

Z2 = characteristic impedance of port 2, in ohms

Z3 = characteristic impedance of port 3, in ohms

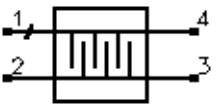
Notes

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 Microstrip Components 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” on page 3-1.

Passive Circuit DG - RLC

MICapP (Microstrip 4-port Interdigital Capacitor)

Symbol



Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

C = capacitance, in farads

W = width of fingers

G = gap between fingers

Ge = gap at end of fingers

Np = number of finger pairs

Wt = width of interconnect (0 if Zt specified)

Zt = characteristic impedance of interconnect lines, in ohms (0 if Wt specified)

Delta = length added to fingers for tuning performance

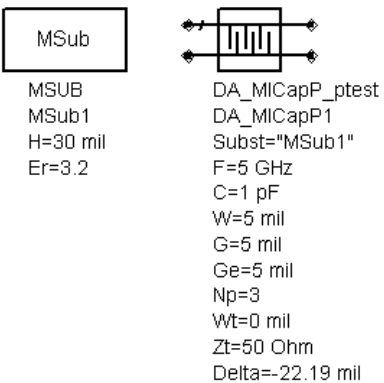
Notes

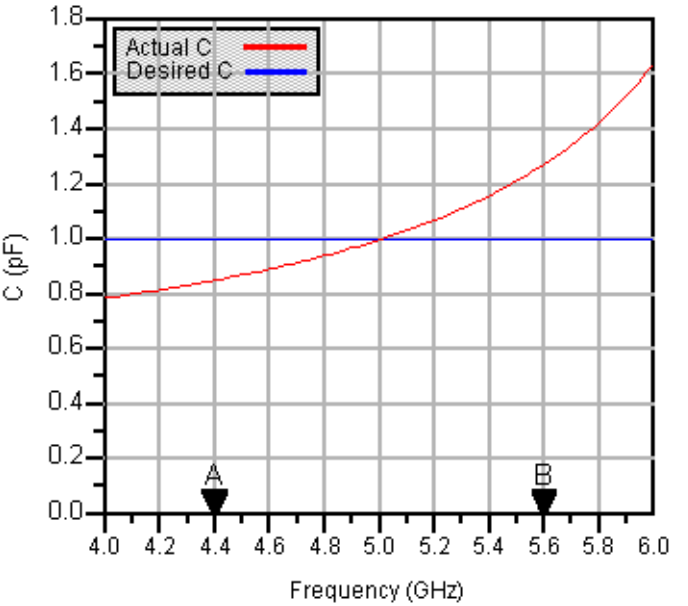
1. MICapP designs a capacitance between two adjacent microstrip lines using interdigital fingers. The underlying design uses the MICAP2 component contained in the Tlines-Microstrip palette.
2. The design is accomplished using a simple model that specifies the length required to achieve the capacitance 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 Microstrip Components 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" on page 3-1.

Example

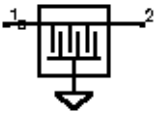
A MICapP component was used to design a 1 pF capacitance between two 50 ohm lines at a center frequency of 5 GHz. Optimization yielded a value of Delta = -22.19 mil.




Actual C
Desired C
C (pF)
1.8
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
A
B
4.0
4.2
4.4
4.6
4.8
5.0
5.2
5.4
5.6
5.8
6.0
Frequency (GHz)

MICapPG (Microstrip Grounded 2-port Interdigital Capacitor)

Symbol



Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

C = capacitance, in farads

W = width of fingers

G = gap between fingers

Ge = gap at end of fingers

Wt = width of interconnect (0 if Zt specified)

Zt = characteristic impedance of interconnect lines, in ohms (0 if Wt specified)

Delta = length added to fingers for tuning performance

Notes

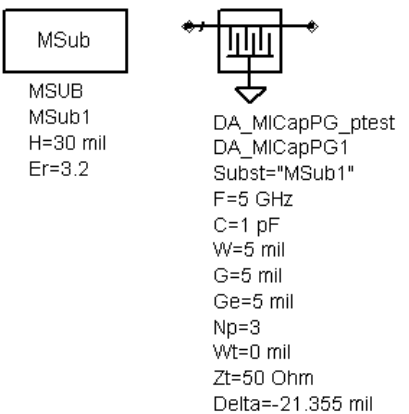
1. MICapPG designs a capacitance between a microstrip line and ground using interdigital fingers. The underlying design uses the MICAP4 component contained in the Tlines-Microstrip palette.
2. The design is accomplished using a simple model that specifies the length required to achieve the capacitance 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 Microstrip Components 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

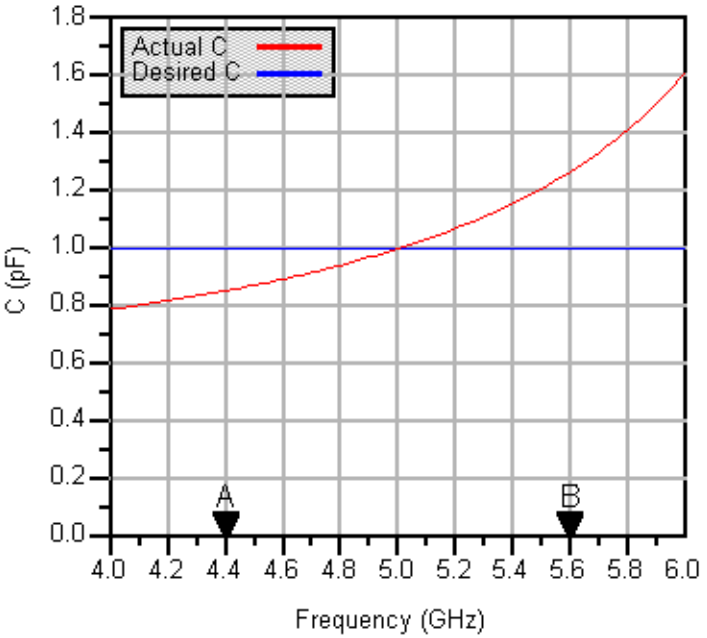
tune the design within the Simulation Assistant and subsequently use the optimizer to perform the fine tuning.

6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to “Design Assistant” on page 3-1.

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.




Actual C
Desired C
C (pF)
1.8
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
A
B
4.0
4.2
4.4
4.6
4.8
5.0
5.2
5.4
5.6
5.8
6.0
Frequency (GHz)

MICapS (Microstrip 2-port Interdigital Capacitor)

Symbol

Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

C = capacitance, in farads

W = width of fingers

G = gap between fingers

Ge = gap at end of fingers

Np = number of finger pairs

Wf = width of feed line (0 if Zf specified)

Zf = characteristic impedance of feed line, in ohms (0 if Wf specified)

Delta = length added to fingers for tuning performance

Notes

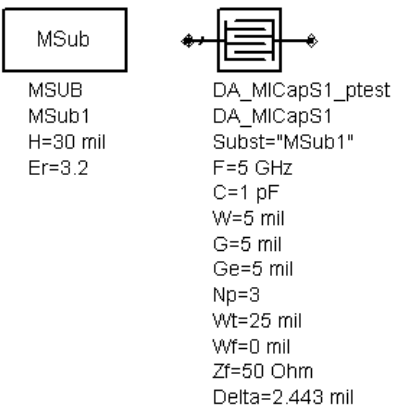
1. MICapS designs a series capacitance within a microstrip line using interdigital fingers. The underlying design uses the MICAP1 component contained in the Tlines-Microstrip palette.
2. The design is accomplished using a simple model that specifies the length required to achieve the capacitance C at 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 Microstrip Components 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

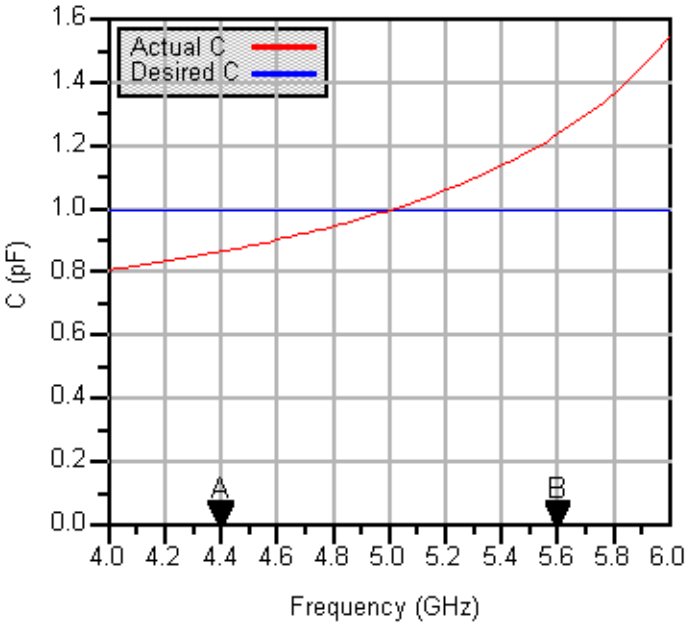
tune the design within the Simulation Assistant and subsequently use the optimizer to perform the fine tuning.

6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to “Design Assistant” on page 3-1.

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.




Actual C
Desired C
C (pF)
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
A
B
4.0
4.2
4.4
4.6
4.8
5.0
5.2
5.4
5.6
5.8
6.0
Frequency (GHz)

MICapSG (Microstrip 1-port Interdigital Capacitor)

Symbol

Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

C = capacitance, in farads

W = width of fingers

G = gap between fingers

Ge = gap at end of fingers

Np = number of finger pairs

Wt = width of interconnect

Wf = width of feed line (0 if Zf specified)

Zf = characteristic impedance of feed line, in ohms (0 if Wf specified)

Delta = length added to fingers for tuning performance

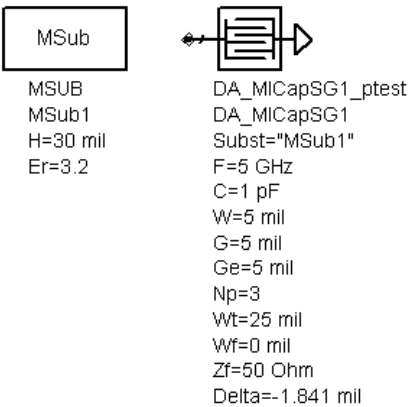
Notes

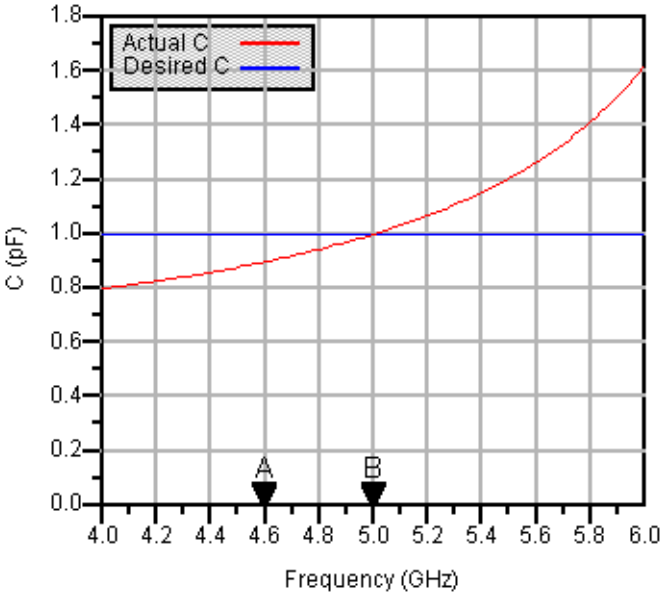
1. MICapSG designs a series capacitance between a microstrip line and ground using interdigital fingers. The underlying design uses the MICAP3 component contained in the Tlines-Microstrip palette.
2. The design is accomplished using a simple model that specifies the length required to achieve the capacitance 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 Microstrip Components 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” on page 3-1.

