™

Empipe™

User's Manual

Optimization Systems Associates Inc.

Empipe™

User's Manual

Version 4.0

July 1997

Optimization Systems Associates Inc.

LIABILITY AND WARRANTY

Copyright

Empipe User's Manual Version 4.0 first published in 1997. Printed in Canada.

Optimization Systems Associates Inc.
P.O. Box 8083, Dundas, Ontario
Canada L9H 5E7

Tel 905 628 8228
Fax 905 628 8225

Trademarks of Optimization Systems Associates Inc.

Empipe
OSA90
OSA90/hope
Datapipe
Geometry Capture
Space Mapping
Emtrack
HarPE

Other Trademarks

em, *xgeom* and Sonnet are registered trademarks of Sonnet Software, Inc.
UNIX is a registered trademark of AT&T.
X-Windows is a trademark of the Massachusetts Institute of Technology.
SPARCstation, SunOS and Solaris are trademarks of Sun Microsystems, Inc.
HP and HP-UX are registered trademarks of Hewlett-Packard Company.
PostScript is a trademark of Adobe Systems Inc.

Empipe™ User's Manual

Table of Contents

6 Tutorial: A Resistor

7 Tutorial: An MIM Capacitor

8 Empipe Geometry Capture

9 OSA90 Environment

10 Empipe Database

11 Response Interpolation

12 Empipe Library Elements

Index

1

Installation

1

Installation

1.1 Introduction

The installation of Empipe can be done by either a system administrator or a user.

In a multi-user environment, the system administrator should install Empipe so that it is accessible to all users. If, on the other hand, Empipe is to be used by a single user, then that user can carry out the installation himself/herself.

Sections 1.2 and 1.3 describe the procedure of loading the software. Only the person who performs the installation needs to follow this procedure.

However, *every* user of Empipe must follow the instructions provided in Sections 1.4 and 1.5 regarding setting UNIX environment variables and copying the tutorial examples.

Empipe drives the electromagnetic simulator ***em*** from Sonnet Software, Inc. You must have an active Sonnet installation on the same computer or accessible through a computer network.

If you experience difficulties in the installation of Empipe, you can contact our technical support staff at

Tel 905 628 8228
Fax 905 628 8225

email osa@osacad.com
Home page http://www.osacad.com

1.2 Creating the OSA Installation Directory

First, you must allocate a directory in which the Empipe program and other associated files will be stored. We will refer to this directory as the OSA Installation Directory.

If this is an upgrade to your current Empipe installation or if you are adding Empipe to an existing installation of other CAD software products from OSA, then the OSA Installation Directory should already exist on your computer. You should find that directory and install Empipe within it.

Otherwise, you can create the OSA Installation Directory using the UNIX command:

`mkdir` *<OSA Installation Directory>*

If you are a system administrator installing Empipe for multiple users, you should create the OSA Installation Directory from the root or another publicly accessible file system. For example, *<OSA Installation Directory>* can be "`/osa_home`" or "`/usr/osa_home`".

System Administrator: please make sure that the OSA Installation Directory has the "read" and "execute" permission for all Empipe users.

If you install Empipe from your user account, then the OSA Installation Directory will be created within your home directory. In this case, access to Empipe is by default restricted to yourself.

1.3 Loading the Software

Change directory to the OSA Installation Directory (described in Section 1.2):

cd *<OSA Installation Directory>*

Then follow the instructions for the specific media in your case.

Sun SPARCstations SunOS 4.x, Floppy Disks

```
bar xvpfZ /dev/rfd0c
```

Sun SPARCstations SunOS 4.x, Tape

```
tar xvpf /dev/rst0
```

Sun SPARCstations Solaris 2.x, Floppy Disks

For Solaris 2.2 and higher, you need to disable the volume manager first: Log in as root and type

```
/etc/init.d/volmgt stop
```

Then load the software in the OSA Installation Directory:

```
cpio -id -H bar -I /dev/rdiskette
```

After the loading is finished, eject the floppy disk:

```
eject floppy
```

For Solaris 2.2 and higher, you may restart the volume manager (as root):

```
/etc/init.d/volmgt start
```

Sun SPARCstations Solaris 2.x, Tape

```
tar xvpf /dev/rmt/0m
```

HP 700 Series, DAT Tape

```
tar xvpf /dev/rmt/0m
```

Listing the Files Loaded

You can visually inspect the files loaded by typing

```
ls -la
```

which lists the files in the OSA Installation Directory.

You should be able to see the following subdirectory names listed:

```
bin
empipe_examples
osa90msg
```

There may also be other files if you have other OSA products.

You can also list the files in the "bin" subdirectory:

```
ls -la bin
```

You should see the following executable files among the listing:

```
dat2dbs
dbs2dat
empipe
emtrack
osa90
```

License File

The authorization of using Empipe is controlled by an encoded license file. Usually the appropriate license file is included in the package supplied to you.

Under some circumstances, such as when the software expiry date is being extended, you may be required to enter a new key in the license file.

In this case, change directory to the OSA Installation Directory, then type

```
cd osa90msg
mv license.osa license.osa.old
vi license.osa
```

Type "i" to enter the "insert" mode; type the key code *exactly* as provided; press <Esc> to exit the "insert" mode; type ":wq" to write to the file and quit the editor.

1.4 Setting UNIX Environment Variables

Every user of Empipe must follow the instructions in this Section before attempting to run Empipe for the first time.

You must define a UNIX environment variable OSA_DIR to point to the OSA Installation Directory. If you do not know the name of that directory, please ask the person who carried out the software installation.

You should also modify the PATH environment variable so that you can execute Empipe from your own directory.

The UNIX environment variable DISPLAY must also be properly defined. For this you need to know the host name of your computer. To obtain the host name, type

```
uname -n
```

Instructions for csh Users

Include the following lines in the .login file in your home directory:

```
setenv OSA_DIR <OSA Installation Directory>
setenv DISPLAY host_name:0
set path=($path $OSA_DIR/bin)
```

Instructions for sh and ksh Users

Include the following lines in the .profile file in your home directory (HP VUE users should include these lines in the .vueprofile file instead):

```
OSA_DIR=<OSA Installation Directory>;  export OSA_DIR
DISPLAY=host_name:0;  export DISPLAY
PATH=$PATH:$OSA_DIR/bin;  export PATH
```

Effecting the Environment Changes

After completing the changes to the environment variables, you need to log out and then log in again in order for the changes to take effect.

1.5 Copying the Tutorial Examples

A set of Empipe examples are provided to you in the subdirectory

<*OSA Installation Directory*>`/empipe_examples`

Chapters 3 to 7 of this Manual utilizes these examples in a series of tutorials to help you learn the features of Empipe. You are encouraged to follow the tutorials step by step on the computer.

It is not advisable to practice with the original set of the examples for they may get inadvertently altered. You should make a copy of the examples.

First, create a suitable subdirectory within your own home directory. You can use the UNIX command:

`mkdir` <*example directory*>

where <*example directory*> is an arbitrary name, such as "`empipe_examples`".

Then change directory to <*example directory*> by typing

`cd` <*example directory*>

and copy the Empipe examples by typing

`cp` <*OSA Installation Directory*>`/empipe_examples/* .`

The total size of all the examples for Empipe Version 4.0 is about 100 kilobytes.

2

Technical Overview

2.

Technical Overview

Empipe is a powerful and friendly software system for automated electromagnetic (EM) design optimization, driving the EM simulator ***em***.

em is a full-wave EM field simulator from Sonnet Software, Inc., for predominantly planar circuits. ***em*** accommodates arbitrary geometries and accounts for dispersion, coupling, surface waves, radiation, metallization and dielectric losses. It is highly recommended for its accuracy and speed (see, e.g., D.G. Swanson, Jr., "Simulating EM fields," *IEEE Spectrum*, November 1991, pp. 34-37, and "CAD benchmark: electromagnetic simulators", *Microwave Engineering Europe*, November 1994, pp. 11-20).

Empipe allows you to designate geometrical and material parameters as candidate variables for optimization in an intuitive and friendly manner. You introduce a set of incremental changes using ***xgeom*** (the same graphical editor you use to set up an ***em*** analysis). The information is then processed by Empipe to parameterize the structure.

To put it simply, any arbitrary structures that you can simulate using ***em*** you can now optimize using Empipe!

Empipe employs the sophisticated optimizers in OSA90, a program supplied as part of the Empipe package. OSA90 offers a comprehensive set of optimizers, including ℓ_1, ℓ_2 (least squares), minimax, Huber, quasi-Newton, conjugate gradient, simplex, simulated annealing and random algorithms, with proven track records in engineering applications. OSA90 also offers a wealth of options for mathematical expressions, response postprocessing, statistical analysis and yield optimization, as well as many graphical display and visualization formats.

Fig. 2.1 illustrates the data flow of the Empipe software system.

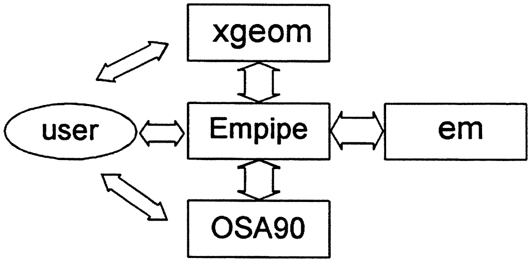


Fig. 2.1 Data flow schematic of the Empipe system.

Functional Relationship Between Empipe and OSA90

The Empipe package includes Empipe and OSA90 as two separate executable programs. Their functional relationship is as follows.

As a user, you start the operation with Empipe. You define the structure to be optimized as an Empipe element. Once the definition of parameters, variables and specifications is completed, you click on a button and OSA90 is invoked. You interact with the OSA90 environment to carry out simulation and optimization. Whenever necessary, OSA90 will call Empipe to obtain *em* analysis results. After you are done with the optimization, you exit from OSA90 and return to Empipe.

A series of tutorials is provided in Chapters 3 to 7 to illustrate the various features of Empipe. The tutorials are organized in a progress order, each one introducing something new. We highly recommend that you follow all the tutorials.

Automatic and Intelligent Discretization and Interpolation

em requires discretization of geometrical parameters with respect to a predefined grid (mesh). Empipe automatically handles the discretization so that when *em* is invoked all the geometrical coordinates are on the grid. Response interpolation is employed to produce the results for off-grid points. This also facilitates the efficient calculation of gradients for optimization.

The user can select from a number of schemes of interpolation, including linear and quadratic interpolations, based on *S*, *Y* or *Z* parameters.

Empipe Database

The results from all *em* analyses are saved in Empipe databases. This eliminates duplicated *em* simulation and, when combined with interpolation, are particularly valuable for making EM-based statistical analysis and yield optimization practical.

Flexibility in Analyzing Large Structures

Empipe gives you flexibility in analyzing large, complicated structures. You can decompose a large structure into several substructures, individually simulated by *em* and then connected via circuit theory. This may produce less accurate results than analyzing the whole structure as one piece if parasitic couplings exist between the substructures, but it can significantly reduce the CPU time needed for the *em* analysis.

Preprogrammed Empipe Library

Empipe provides a library of preprogrammed microstrip structures, including

asymmetrical gap
bend
mitered microstrip bend
microstrip coupled line filter
cross junction
interdigital capacitor (two models)
double patch capacitor
double stub (two models)
folded double stub (two models)
symmetrical gap
microstrip line
overlay double patch capacitor (four models)
open stub
rectangular structure
spiral inductor
step junction
T-junction

Optional Equivalent Circuit Model Library

The Empipe software system can be upgraded with an optional OSA90/hope module. It offers a comprehensive library of equivalent circuit models, including lumped elements, controlled sources, transmission lines and empirical models for microstrip components.

With this option, you can freely mix within the same circuit *em*-simulated structures with equivalent circuit elements. You can use optimization to automate calibration, refinement and new development of models for novel structures.

Nonlinear Harmonic Balance Simulation and Optimization

The optional OSA90/hope upgrade also provides a wealth of nonlinear modeling, simulation and optimization capabilities. It includes a library of nonlinear device models for diodes, FETs, bipolar transistors, HEMTs and user-definable nonlinear controlled sources.

You can perform small-signal, DC and large-signal AC simulation. You can combine EM analysis with harmonic balance analysis to accurately simulate and optimize nonlinear circuits.

3

Tutorial: A Microstrip Line

3

Tutorial: A Microstrip Line

3.1 Introduction

This tutorial chapter is intended to help you learn to use Empipe by going through a simple example step by step. Additional tutorials are provided in Chapters 4 through 7, gradually and systematically introducing you to the various features of the program.

Before we proceed, please make sure that the following steps as described in Chapter 1 have been carried out:

1 Empipe has been properly installed.

2 You have set up the environment variables OSA_DIR, PATH and DISPLAY in your own user account.

3 You have made a copy of the Empipe examples in your own user account.

What You Will Learn From This Tutorial

1 The basics of the user interface of Empipe.

2 How to parameterize a geometrical dimension as an optimization variable.

3 How to define a performance specification on the *S* parameters.

4 How to start an ***em*** simulation through Empipe.

5 How to start an optimization.

6 How to save the optimized geometry.

This introductory tutorial is not at all time consuming. Each ***em*** analysis takes less than 1 second on a SPARCstation 10. The optimization takes less than 10 seconds in total.

When this symbol appears on the left-hand side column, it highlights text that describes hands-on actions. You can take a "short-cut" through the tutorial by following this symbol and skip over the commentaries.

Description of the Example

For this introductory tutorial, we consider a very simple example: a microstrip line, as illustrated in Fig. 3.1.

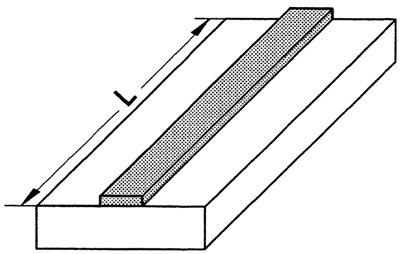


Fig. 3.1 The microstrip line example.

It is nominally a 50 Ω line. The substrate is 25 mil thick, with a dielectric constant of 9.8. The line width is 25 mils. The metallization is assumed to be lossless.

The objectives of our exercise include:

1 Define the length of the line as a designable parameter, identified by the name *L*.

2 Define the line as a two-port, with a port defined at each end of the line.

3 Optimize the parameter *L* to satisfy the specification

PS21 = -120 at frequency = 10 GHz

where PS21 represents the phase of S_{21} in degrees.

Are You in the Right Subdirectory?

All the operations in this tutorial should take place within the subdirectory which contains your copy of the Empipe examples.

If you are not already in that subdirectory, type

cd *<Your Empipe Example Subdirectory>*

3.2 Creating the Geo Files

Structures to be simulated by ***em*** must be described by a ".geo" file. Sonnet Software, Inc. provides the ***xgeom*** program to let you draw the geometry graphically. A utility program to convert GDSII files to ".geo" files is also available from Sonnet Software, Inc.

A ".geo" file contains the description of the geometry by means of a set of coordinates. It also contains other relevant information, such as substrate data and port definitions.

We assume that you already know how to create a ".geo" file to represent a given structure.

For the microstrip line tutorial example, we have created a ".geo" file, under the name em_line0.geo, which is included in your copy of the Empipe example files.

You can use ***xgeom*** to view the ".geo" file by typing

```
xgeom em_line0.geo
```

The ***xgeom*** display is depicted in Fig. 3.2.

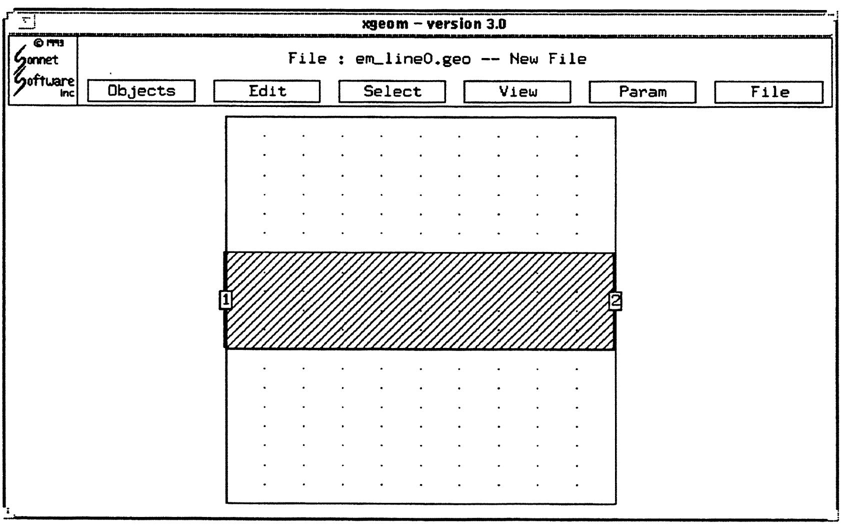


Fig. 3.2 The microstrip line described in em_line0.geo.

