

## *Wireless Circuits and Systems Laboratory*

### Procedure #10 Circuit Optimization using ADS

This procedure provides a demonstration of how to use the optimization feature of ADS. Specifically, you will first create two schematics. The first will simply be used to reference a set of measured S-parameters for a chip capacitor mounted on an FR4 substrate (relative dielectric constant of 4.3), and the second will represent an equivalent circuit model for the capacitor. In the equivalent circuit model, the values of certain parameters will be specified such that they can lie within a given range; the optimizer will attempt to determine the best combination of parameter values that result in a match between the measured S-parameters and the S-parameters of the equivalent circuit. After creating these two schematics you will create a third, from which the “data circuit” and “model circuit” are referenced and the optimization process is controlled.

The measured S-parameter data referred to in this procedure pertains to a capacitor that was mounted on a microstrip “test fixture”. As indicated in Figure 1, the measured data is referenced to the outer edges of microstrip tapers, such that the data **includes the taper effects**. Because of this, you will have to include an ADS library model of a microstrip taper on either side of the capacitor equivalent circuit model that you generate.

In order to keep this procedure brief, extensive references are made to Procedures #1-3 and 6.

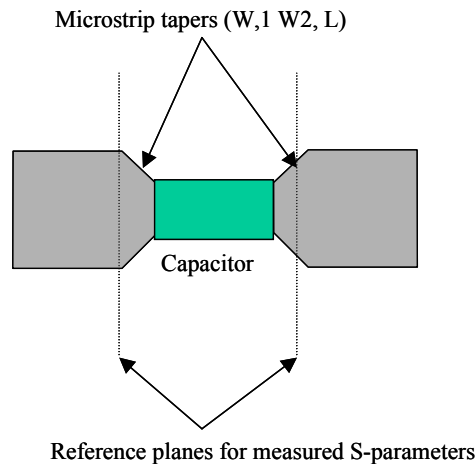


Figure 1. Microstrip test fixture used to measure the chip capacitor.

### Setting up the Schematics

- 1) Use the approach outlined in Procedure #3 to create a circuit schematic that references a .s2p data file. Attach a Port (not Termination) to each side of the schematic. (See Figure 2). Use the schematic name cap\_data\_1p2pF\_XXX, **where XXX are your initials**.
- 2) Create the equivalent circuit model using lumped elements (see Procedure #1) and microstrip tapered-line elements (see Procedure #6). Attach a Port (not Termination) to each side of the schematic (see Figure 3.) Use the schematic name cap\_model\_1p2pF\_XXX.
  - a) **To create this schematic, first select Data Items from the Component Palette pull-down menu.**