Example

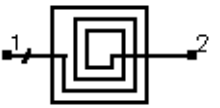
A MICapSG component was used to design a 1 pF capacitance for a 50 ohm line at a center frequency of 5 GHz. Optimization yielded a value of Delta = -1.841 mil.




Actual C
Desired C
C (pF)
1.8
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
A
B
4.0
4.2
4.4
4.6
4.8
5.0
5.2
5.4
5.6
5.8
6.0
Frequency (GHz)

MREInd (Microstrip Elevated Rectangular Inductor)

Symbol



Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

L = inductance, in henries

Ln = length of innermost segment (0 means full length)

Ln1 = length of second innermost segment

Ln2 = length of second innermost segment

W = conductor width

Ri = resistivity (relative to gold) of conductors

Sx = spacing limit between support posts (0 to ignore posts)

Cc = coefficient for capacitance of corner support posts

Cs = coefficient for capacitance of support posts along segment

Wu = width of underpass strip conductor

Au = angle of departure from innermost segment

UE = extension of underpass beyond inductor

Delta = incremental number of segments for tuning inductance (need not be integer)

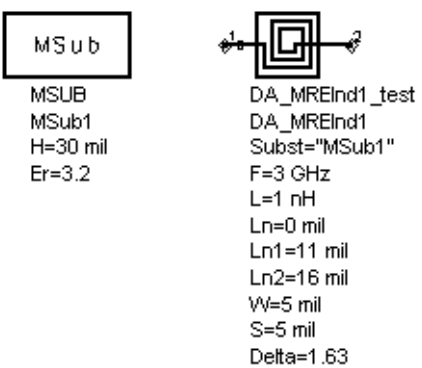
Notes

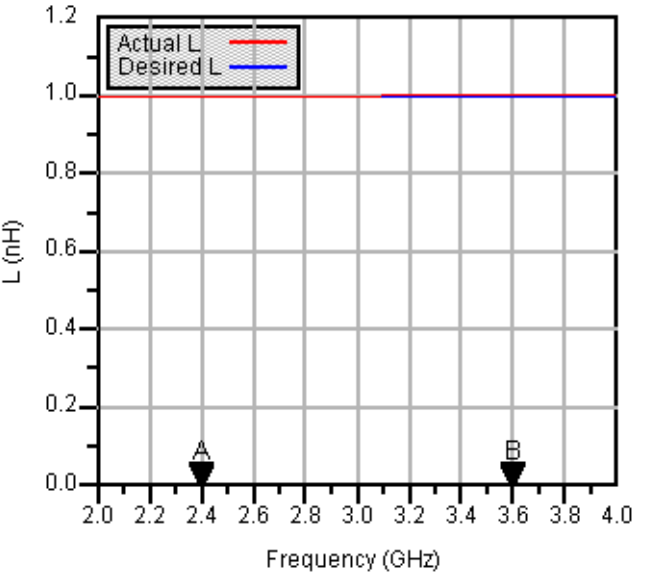
1. MREInd designs an elevated microstrip rectangular inductor. The underlying design uses the MRINDELA component contained in the Tlines-Microstrip palette.
2. The design is accomplished using a simple model that specifies the number of segments in the rectangular inductor required to achieve the inductance 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.
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 Microstrip Components 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 "Optimization Assistant" on page 3-4, 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" on page 3-1.

Example

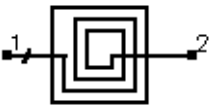
A MREInd component was used to design a 1 nH inductor at a center frequency of 3 GHz. The design used a full-length innermost segment. By tuning the number of segments within the Simulation Assistant, it was determined that a value of Delta = 1.63 would achieve the desired inductance.




Actual L
Desired L
L (nH)
1.2
1.0
0.8
0.6
0.4
0.2
0.0
A
B
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.4
3.6
3.8
4.0
Frequency (GHz)

MRInd (Microstrip Rectangular Inductor)

Symbol



Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

L = inductance, in henries

IndType = inductance type (no bridge or wire bridge)

Ln = length of innermost segment (0 means full length)

Ln1 = length of second innermost segment

Ln2 = length of second innermost segment

W = conductor width

S = conductor spacing

Rb = resistivity (relative to gold) of bridge wire (for wire bridge)

Hw = height of wire above inductor (for wire bridge)

Aw = angle of departure from innermost segment (for wire bridge)

WE = extension of bridge beyond inductor (for wire bridge)

Delta = incremental number of segments for tuning inductance (need not be integer)

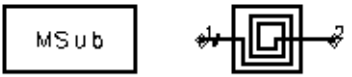
Notes

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.
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 Microstrip Components 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 “Optimization Assistant” on page 3-4, 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” on page 3-1.

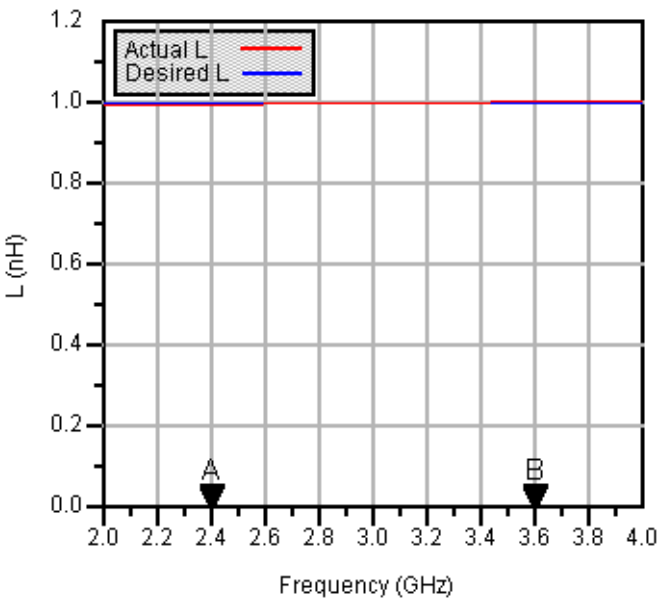
Example

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.



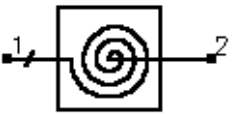
MSUB
MSub1
H=30 mil
Er=3.2

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

Subst = microstrip substrate name

F = center frequency, in hertz

Ri = inner radius measured to center of conductor

W = conductor width

S = conductor spacing

W1 = width of strip connected to pin 1

W2 = width of strip connected to pin 2

Delta = incremental number of turns for tuning inductance (need not be integer)

Notes

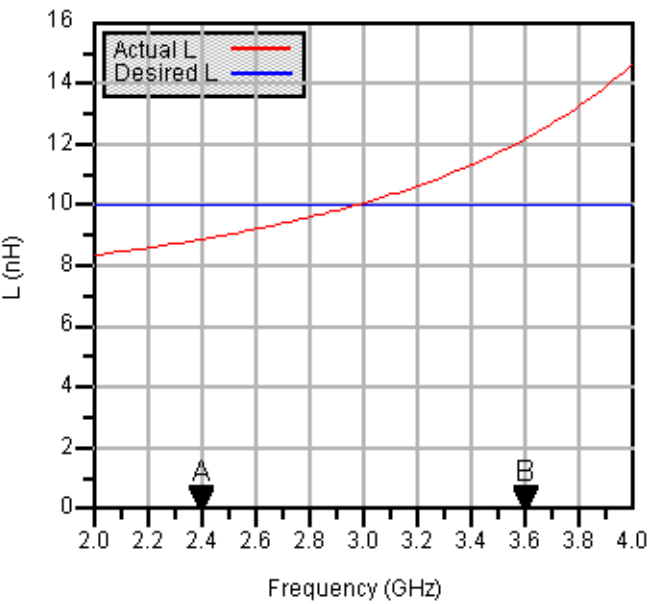
1. MSInd designs a microstrip spiral inductor. The underlying design uses the MSIND component contained in the Tlines-Microstrip palette.
2. The design is accomplished using a simple model that specifies the number of turns in the spiral inductor required to achieve the inductance L at the design center 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 Microstrip Components 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).
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” on page 3-1.

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.

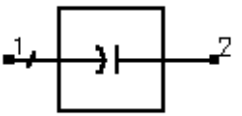
MSUB
MSub1
H=30 mil
Er=3.2

DA_MSInd1_test
DA_MSInd1
Subst="MSub1"
F=3 GHz
L=10 nH
Ri=50 mil
W=10 mil
S=10 mil
W1=25 mil
W2=25 mil
Delta=-0.53



MTFC (Microstrip Thin Film Capacitor)

Symbol



Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

C = capacitance, in farads

W = conductor width (set to 0 if Zo specified)

Zo = characteristic impedance of line for computing W (set to 0 if W specified)

CPUA = capacitance per unit area, in pf/mm^2

T = thickness of capacitor dielectric

RsT = sheet resistance of top metal plate, in ohms

RsB = sheet resistance of bottom metal plate, in ohms

TT = thickness of top metal plate

TB = thickness of bottom metal plate

COB= bottom conductor overlap

COT = top conductor overlap

DO = dielectric overlap

Delta = length added to conductor for tuning capacitance

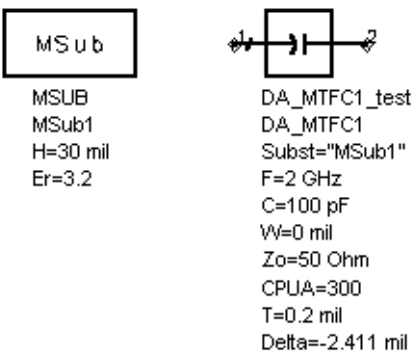
Notes

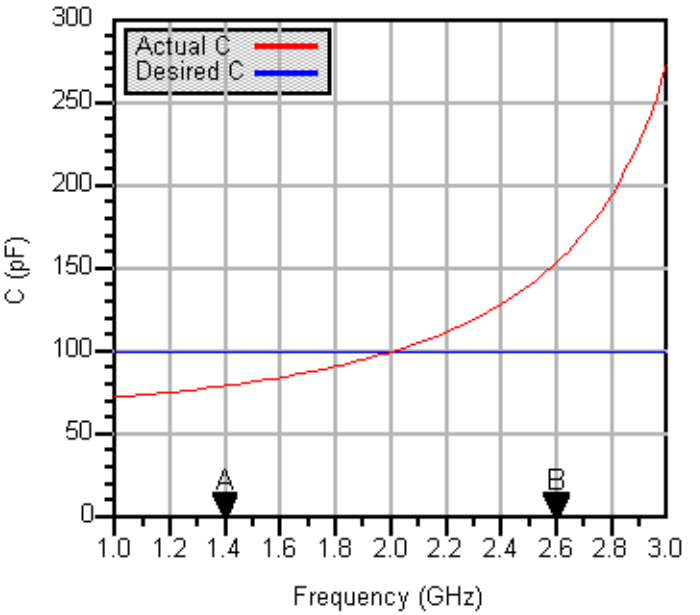
1. MTFC designs a microstrip thin film capacitor. The underlying design uses the MTFC component contained in the Tlines-Microstrip palette.
2. The design is accomplished by determining the length required to achieve the desired capacitance using the capacitance per unit area (CPUA) in conjunction with 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" on page 3-1.