Creating a Geo File to Represent an Incremental Change

In the ".geo" file shown in Fig. 3.2, we set the microstrip line length to 100 mils. We call it the nominal value of the length parameter. The nominal value of a parameter may come from an empirical design or it can simply be a reasonable and convenient value.

For Empipe to capture the microstrip line length as a parameter, we need to create an additional ".geo" file to represent an incremental change. For instance, we may change the length from its nominal value of 100 mils to 110 mils, and create a new geometry to correspond to the new line length. The name for the new ".geo" file is em_line1.geo.

Fig. 3.3 illustrates the geometries before and after making the incremental change.

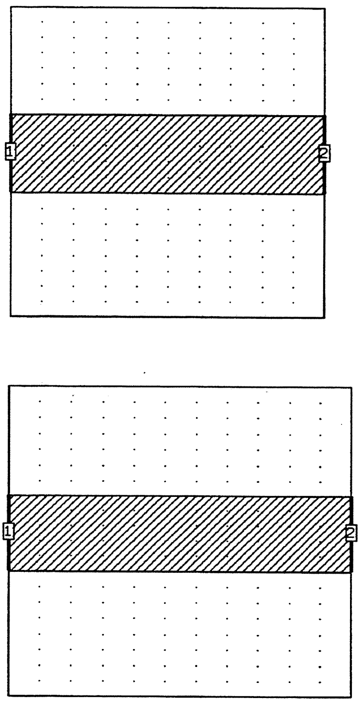


Fig. 3.3 Geometries for L=100 mils and L=110 mils, respectively.

By comparing the two ".geo" files, Empipe automatically extracts the information necessary for translating a given parameter value to the corresponding geometrical coordinates.

Steps of Editing the Geometry

In the following, we show the steps involved in creating the file em_line1.geo which represents an incremental change in the microstrip line length.

It is not necessary for you to actually carry out these steps. You can gain an idea of the process by simply reading the descriptions.

1 Invoke xgeom, starting with the nominal ".geo" file:

```
xgeom em_line0.geo
```

2 Noting that the length of the substrate box is the same as the line length, we first enlarge the box size. Use this sequence of ***xgeom*** menu options:

```
Param→Box→Area→Keyboard
```

Then, select to keep constant the Cell Size. Next you are prompted to enter the new X dimension of substrate in mils. Enter 110 (same as the new line length). At the next prompt for the Y dimension, keep the present value by simply pressing <Enter>. Click on the button Done to finish the modification of the box size.

3 You should be able to see that one end of the line (port 2) is now inside the box. We need to move this end to the edge of the box. Click the left-hand mouse button on one of the vertices; then click and hold the left mouse button on the other vertex and drag it to the edge of the box; release the mouse button. The line length is thus increased from 100 mils to 110 mils.

4 Save the modified file under a new name. *Do not actually carry out this step* so that you will not overwrite the file supplied with the Empipe examples. The ***xgeom*** menu options

```
File→Save→Another Name
```

allows you to save a file under the new name em_line1.geo.

Rules for Making Incremental Changes

- The cell size must remain the same as in the nominal ".geo" file.
- The incremental change must be a multiple of the grid size, i.e., the geometry must remain on the grid after the change.
- You cannot add or delete polygons.
- The basic shape of the polygons must remain the same (e.g., you cannot change a polygon from a rectangle to a triangle).

These rules are specified in more precise details in Chapter 8.

3.3 Geometry Capture Form Editor

This section shows how to use Empipe to capture the microstrip line length parameter from the geometries described by the files em_line0.geo and em_line1.geo.

Type

```
empipe em_line
```

where em_line is the name for the Empipe element to be created.

The Empipe Geometry Capture form editor is displayed, as depicted in Fig. 3.4.

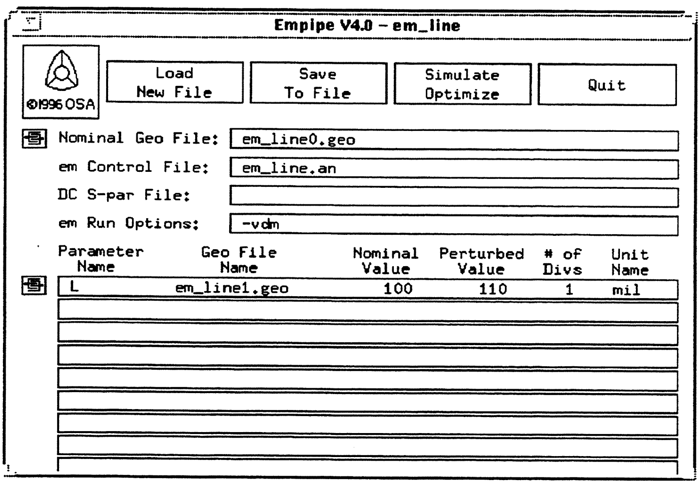


Fig. 3.4 Empipe Geometry Capture form editor.

For this tutorial example, the data is already set up for you. Otherwise, if we were starting a new project from scratch, the form editor would have mostly blank fields. Now we will look at the meaning of each of the entries.

Nominal Geo File

This entry identifies the ".geo" file which represents the structure with all the parameters set to their nominal values.

For the microstrip line tutorial example, this entry should be the file name em_line0.geo.

For a new project, you will see a blank entry box like this:

```
Nominal Geo File: [                                        ]
```

Click the left-hand mouse button on the box, and then type in the appropriate file name.


This button appears adjacent to the entry box. If you click on it, ***xgeom*** will be invoked to display the nominal ".geo" file.

em Control File

This entry identifies the ***em*** analysis control file. An analysis control file is required by ***em***. It defines the frequencies for the ***em*** analysis.

An analysis control file has been created for you and supplied with the microstrip line tutorial example, under the name em_line.an. Its contents are as follows

```
VER 3.0
GHZ
FRE 10
```

It defines a single frequency of 10 GHz for the ***em*** analysis.

For a new project, you need to prepare an appropriate analysis control file in accordance with the syntax described in the Sonnet User's Manual. Then you click the left-hand mouse button on the entry box and fill in the name of the analysis control file.

DC S-Parameter File

This field allows you to supply a separate *S*-parameter file for DC. This is necessary only if the Empipe element is to become part of a nonlinear circuit for harmonic balance simulation. It does not apply to our example and should be simply left blank.

em Run Options

This field allows you to specify the run-time options for ***em***. The default options are -vdm (verbose, automatic de-embedding and memory saver).

Parameter Definition Data

The entries under the heading

```
Parameter   Geo File   Nominal   Perturbed   # of   Unit
  Name        Name      Value      Value     Divs   Name
```

contain the data necessary for defining the parameters. Each line defines one parameter.

For the microstrip line tutorial example, we have just one parameter, namely the line length. The entry that defines this parameter is

```
L    em_line1.geo    100    110    1    mil
```

The meaning of the items is as follows.

Parameter Name is an arbitrary ASCII string of no more than 32 characters. We choose the name "L" to identify the microstrip line length parameter.

Geo File Name identifies the ".geo" file which describes the geometry of the structure after an incremental change in the parameter value is made. The file that represents an incremental change in the microstrip line length parameter is em_line1.geo.

Nominal Value refers to the nominal value of the parameter which is consistent with the nominal ".geo" file. For the microstrip line length parameter, this value is 100 (it should be entered as a plain number, since the physical unit is entered as a separate item).

Perturbed Value refers to the parameter value after the incremental change. For the microstrip line length parameter, this value is 110.

Number of Divs measures the incremental change in terms of the ***em*** grid size. It is obtained by dividing the difference between the nominal value and the perturbed value by the ***em*** cell size along the appropriate dimension. For the microstrip line length parameter, the incremental change is 10 mils. Since the geometrical change is along the X dimension of the layout and the grid size for the X dimension is 10 mils, the number of divisions is 1.

If we had changed the line length from 100 mils to 120 mils instead of 110 mils, then the number of divisions would be 2 instead of 1.

This information is needed to ensure that the geometry Empipe generates for ***em*** analysis is always on the grid (interpolation is used for off-grid points).

Unit Name identifies the physical unit of the parameter. Permissible unit names include IN (inch), MIL (milli-inch), M (meter), CM (centimeter), MM (millimeter), UM (micron) and NONE (without unit). The unit name for the line length parameter is MIL.

Suppose that you had started from scratch and had to enter this data on a blank line. You would first click the left-hand mouse button on the box and then type

```
L em_line1.geo 100 110 1 mil
```

Note that you do not have to manually line up the text with the heading. When you have completed the entry, press the <Enter> key and the text will be automatically formatted to align with the heading.

This button appears on each line that has a parameter defined. If you click on it, ***xgeom*** will be invoked to display the ".geo" file specified on that line.

What's Next?

At this point, Empipe has been given enough information to define the microstrip line structure with an adjustable length parameter. The following section shows how to select the length parameter as an optimization variable and to define a performance specification.

3.4 Defining Variables and Specifications

In the Empipe form editor window, click on the button

Simulate
Optimize

Selecting Optimization Variables

Two new windows appear on the screen, entitled "Empipe Select Variables" and "Empipe Specifications", respectively.

The initial appearance of the "Select Variables" window is depicted in Fig. 3.5. Listed in the window is the parameter "L" defined for the microstrip line example.

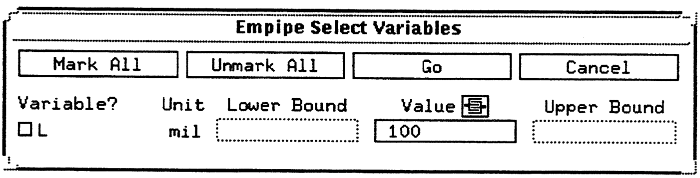


Fig. 3.5 The "Select Variables" window.

Click on the small square box next to the parameter name "L", a check mark will appear in the box to indicate that the parameter L has been selected as an optimization variable, as illustrated in Fig. 3.6.

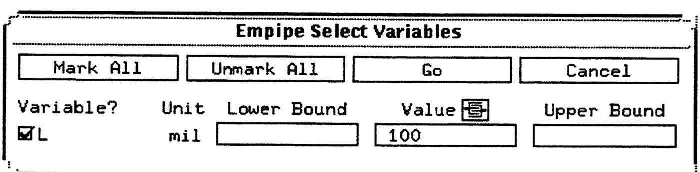


Fig. 3.6 The parameter L is selected as a variable.

In a general case, all parameters defined through Empipe are candidates for optimization variables. You can select an individual parameter as a variable by clicking on the check mark box associated with that parameter. Alternatively, you can click on the button labelled <Mark All> to select all the parameters as variables.

You can undo the selection of a parameter as a variable by clicking on the check mark box again. You can also use the <Unmark All> button.

Value

The current value for each parameter is shown under the heading "Value". In optimization, this value is used as the starting point. By default, it is set to the parameter's nominal value. If you wish to change it, click on the entry box under "Value" and enter the desired value. You can also click on the button to view the geometry of the current point.

Bounds on the Variables

You can specify upper and/or lower bound to limit the parameter value range during optimization. Setting suitable bounds on all the variables is advisable since this prevents the optimizer from changing the structure beyond what can be realized physically.

If the bounds are not given explicitly, they will be determined by the program according to the starting point. Suppose that the starting point of a selected variable is x.

If the lower bound is not given explicitly, it is set to 0 if $x \geq 0$, or $-\infty$ if $x < 0$.

If the upper bound is not given explicitly, it is set to $+\infty$ if $x \geq 0$, or 0 if $x < 0$.

For the tutorial example, we leave all the entries at their default setting, which means that the starting point for the parameter L is 100 (mils), the lower bound is 0 and the upper bound is $+\infty$.

Specifications for Optimization

The initial appearance of the "Specifications" window is depicted in Fig. 3.7.

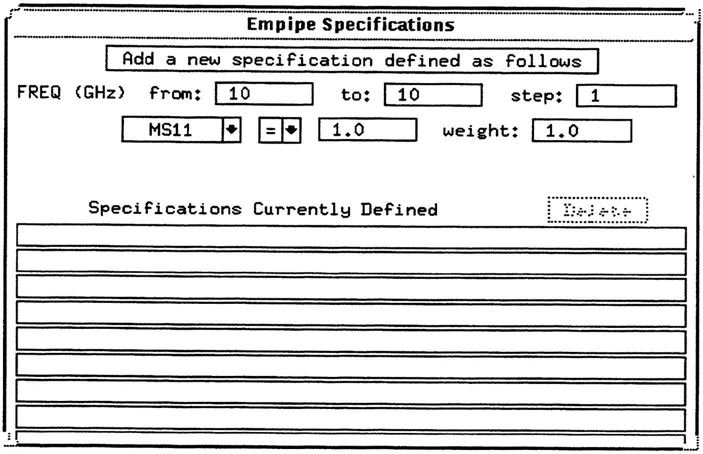


Fig. 3.7 The "Specifications" window.

In general, the definition of a specification involves the following steps:

1 Select a frequency range.

2 Select an *S*-parameter response.

3 Select a specification type (upper, lower or equality specification).

4 Enter a numerical value as the goal.

5 Optionally, enter a weighting factor.

Recall that in Section 3.1 the specification for this tutorial example is stated as

PS21 = -120 at frequency = 10 GHz

where PS21 represents the phase of S_{21} in degrees.

Selecting a Frequency Range

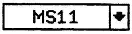
The line that appears as

allows you to select a frequency range from those defined in the *em* analysis control file.

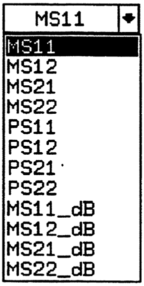
For the tutorial example, we have defined only a single frequency in the analysis control file, namely 10 GHz, which is automatically selected. You do not need to change anything.

Selecting an S-Parameter Response

This is the entry box for selecting an *S*-parameter response:



Click on the arrow and you will see the list of available responses:



The label MS*ij* represents the magnitude of S_{ij}, PS*ij* represents the phase of S_{ij} in degrees, and MS*ij*_dB represents the magnitude of S_{ij} in decibels.

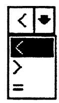
To select a response from the list, simply click on it.

Click on the label PS21.

Selecting the Type of Specification

The entry box for selecting the type of specification appears as

Click on the arrow and you will see the list of available specification types:



The symbols "<", ">" and "=" represent upper, lower and equality specifications, respectively.

Click on the symbol "=".

Entering a Numerical Value as the Goal

The third field on the specification line represents the goal. For the tutorial example, the goal for the response PS21 is -120.

Click on the box and then type in the number -120.

While entering the number, you may use the cursor keys, the <Back Space> key and the <Delete> key to edit the text, if necessary.

Optional Weighting Factor

You can enter an optional weighting factor. The default value is 1. If given, the weighting factor must be a positive number.

For the tutorial example, leave the weighting factor at the default value of 1.

Completing the Definition of a Specification

Click on the button

Add a new specification defined as follows

The specification we have just defined will be recorded under the heading "Specifications Currently Defined". The "Specifications" window at this point is shown in Fig. 3.8.

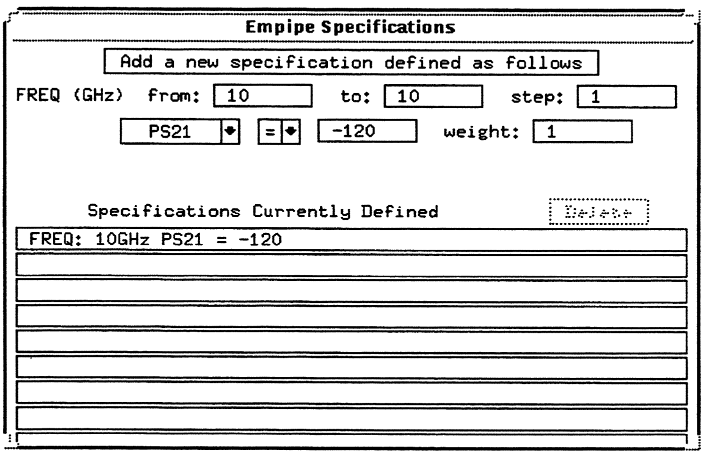


Fig. 3.8 The specification defined for the microstrip line tutorial.