- b) Click on the **Vareqn** button on the left-hand side of the window and place a Variable block into your schematic. (You can also simply click on the **Var** button found to the immediate left of the library (books) icon.)
  - c) Double-click on the Variable block. Type **C1** in the **Name** box.
  - d) Click on the **Optimization/Statistics** setup button. From the **Optimization tab**, click on the pull-down menu next to **Optimization Status** and select **Enabled**. Select **Type** as **Continuous** and **Format** as **Min/Max**. (The continuous type means the variable does not have to take on specific discrete values. Min/Max means we will specify a nominal value and then a range on either side of the nominal through which the optimizer can vary the value of C1.)
  - e) Set the **Nominal** value to **1.2**, the **minimum** to **0.3** and the **maximum** to **3**. (These numbers will come to represent capacitance values in pF; refer to later steps.)
  - f) You also have the option to calculate certain statistics for the C1 variable, but this is not necessary for the procedure.
  - g) Click OK to finish the setup for the C1 variable.
  - h) Add the following variables to this Variable block, using the indicated nominal/min/max values: L1 (1/0/5), R1 (1/0/5), C2 (1/0/2), C3 (0.1/0/2).
  - i) **Now select Tlines-Microstrip from the Component Palette pull-down menu.**
  - j) Insert a **microstrip substrate** with the parameters shown in .
  - k) Now scroll down the menu on the left-hand side of the window until you find the **Mtaper** (microstrip taper) button. Place two tapers into your schematic, referencing each the MSUB1 and having the parameters W1=0.67 mm, W2=1.5 mm, and L=1.0 mm. Rotate the mtaper element on the right-hand side of the schematic by 180 degrees.
  - l) Now select Lumped-Components from the Component Palette pull-down menu.
  - m) Place capacitors, an inductor and a resistor into the schematic to build up the circuit shown in Figure 3. When you specify the parameters for each element, **assign the variable names** (e.g., C1) instead of numerical values. Use units of pF, nH, and Ohms. (For the inductor element you can leave the R parameter unspecified.) Be certain to check the units, as ADS will often default to “none” for the units when the variable value (or name) is modified.
  - n) Place the two grounds into the schematic and use the wire tool to connect the elements and complete the schematic.
- 3) Create a schematic that references the “data circuit” (from step 1) and the “model circuit” (from step 2). Refer to Procedure #2. For clarity, you may want to use a schematic name such as cap\_main\_1p2pF.
- a) The first step is to add references to the data circuit and model circuit, and attach ports (as was done in Procedure #2).
  - b) Add the S-Parameter simulation control (selected from the Simulation-S\_Param palette list). Set the minimum frequency to 0.05 GHz, the maximum frequency to 6 GHz, and the step to 0.25 GHz.
  - c) From the Component Palette pull-down menu, select **Optim/Stat/Yield**.
  - d) The first thing to add is optimization “**Goals**.” These are the goals that ADS attempts to meet as it adjusts the variables in your equivalent circuit model to match the frequency response of the data circuit.

- i) Click on the **Goal** button on the left-hand side of the window and place a goal element in your schematic.
  - ii) Double-click on the **Goal**.
  - iii) The first “expression” that we are trying to optimize is the **magnitude** of the **vector difference** between S11 of the data circuit and S11 of the model circuit. [Note that, as shown in Figure 4, S11 of the model circuit will be treated as S33 by ADS, due to the port numbering scheme.]
  - iv) In the **Expr** (expression) box, type in **mag(S11-S33)**.
  - v) As it is possible to have multiple (S-parameter, or other) simulation control blocks in a given schematic (e.g., each may specify different frequency bands) you must indicate the simulation instance name (**SimInstanceName**, which refers to an (S-parameter) control block) to use. In this case, your only option should be **SP1**.
  - vi) You also need to specify the minimum and maximum values for the expression that will define a “successful” optimization. Since we want S11 and S33 to be equal, specify the **Min** as 0 and the **Max** as 1e-3. [Note: During optimization you will specify that a certain number of ‘trials’ are run to try to meet the expression goals. The simulator will most likely report that “simulation goals were not met” since we’ve specified a low allowable Max value. It is not likely that the optimized circuit model will perfectly match the measured data.]
  - vii) The remaining parameter to specify is the **Weight**. If you have multiple optimization goals, you can have the optimizer place more emphasis on certain goals by giving them a higher weight (usually on a scale of 0 to 1). In this case enter **1**.
  - viii) Repeat the previous steps to insert a **Goal** to optimize the magnitude of the vector differences in S21 (here specified as **S21** and **S43**). Use a weight of **1**.
- e) Now we have to add an optimization control block.
- i) Click on the Optim (Nominal Optimization) on the left-hand side of the window and place a copy of the element in your schematic.
  - ii) Double-click on the optimization control block and select the **Setup** tab. If you click on the Optimization Type pull-down menu, you will see a list of the different optimization methods that can be selected. Generally, it is best to begin with Random, perform 100 trials or so, update the schematic variables (see below), and then optimize with Gradient or other types. In this case, however, start by selecting **Gradient**. [Note: you can read more about suggested optimization approaches in the ADS on-line documentation.]
  - iii) In the OptGoal box, you specify which optimization goals are to be used. If none are selected, the default is to use all Goals in your schematic – **this will be our approach (i.e., do not explicitly specify any goals)**. If you want to see a list of available goals, click on the pull-down menu.
  - iv) At the bottom of the window you specify the stopping criterion (number of iterations that the optimizer will perform). Enter **100**.
  - v) Now click on the **Parameters** tab in the optimization control pop-up window.
  - vi) The Data to Save indicates which results of your optimization run will be stored to a file. Select **ONLY Solutions to dataset**.