Example

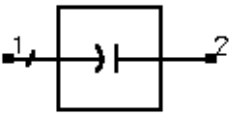
A MTFC component was used to design a 100 pF capacitor at a center frequency of 2 GHz. The conductor width corresponds to that of a 50 ohm microstrip line fabricated on MSub1. Optimization yielded a value of Delta = -2.411.




Actual C
Desired C
C (pF)
300
250
200
150
100
50
0
A
B
1.0
1.2
1.4
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0
Frequency (GHz)

TFC (Thin Film Capacitor)

Symbol



Parameters

F = center frequency, in hertz

C = capacitance, in farads

W = conductor width

T = thickness of capacitor dielectric

Er = relative dielectric constant

Rho = resistivity of conductor (relative to gold)

TanD = dielectric loss tangent

CO = conductor overlap

DO = dielectric overlap

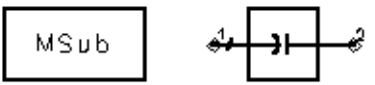
Delta = length added to conductor for tuning capacitance

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” on page 3-1.

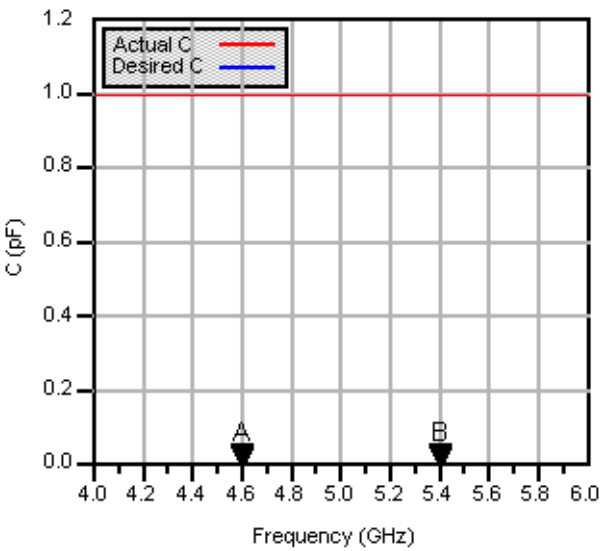
Example

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.



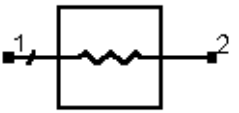
MSUB
MSub1
H=30 mil
Er=3.2

DA_TFC1_test
DA_TFC1
F=5 GHz
C=1 pF
W=25 mil
T=0.2 mil
Er=9.6
Delta=-0.000178 mil



TFR (Thin Film Resistor)

Symbol



Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

R = resistance, in ohms

W = conductor width (set to 0 if Zo specified)

Zo = characteristic impedance of line assuming Rs = 0 (set to 0 if W specified)

Rs = sheet resistivity, in ohms/square

CO = conductor overlap

Delta = length added to conductor for tuning resistance

Notes

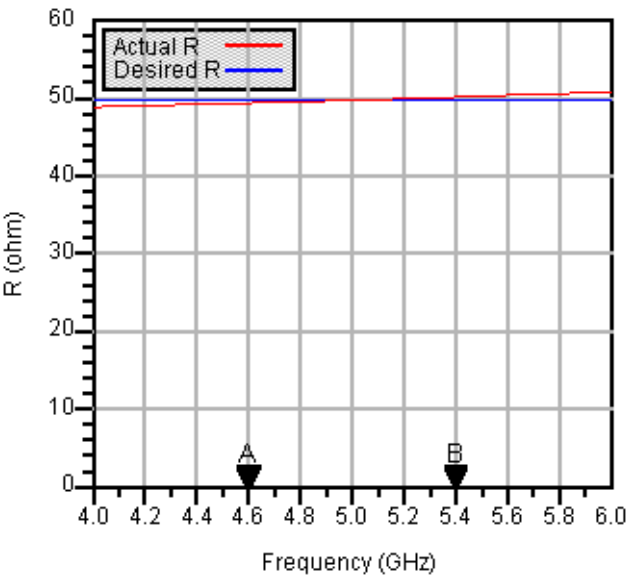
1. TFR designs a thin film resistor. The underlying design uses the TFR component contained in the Tlines-Microstrip palette.
2. The design is accomplished by determining the length required to achieve the desired resistance using the sheet resistivity Rs in conjunction with the strip width W. The parameters 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” on page 3-1.

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.

MSUB
MSub1
H=30 mil
Er=3.2

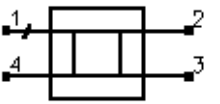
DA_TFR1_test
DA_TFR1
Subst="MSub1"
F=5 GHz
R=50 Ohm
W=0 mil
Zo=50 Ohm
Rs=50 Ohm
Delta=-3.549 mil



Passive Circuit DG - Couplers

BLCoupler (Branch-Line Coupler)

Symbol



Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

DeltaF = total frequency bandwidth, in hertz

Zo = characteristic impedance, in ohms

ResponseType = type of frequency response

N = number of coupler sections (set N = 0 to compute N)

Rmax = maximum voltage reflection coefficient at the input

C = coupling coefficient, in dB

Delta = length added to branches for tuning performance

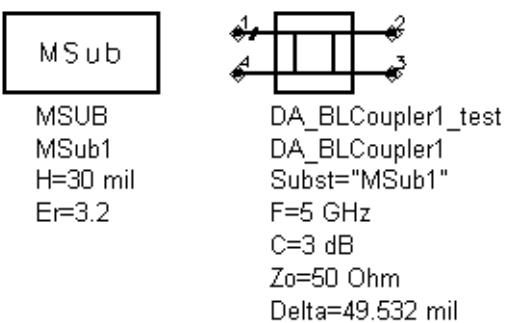
Notes

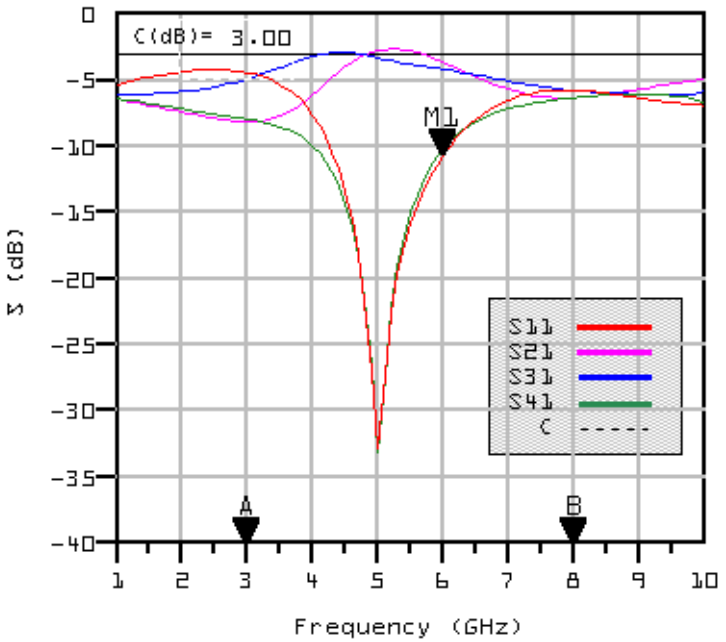
1. A branch-line coupler outputs from the coupled port (pin 3) a fraction of the power presented at the input (pin 1). The remainder of the power is passed through to the output port (pin 2). At the center frequency the phase difference between the outputs is 90 degrees, with the coupled port representing 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 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.
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 may 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” on page 3-1.
7. 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

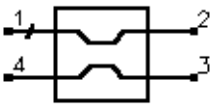
A single-section branch-line coupler was designed for a center frequency of 5 GHz with an equal power split between the I and Q ports. Tuning using the Optimization Assistant yielded a value of Delta = 49.532 mil.




C(dB)= 3.00
M1
A
B
S11
S21
S31
S41
C
S (dB)
Frequency (GHz)
0
-5
-10
-15
-20
-25
-30
-35
-40
1
2
3
4
5
6
7
8
9
10

CLCoupler (Coupled-Line Coupler)

Symbol



Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

C = coupling coefficient, in dB

Zo = characteristic impedance, in ohms

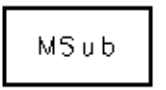
Delta = length added to branches for tuning performance

Notes

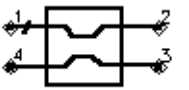
1. A coupled-line coupler outputs from the coupled port (pin 4) a fraction of the power presented at the input (pin 1). The remainder of the power is passed through to the output port (pin 2). The coupling coefficient specifies the ratio of the input power to the coupled power (P_1/P_4). The remaining 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 may 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” on page 3-1.
5. 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

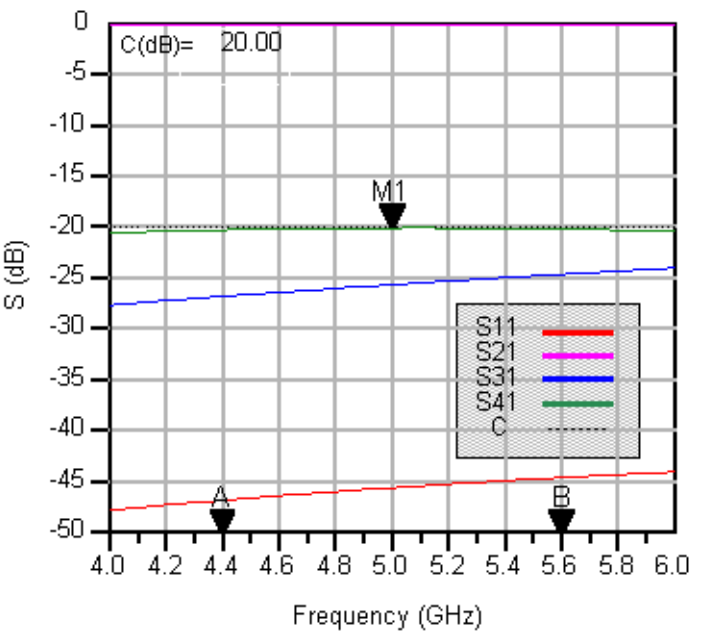
A coupled-line coupler was designed for a center frequency of 5 GHz with 20 dB of coupling. Tuning using the Optimization Assistant yielded a value of Delta =-1.087 mil.



MSUB
MSub1
H=30 mil
Er=3.2

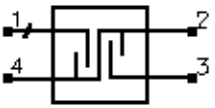


DA_CLCoupler1_test
DA_CLCoupler1
Subst="MSub1"
F=5 GHz
C=20 dB
Zo=50 Ohm
Delta=-1.087 mil



LCoupler (Lange Coupler)

Symbol



Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

C = coupling coefficient, in dB

N = number of fingers (4, 6, or 8)

Zo = characteristic impedance, in ohms

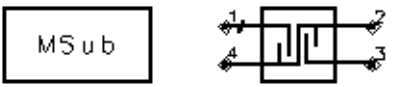
Delta = length added to fingers for tuning performance

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 may 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” on page 3-1.
6. 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.