What's Next

We are ready to start EM simulation and optimization. Go to the "Select Variables" window and click on the button

Go

3.5 Simulation

After you click on the "Go" button, Empipe invokes the OSA90 simulation/optimization environment. The OSA90 window is depicted in Fig. 3.9.

```
                                  OSA90
File Parsing Completed                    | OSA      Fri Sep  6 15:49:07 1996
                                          | /osa/empipe_examples/em_line.ckt

! EM optimization of user-defined structure: EM_LINE

Model
#include "em_line.inc";

   EM_LINE_L: ?100?;

   EM_LINE  1 2 0
      L=(EM_LINE_L * 1mil);

   PORTS 1 0 2 0;

   CIRCUIT;

   MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);
end

Sweep
   AC: FREQ: 10GHz
       MS MS_DB PS
      {XSWEEP Title="PS21 and Spec"
       Y=PS21
       XMIN=9GHz XMAX=11GHz NXTICKS=2 X_title=FREQ
       SPEC=(at 10GHz, = -120)};
end

Spec
   AC: FREQ: 10GHz PS21 = -120;
end

Control
   Perturbation_Scale=1.0e-4;
end

File: reads, edits, parses and saves files                          <F1> Help
OSA90> File Display Optimize Macro Sensitivity monteCarlo Learn
```

Fig. 3.9 OSA90 window.

At the top of the window is the message area, where the date, time and file name are displayed. At the bottom of the window is the menu area, where the menu options are presented. In the middle is the text area, where the input file (netlist) is shown.

OSA90 Input File (Netlist)

The OSA90 input file (netlist) consists of a number of "blocks". Each block begins with a block identifier, such as "Model", "Sweep", "Spec" and "Control", and ends with the keyword "end". The Model block describes the circuit, the Sweep block selects the simulation outputs, the Spec block defines the specifications for optimization, and the Control block contains operation options (the input file structure is further discussed in Chapter 9).

In the Model block, notice the statement

```
EM_LINE_L: ?100?;
```

The label EM_LINE_L identifies the parameter L of the element EM_LINE. The pair of question marks denotes an optimization variable, and the value between the question marks is the starting point.

OSA90 Menu Options

At the bottom of the OSA90 window is a list of the menu options:

```
File: reads, edits, parses and saves files                     <F1> Help
OSA90> File Display Optimize Macro Sensitivity monteCarlo Learn
```

As you move the cursor through the menu options, the line immediately above the menu shows a brief description of the option. To activate a menu option, move the cursor to highlight the option and then click the left-hand mouse button or press the <Enter> key.

Starting EM Simulation

First, we perform an EM simulation and check the response against the specification.

Click on the menu option Display.

Empipe will invoke *em* in the background. For this simple structure, the *em* analysis should be completed very quickly (less than one second on a SPARCstation 10).

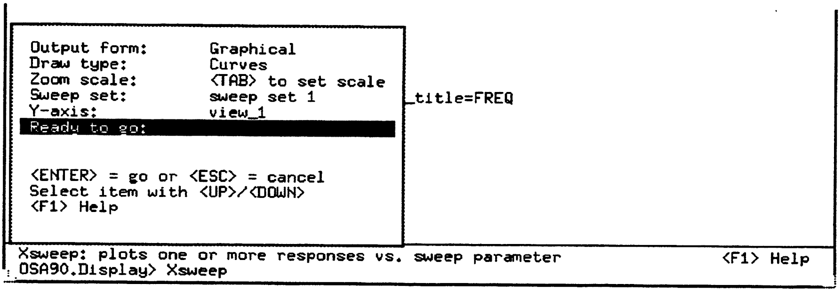
When the *em* analysis is finished, the menu at the bottom of the window will change to

```
Xsweep: plots one or more responses vs. sweep parameter   <F1> Help
OSA90.Display> Xsweep Parametric Array Waveform Smith Visual
```

where the prefix OSA90.Display> indicates that this is the sub-menu under the main menu option Display, and Xsweep, Parametric, ..., represent the various options for displaying the simulation results.

Displaying the Response PS21

Click on the display menu option **Xsweep**. A pop-up window appears, showing the default setting for the display options.



Simply press <Enter> to accept the default setting. The display is illustrated in Fig. 3.10.

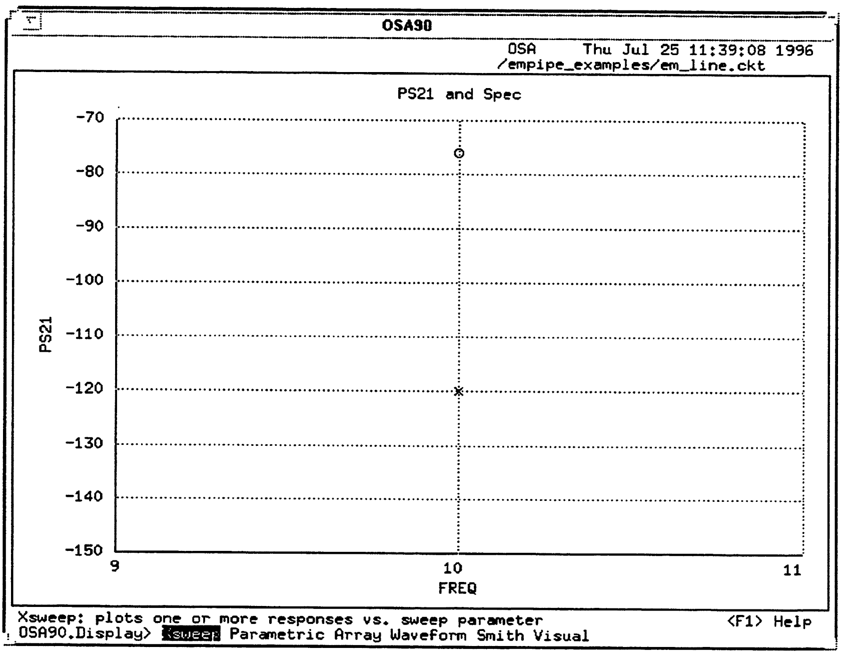


Fig. 3.10 The PS21 response before optimization.

3.6 Optimization

The *em* simulation result shows that the design specification of PS21 = -120 is not satisfied. In order to meet the specification, we need to perform optimization.

Exit from the display menu by pressing the <Esc> key. You should be back at the main menu

```
File: reads, edits, parses and saves files                    <F1> Help
OSA90> File Display Optimize Macro Sensitivity monteCarlo Learn
```

To start optimization, click on the menu option Optimize. A pop-up window appears:

```
PS2 -110

  Optimizer:                L1
  objective Function:       L1
  Number of iterations:     30
  Accuracy of solution:     0.0001
  Display option:           every iteration
  Ready to go:

  <ENTER> = go or <ESC> = cancel
  Select item with <UP>/<DOWN>
  <F1> Help
                                                              11
Optimize: optimizes variables according to specifications     <F1> Help
OSA90>                  Optimize
```

The pop-up window shows a number of options, such as the choice of optimizer, the maximum number of iterations, etc.

Press the <Enter> key to accept the default setting.

During the optimization, *em* is invoked in the background whenever a new EM analysis is required. On the screen, the progress of optimization is reported:

```
Iteration   1/30 L1 Objective=43.67
Iteration   2/30 L1 Objective=42.8941
Iteration   3/30 L1 Objective=41.3189
Iteration   4/30 L1 Objective=38.0724
Iteration   5/30 L1 Objective=31.2506
Iteration   6/30 L1 Objective=15.9797
Iteration   7/30 L1 Objective=1.39943
Iteration   8/30 L1 Objective=0.00756073
Iteration   9/30 L1 Objective=1.52588e-05
Solution  L1 Objective=1.52588e-05
```

Simulation of the Optimized Microstrip Line

Click on the menu option Display. Empipe starts the simulation of the optimized microstrip line. When it is finished, you will see the OSA90.Display menu. Click on the menu option Xsweep. After the pop-up window appears, press <Enter> to accept the default setting.

The display is depicted in Fig. 3.11. It shows that the specification on PS21 is met.

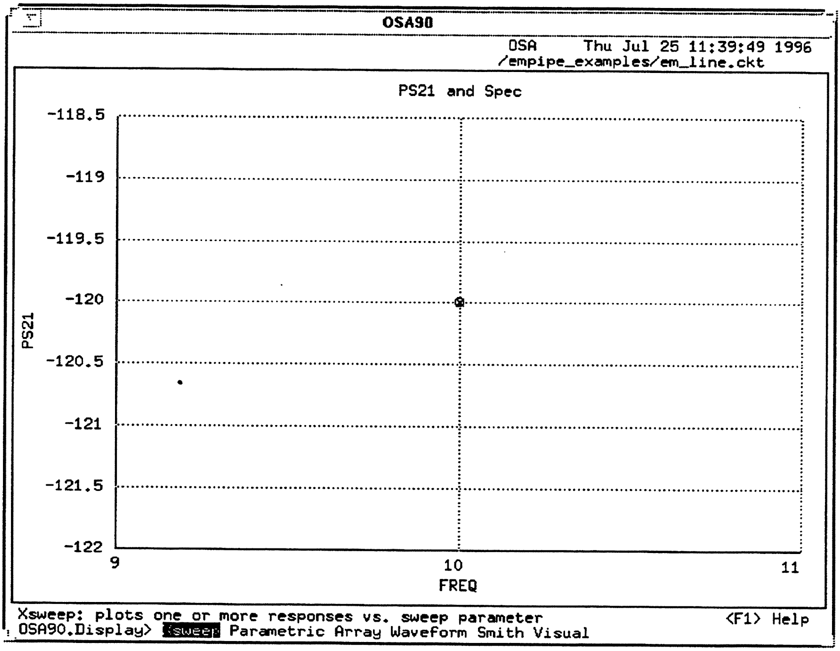


Fig. 3.11 The PS21 response after optimization.

3.7 Saving the Optimized Geometry

We have completed the tasks within the OSA90 simulation/optimization environment.

Exit from the display menu by pressing the <Esc> key (or by clicking the right-hand mouse button). At the OSA90 main menu, press the <Esc> key again.

You will see this prompt near the bottom of the OSA90 window:

```
Exit from OSA90 (Y/<N>):
```

Click the left-hand mouse button to confirm (or press the <Y> key).

Upon exit from OSA90, the Empipe windows reappear on the screen. A dialogue box is also displayed, as depicted in Fig. 3.12.

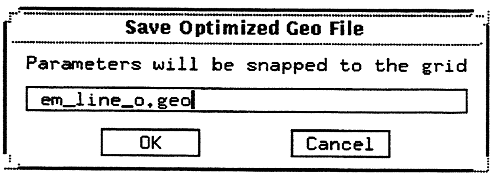


Fig. 3.12 Dialogue box for the optimized ".geo" file name.

The dialogue box allows you to specify the name of a ".geo" file in which the optimized geometry will be saved.

Click on the "OK" button to accept the default file name. The optimized microstrip line will be saved to a disk file under the name "em_line_o.geo", and ***xgeom*** will be automatically invoked to display the geometry.

When saving the ".geo" file, the optimized parameters are snapped to the nearest grid. In the microstrip line example, the optimized value for the length parameter is 157.237 mils and the grid size is 10 mils. Hence, the line length saved in em_line_o.geo is 160 mils.

The optimized solution without truncation is displayed in the "Select Variables" window, as shown in Fig. 3.13.

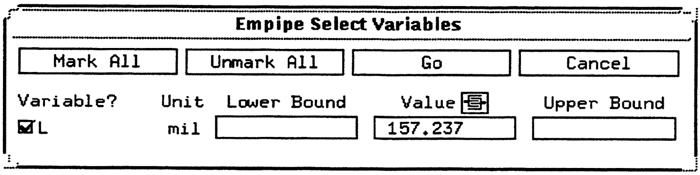


Fig. 3.13 The optimized solution displayed.

Concluding the Tutorial

Click on the "Cancel" button in the "Select Variables" window. This brings you back to the main window of Empipe. Click on the "Quit" button. You will be prompted whether you wish to save the project file. Click "No" and Empipe will exit.

4

Tutorial: Double Folded Stub Filter

4

Tutorial: Double Folded Stub Filter

4.1 Introduction

This chapter is the second segment of the series of tutorials which systematically introduces you to the various features of Empipe.

We assume that you have already been given a basic overview of Empipe by going through the introductory tutorial provided in Chapter 3. If you have not, then please follow Chapter 3 first.

What You Will Learn From This Tutorial

1 In Chapter 3, you have learned how to apply Geometry Capture to a simple structure with one variable. This chapter demonstrates a more complex example which involves three designable parameters.

2 The example in Chapter 3 was analyzed at a single frequency. In this chapter, we learn how to define upper and lower specifications over different frequency bands.

3 How to select the minimax optimizer for filter design.

4 Another feature of Empipe: tracking ***em*** simulation in real time and plotting the *S* parameters while the ***em*** simulation is still in progress.

The ***em*** simulation of the double folded stub filter takes approximately 1/4 second per frequency on a SPARCstation 10 and the total number of frequencies is 61. All the ***em*** analysis results necessary for this tutorial have been saved in a database, therefore we can carry out the tutorial without actually invoking ***em***. In Section 4.7, however, we will deliberately bypass the database in order to track an ***em*** simulation in real time.

When this symbol appears on the left-hand side column, it highlights text that describes hands-on actions. You can take a "short-cut" through the tutorial by following this symbol and skip over the commentaries.

Description of the Example

The example we will use is a double folded stub microstrip bandstop filter, as depicted in Fig. 4.1 (J.C. Rautio, Sonnet Software, Inc., 1992).

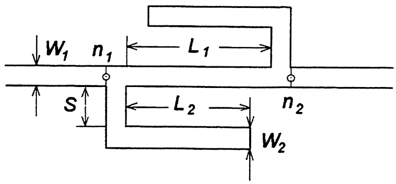


Fig. 4.1 The double folded stub microstrip filter.

The substrate is 5 mil thick with a relative dielectric constant of 9.9. Three designable parameters are defined: L_1, L_2 and S. The widths W_1 and W_2 are fixed at 4.8 mils.

The design specifications for the double folded stub filter are

$|S_{21}| \leq -30$ dB for 12 GHz $\leq f \leq$ 14 GHz

$|S_{21}| \geq -3$ dB for $f \leq$ 9.5 GHz or $f \geq$ 16.5 GHz

where f represents frequency.

The frequency band for ***em*** simulation is chosen to be from 5 GHz to 20 GHz with a step of 0.25 GHz, for a total of 61 frequencies.

Are You in the Right Subdirectory?

All the operations in this tutorial should take place within the subdirectory which contains your copy of the Empipe examples (the instructions for coping the Empipe examples are found in Chapter 1). If you are not already in that subdirectory, type

cd <*Your Empipe Example Subdirectory*>

4.2 Geometry Capture From a Set of Geo Files

First, we choose a nominal geometry for the filter with the following parameter values:

$L_1 = 86.4$ mils

$L_2 = 81.6$ mils

$S = 4.8$ mils

(The definition of these parameters are illustrated in Fig. 4.1.)

The nominal geometry is described by the file dfstub0.geo, which is included in the set of Empipe example files.

You can use *xgeom* to view the geometry by typing

```
xgeom dfstub0.geo
```

The *xgeom* display is shown in Fig. 4.2.

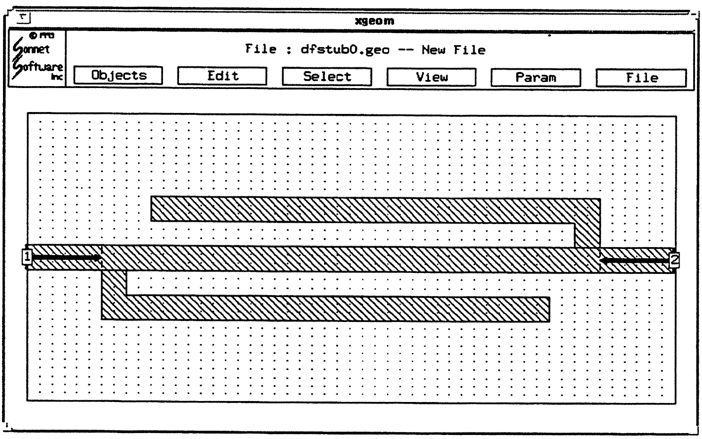


Fig. 4.2 The nominal geometry of the double folded stub filter.

The cell size defined in dfstub0.geo is 2.4 × 1.6 mils.

Creating the Incremental Change Geo Files

We need to create three additional ".geo" files to represent incremental changes in the parameters with respect to their nominal values. You can find these files among the **Empipe** example files under the names dfstub1.geo, dfstub2.geo and dfstub3.geo, representing the parameters L_1, L_2 and S, respectively.

For instance, Fig. 4.3 compares the geometries described by dfstub0.geo and dfstub3.geo. The value of the parameter S is 4.8 mils in `dfstub0.geo` and is changed to 11.2 mils in dfstub3.geo.