- vii) The **Levels** parameter indicates the amount of information that is displayed in a simulation pop-up window during the optimization process (0-4, with more information displayed as the level increases). Enter a value of **4**.
- viii) In the **Other** field you enter parameters that control the optimization process. **Seed** is an initial value used for Random optimization; if no value is specified the optimizer will choose its own (different) value each time an optimization is performed. The **order of optimization** norm can be 2, 4, 8 or 16. The parameter is related to the error function formulation, and you can find more information about it in the on-line documentation. The **desired error** is how close we want our expression to meet the specified min/max range. In these three boxes enter **nothing** (seed), **2** (order) and **0** (error).
- ix) Now click on the **Display** tab in the optimization control pop-up window. This window lets you control which of the optimization control block specifications appear on the circuit schematic. You may choose to select all of them.

At this point you should have schematics that resemble Figure 2 through Figure 4.

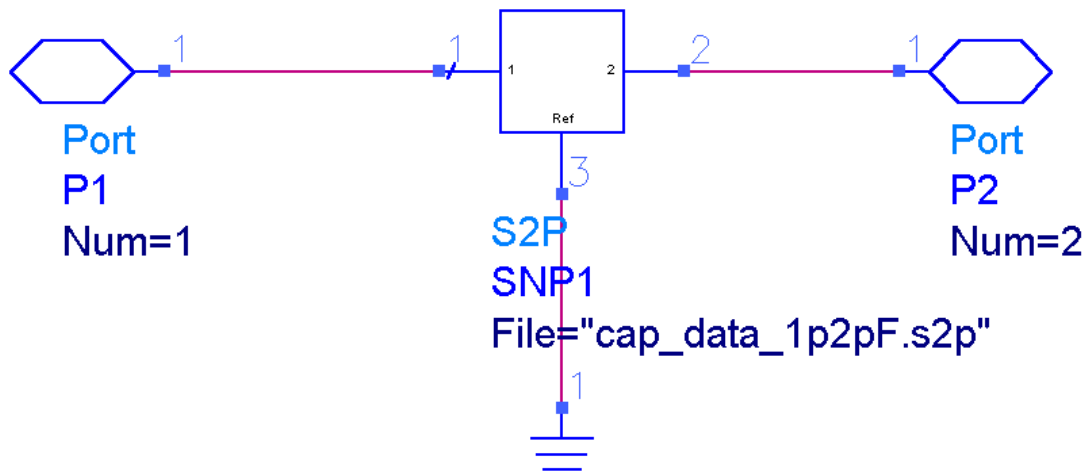


Figure 2. Schematic used to reference an s2p datafile ("data circuit").