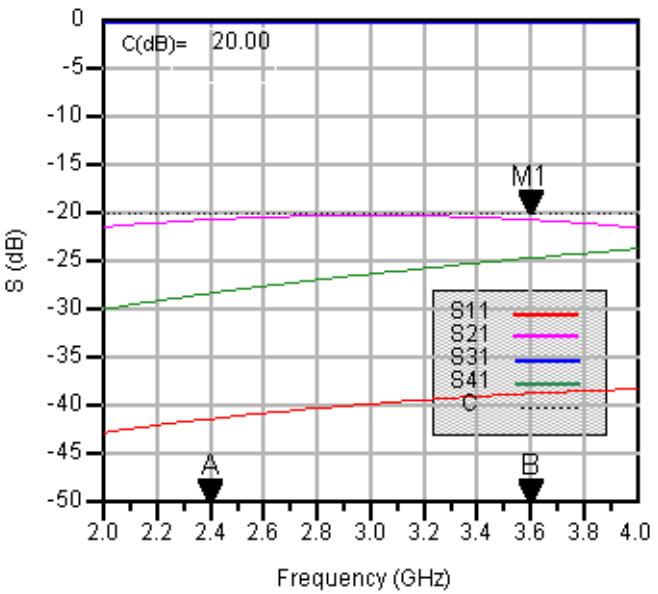
Example

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.



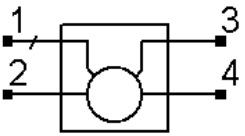
MSUB
MSub1
H=30 mil
Er=3.2

DA_LCoupler1_test
DA_LCoupler1
Subst="MSub1"
F=3 GHz
C=20 dB
N=6
Zo=50 Ohm
Delta=-4.57 mil



RRCoupler (Rat-Race Coupler)

Symbol



Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

Zo = characteristic impedance, in ohms

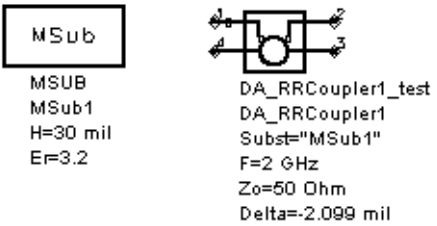
Delta = length added to ring branches for tuning

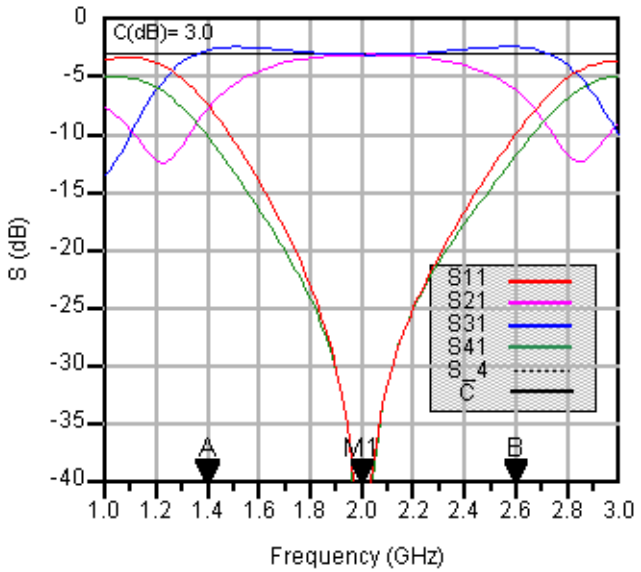
Notes

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” on page 3-1.
5. 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

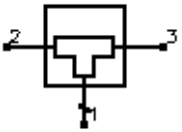
A rat-race coupler was designed for a 50 ohm system impedance at a center frequency of 2 GHz. Tuning using the Optimization Assistant yielded a value of Delta =-2.099 mil.


MSub
MSUB
MSub1
H=30 mil
Er=3.2
DA_RRCoupler1_test
DA_RRCoupler1
Subst="MSub1"
F=2 GHz
Zo=50 Ohm
Delta=-2.099 mil


C(dB)= 3.0
S (dB)
Frequency (GHz)
S11
S21
S31
S41
S_4
C
A
M1
B
0
-5
-10
-15
-20
-25
-30
-35
-40
1.0
1.2
1.4
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0

TCoupler (Tee Power Divider)

Symbol



Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

Zo1 = characteristic impedance of input port 1, in ohms

Zo2 = characteristic impedance of output port 2, in ohms

K = ratio of power out port 2 to power out port 3

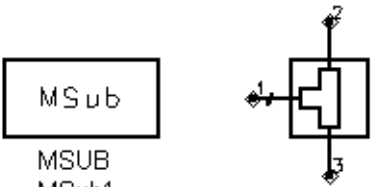
Delta = length added to quarter-wave sections for tuning performance

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) may 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 may 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” on page 3-1.
6. 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.

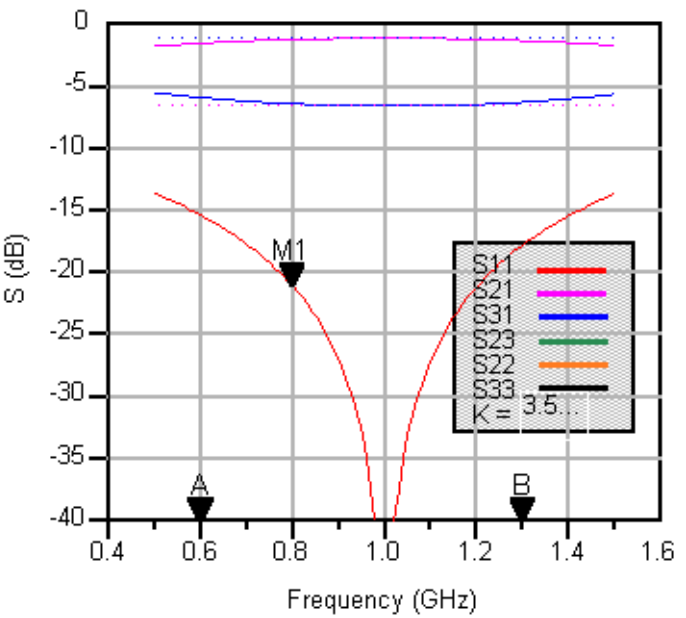
Example

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.



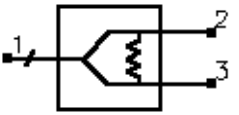
MSUB
MSub1
H=30 mil
Er=3.2

DA_TCoupler1_test
DA_TCoupler1
Subst="MSub1"
F=1 GHz
Zo1=50 Ohm
Zo2=50 Ohm
Zo3=50 Ohm
Delta=-34.633 mil



WDCoupler (Wilkinson Divider)

Symbol



Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

DeltaF = frequency bandwidth, in hertz

Zo = characteristic impedance, in ohms

ResponseType = type of frequency response

N = number of quarter-wave sections (set N=0 to compute N)

Rmax = maximum voltage reflection coefficient on input port

K = ratio of power out port 2 to power out port 3

Wgap = width of gap for resistor

Delta = length added to quarter-wave branches for tuning performance

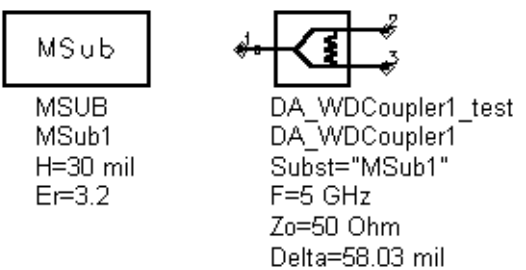
Notes

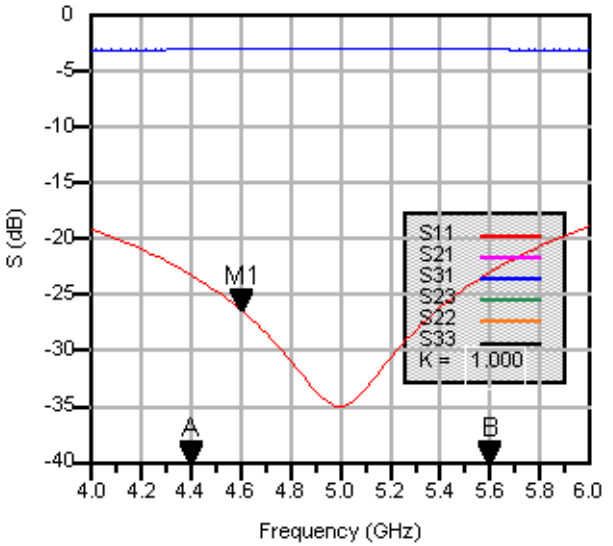
1. A Wilkinson power divider splits the power at the input (pin 1) between the two outputs (pins 2 and 3). Unequal or equal power splits can be realized. The signals at the outputs are in phase. All three ports will be matched, and ports 2 and 3 will in general be well isolated from each other.
2. For 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 may 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" on page 3-1.
8. 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.

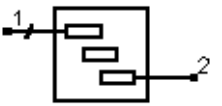



0
-5
-10
-15
-20
-25
-30
-35
-40
S (dB)
M1
A
B
S11
S21
S31
S23
S22
S33
K = 1.000
4.0
4.2
4.4
4.6
4.8
5.0
5.2
5.4
5.6
5.8
6.0
Frequency (GHz)

Passive Circuit DG - Filters

CLFilter (Coupled-Line Filter)

Symbol



Parameters

Subst = microstrip substrate name

Fs1 = lower stopband edge frequency, in hertz

Fp1 = lower passband edge frequency, in hertz

Fp2 = upper passband edge frequency, in hertz

Fs2 = lower stopband edge frequency, in hertz

Ap = passband edge attenuation (or ripple for Chebyshev), in dB

As = stopband edge attenuation, in dB

N = number of filter sections (or 0 to compute N)

ResponseType = type of frequency response (maximally flat or Chebyshev)

Zo = desired input/output impedance, in ohms

Delta = length added to coupled sections for tuning performance

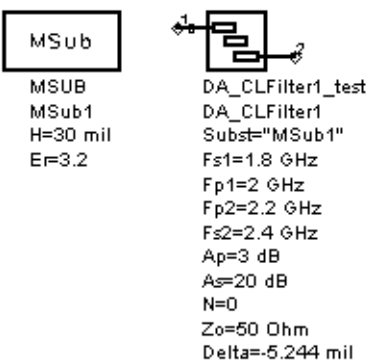
Notes

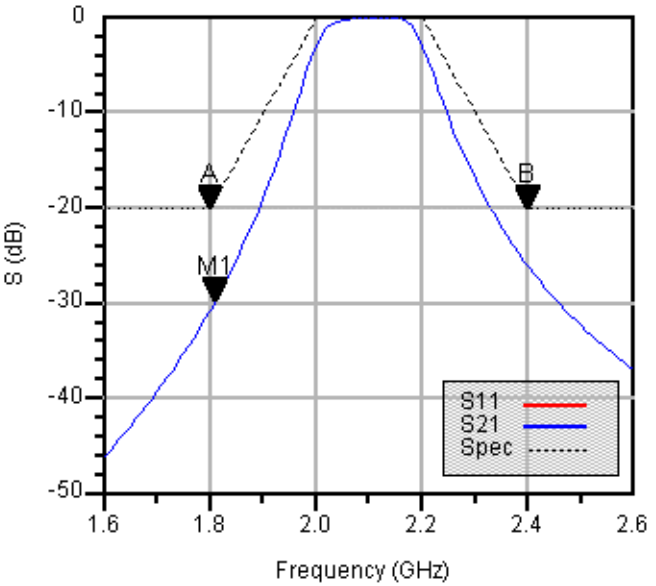
1. A coupled-line filter provides a bandpass frequency response between the input and output ports. N coupled-line sections produces an N-1 order filter response. Additional numbers of sections can be used to steepen the transition band roll off or widen the pass bandwidth.
2. Because of the heavy computational 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. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to "Design Assistant" on page 3-1.
7. 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.
8. 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.