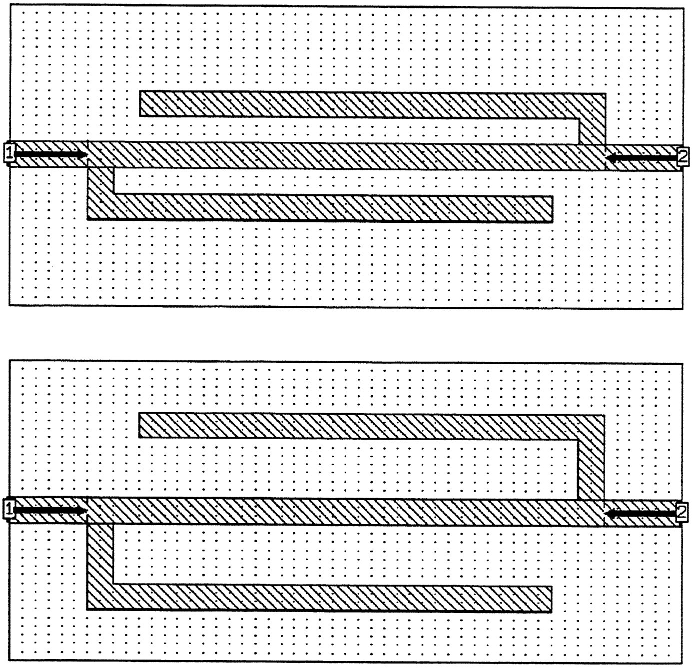


Fig. 4.3 Geometries for S=4.8 mils and S=11.2 mils, respectively.

Note that when we change the value of S we have the option of keeping the size of the substrate box constant or increasing the box size proportionally along the Y dimension. We can make a choice based on convenience or other practical consideration. In dfstub3.geo, we chose to keep the size of the substrate box constant. As a consequence, we should impose bounds on the value of S such that the folded stub will not be increased beyond the substrate box. We will show how to define the bounds in Section 4.3.

Using the Geometry Capture Form Editor

Type

```
empipe dfstub
```

where dfstub identifies the Empipe element to be created for the double folded stub filter.

The Empipe Geometry Capture form editor is displayed, as depicted in Fig. 4.4.

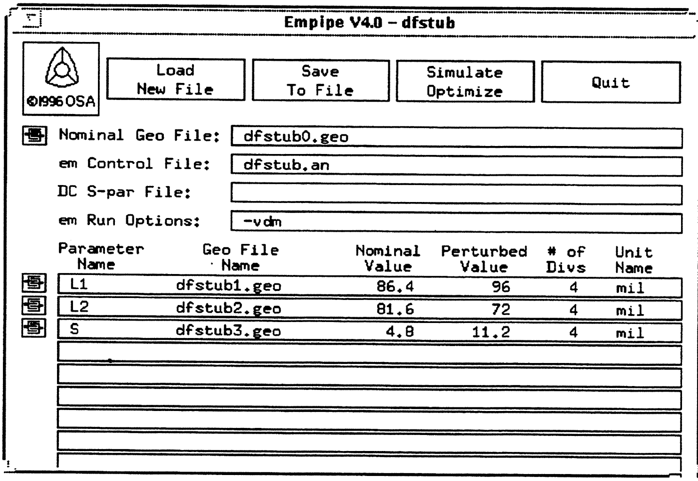


Fig. 4.4 Empipe Geometry Capture form editor.

The data shown in the form editor is retrieved from the file dfstub.inc which is provided with the Empipe example files. Once a structure has been captured, Empipe stores the relevant data in a "*name*.inc" file, where *name* is the identifier (which is dfstub in our example).

When you start Empipe by typing

Empipe *name*

It first looks for the file *name*.inc on the disk. If the file exists, Empipe will retrieve the data, otherwise it will assume that you are starting a new project and accordingly present the form editor with blank fields.

Nominal Geo File

This entry identifies the nominal ".geo" file which in this case is dfstub0.geo.

This button appears adjacent to the entry box. If you click on it, ***xgeom*** will be invoked to display the nominal ".geo" file.

em Control File

This entry identifies the ***em*** analysis control file which defines the frequencies.

The analysis control file which has been created for the double folded stub filter is named dfstub.an. It contains

```
VER 3.0
GHZ
FRE 5.0  20.0  0.25
```

which defines a frequency sweep from 5 GHz to 20 GHz with a step of 0.25 GHz, for a total of 61 frequencies.

DC S-Parameter File

This field allows you to supply a separate *S*-parameter file for DC. This is necessary only if the Empipe element is to become part of a nonlinear circuit for harmonic balance simulation. It does not apply to our example and should be simply left blank.

em Run Options

This field allows you to specify the run-time options for ***em***. The default options are -vdm (verbose, automatic de-embedding and memory saver).

Parameter Definition Data

Parameter Name	Geo File Name	Nominal Value	Perturbed Value	# of Divs	Unit Name
L1	dfstub1.geo	86.4	96	4	mil
L2	dfstub2.geo	81.6	72	4	mil
S	dfstub3.geo	4.8	11.2	4	mil

Parameter Name is an arbitrary ASCII string of no more than 32 characters. We choose the names L1, L2 and S to identify the parameters L_1, L_2 and S, respectively.

Geo File Name identifies the ".geo" file which describes the geometry of the structure after an incremental change in the parameter value is made. The files in this case are dfstub1.geo, dfstub2.geo and dfstub3.geo for the parameters L1, L2 and S, respectively.

Nominal Value refers to the nominal value of the parameter which is consistent with the nominal ".geo" file. The nominal values for L1, L2 and S is 86.4, 81.6 and 4.8, respectively (these should be entered as plain numbers, since the physical unit mil is entered as a separate item).

Perturbed Value refers to the parameter value after the incremental change. The perturbed values for L1, L2 and S are 96, 72 and 11.2 (mils), respectively. Note that the incremental change does not always have to be an increase in the parameter value. For instance, we decrease the value of L2 from 81.6 to 72 (mils).

Number of Divs is obtained by dividing the incremental change by the *em* cell size along the appropriate dimension. For instance, the incremental change in the parameter L1 is 9.6 mils. This change is along the X dimension of the layout and the grid size for the X dimension is 2.4 mils, therefore the number of divisions is 4.

We chose the increments to be 4 times the grid size instead of 1 merely for the purpose of illustration. In general, you can choose any reasonable and convenient number. Also, the number may vary for different parameters.

This information is needed to ensure that the geometry Empipe generates for *em* analysis is always on the grid (interpolation is used for off-grid points).

Unit Name identifies the physical unit of the parameter. For this example, the unit name is MIL for all three parameters.

4.3 Selecting the Optimization Variables

In the Empipe form editor window, click on the button

Simulate
Optimize

Selecting Optimization Variables

Two new windows appear on the screen, entitled "Empipe Select Variables" and "Empipe Specifications", respectively.

The parameters L1, L2 and S are listed in the "Select Variables" window. Initially none of the parameters are selected.

Click on the button labelled <Mark All> to select all the parameters as variables.

You should see a check mark appear in the check boxes next to the parameters, indicating that they have been selected as optimization variables, as depicted in Fig. 4.5.

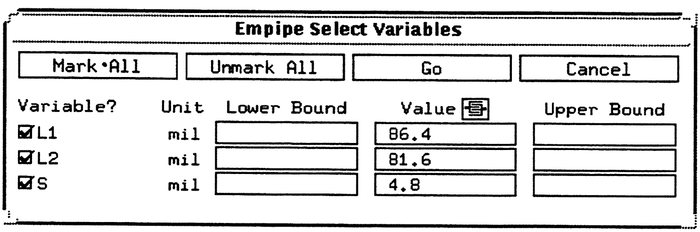


Fig. 4.5 The "Select Variables" window.

Value

The current value for each parameter is shown under the heading "Value". In optimization, this value is used as the starting point. By default, it is set to the parameter's nominal value. If you wish to change it, click on the entry box under "Value" and enter the desired value. You can also click on the button to view the geometry of the current point.

Imposing an Upper Bound on S

Recall that when we made the incremental change to the parameter S, we kept the substrate box size constant. Consequently, we need to impose an upper bound on the variable S so that during optimization the filter structure will not grow beyond the substrate box (see also the discussion following Fig. 4.3 in Section 4.2).

Click on the entry box on the line for the parameter S and in the column under the heading "UpperBound". Then type "16" followed by <Enter>. In other words, we impose an upper bound of 16 mils on S, so that during optimization the value of S is not allowed to exceed 16 mils.

The "Select Variables" window showing the upper bound on S is illustrated in Fig. 4.6.

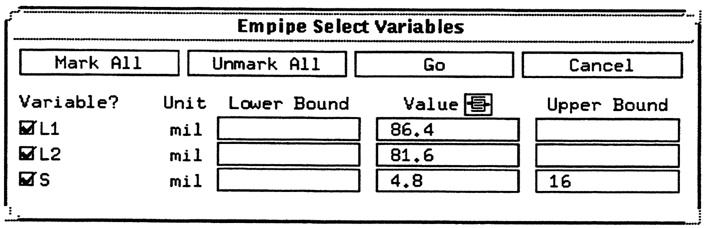


Fig. 4.6 Window showing the upper bound on the variable S.

We leave all the other entries at their default setting.

4.4 Passband and Stopband Specifications

In general, to define a specification we need to carry out the following steps in the "Specifications" window:

1 Select a frequency range.

2 Select an *S*-parameter response.

3 Select a specification type (upper, lower or equality specification).

4 Enter a numerical value as the goal.

5 Optionally, enter a weighting factor.

The design specifications for the double folded stub filter are

$|S_{21}| \leq -30$ dB for 12 GHz $\leq f \leq$ 14 GHz

$|S_{21}| \geq -3$ dB for $f \leq$ 9.5 GHz or $f \geq$ 16.5 GHz

where f represents frequency.

First, we show step by step how to define the stopband specification

$|S_{21}| \leq -30$ dB for 12 GHz $\leq f \leq$ 14 GHz

Selecting the Frequency Range

Initially the frequency entry line appears as

FREQ (GHz) from: [5] to: [20] step: [0.25]

The default setting represents the ***em*** analysis frequencies defined in the file dfstub.an.

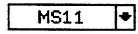
To define the stopband, click on the "from:" box and type "12"; then click on the "to:" box and type "14".

The frequency range is now shown as

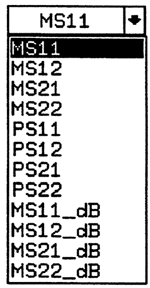
FREQ (GHz) from: [12] to: [14] step: [0.25]

Selecting the S-Parameter Response

This is the entry box for selecting an *S*-parameter response:



Click on the arrow and you will see the list of available responses:



The label MS*ij* represents the magnitude of S_{ij}, PS*ij* represents the phase of S_{ij} in degrees, and MS*ij*_dB represents the magnitude of S_{ij} in decibels.

We wish to select the magnitude of S_{21} in dB. Click on the label MS21_dB.

Selecting the Type of Specification

The entry box for selecting the type of specification appears as

The symbol "<" represents an upper specification. Since this is what we need, do not change it.

Entering a Numerical Value as the Goal

The third field on the specification line represents the goal. The goal for MS21_dB in the stopband is -30.

Click on the box and then type "-30".

Optional Weighting Factor

You can enter an optional weighting factor. The default value is 1. If given, the weighting factor must be a positive number.

For the tutorial example, leave the weighting factor at the default value of 1.

Completing the Definition of the Stopband Specification

Click on the button

Add a new specification defined as follows

The specification we have just defined will be recorded under the heading "Specifications Currently Defined". The "Specifications" window at this point is shown in Fig. 4.7.

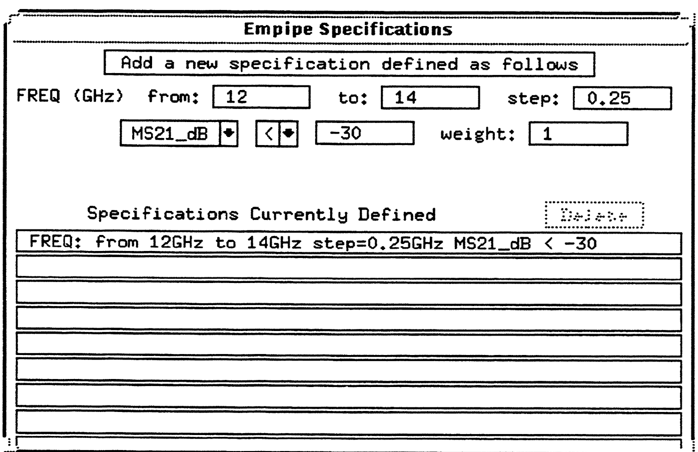


Fig. 4.7 The "Specifications" window.

Defining the Passband Specifications

In similar manner we define the specifications for the lower passband, as

$|S_{21}| \geq -3$ dB for $f \leq 9.5$ GHz

and the upper passband, as

$|S_{21}| \geq -3$ dB for $f \geq 16.5$ GHz

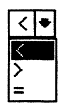
where f represents frequency.

The procedure for defining the passband specifications is described step by step as follows.

1 Modify the frequency range. Click on the "from:" box and type "5"; then click on the "to:" box and type "9.5".

2 Change the specification type. The entry box for selecting the type of specification appears as

Click on the arrow and you will see the list of available specification types:



Click on the symbol ">".

3 Change the goal from -30 to -3. Click on the goal box and type "-3".

4 Click on the button

`Add a new specification defined as follows`

The lower passband specification is added to the list of "Specifications Currently Defined".

5 Modify the frequency range. Click on the "from:" box and type "16.5"; then click on the "to:" box and type "20".

6 Click on the button

`Add a new specification defined as follows`

The complete set of specifications is shown in Fig. 4.8.

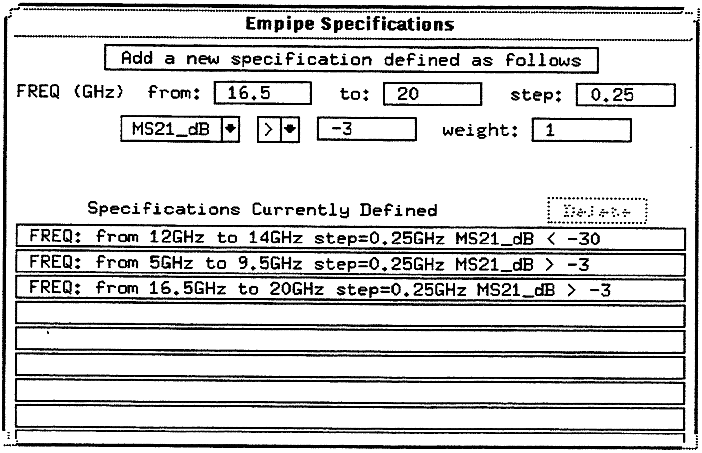


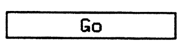
Fig. 4.8 The complete set of specifications.

Deleting an Existing Specification

In case you made a mistake, you can delete a specification and redefine it. To delete an entry under the heading "Specifications Currently Defined", click on the entry. The button "Delete" will turn from a shade of grey to a solid color ("clickable"). A click on this button will delete the highlighted specification line.

What's Next

We are ready to start EM simulation and optimization. Go to the "Select Variables" window and click on the button



4.5 Minimax Optimization

After you click on the "Go" button, Empipe activates the OSA90 simulation/optimization environment. The OSA90 window is depicted in Fig. 4.9.

```
                                  OSA90
File Parsing Completed                      OSA      Fri Sep  6 15:50:59 1996
                                            /osa/empipe_examples/dfstub.ckt

! EM optimization of user-defined structure: DFSTUB

Model
#include "dfstub.inc";

   DFSTUB_L1: ?86.4?;
   DFSTUB_L2: ?81.6?;
   DFSTUB_S: ?0 4.8 16?;

   DFSTUB  1 2 0
      L1=(DFSTUB_L1 * 1mil)  L2=(DFSTUB_L2 * 1mil)
      S=(DFSTUB_S * 1mil);

   PORTS 1 0 2 0;

   CIRCUIT;

   MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);
   MS21_DB = MS_DB[2,1];
end

Sweep
   AC: FREQ: from 5GHz to 20GHz step=0.25GHz
       MS MS_DB PS MS21_dB
      {XSWEEP Title="MS21_dB and Spec"
       Y=MS21_dB X=FREQ
       SPEC=(from 12GHz to 14GHz, < -30) &
            (from 5GHz to 9.5GHz, > -3) &
            (from 16.5GHz to 20GHz, > -3)};
end

Spec
   AC: FREQ: from 12GHz to 14GHz step=0.25GHz MS21_dB < -30;
   AC: FREQ: from 5GHz to 9.5GHz step=0.25GHz MS21_dB > -3;
   AC: FREQ: from 16.5GHz to 20GHz step=0.25GHz MS21_dB > -3;
end

File: reads, edits, parses and saves files                          <F1> Help
OSA90> File Display Optimize Macro Sensitivity monteCarlo Learn
```

Fig. 4.9 OSA90 window.