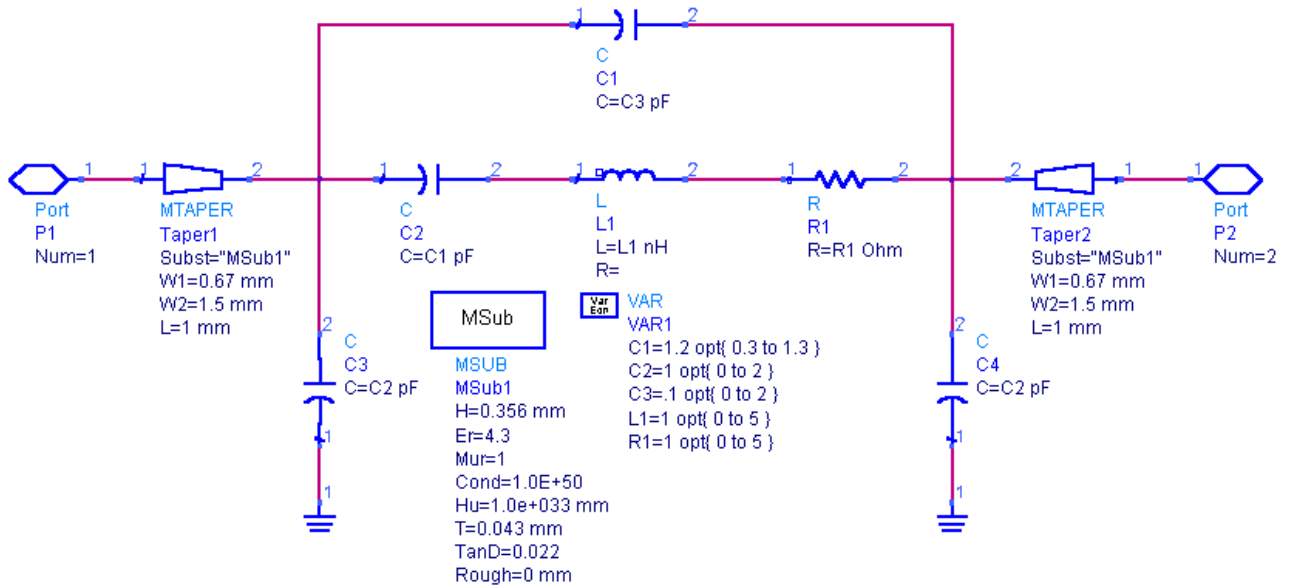


Figure 3. Schematic representing the equivalent circuit (“model circuit”). The variable values shown have no specific significance.

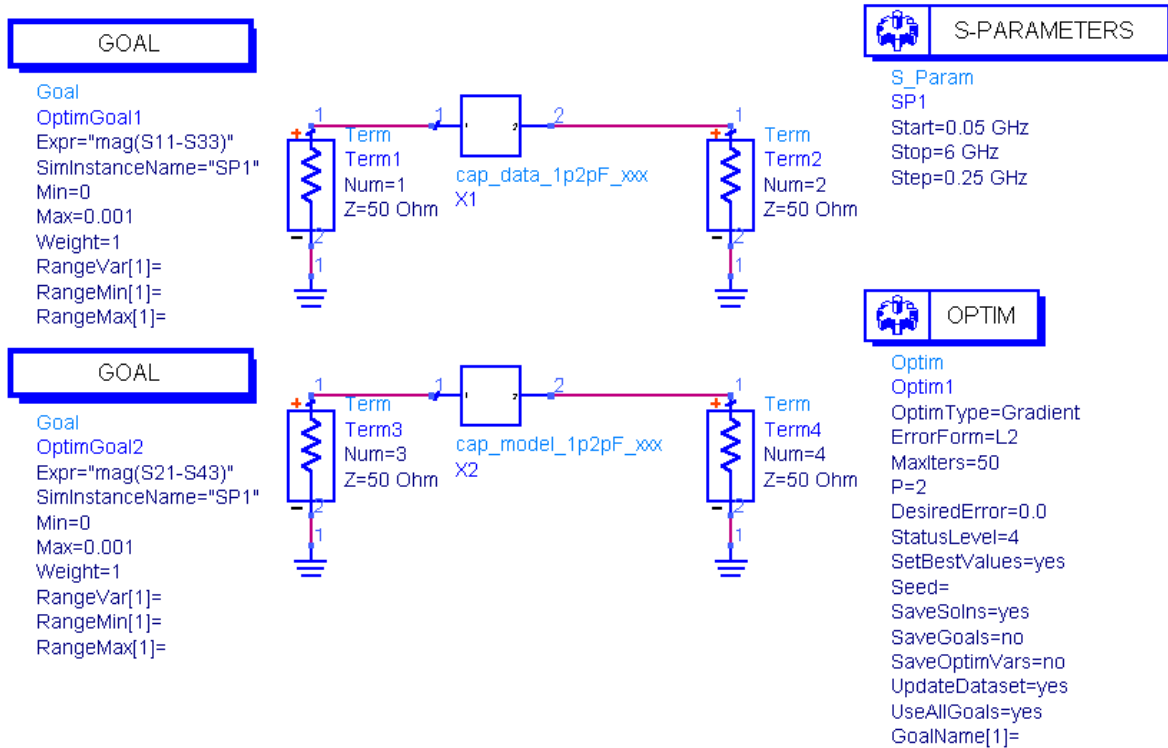


Figure 4. Schematic used to reference the data circuit (cap\_data\_1p2pF\_XXX) and model circuit (cap\_model\_1p2pF\_XXX). S-parameter and optimization simulation controls are included.