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.




0
-10
-20
-30
-40
-50
S (dB)
A
B
M1
S11
S21
Spec
1.6
1.8
2.0
2.2
2.4
2.6
Frequency (GHz)

CMFilter (Comb-Line Filter)

Symbol

Parameters

Subst = microstrip substrate name

Fs1 = lower stopband edge frequency, in hertz

Fp1 = lower passband edge frequency, in hertz

Fp2 = upper passband edge frequency, in hertz

Fs2 = lower stopband edge frequency, in hertz

Ap = passband edge attenuation (or ripple for Chebyshev), in dB

As = stopband edge attenuation, in dB

N = number of filter sections (or 0 to compute N)

ResponseType = type of frequency response (maximally flat or Chebyshev)

Zo = desired input/output impedance, in ohms

Lelec = electrical length of filter resonators (the units of Leclec are in fractions of a wavelength i.e., 0.25 = 1/4 wavelength)

ya = normalized interior resonator admittance ($0 < ya < 1$)

CouplingType = type of input/output coupling (coupled line or tapped line transformer)

Delta = length added to coupled sections for tuning performance

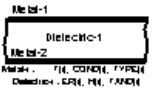
Notes

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” on page 3-1.
9. 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.

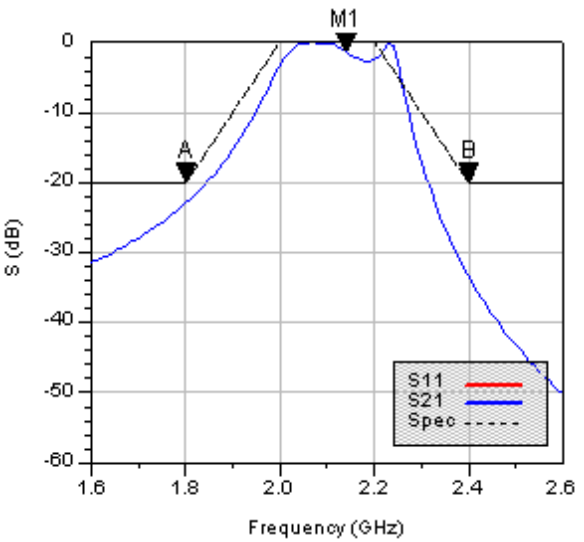
Example

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



HPFilter (Hairpin Filter)

Symbol

Parameters

Subst = microstrip substrate name

Fs1 = lower stopband edge frequency, in hertz

Fp1 = lower passband edge frequency, in hertz

Fp2 = upper passband edge frequency, in hertz

Fs2 = lower stopband edge frequency, in hertz

Ap = passband edge attenuation (or ripple for Chebyshev), in dB

As = stopband edge attenuation, in dB

N = number of filter sections (or 0 to compute N)

ResponseType = type of frequency response (maximally flat or Chebyshev)

Zo = desired input/output impedance, in ohms

Sphys = physical spacing between legs within hairpin resonator (set to zero if Selec specified)

Selec = spacing between legs within hairpin resonator in wavelengths (set to zero if Sphys specified)

CouplingType = type of input/output coupling (coupled line or tapped line transformer)

Delta = length added to coupled sections for tuning performance

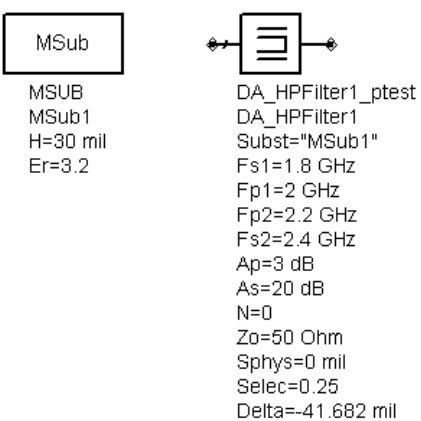
Notes

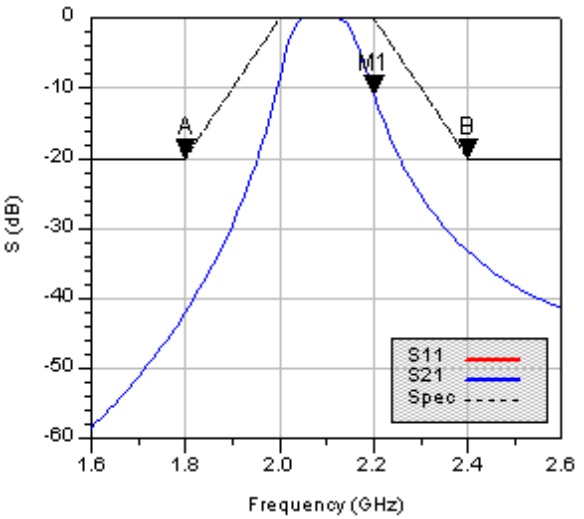
1. A hairpin filter provides a bandpass frequency response between the input and output ports. N coupled-line sections produce an N-1 order filter response. Additional numbers of sections can be used to steepen the transition band roll off or widen the pass bandwidth.
2. Because of the heavy computational 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” on page 3-1.
8. 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.




0
-10
-20
-30
-40
-50
-60
S (dB)
1.6
1.8
2.0
2.2
2.4
2.6
Frequency (GHz)
A
M1
B
S11
S21
Spec

IDFilter (Interdigital Filter)

Symbol

Parameters

Subst = microstrip substrate name

Fs1 = lower stopband edge frequency, in hertz

Fp1 = lower passband edge frequency, in hertz

Fp2 = upper passband edge frequency, in hertz

Fs2 = lower stopband edge frequency, in hertz

Ap = passband edge attenuation (or ripple for Chebyshev), in dB

As = stopband edge attenuation, in dB

N = number of filter sections (or 0 to compute N)

ResponseType = type of frequency response (maximally flat or Chebyshev)

Zo = desired input/output impedance, in ohms

ya = normalized interior resonator admittance ($0 < ya < 1$)

CouplingType = type of input/output coupling (coupled line or tapped line transformer)

Delta = length added to coupled sections for tuning performance

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.

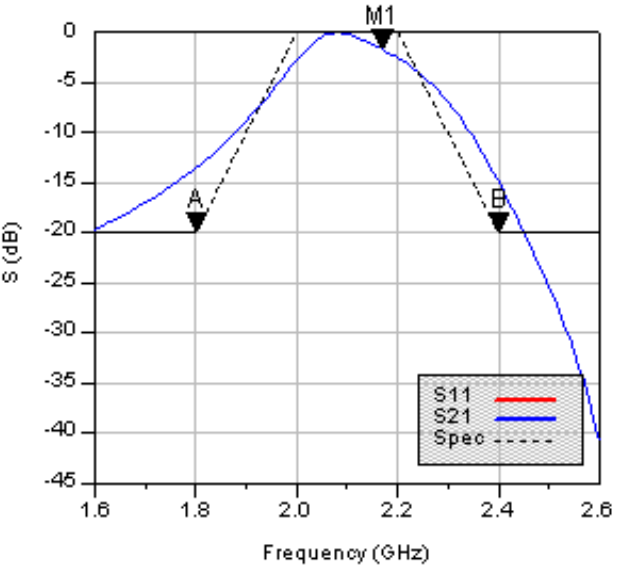
5. If N is zero, the number of filter sections will be computed from the frequency/attenuation information. If N is non-zero, the design will use the frequency/attenuation parameters only for determining the design center frequency.
6. Using a CouplingType of “Coupled Line Transformer Input” will use 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” on page 3-1.
9. 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.

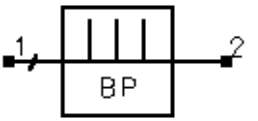
MLSUBSTRATE2
Subst1
Er=3.2
H=30 mil

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


M1
A
B
S (dB)
0
-5
-10
-15
-20
-25
-30
-35
-40
-45
1.6
1.8
2.0
2.2
2.4
2.6
Frequency (GHz)
S11
S21
Spec

SBFilter (Stub Bandpass Filter)

Symbol



Parameters

Subst = microstrip substrate name

Fs1 = lower stopband edge frequency, in hertz

Fp1 = lower passband edge frequency, in hertz

Fp2 = upper passband edge frequency, in hertz

Fs2 = lower stopband edge frequency, in hertz

Ap = passband edge attenuation (or ripple for Chebyshev), in dB

As = stopband edge attenuation, in dB

N = number of filter sections (or 0 to compute N)

ResponseType = type of frequency response (maximally flat or Chebyshev)

StubConfig = configuration of interior shunt stubs

Zo = desired input/output impedance, in ohms

D = impedance control parameter ($0 < D < 1$)

Delta = length added to stubs for tuning performance

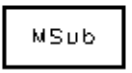
Notes

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 may 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.

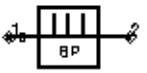
3. For a Chebyshev (equal ripple) frequency response, ripple levels greater than about 1 dB are not recommended. Exceeding this value will typically deform the shape of the passband characteristics.
4. The parameter StubConfig specifies whether the interior stubs (all but those closest to the source 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 $0 < D < 1$ may 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” on page 3-1.
9. 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

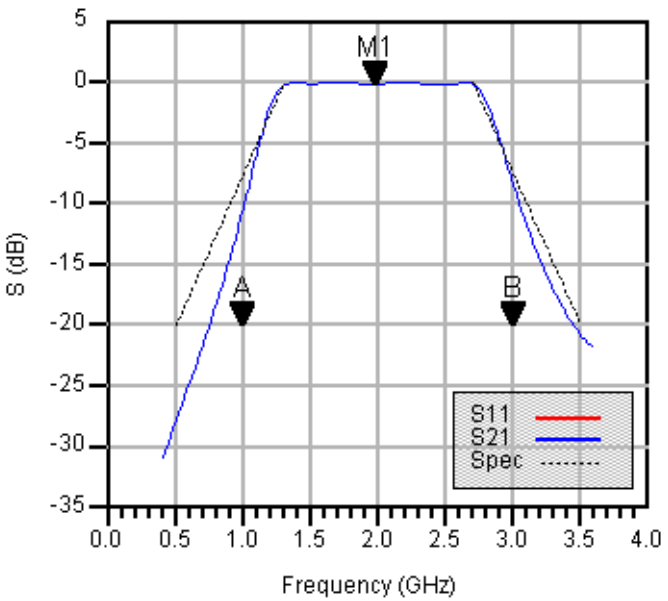
A stub bandpass filter was designed for a Chebyshev response with a 0.1 dB ripple. The passband edge frequencies are at 1.3GHz and 2.7 GHz respectively. The design uses two parallel stubs in the interior regions with D = 1. The design required 4 stubs. Tuning using the Optimization Assistant yielded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