In the OSA90 input file (netlist), the Model block contains the labels representing the optimization variables, the DFSTUB element and the port definitions. The Sweep block contains the simulation range and outputs. The Spec block contains the three specifications we have just defined.

Simulation and Display the Response

Click on the menu option "Display".

You will notice that the simulation seems to finish very quickly. This is because that the *em* analysis has already been done when we prepared the tutorial example and the results have been stored in a database. Empipe simply retrieved the appropriate data from the database, without actually invoking *em*. An actual *em* simulation of this filter would take approximately 14 minutes of CPU time on a SPARCstation 10.

Now the menu area at the bottom of the window shows

```
Xsweep: plots one or more responses vs. sweep parameter    <F1> Help
OSA90.Display> Xsweep Parametric Array Waveform Smith Visual
```

Click on the display menu option Xsweep. A pop-up window appears, showing the default setting for display. Simply press <Enter> to accept the default setting.

The graphical display is depicted in Fig. 4.10. It shows the response MS21_dB versus the frequency. The specification is also shown. Clearly, the filter response does not satisfy the specifications with the initial parameter values.

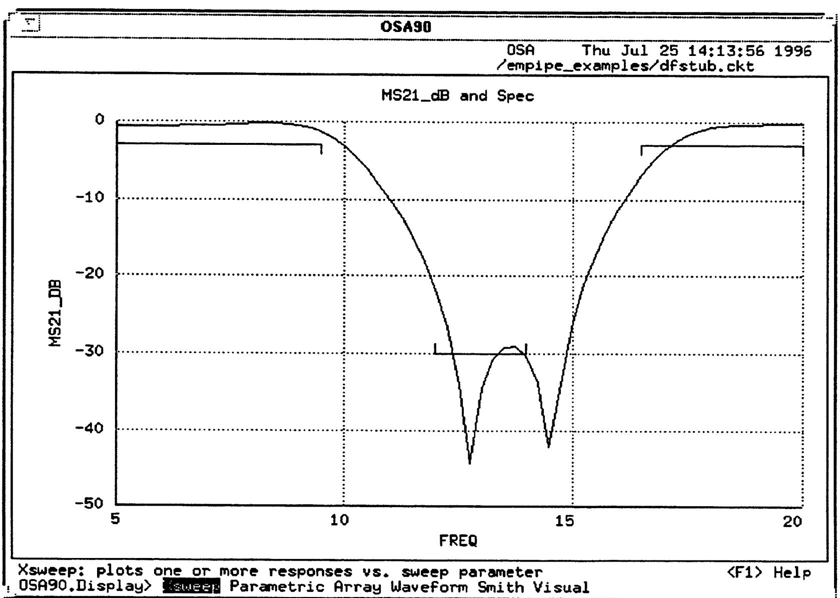
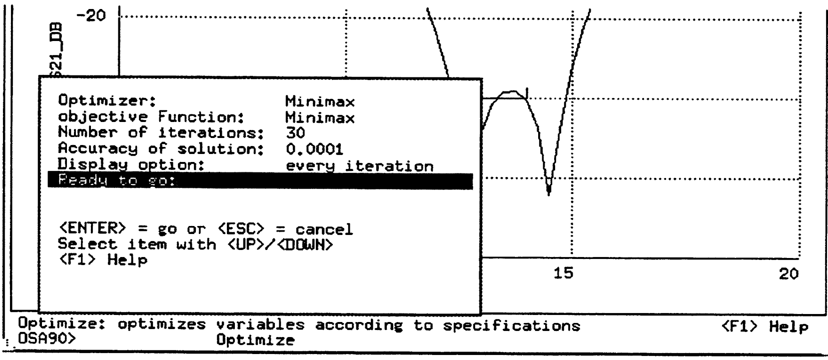


Fig. 4.10 Double folded stub filter response before optimization.

Start Optimization

Exit the display menu by pressing the <Esc> key. This brings you back to the OSA90 main menu. Click on the menu option Optimize.

A pop-up window appears:



Notice that the optimizer is shown as "Minimax". Empipe automatically chooses the optimizer for you according to the type of specifications defined. For the microstrip line example in Chapter 3, the "L1" optimizer was chosen, since the specification was to match the calculated PS21 to -120 degrees. The filter example we are considering here involves upper and lower specifications, hence Empipe chooses the minimax optimizer.

It is possible for you to change the default optimizer as well as the setting for the maximum number of iterations and the desired accuracy of solution (see Chapter 9).

Press <Enter> to accept the default setting.

The progress of optimization is reported on the screen:

```
Iteration   1/30 Max Error=8.14511
Iteration   2/30 Max Error=6.34936
Iteration   3/30 Max Error=1.53442
Iteration   4/30 Max Error=0.553539
Iteration   5/30 Max Error=-0.19606
Iteration   6/30 Max Error=-0.213403
Iteration   7/30 Max Error=-0.213773
Solution  Max Error=-0.213773
```

The optimization seems to go very quickly. This is because all the necessary *em* analysis results are already available from the database.

The numerical values you actually see may be slightly different from those shown here, due to differences in the computer hardware and/or software versions.

Simulation of the Optimized Filter

Click on the menu option Display. When the simulation is completed and you see the OSA90.Display menu, click on the option Xsweep. As the pop-up window appears, press the <Enter> key to accept the default setting. The response shown in Fig. 4.11 indicates that the specifications are satisfied after optimization.

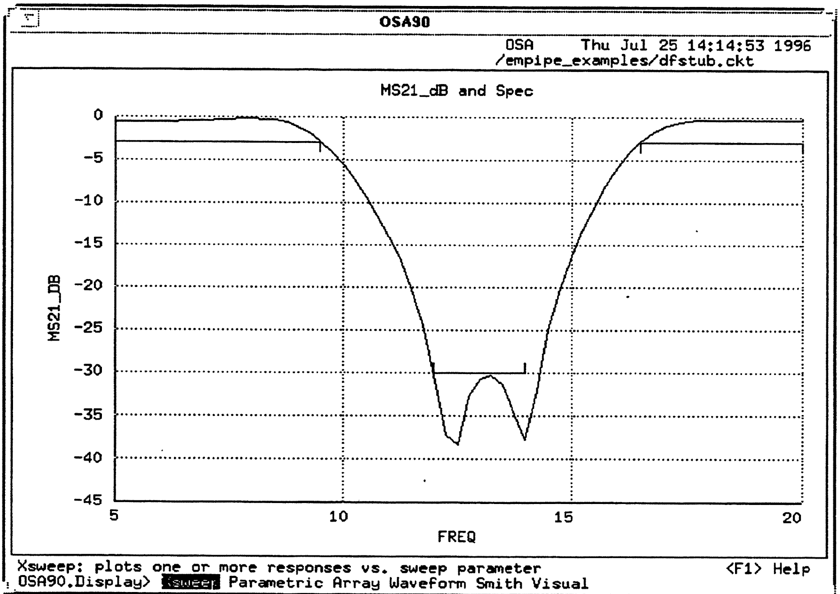


Fig. 4.11 Double folded stub filter response after optimization.

4.6 Saving the Optimized Geometry

We have completed the tasks within the OSA90 simulation/optimization environment.

Exit from the display menu by pressing the <Esc> key (or by clicking the right-hand mouse button). At the OSA90 main menu, press the <Esc> key again.

You will see this prompt near the bottom of the OSA90 window:

```
Exit from OSA90 (Y/<N>):
```

Press the <Y> key to confirm (or click the left-hand mouse button).

Upon exit from OSA90, the Empipe windows reappear on the screen. A dialogue box appears to prompt you for the file name for saving the optimized geometry. Click on the "OK" button to accept the default file name "dfstub_o.geo". The optimized filter will be saved on the disk, and *xgeom* will be invoked to display the geometry.

When saving the ".geo" file, the optimized parameters are snapped to the nearest grid. For example, the optimized value for the parameter L1 is 92.1059 mils. In dfstub_o.geo, L1 is rounded to 91.2 mils, since the corresponding grid size is 2.4 mils.

However, you can see the optimized solution without truncation in the "Select Variables" window, as shown in Fig. 4.12.

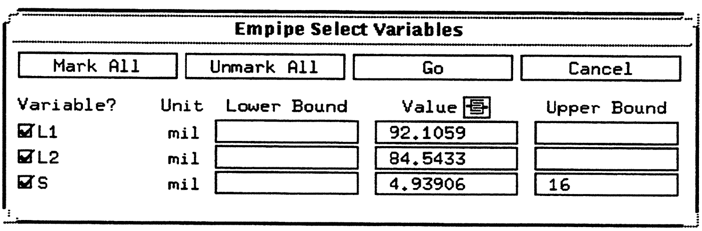


Fig. 4.12 The optimized solution without truncation.

4.7 Tracking EM Simulation in Real Time

So far in this chapter we have not actually started any *em* analysis, since all the necessary results are available from the database.

In this section, we will deliberately bypass the database and invoke *em* to simulate the double folded stub filter. This gives you an idea of the time scale of the actual *em* simulation. It will also demonstrate the Empipe feature of tracking *em* simulation in real time and plotting the *S* parameters while the *em* simulation is in progress.

In the "Select Variables" window, click on the button

Go

This brings up the OSA90 window.

OSA90 Input File Editor

At the OSA90 main menu, click on the menu option File to activate the input file editor. You can use the cursor keys and the mouse to move around the file; use the <Pg Up> and <Pg Dn> keys to scroll the text; use the <Back Space> and <Delete> keys to delete text; and use any alphanumeric keys to enter new text.

Database Index

In order to bypass the database, we need to modify the database index in the OSA90 input file.

Locate this statement in the file:

```
DFSTUB  1 2 0
   L1=(DFSTUB_L1 * 1mil)  L2=(DFSTUB_L2 * 1mil)
   S=(DFSTUB_S * 1mil);
```

Add a line to this statement so that it looks like this:

```
DFSTUB  1 2 0
   INDEX=0
   L1=(DFSTUB_L1 * 1mil)  L2=(DFSTUB_L2 * 1mil)
   S=(DFSTUB_S * 1mil);
```

The phrase "INDEX=0" disables the Empipe database feature (this is more thoroughly described in Chapter 10).

Start a New EM Simulation

Click the left-hand mouse button anywhere within the file editor. A menu box appears on the screen. Click on the option "Exit from editor". Now you are back at the OSA90 main menu at the bottom of the OSA90 window. Click on the Display menu option.

On the top of the window, you should see this message:

```
Simulation ... Press any key to interrupt
```

Be patient, since now we have started a new *em* simulation of the double folded stub filter. It takes approximately 1/4 minute per frequency on a SPARCstation 10.

After 30 seconds or so (on a SPARCstation 10), a new window appears on the screen, as depicted in Fig. 4.13.

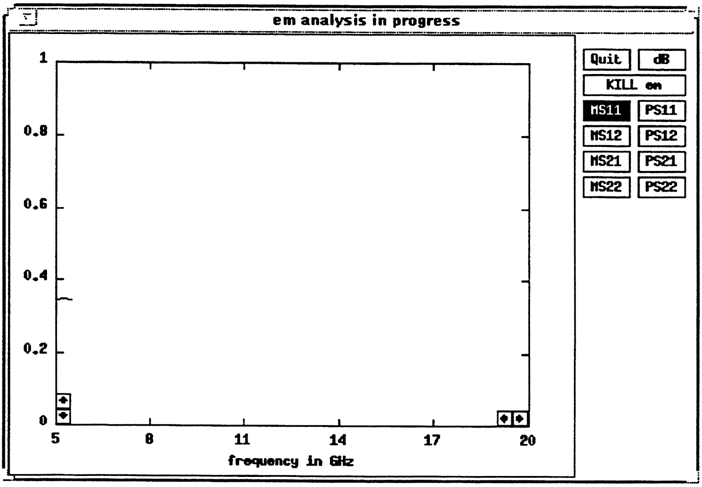


Fig. 4.13 Tracking em simulation in progress.

This window represents Emtrack, a companion utility program for Empipe. Emtrack plots the *S* parameters calculated by *em* while the *em* simulation is still in progress. As *em* completes the analysis for each frequency, it writes the *S* parameters to the output file. Empipe monitors the output file and reads the *S* parameters as soon as they become available for each frequency.

On the right-hand side of the Emtrack window there is a set of buttons, each representing one of the *S* parameter responses (magnitude and phase). By clicking on these buttons, you can select one or more responses to be plotted. The curves plotted are periodically updated as new results become available.

Interrupting the EM Simulation

The button labelled "KILL em" allows you to interrupt the *em* simulation by killing the UNIX process.

If you do not wish to wait until the *em* simulation finishes, you can abandon this part of the tutorial exercise by click on the "KILL em" button. The Emtrack window will disappear. A message will be displayed on the top of the OSA90 window indicating that the simulation has been terminated by the child program. You can skip the rest of this section and go directly to the end of the chapter for "Concluding the Tutorial".

Quitting Emtrack

The button labelled "Quit" allows you to close the Emtrack window without terminating the *em* simulation. Do not touch this button for this tutorial.

Showing the Magnitude of S Parameters in dB

The Emtrack window has a button labelled "dB". It allows you to toggle on and off the option which displays the magnitude(s) of selected *S* parameter(s) in dB.

In Fig. 4.14, MS11 and MS21 are selected and plotted in dB. It also shows that the latest frequency completed by *em* is 10 GHz.

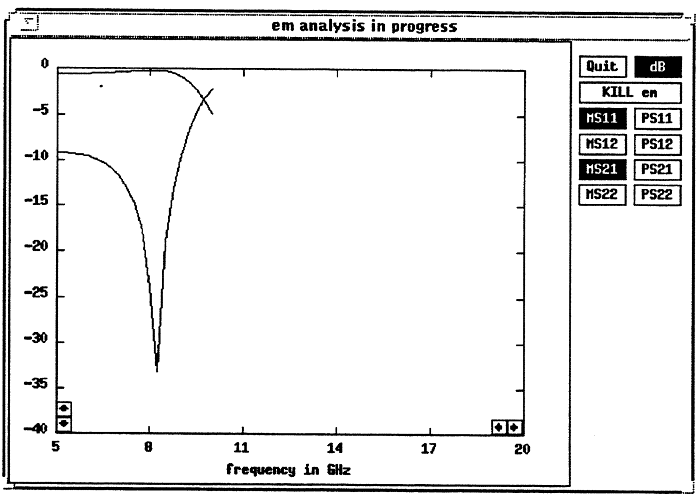


Fig. 4.14 MS11 and MS21 plotted in dB.

Changing the Y-Axis Scale or the X-Axis Divisions

In the Emtrack window you can see four small solid arrow buttons. They allow you to change the display scale of the Y-axis and the number of divisions (ticks) of the X-axis.

The two arrow buttons lined up vertically near the left-bottom corner of the window control the Y-axis scale. They are active only when the "dB" option is selected. When the "dB" option is not selected, the Y-axis scale is always from 0 to 1.

The two arrow buttons lined up horizontally control the number of divisions (ticks) of the X-axis. Initially, the X-axis, which represents the frequency, is marked by 5 divisions. Depending on the frequency range, you may find a more convenient number of divisions for the X-axis.

For instance, the frequency range in our example is from 5 to 20 GHz. In Fig. 4.15, the X-axis has 15 divisions, therefore the display is clearly ticked for every 1 GHz.

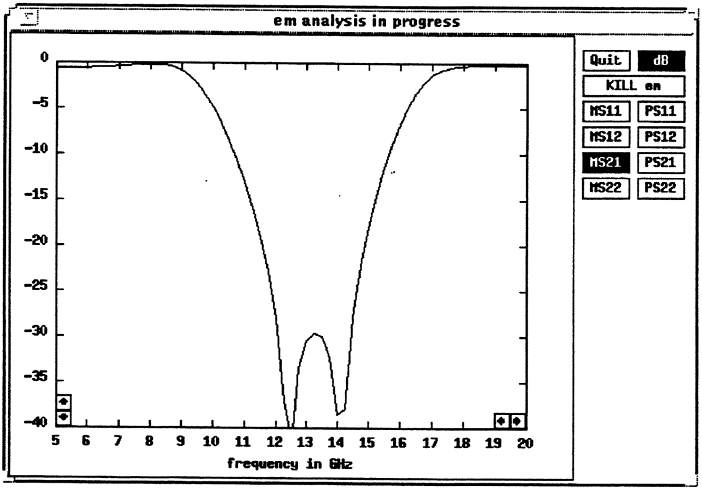


Fig. 4.15 Illustration of X-axis divisions.

Completion of the EM Simulation

The *em* simulation of the double folded stub filter over the whole frequency range should take approximately 14 minutes on a SPARCstation 10. Once it is completed, the Emtrack window is automatically closed and you will be at the "OSA90.Display>" menu.