## Performing Circuit Optimization

Some introductory remarks are in order before we perform a simulation/optimization:

- ◆ If you have an optimization control block properly set up in a schematic, the optimization is automatically performed when you click on the Simulate button. If you do NOT want to perform optimization, but merely calculate and display S-parameters, you must **deactivate** the optimization block before starting the simulation. One way to deactivate an element is by selecting Edit>Component>Deactivate and then clicking the element(s) you want deactivated. The other (faster) way is to click on the Deactivate button (see Figure 5) and then click on the element.
- ◆ **VERY IMPORTANT:** After you perform an optimization run and it completes, the new parameter values (variables) calculated by ADS are only stored in memory (by default). You can display the S-parameters of the optimized “model circuit”, but the variables within the model circuit schematic themselves are not automatically changed. In order to have the new values stored into the schematic, you must click on Simulate>Update Optimization Values. After doing this, you can look into the model circuit schematic and see the new values. **IF** you perform a new simulation or optimization without updating the optimization values, ADS will resort to the last saved values in the schematic (not the new ones stored in memory). Note that this feature can be useful if you don’t happen to like the results of an optimization run, i.e., if you don’t like the appearance of the obtained frequency response of the model circuit.



Figure 5. Buttons used to activate (left) and deactivate (right) elements in a schematic.



Figure 6. Buttons used to automatically “push into” a lower level schematic (right) or “pop out of” a lower level schematic (left).

- 1) Begin by deactivating the optimization control block and perform a simple S-parameter simulation.
- 2) Open the data display and create the following graphs (note that multiple graphs can be placed in one display window):
  - a) S11, S21, S33, and S43 in dB format
  - b) S11, S21, S33, and S43 in phase (angle) format
- 3) Return to the schematic and activate the optimization control block (using the activate button, or Edit>Component>Activate). Click on the simulate button to begin the optimization
  - a) When the optimization is complete you should see multiple traces on each graph in the display window. Each trace represents the frequency response of the “data” and “model” circuits at certain iterations during the optimization. (Of course, the response of the data circuit is not changing, since it contains no optimizable parameters.) Generally, new traces are added to the graphs only when the optimizer finds an improved solution toward meeting the goals.

- b) Return to the schematic window, click on “push into the schematic” button (see Figure 6 for an illustration of this button) and then click on the cap\_model\_1p0pF\_xxx component in your schematic window. Alternatively, click on the cap\_model\_1p2pF\_xxx component and then click View>Push into Hierarchy.
  - i) If you examine the Variables block, you will see that the values have not been modified since you initially entered them.
  - ii) Now, click on Simulate>Update Optimization Values. You will see the new values appear. (Note that you can update optimization values from the higher-level schematic as well.)
  - iii) Save the modified model circuit.
  - iv) Now “pop out of” the model circuit back into the schematic with the optimization control block (click the “pop out of” button shown in Figure 6). Deactivate the optimization control block and click on the simulate button.
  - v) In the data display window you should now see only one trace for each scattering parameter. If you have followed the procedure closely, you should see a very good match between the measured scattering parameters and those of the new, optimized model circuit (see Figure 7).

**WORD OF CAUTION:** After generating an optimized equivalent circuit model, it is often the case that you will use this model as part of a larger circuit design. **IF** you reference this equivalent circuit model in another circuit and **IF** for some reason you perform optimization on the new circuit, the optimizer will try to change the parameter values that you just finished determining!

Presumably any optimization that you perform on the ‘new’ circuit will be for purposes other than improving your capacitor equivalent circuit model. However, when using hierarchical designs, it is easy to forget that ‘optimizable’ variables may be embedded in lower level schematics.

A safe process to follow is to save your optimized equivalent circuit to a new schematic name, such as model\_fixedvalue. In model\_fixedvalue, you edit the Variable block and disable optimization on all of the variables.