MSUB
MSub1
H=30 mil
Er=3.2

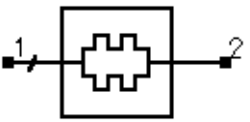


DA_SBFilter1_test
DA_SBFilter1
Subst="MSub1"
Fs1=0.5 GHz
Fp1=1.3 GHz
Fp2=2.7 GHz
Fs2=3.5 GHz
Ap=0.1 dB
As=20 dB
N=0
Zo=50 Ohm
Delta=-19.316 mil



SIFilter (Stepped Impedance Lowpass Filter)

Symbol



Parameters

Subst = microstrip substrate name

Fp = frequency at passband edge, in hertz

Ap = passband edge attenuation (or ripple for Chebyshev), in dB

Fs = frequency at stopband edge, in hertz

As = stopband edge attenuation, in dB

N = number of filter sections (or 0 to compute N)

ResponseType = type of frequency response (maximally flat or Chebyshev)

FElement = first filter component (automatic, capacitive, or inductive)

Zo = desired input/output impedance, in ohms

ZL = characteristic impedance of low impedance sections, in ohms

ZH = characteristic impedance of high impedance sections, in ohms

Delta = length added to filter sections for tuning performance

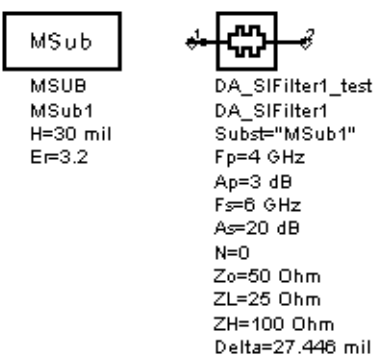
Notes

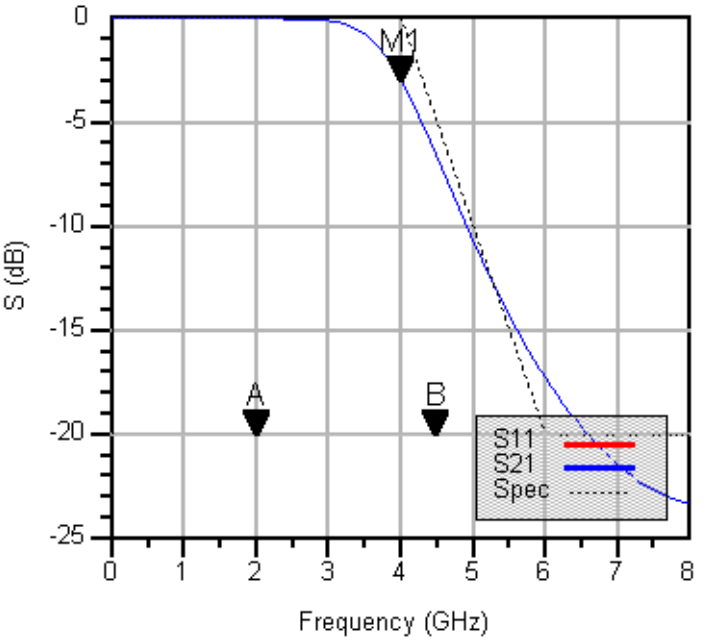
1. A stepped impedance filter provides a lowpass frequency response between the input and output ports. The design is realized using alternating wide and narrow microstrip lines.
2. The stepped impedance filter uses wide microstrip lines to approximate shunt capacitors and narrow lines to approximate 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 may 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" on page 3-1.
9. 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

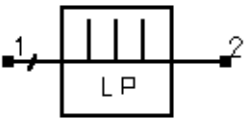
A stepped impedance lowpass filter was designed for a maximally flat response with a 3 dB attenuation at the passband edge frequency of 4 GHz. Choosing FElement as Automatic results in a capacitive first component. Tuning using the Optimization Assistant yielded a value of Delta = 27.446 mil.




0
-5
-10
-15
-20
-25
S (dB)
M1
A
B
S11
S21
Spec
0
1
2
3
4
5
6
7
8
Frequency (GHz)

SLFilter (Stub Lowpass Filter)

Symbol



Parameters

Subst = microstrip substrate name

Fp = frequency at passband edge, in hertz

Ap = passband edge attenuation (or ripple for Chebyshev), in dB

Fs = frequency at stopband edge, in hertz

As = stopband edge attenuation, in dB

N = number of filter sections (or 0 to compute N)

ResponseType = type of frequency response (maximally flat or Chebyshev)

StubType = type of stubs (commensurate or variable length)

FElement = first filter component (automatic, capacitive, or inductive)

Zo = desired input/output impedance, in ohms

ZS = characteristic impedance of stubs, in ohms

ZH = characteristic impedance of connecting sections, in ohms

Delta = length added to stubs for tuning performance

Notes

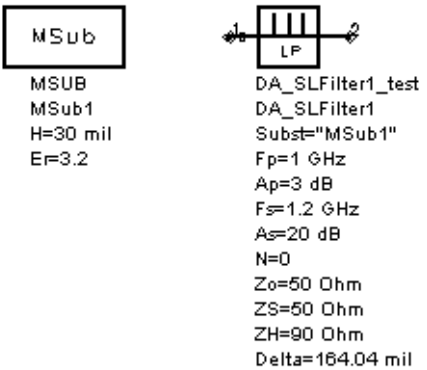
1. A stub lowpass filter provides a lowpass frequency response between the input and output ports. The design is realized using narrow lines to approximate series inductances and shunt open circuited stubs to realize shunt capacitances.
2. The stub lowpass filter uses narrow microstrip lines that approximate 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

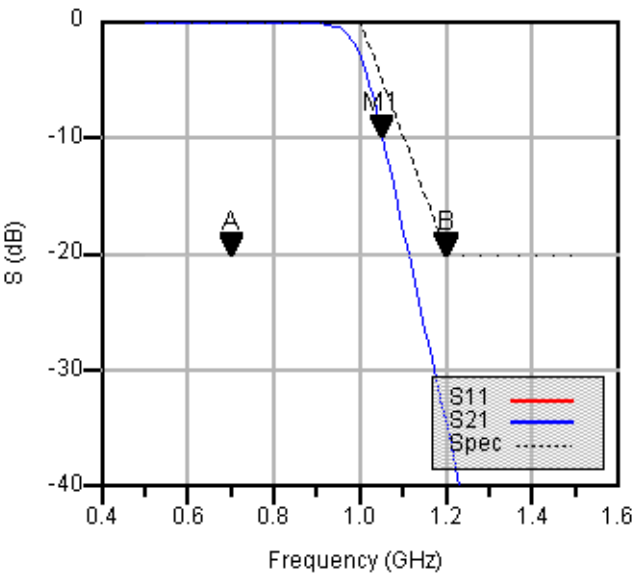
characteristics. In order to obtain an impedance match, Chebyshev designs must use an odd number of components (N).

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 may 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" on page 3-1.
9. 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

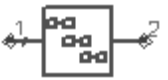
A stub lowpass filter was designed for a maximally flat response with a 3 dB attenuation at the passband edge frequency of 1 GHz. A variable length StubType and automatic FElement resulted in 13 components for the design. Tuning using the Optimization Assistant yielded a value of Delta = 164.04 mil.


MSub
MSUB
MSub1
H=30 mil
Er=3.2
LP
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


0
-10
-20
-30
-40
S (dB)
M1
A
B
S11
S21
Spec
0.4
0.6
0.8
1.0
1.2
1.4
1.6
Frequency (GHz)

SRFilter (Stepped Impedance Resonator Filter)

Symbol



Parameters

Subst = microstrip substrate name

Fs1 = lower stopband edge frequency, in hertz

Fp1 = lower passband edge frequency, in hertz

Fp2 = upper passband edge frequency, in hertz

Fs2 = lower stopband edge frequency, in hertz

Ap = passband edge attenuation (or ripple for Chebyshev), in dB

As = stopband edge attenuation, in dB

N = number of filter sections (or 0 to compute N)

ResponseType = type of frequency response (maximally flat or Chebyshev)

Zo = desired input/output impedance, in ohms

Fsp = center frequency of first spurious passband

Delta = length added to filter sections for tuning performance

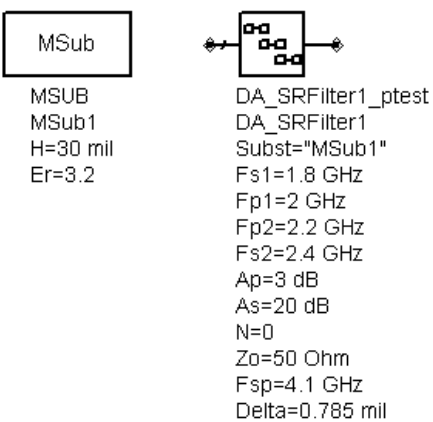
Notes

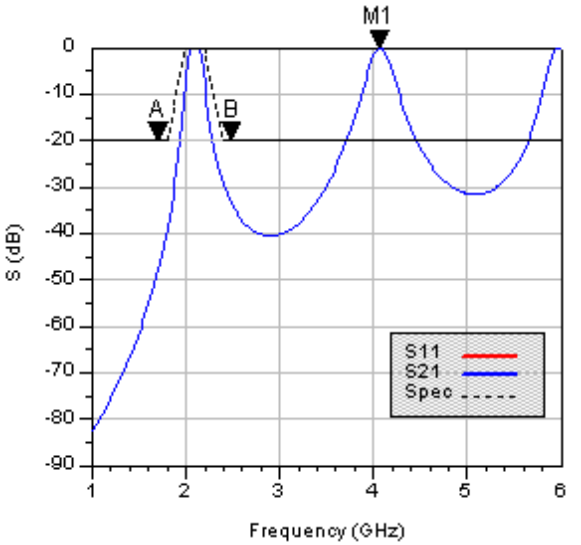
1. A stepped impedance resonator filter provides a bandpass frequency response between the input and output ports. N coupled-line sections produce an N–1 order filter response. Additional numbers of sections can be used to steepen the transition band roll off or widen the pass bandwidth. The use on non-uniform 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" on page 3-1.
8. 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.