Press the <Esc> key to exit from the OSA90.Display menu.

Turning Off Emtrack

By default, Emtrack is activated whenever a new *em* analysis involving multiple frequency points is invoked by Empipe. This feature can be turned off, if you prefer, by setting the keyword "TRACK" to 0 in the OSA90 input file.

For example:

```
DFSTUB  1 2 0
   TRACK=0
   L1=(DFSTUB_L1 * 1mil)  L2=(DFSTUB_L2 * 1mil)
   S=(DFSTUB_S * 1mil);
```

You can set "TRACK" to 0, 1 or 2. Setting it to 0 disables the Emtrack feature. The default setting of 1 activates Emtrack for any new *em* simulations, and the Emtrack display will be closed automatically as soon as the *em* analysis is completed. Setting "TRACK" to 2 also activates Emtrack, and the Emtrack display will stay on even after the *em* analysis is completed (until you click on the "Quit" button).

Concluding the Tutorial

Press the <Esc> key to exit from OSA90. At the prompt:

```
Exit from OSA90 (Y/<N>):
```

Click the left-hand mouse button or press the <Y> key. Back in Empipe, click on the "Cancel" button in the "Select Variables" window. Then, click on the "Quit" button in the main Empipe window. You will be prompted whether you wish to save the project file. Click "No" and Empipe will exit.

5

Tutorial: 10-dB Distributed Attenuator

5

Tutorial: 10-dB Distributed Attenuator

5.1 Introduction

This chapter is the third segment of the series of tutorials which systematically introduces you to the various features of Empipe.

We recommend that you follow the tutorials in the order that they are presented. At least you should study the introductory tutorial in Chapter 3 before any other chapters.

What You Will Learn From This Tutorial

1 How to capture parameters with symmetrical incremental changes in order to preserve geometrical symmetry during optimization.

2 In the preceding chapters the design specifications involve only a single *S* parameter response. This chapter demonstrates specifications on multiple responses.

3 How to assign weighting factors to different responses.

4 How to change the display options in OSA90 to view the simulation results in different formats.

The ***em*** simulation of the attenuator takes approximately 3 minutes per frequency on a SPARCstation 10 and the total number of frequencies is 5. All the ***em*** analysis results necessary for this tutorial have been saved in a database, therefore we can carry out the tutorial without actually invoking ***em***.

When this symbol appears on the left-hand side column, it highlights text that describes hands-on actions. You can take a "short-cut" through the tutorial by following this symbol and skip over the commentaries.

Description of the Example

We consider the 10-dB distributed attenuator depicted in Fig. 5.1 (D.G. Swanson, Jr., Watkins-Johnson Company, Palo Alto, CA 94304-1204, private communication, 1994).

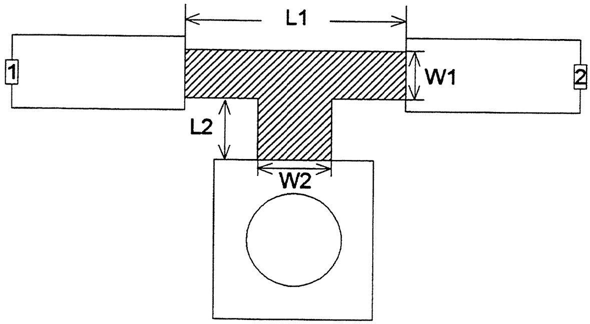


Fig. 5.1 The 10-dB distributed attenuator.

The substrate is 15 mil thick with a relative dielectric constant of 9.8. Two types of metallization are defined: the shaded area in Fig. 5.1 has a high resistivity of 50 Ohms/sq; the feed lines and the grounding pad are assumed to be lossless.

Four designable parameters are defined: L1, L2, W1 and W2.

The design specifications for the attenuator are

$$|S_{21}| = -10 \text{ dB}$$

$$|S_{11}| \leq -10 \text{ dB}$$

for the frequency range of 2 GHz to 18 GHz. Choosing a step of 4 GHz, we have a total of 5 frequencies.

Are You in the Right Subdirectory?

All the operations in this tutorial should take place within the subdirectory which contains your copy of the Empipe examples (the instructions for coping the Empipe examples are found in Chapter 1). If you are not already in that subdirectory, type

cd <*Your Empipe Example Subdirectory*>

5.2 Geometry Capture

First, we choose the nominal parameter values for the attenuator as

L1 = 22 mils
L2 = 7 mils
W1 = 11 mils
W2 = 10 mils

(The definition of these parameters is illustrated in Fig. 5.1.)

The geometry which corresponds to the nominal parameter values is defined by the file tpad0.geo (we pick the name "tpad" for the fact that the resistive pad is shaped like a tee). You can find the file tpad0.geo among the Empipe example files.

You can use *xgeom* to view the geometry by typing

```
xgeom tpad0.geo
```

The *xgeom* display is shown in Fig. 5.2.

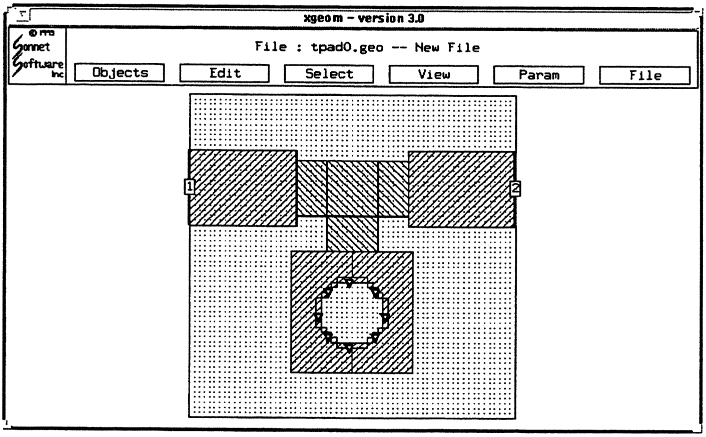


Fig. 5.2 The nominal geometry of the 10-dB distributed attenuator.

The cell size defined in tpad0.geo is 1 mil × 1 mil.

Creating the Incremental Change Geo Files

We need to create additional ".geo" files to represent incremental changes in the parameters with respect to their nominal values. The basic approach to creating the incremental change ".geo" files has already been illustrated in the tutorials in Chapters 3 and 4.

The ".geo" files for the attenuator can be found among the Empipe example files, under the names tpad1.geo, tpad2.geo, tpad3.geo and tpad4.geo, representing the parameters L1, L2, W1 and W2, respectively.

The next step is to invoke the Empipe Geometry Capture form editor to process the set of ".geo" files.

Type

```
empipe tpad
```

The Empipe Geometry Capture form editor window is depicted in Fig. 5.3. The data shown in the window is retrieved from the file tpad.inc which is provided with the Empipe example files. Once a structure has been captured, Empipe stores the relevant data in a "*name*.inc" file, where *name* is the identifier (which is tpad in this case).

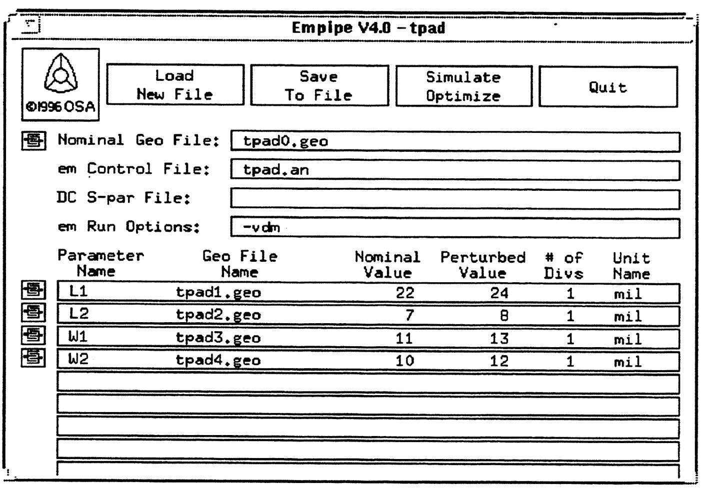


Fig. 5.3 Empipe Geometry Capture form editor.

Parameters with Symmetrical Perturbations

We assume that you already understand the meaning of the entries in the Empipe Geometry Capture form editor. If necessary, please review the descriptions in Chapters 3 and 4.

Here, we would like to focus our attention on a new issue, namely the definition of parameters with symmetrical perturbations.

Note this entry in the Empipe Geometry Capture window:

Parameter Name	Geo File Name	Nominal Value	Perturbed Value	# of Divs	Unit Name
L1	tpad1.geo	22	24	1	mil

It indicates that the value of the parameter L1 has been changed from 22 mils to 24 mils, which means the incremental change is 2 mils. Since the *em* cell size is 1 mil × 1 mil, the number of divisions would seem to be 2. Why, then, is the entry under "# of Divs" 1 instead of 2? It is because we wish to preserve the geometrical symmetry when the parameter L1 changes. This is illustrated in Fig. 5.4.

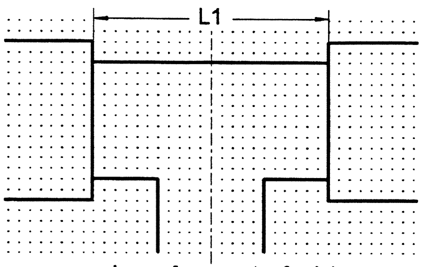


Fig. 5.4 Preserving the symmetry when L1 changes from 22 mils to 24 mils.

As shown in Fig. 5.4, in order to preserve geometrical symmetry, when we increment L1, we have to modify the geometry on both sides of the plane of symmetry. Since the smallest on-grid adjustment on each side is 1 mil (one ***em*** cell), the perturbation to L1, when taking both sides into account, is 2 mils. Since this is the smallest increment that can be made to L1 without destroying the symmetry, we define the number of divisions as 1, although the perturbation is twice the ***em*** cell size.

In other words, the "# of Divs" entry should be interpreted as the parameter discretization grid. It represents a unit change to the parameter while satisfying a set of implied conditions. In many cases, we are concerned with only one condition, namely to stay on the ***em*** grid. In such cases, the parameter discretization grid coincides with the ***em*** grid. If the conditions also include the preservation of symmetry, as in the case of the parameter L1 of the attenuator, then the parameter discretization grid may be different from the ***em*** grid. Typically in such cases the parameter discretization grid is two times the ***em*** cell size along the appropriate dimension.

For the attenuator, three of the four parameters, namely L1, W1 and W2, are defined with symmetrical perturbations.

5.3 Variables, Specifications and Weighting Factors

In the Empipe main window, click on the button

```
Simulate
Optimize
```

As the "Select Variables" window appears, click on the <Mark All> button. You should see all the parameters marked as optimization variables, as illustrated in Fig. 5.5.

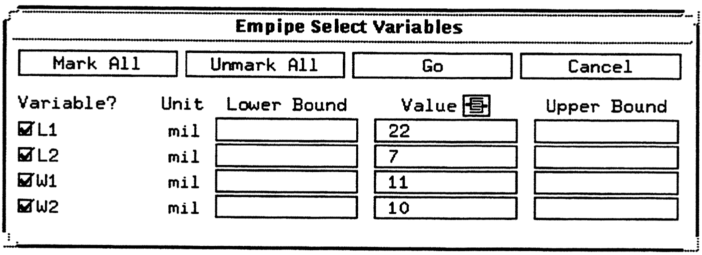


Fig. 5.5 The "Select Variables" window.

Reformulating the Specifications

The design specifications for the 10-dB distributed attenuator are

$$|S_{21}| = -10 \text{ dB}$$

$$|S_{11}| \leq -10 \text{ dB}$$

for the frequency range from 2 GHz to 18 GHz.

The equality specification on $|S_{21}|$ is poorly suited for the minimax optimizer which we wish to use. It can be reformulated into a pair of upper and lower specifications:

$$|S_{21}| \leq -9 \text{ dB}$$

$$|S_{21}| \geq -11 \text{ dB}$$

We will assign a weighting factor of 5 to this pair of specifications. By doing so we attach a greater emphasis to these specifications than the specification on $|S_{11}|$ which by default has a weighting factor of 1.

Steps for Entering the Specifications

1 Select the frequency range. In the "Specifications" window, the default frequency range is shown as

```
FREQ (GHz)  from: [ 2 ]   to: [ 18 ]   step: [ 4 ]
```

No change is needed.

2 Select the *S*-parameter response. This is the entry box for selecting an *S*-parameter response:

```
[ MS11 ▾]
```

Click on the arrow and you will see the list of available responses:

```
[ MS11 ▾]
MS11
MS12
MS21
MS22
PS11
PS12
PS21
PS22
MS11_dB
MS12_dB
MS21_dB
MS22_dB
```

The label MS*ij* represents the magnitude of S_{ij}, PS*ij* represents the phase of S_{ij} in degrees, and MS*ij*_dB represents the magnitude of S_{ij} in decibels.

We wish to select the magnitude of S_{21} in dB. Click on the label MS21_dB.

3 Selecting the type of specification. Initially, the specification type box shows "<". Since this happens to be what we need for the first specification, no change is needed.

4 Enter a numerical goal. Click on the goal box (the third field on the specification line) and type "-9".

5 Enter the weighting factor. Click on the box labelled "weight:" and type "5".

6 Click on the button labelled "Add a new specification defined as follows". The first specification is added to the list under the heading "Specifications Currently Defined".

7 Make the necessary changes for the second specification.

Change the specification type symbol from "<" to ">".

Click on the numerical goal box and type "-11".

Click on the button labelled "Add a new specification defined as follows". The second specification is added to the list under the heading "Specifications Currently Defined".

8 Make the necessary changes for the third specification.

Change the response label from MS21_dB to MS11_dB.

Change the specification type symbol from ">" to "<".

Click on the numerical goal box and type "-10".

Click on the box labelled "weight:" and type "1".

Click on the button labelled "Add a new specification defined as follows". All three specifications are now defined in the "Specifications" window, as depicted in Fig. 5.6.

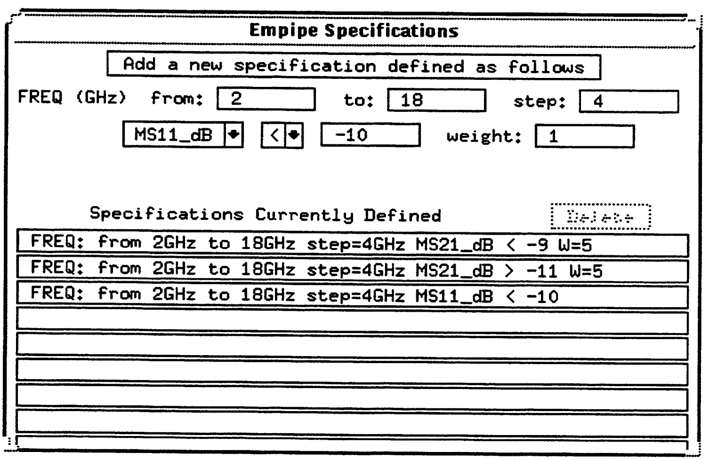


Fig. 5.6 The "Specifications" window.

What's Next

We are ready to start EM simulation and optimization. Proceed by clicking on the "Go" button in the "Select Variables" window.

5.4 Selecting Display Options

After you click on the "Go" button, Empipe activates the OSA90 simulation/optimization environment. The OSA90 window is depicted in Fig. 5.7.

```
OSA90
File Parsing Completed                    OSA      Fri Sep  6 15:54:44 1996
                                          /osa/empipe_examples/tpad.ckt

! EM optimization of user-defined structure: TPAD

Model
#include "tpad.inc";

   TPAD_L1: ?22?;
   TPAD_L2: ?7?;
   TPAD_W1: ?11?;
   TPAD_W2: ?10?;

   TPAD  1 2 0
      L1=(TPAD_L1 * 1mil)  L2=(TPAD_L2 * 1mil)
      W1=(TPAD_W1 * 1mil)  W2=(TPAD_W2 * 1mil);

   PORTS 1 0 2 0;

   CIRCUIT;

   MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);
   MS11_DB = MS_DB[1,1];
   MS21_DB = MS_DB[2,1];
end

Sweep
   AC: FREQ: from 2GHz to 18GHz step=4GHz
       MS MS_DB PS MS21_dB MS11_dB
      {XSWEEP Title="MS21_dB and Spec"
       Y=MS21_dB X=FREQ
       SPEC=(from 2GHz to 18GHz, < -9) &
            (from 2GHz to 18GHz, > -11)}
      {XSWEEP Title="MS11_dB and Spec"
       Y=MS11_dB X=FREQ
       SPEC=(from 2GHz to 18GHz, < -10)}
      {XSWEEP Title="All Specifications"
       Y=MS21_dB & MS11_dB};
end

File: reads, edits, parses and saves files                        <F1> Help
OSA90> File Display Optimize Macro Sensitivity monteCarlo Learn
```

Fig. 5.7 OSA90 window.