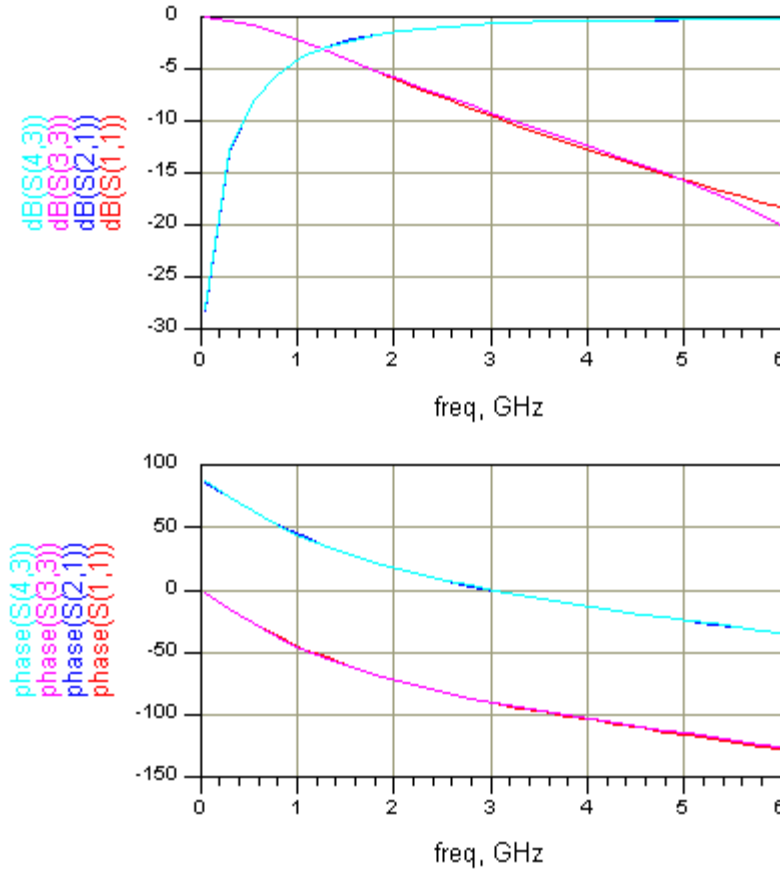


Figure 7. Comparison between measured data (S11, S21) and the optimized model circuit (S33, S43).

### Some Parting Comments

Let's consider what the simulator is required to do during optimization. Essentially, we are creating an N-dimensional space, where each dimension corresponds to one of the optimizable parameters. Within the N-D space, there are one or more solutions to the problem (i.e., meeting the optimization goal). There may also be, and probably are, several local minima or “near solutions”. The number of solutions and local minima clearly will increase as the number of dimensions increases and as the allowable range of each of the optimizable parameters increases. In other words, unless you are lucky, you cannot dive blindly into circuit optimization and hope for an easy solution!

Some basic guidelines that you can follow to help with circuit optimization are as follows:

a - Limit the number of optimizable parameters as much as possible. In some cases, circuits can also be built up slowly; start with a sub-network of your overall design and slowly add elements as the values (or reasonable ranges of values) for the preliminary elements are determined.

b- Try to reasonable starting values for all parameters. A little bit of homework up-front should usually allow you to determine an upper and lower limit for each element. **If the optimizer tends to push the value of a particular element toward either limit that you specified, simply increase the range.**

c- Be selective in the type of optimization you use. There are several different types of routines that can be used: Random, Random Min/Max, Gradient, Gradient Min/Max, etc. The random type is the fastest. I usually begin with this and hope that a decent set of starting values can be found. Then I proceed with Gradient or Gradient Min/Max. These are slower optimizers, but if they lock onto a solution they will converge very quickly. In a large N-space, the



gradient routines will often get caught in local valleys; if it is apparent this is not the solution you are looking for, you might choose to revert to Random to help “escape the local valley”. I realize this is very general/vague information, but optimization can be somewhat of an art and you really have to get the feel for each particular circuit.