M1
A
B
0
-10
-20
-30
-40
-50
-60
-70
-80
-90
S (dB)
1
2
3
4
5
6
Frequency (GHz)
S11
S21
Spec

ZZFilter (Zig-Zag Coupled-Line Filter)

Symbol

Parameters

Subst = microstrip substrate name

Fs1 = lower stopband edge frequency, in hertz

Fp1 = lower passband edge frequency, in hertz

Fp2 = upper passband edge frequency, in hertz

Ap = passband edge attenuation (or ripple for Chebyshev), in dB

As = stopband edge attenuation, in dB

N = number of filter sections (or 0 to compute N)

ResponseType = type of frequency response (maximally flat or Chebyshev)

Zo = desired input/output impedance, in ohms

Delta = length added to coupled sections for tuning performance

Notes

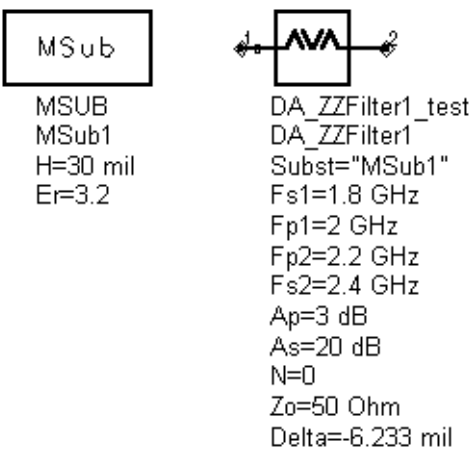
1. A zig-zag coupled-line filter provides a bandpass frequency response between the input and output ports. The design uses a coupled-line filter topology but includes 90 degree bend between each coupled section so that the layout is more compact.
2. Because of the heavy computational burden in determining 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

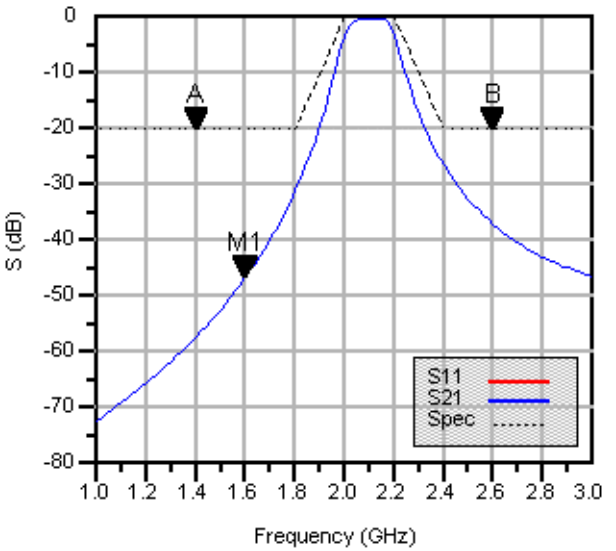
frequencies. Because only the line lengths are changed, this tuning will typically center the response within the specified passband. More advanced shaping of the passband response can be accomplished by manually tuning the widths and spacings of the coupled filter sections.

6. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to “Design Assistant” on page 3-1.
7. 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.




0
-10
-20
-30
-40
-50
-60
-70
-80
S (dB)
A
B
M1
S11
S21
Spec
1.0
1.2
1.4
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0
Frequency (GHz)

Passive Circuit DG - Matching

DSMatch (Double-Stub Match)

Symbol

Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

Zin = desired complex input impedance, in ohms

Zload = complex load impedance to match, in ohms

Zstub1 = characteristic impedance of stub line 1 (nearest input), in ohms

Zstub2 = characteristic impedance of stub line 2 (nearest output), in ohms

Zline = characteristic impedance of line between stubs, in ohms

Zfeed1 = characteristic impedance of line connected to port 1, in ohms

Zfeed2 = characteristic impedance of line connected to port 2, in ohms

Stub1Type = stub 1 type (open or short circuit)

Stub2Type = stub 2 type (open or short circuit)

Delta = length added to stubs for tuning performance

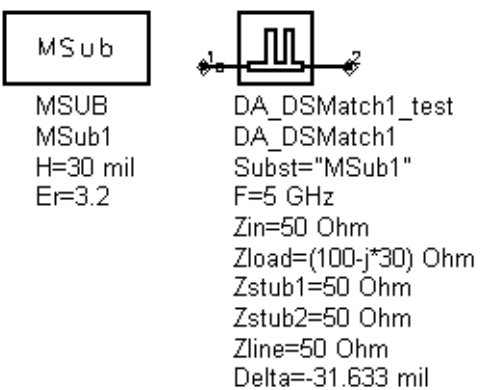
Notes

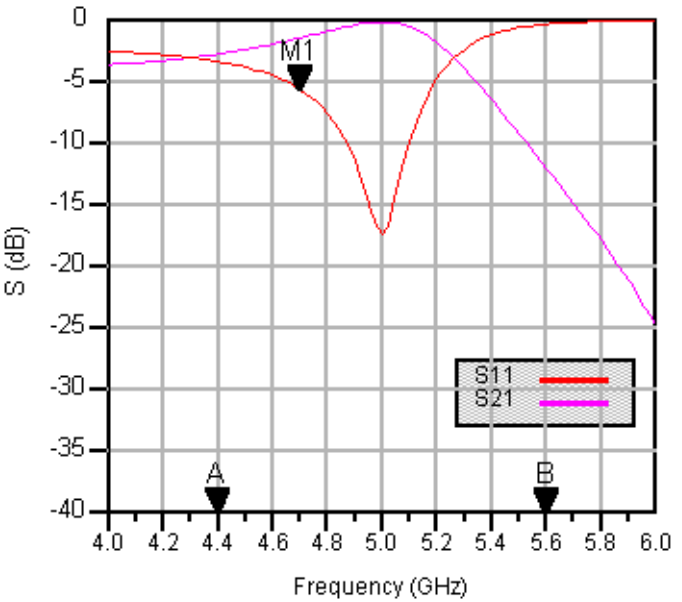
1. A double-stub matching network matches a complex load impedance (Zload) to a desired complex input impedance (Zin) using two shunt stubs and a connecting line.
2. An impedance match can be realized using any combination of stub types, although some combinations may be more realizable.
3. Zfeed1 and 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" on page 3-1.
10. 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

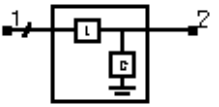
A double-stub matching network was designed to match a load impedance of 100 - j30 ohms to a 50 ohm line at a center frequency of 5 GHz with open circuited stubs. Tuning using the Optimization Assistant yielded a value of Delta =-31.633 mil.




0
-5
-10
-15
-20
-25
-30
-35
-40
S (dB)
M1
A
B
S11
S21
4.0
4.2
4.4
4.6
4.8
5.0
5.2
5.4
5.6
5.8
6.0
Frequency (GHz)

LEMatch (Lumped Component Match)

Symbol



Parameters

F = center frequency, in hertz

Zin = desired complex input impedance, in ohms

Zload = complex load impedance to match, in ohms

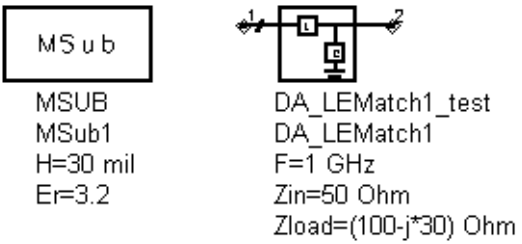
NetworkType = type of network (source to load)

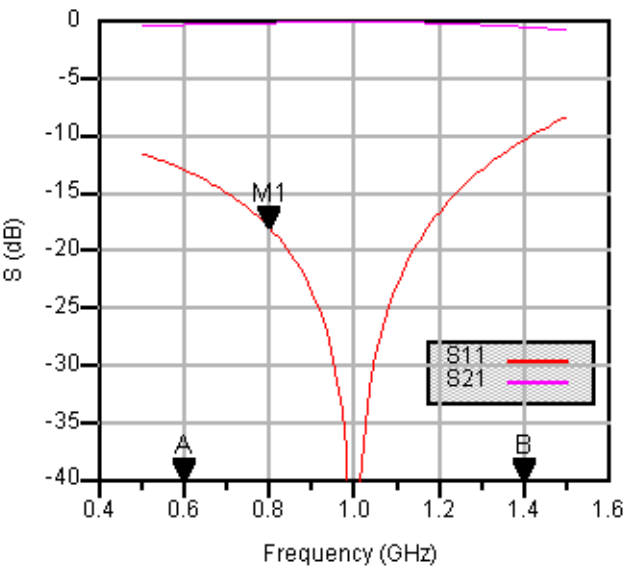
Notes

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” on page 3-1.
7. 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

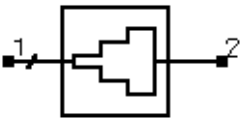
A lumped component matching network was designed to match a load impedance of 100 - j30 ohms to an input impedance of 50 ohms at a center frequency of 1 GHz. A Series L Shunt C configuration was used to realize this match.





QWMatch (Quarter-Wave Match)

Symbol



Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

DeltaF = total frequency bandwidth, in hertz

Zo = desired input impedance, in ohms

Rload = load impedance to match, in ohms

ResponseType = type of frequency response

N = number of quarter-wave sections (set N=0 to compute N)

Rmax = maximum voltage reflection coefficient

Delta = length added to transformer sections for tuning performance

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)

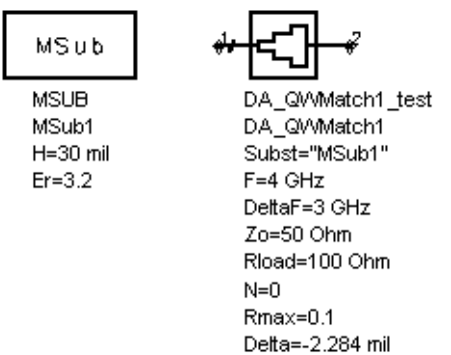
versus frequency at the center frequency by changing the length of each quarter-wave section. All sections are tuned by the same length during the optimization.

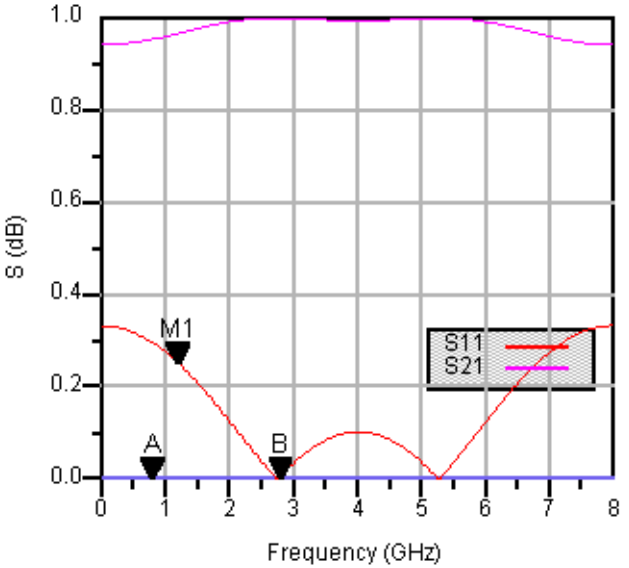
7. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to “Design Assistant” on page 3-1.

8. 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

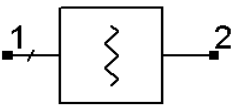
A quarter-wave matching network was designed to match a load impedance of 100 ohms to a 50 ohm line at a center frequency of 4 GHz. Specifying a Chebyshev frequency response for which the reflection coefficient remains below 0.1 over a 3 GHz bandwidth dictates 3 quarter-wave sections. Tuning using the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




1.0
0.8
0.6
0.4
0.2
0.0
S (dB)
M1
A
B
S11
S21
0
1
2
3
4
5
6
7
8
Frequency (GHz)

RAtten (Resistive Attenuator)

Symbol



Parameters

Loss = attenuation, in dB

Rin = input resistance, in ohms

Rout = output resistance, in ohms

DesignType = length added to ring branches for tuning

Notes

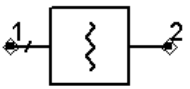
1. A resistive attenuator achieves a specified attenuation while maintaining desired input and output impedance levels.
2. The design specifies the resistance values to achieve the specified level of attenuation for the input and output resistances.
3. The input resistance Rin must be greater than or 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:

$$MinimumLoss = 20\log\left(\sqrt{\frac{Rin}{Rout}} + \sqrt{\frac{Rin}{Rout} - 1}\right)$$

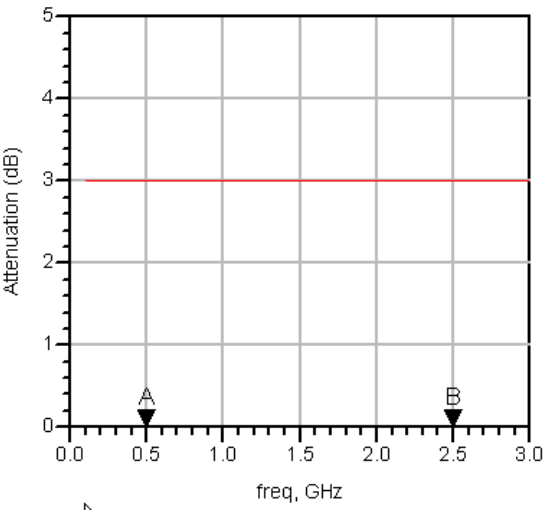
5. A SmartComponent subnetwork is empty until the Design Assistant is used to generate the design. Refer to “Design Assistant” on page 3-1.