In the OSA90 input file (netlist), the Model block contains the labels representing the optimization variables, the TPAD element and the port definitions. The Sweep block contains the simulation range and output labels. The Spec block (not visible in Fig. 5.7) contains the specifications for optimization.

Simulation and Display of the Response

Click on the menu option "Display".

You will notice that the simulation seems to finish very quickly. This is because that the *em* analysis has already been performed when we prepared the tutorial example and the results have been stored in a database. An actual *em* simulation of the attenuator takes approximately 3 minutes of CPU time per frequency (15 minutes for all five frequencies) on a SPARCstation 10.

Now the menu area at the bottom of the window shows

```
Xsweep: plots one or more responses vs. sweep parameter    <F1> Help
OSA90.Display> Xsweep Parametric Array Waveform Smith Visual
```

Click on the display menu option Xsweep. A pop-up window appears, showing the default setting for display. Press <Enter> to accept the default setting.

The graphical display is depicted in Fig. 5.8.

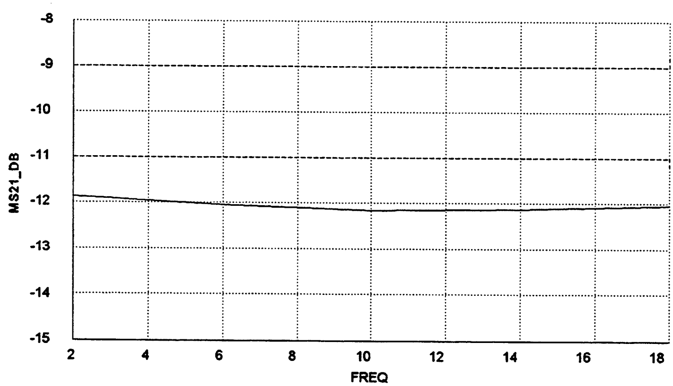


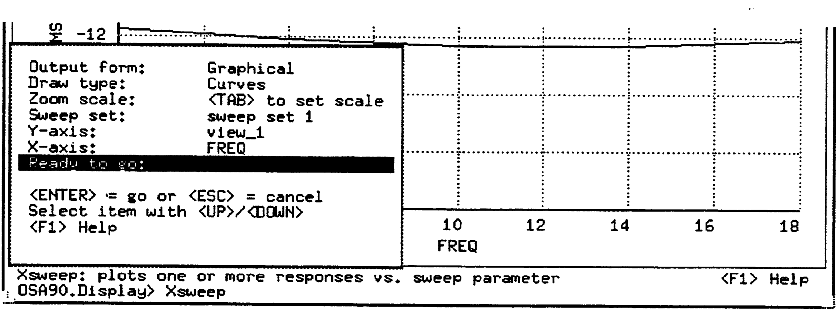
Fig. 5.8 Attenuator response MS21_dB.

Fig. 5.8 shows the response MS21_dB versus the frequency. The specifications on this response are also shown as dashed lines (on the screen they are shown in a distinct color). Clearly, the specifications are not satisfied at this point.

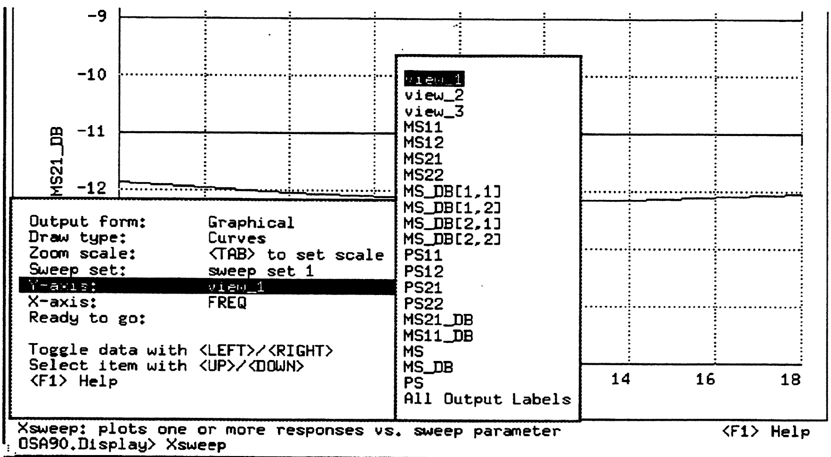
We also have a specification on MS11_dB. Next, we will show how to select the response MS11_dB for display.

Selecting Responses for Display

Click on the display menu option Xsweep. A pop-up window appears as



Click on the option "Y-axis" and you will see a list of choices.



Click on the choice "view_2". The "Y-axis" option in the pop-up window now shows "view_2". Click on the line that says "Ready to go". The response MS11_dB is displayed, as depicted in Fig. 5.9.

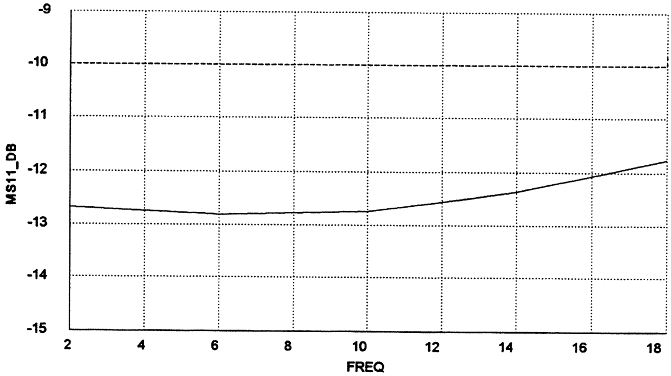


Fig. 5.9 Attenuator response MS11_dB.

Displaying Both MS21_dB and MS11_dB

Click on the display menu option Xsweep. As the pop-up window appears, click on the option "Y-axis". From the list of choices, click on "view_3". The "Y-axis" option now shows "view_3". Click on the "Ready to go" line.

The graphical display, as depicted in Fig. 5.10, now contains both responses MS21_dB and MS11_dB (on the screen the two curves are shown in two different colors).

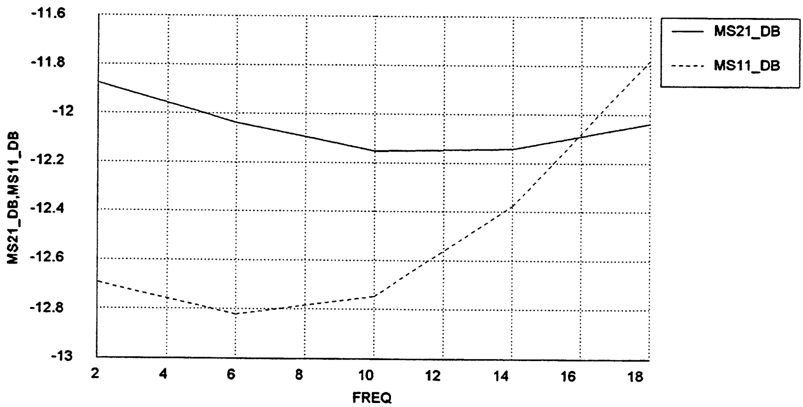


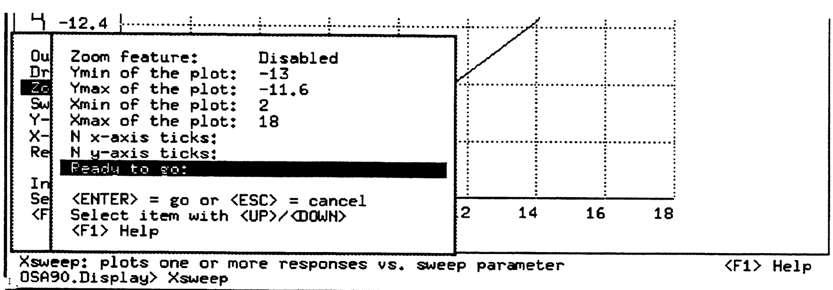
Fig. 5.10 Graphical display of both MS21_dB and MS11_dB.

Customizing the Display Scale

By default, the scale of the Y-axis is automatically determined by the program according to the minimum and maximum values of the responses.

For instance, in Fig. 5.10, the Y-axis has the range of -11.6 to -13 with ticks for 0.2 intervals. On this scale, it is difficult to check the responses against the specifications. We would like to change the default setting to a more convenient one.

Click on the display menu option **Xsweep**. As the pop-up window appears, click on the option "**Zoom scale**". This leads to another pop-up window.



The parameters "Ymin", "Ymax", "Xmin" and "Xmax" define the corners of the display. The parameters "N x-axis ticks" and "N y-axis ticks" specify the number of intervals (divisions) on the X-axis and Y-axis, respectively. These parameters can be modified to customize the display scale.

We will change the display scale as follows.

1. Move the cursor (using the mouse or the cursor keys) to the line labelled "Ymin:" and type "-20".

2. Move the cursor to the line labelled "Ymax:" and type "0".

3. Move the cursor the line labelled "N y-axis ticks:" and type "4".

4. Click on the "Ready to go" line. The "Zoom scale" window disappears. Click on the "Ready to go" line in the "Xsweep" window.

The graphical display with the modified scale is depicted in Fig. 5.11. The new scale allows us to see more clearly the position of the responses relative to the specifications.

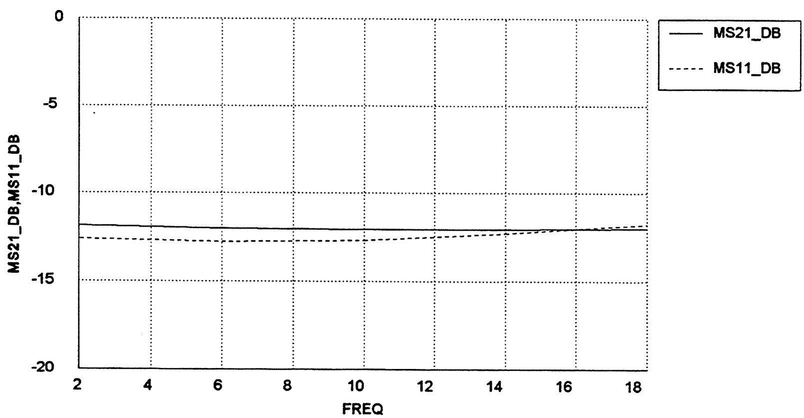


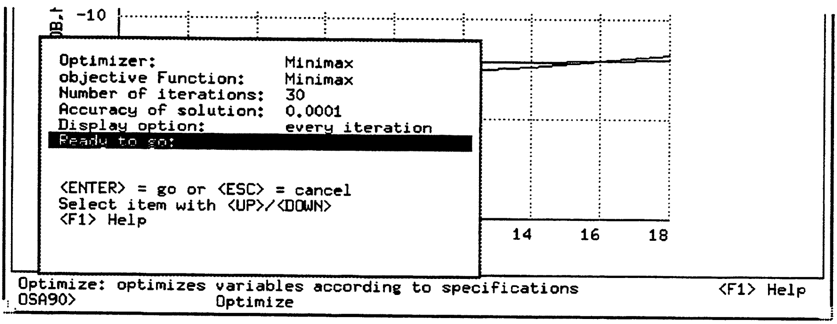
Fig. 5.11 Graphical display with customized scale.

What's Next

We are ready to proceed to optimization. Exit the display menu by pressing the <Esc> key.

5.5 Minimax Optimization

At the OSA90 main menu, click on the menu option Optimize. A pop-up window appears:



Notice that the optimizer is shown as "Minimax". Empipe automatically chooses the optimizer for you according to the type of specifications defined. This tutorial example involves upper and lower specifications, hence Empipe chooses the minimax optimizer.

Press <Enter> to accept the default setting.

The progress of optimization is reported on the screen:

```
Iteration   1/30 Max Error=5.74648
Iteration   2/30 Max Error=5.26174
Iteration   3/30 Max Error=4.31002
Iteration   4/30 Max Error=2.45741
Iteration   5/30 Max Error=-0.987392
      ...
      ...

Iteration  25/30 Max Error=-4.94847
Iteration  26/30 Max Error=-4.95548
Iteration  27/30 Max Error=-4.94672
Iteration  28/30 Max Error=-4.95954
Iteration  29/30 Max Error=-4.93807
Iteration  30/30 Max Error=-4.94534
Solution  Max Error=-4.95954
```

The numerical values you actually see may be slightly different from those shown here, due to differences in the computer hardware and/or software versions.

The optimization goes very quickly, since all the necessary *em* analysis results are already available from the database.

Displaying the Optimized Attenuator Responses

Click on the menu option Display. When the simulation is completed and the OSA90.Display menu appears, click on the option Xsweep. In the pop-up window click on the "Y-axis" option. From the list of available choices, click on "view_3" to display both MS21_dB and MS11_dB. Then click on the "Ready to go" line.

The graphical display is depicted in Fig. 5.12. It shows that after optimization the response MS21_dB is almost exactly -10 dB and the response MS11_dB is well below -10 dB.

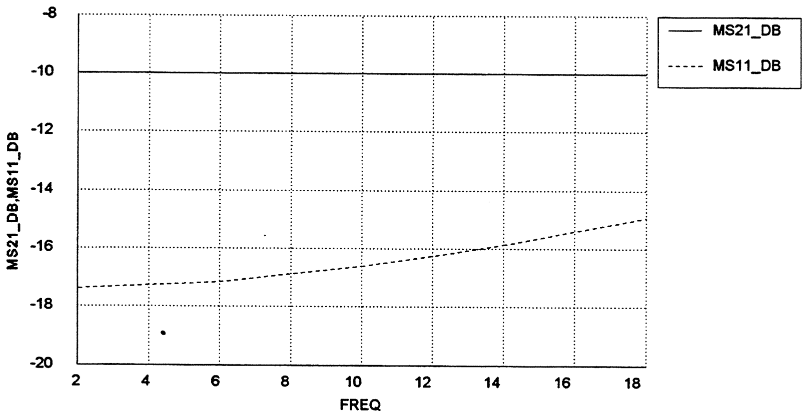


Fig. 5.12 Responses of the optimized attenuator.

Saving the Optimized Geometry

Exit from the OSA90.Display menu by pressing the <Esc> key. At the OSA90 main menu, press <Esc> again. When you see this prompt

```
Exit from OSA90 (Y/<N>):
```

click the left-hand mouse button to confirm (or press the <Y> key).

Upon exit from OSA90, the Empipe windows reappear on the screen. A dialogue box appears, allowing you to specify the file name for saving the optimized geometry.

Click on the "OK" button to accept the default file name "tpad_o.geo".

Empipe automatically invokes ***xgeom*** to display the geometry, as illustrated in Fig. 5.13.

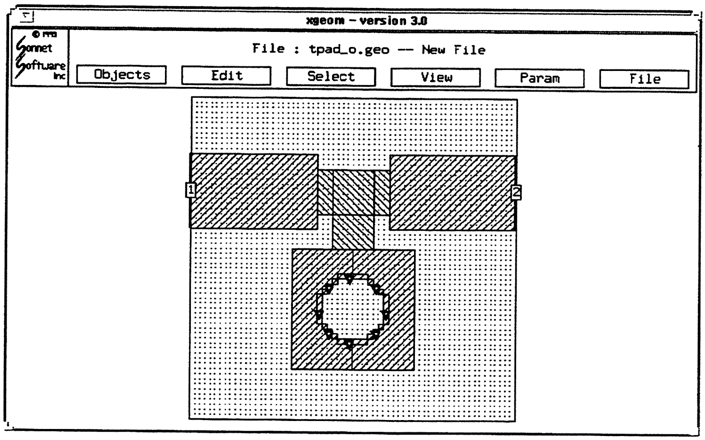


Fig. 5.13 The optimized geometry of the attenuator.

The optimized geometry saved in tpad_o.geo is snapped to the nearest grid. For example, the optimized value for the parameter L1 is 13.1518 mils. In tpad_o.geo, L1 is rounded, with respect to the cell size 1 mil × 1 mil, to 13 mils.

However, you can see the optimized solution without truncation in the "Select Variables" window, as shown in Fig. 5.14.

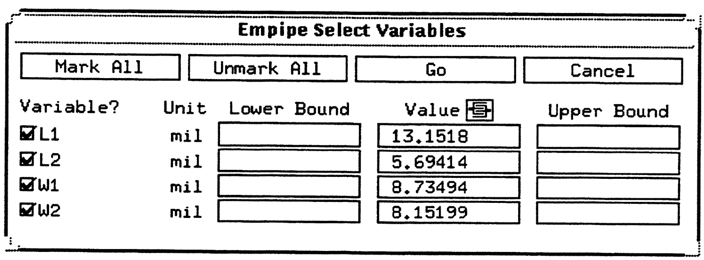


Fig. 5.14 The optimized solution without truncation.