Example

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



DA_RAtten1_test
DA_RAtten1
Loss=3 dB
Rin=50 Ohm
Rout=50 Ohm



SSMatch (Single-Stub Match)

Symbol

Parameters

Subst = microstrip substrate name

F = center frequency, in hertz

Zin = desired complex input impedance, in ohms

Zload = complex load impedance to match, in ohms

Zstub = characteristic impedance of stub line, in ohms

Zline = characteristic impedance of line, in ohms

Zfeed = characteristic impedance of line connected to port 1 or 2, in ohms

StubType = stub type (open or short circuit)

NetType = network type (source to load)

Delta = length added to stub for tuning performance

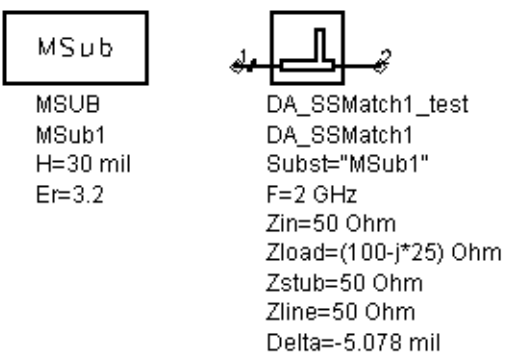
Notes

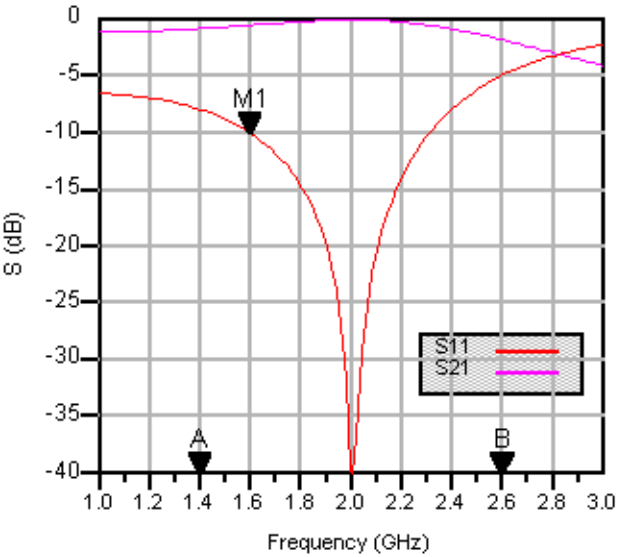
1. A single-stub matching network matches a complex load impedance (Zload) to a desired complex input impedance (Zin) using a single shunt stub and length of line.
2. The parameter NetType can be Automatic, stub-line, or line-stub, with the latter two choices representing configurations that have a stub 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" on page 3-1.
9. 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

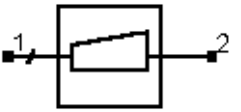
A single-stub matching network was designed to match a load impedance of 100 - j25 ohms to a 50 ohm line at a center frequency of 2GHz. Choosing an open-circuit stub with automatic selection of the NetType resulted in a stub-line configuration. Tuning using the Optimization Assistant yielded a value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




0
-5
-10
-15
-20
-25
-30
-35
-40
S (dB)
M1
A
B
S11
S21
1.0
1.2
1.4
1.6
1.8
2.0
2.2
2.4
2.6
2.8
3.0
Frequency (GHz)

TLMatch (Tapered-Line Match)

Symbol



Parameters

Subst = microstrip substrate name

F = design frequency, in hertz

Zo = desired input impedance, in ohms

Rload = load impedance to match, in ohms

ResponseType = type of frequency response (exponential, triangular, Klopfenstein)

L = length of tapered line in wavelengths at frequency F (set L=0 to compute L)

Rmax = maximum voltage reflection coefficient

NSection = number of linear taper sections per wavelength

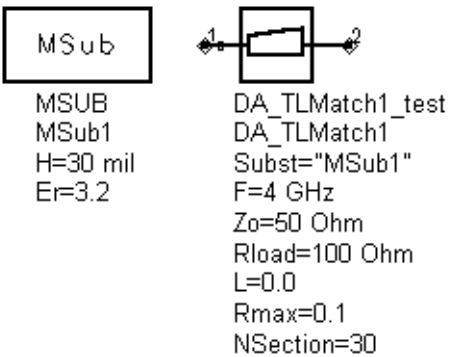
Notes

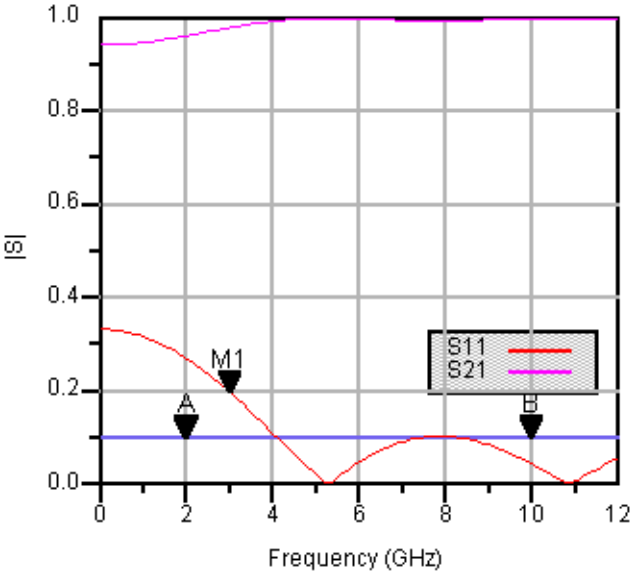
1. A tapered-line matching network matches a real load impedance (Rload) to a desired real input impedance (Zo) using a continuously varying line characteristic impedance to realize a specified frequency response. The reflection coefficient remains below the specified maximum value for all frequencies 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” on page 3-1.
7. 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 ohm 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.




1.0
0.8
0.6
0.4
0.2
0.0
|S|
0
2
4
6
8
10
12
Frequency (GHz)
M1
A
B
S11
S21

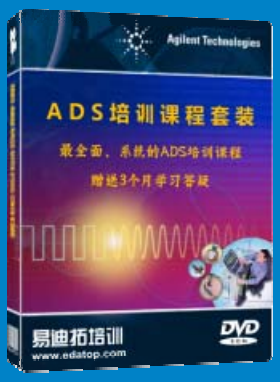
Index

Z

ADS视频培训课程推荐

ADS - Advanced Design System 是由原美国安捷伦科技(现更名为是德科技)推出的微波射频电路、通信系统和 MMIC/RFIC 仿真设计软件，其功能强大、应用广泛，被国内高校、科研院所和大型科技公司使用广为使用。掌握 ADS 无疑能提升相关设计领域工程师的技术实力、提高工作效率。

为了帮助工程技术人员更好、更快的学习掌握 ADS 的使用，易迪拓培训(www.edatop.com)特聘多年 ADS 使用经验的资深专家精心制作推出了多套 ADS 视频培训课程，由浅入深、全面系统地讲授了 ADS 在微波射频电路设计、通信系统设计和电磁仿真设计方面的仿真设计和应用操作。其中，视频课程多以设计实例边操作边讲解，工程实践强，且直观易学，能够帮助您在最短的时间内学会使用 ADS，并把 ADS 真正应用到设计研发工作中去...



ADS 学习培训课程套装

该套装是易迪拓培训和微波 EDA 网联合推出的迄今为止国内最全面、最权威的 ADS 培训教程，共包含 10 门 ADS 学习培训课程。课程是由具有多年 ADS 使用经验的微波射频与通信系统设计领域资深专家讲解，并多结合设计实例，由浅入深、详细而又全面地讲解了 ADS 在微波射频电路设计、通信系统设计和电磁仿真设计方面的内容。能让您在最短的时间内学会使用 ADS，迅速提升个人技术能力，把 ADS 真正应用到实际研发工作中去，成为 ADS 设计专家...

课程网址： http://www.edatop.com/peixun/ads/13.html

更多 ADS 视频培训课程：

- **两周学会 ADS — 中文视频培训教程**
  最新版 ADS 的入门和进阶培训课程，适合 ADS2011 ~ ADS2014 以及更新版本 ADS 的学习。是 ADS 初学者的最佳课程，网址：http://www.edatop.com/peixun/ads/14.html
- **ADS 射频模拟电路设计详解 — 中文视频教程**
  本课程基于新版的 ADS 软件，由李明洋老师讲授，讲解了 ADS 在微波射频模拟电路设计中的具体应用，视频课程，直观易学，网址：http://www.edatop.com/peixun/ads/15.html
- **ADS 高低阻抗线微带滤波器设计（ADS2014 版）— 中文视频教程**
  该门课程旨在帮助学员快速、全面、透彻地理解高低阻抗线微带滤波器的设计原理和设计步骤，帮助学员学会并掌握使用 ADS 软件仿真分析和优化设计微带线滤波器的实际操作；课程网址：http://www.edatop.com/peixun/filter/128.html
- **更多 ADS 培训课程**，敬请浏览：http://www.edatop.com/peixun/ads

关于易迪拓培训：

易迪拓培训(www.edatop.com)由数名来自于研发第一线的资深工程师发起成立，一直致力和专注于微波、射频、天线设计研发人才的培养；后于 2006 年整合合并微波 EDA 网(www.mweda.com)，现已发展成为国内最大的微波射频和天线设计人才培养基地，成功推出多套微波射频以及天线设计相关培训课程和 ADS、HFSS 等专业软件使用培训课程，广受客户好评；并先后与人民邮电出版社、电子工业出版社合作出版了多本专业图书，帮助数万名工程师提升了专业技术能力。客户遍布中兴通讯、研通高频、埃威航电、国人通信等多家国内知名公司，以及台湾工业技术研究院、永业科技、全一电子等多家台湾地区企业。

我们的课程优势：

※ 成立于 2004 年，10 多年丰富的行业经验

※ 一直专注于微波射频和天线设计工程师的培养，更了解该行业对人才的要求

※ 视频课程、既能达到现场培训的效果，又能免除您舟车劳顿的辛苦，学习工作两不误

※ 经验丰富的一线资深工程师讲授，结合实际工程案例，直观、实用、易学

联系我们：

※ 易迪拓培训官网：http://www.edatop.com

※ 微波 EDA 网：http://www.mweda.com

※ 官方淘宝店：http://shop36920890.taobao.com