Concluding the Tutorial

Click on the "Cancel" button in the "Select Variables" window. This brings you back to the main window of Empipe. Click on the "Quit" button. You will be prompted whether you wish to save the project file. Click "No" and Empipe will exit.

6

Tutorial: A Resistor

6

Tutorial: A Resistor

6.1 Introduction

This chapter is the fourth segment of the series of tutorials which systematically introduces you to the various features of Empipe.

We recommend that you follow the tutorials in the order that they are presented. At least you should study the introductory tutorial in Chapter 3 before any other chapters.

What You Will Learn From This Tutorial

1. How to capture metallization resistivity loss as an Empipe element parameter.

2. How to modify the OSA90 input file to redefine the ports.

3. How to display *Z* parameters which are converted by OSA90 from the *S* parameters calculated by ***em***.

4. How to define a parameter sweep for simulation and display.

The ***em*** simulation of this simple circuit takes approximately 4 seconds per frequency on a SPARCstation 10.

Are You in the Right Subdirectory?

All the operations in this tutorial should take place within the subdirectory which contains your copy of the Empipe examples (the instructions for copying the Empipe examples are found in Chapter 1). If you are not already in that subdirectory, type

cd <*Your Empipe Example Subdirectory*>

When this symbol appears on the left-hand side column, it highlights text that describes hands-on actions. You can take a "short-cut" through the tutorial by following this symbol and skip over the commentaries.

Description of the Example

The structure considered in this tutorial is based on one of the standard examples provided by Sonnet Software for *em* ("res400.geo" in the Sonnet example subdirectory). It is a simple resistor consisting of metallization squares.

The modified ".geo" file for this tutorial is named "em_res0.geo" in the Empipe example subdirectory.

You can use *xgeom* to view the geometry by typing

```
xgeom em_res0.geo
```

The *xgeom* display is shown in Fig. 6.1.

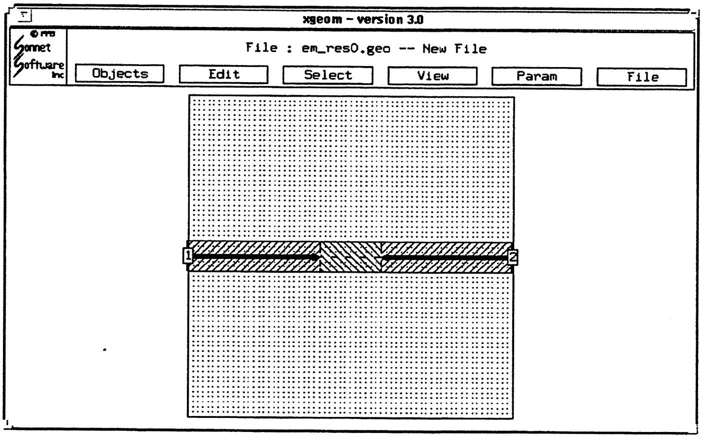


Fig. 6.1 The nominal geometry of the resistor.

The substrate is 100 micron thick with a relative dielectric constant of 12.9. The cell size is 10 × 10 microns.

6.2 Defining Metallization Loss as a Parameter

The resistor is constructed from squares of high resistivity metallization. In em_res0.geo, the DC resistivity is defined as 150 ohms/square and the structure consists of two squares. This produces a nominal resistance of 300 ohms.

We wish to capture two parameters for the resistor: the DC resistivity and the number of squares.

Changing the DC Resistivity Using *xgeom*

We can use *xgeom* to edit the DC resistivity parameter using the command

```
Param→Metallization
```

The *xgeom* window is depicted in Fig. 6.2, where the DC resistivity for the metal type "resis" is shown as 150 ohms/square. To change it, you would simply click the mouse button on the number "150" and *xgeom* will prompt you for a new value.

The Empipe example file set includes the file "em_res1.geo" which represents an incremental change in the DC resistivity. The DC resistivity is changed from 150 ohms/square in em_res0.geo to 160 ohms/square in em_res1.geo.

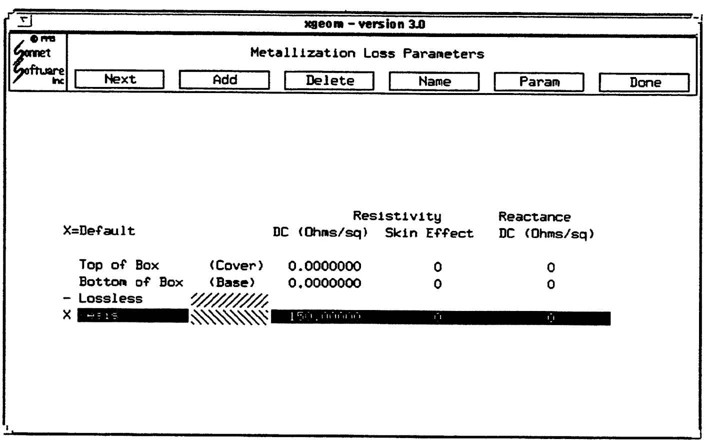


Fig. 6.2 xgeom window for editing metallization parameters.

Parameterizing the Number of Squares

In addition to the resistivity parameter, the resistance of the resistor is also controlled by the length of the metal. But instead of defining the length as a parameter, we choose to define the number of squares as a parameter. This serves to illustrate Empipe's ability to handle abstract, non-geometrical parameters.

The number of squares is 2 for the nominal geometry in em_res0.geo. We created an additional file "em_res2.geo", in which the number of squares is changed to 3, as illustrated in Fig. 6.3.

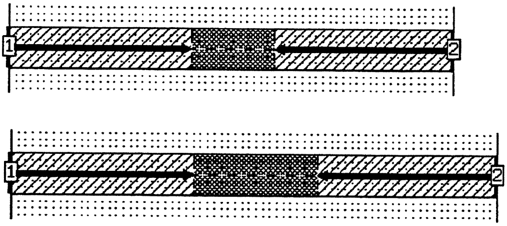


Fig. 6.3 Incremental change in the number of squares.

Geometry Capture by Empipe

Type

```
empipe em_res
```

The Empipe Geometry Capture form editor window is depicted in Fig. 6.4. The data shown in the window is retrieved from the file em_res.inc which is provided with the Empipe example files.

The *em* analysis control file "em_res.an" contains a single frequency of 2 GHz.

We name the parameters RDC and N_SQUARES. These names are of course arbitrary and we could choose other names if so desired.

The incremental change we made to RDC is from 150 to 160, and the number of divisions is specified as 1. This means a grid size of 10. This grid is not imposed by *em*, since it allows arbitrary resistivity values. We assign a discretization grid to the parameter RDC, not because it is necessary, but because it reduces the number of potential *em* simulations. Empipe will invoke *em* only if RDC changes by more than one grid (10 ohms/square). For changes in RDC within one grid, Empipe will rely on interpolation to obtain simulation results. So long as we choose a grid that is reasonably small, this scheme will improve the efficiency without excessive loss of accuracy.

A similar rationale applies to the parameter N_SQUARES. Between the files em_res0.geo and em_res2.geo, the parameter N_SQUARES is increased by 1. The size of one square is 60 × 60 microns. If we directly use the *em* cell size, which is 10 × 10 microns, as the parameter grid, then the perturbation would be 6 times the grid size. Instead we specify the "# of Divs" for N_SQUARES as 1, in order to reduce *em* simulations and expand the interval of interpolation.

Note also the "Unit Name" entry is "none" for both parameters. The parameter N_SQUARES does not have a physical unit, and the unit for RDC (ohms/square) is not used by Empipe. Empipe recognizes physical units for geometrical measurement only, including IN, MIL, M, CM, MM and UM.

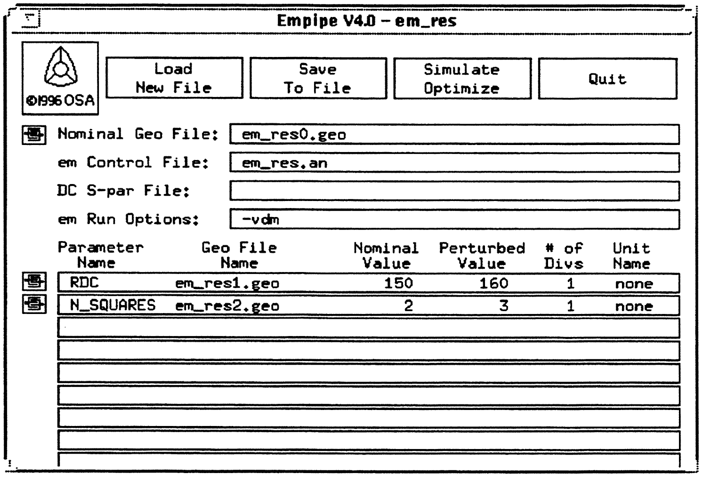


Fig. 6.4 Empipe Geometry Capture form editor.

6.3 S-Parameter Optimization

Our design objective is to find, by automated optimization, the value of RDC such that the resistance is 400 ohms (with N_SQUARES kept constant at 2). The answer to this simple problem is easy to determine (200 ohms/square).

By default, Empipe deals with the *S* parameters. We have to reformulate the specification on the resistance into a specification on the *S* parameters. The desired resistance of 400 ohms translates into a value of 0.8 for MS11 (with respect to the reference impedance of 50 ohms). In Section 6.4, we will show how to obtain the *Z* parameters directly in the OSA90 environment.

In the Empipe main window, click on the button

```
Simulate
Optimize
```

In the "Select Variables" window, click on the check box beside the parameter RDC to select it as an optimization variable. We will keep the parameter N_SQUARES constant, as depicted in Fig. 6.5.

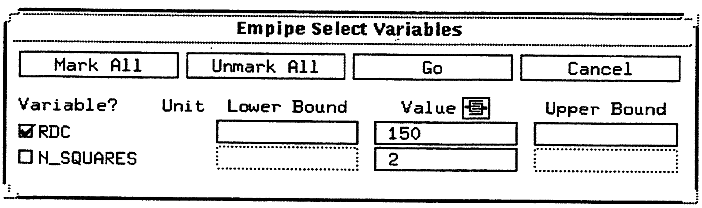


Fig. 6.5 The "Select Variables" window.

In the "Specifications" window, leave the response at the default (MS11). Change the type of specification from the default "<" to "=". Click on the goal box and type "0.8". Then click on the button

```
Add a new specification defined as follows
```

The "Specification" window should now contain

```
Specifications Currently Defined                    Delete
FREQ: 2GHz MS11 = 0.8
```

Simulation Before Optimization

In the "Select Variables" window, click on the button "Go". In the OSA90 window, click on "Display", and then click on "Xsweep". When the pop-up window appears, press <Enter> to accept the default setting. The simulation result is shown in Fig. 6.6.

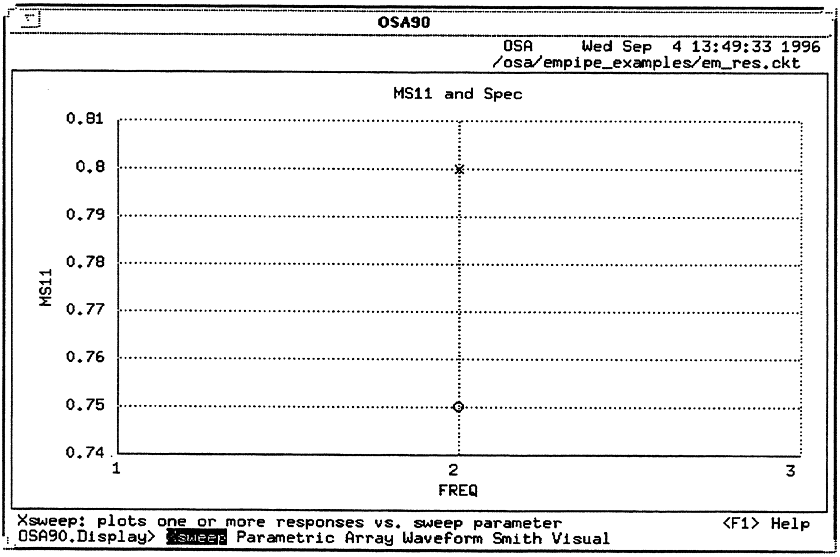


Fig. 6.6 MS11 and specification before optimization.

L1 Optimization

Click on the menu option "Optimize". Notice that the "L1" optimizer is automatically chosen, since we are trying to match the calculated MS11 to a specific value. Press <Enter> to accept the default setting.

The progress of optimization is reported on the screen:

```
Iteration   1/30 L1 Objective=0.050248
Iteration   2/30 L1 Objective=0.0484073
Iteration   3/30 L1 Objective=0.0446703
Iteration   4/30 L1 Objective=0.0370872
Iteration   5/30 L1 Objective=0.0229221
Iteration   6/30 L1 Objective=0.00123237
Iteration   7/30 L1 Objective=1.477e-05
Iteration   8/30 L1 Objective=1.19209e-08
Solution  L1 Objective=1.19209e-08
```

Displaying the Optimized Responses

Click on the menu option "Display" and then click on "Xsweep". Press <Enter> to accept the default setting. As shown in Fig. 6.7, the specification is met.

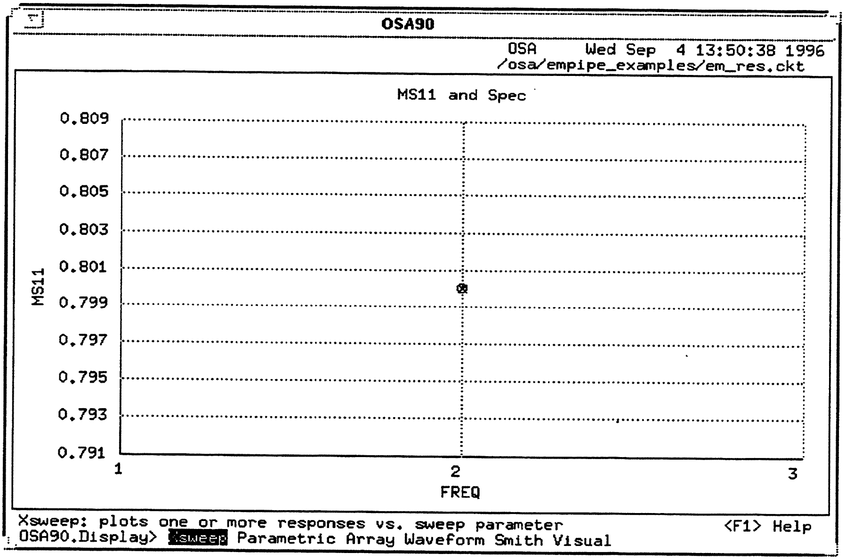


Fig. 6.7 MS11 meets specification after optimization.

6.4 Calculating Z Parameters

Exit from the "Display" menu by pressing <Esc>. At the OSA90 main menu, click on the "File" option.

You are now in the OSA90 input file editor. You can use the cursor keys to move around the file; use the <Back Space>, <Delete> and the alphanumeric keys to modify the text.

The Empipe element is defined in the Model block:

```
Model
#include "em_res.inc";

   EM_RES_RDC: ?200.343?;
   EM_RES_N_SQUARES: 2;

   EM_RES  1 2 0
      RDC=EM_RES_RDC  N_SQUARES=EM_RES_N_SQUARES;

   PORTS 1 0 2 0;

   CIRCUIT;

   MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);
end
```

The labels "EM_RES_RDC" and "EM_RES_N_SQUARES" represent the two parameters of the Empipe element "EM_RES". The pair of question marks following the label "EM_RES_RDC" indicates that the "RDC" parameter is defined as an optimization variable. Notice that the optimized value is 200.343 (ohms/square) which is very close to the expected value of 200 ohms/square.

Redefining the Ports

The Model block contains this statement:

```
PORTS 1 0 2 0;
```

It reflects the fact that in the ".geo" file the resistor is defined as a two-port, as illustrated in Fig. 6.8, where nodes 1 and 2 represent the edges of the metal and node 0 represents the ground.

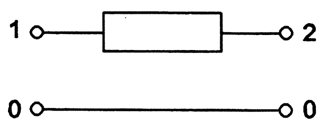


Fig. 6.8 The em port configuration for the resistor.

It will be more convenient to analyze the characteristics of the resistor if we redefine the circuit as a one-port, as shown in Fig. 6.9.

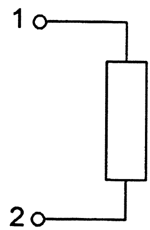


Fig. 6.9 The resistor defined as a one-port.

We can redefine the ports by editing the **Model** block. Change this statement

```
PORTS 1 0 2 0;
```

to

```
PORT 1 2;
```

OSA90 will automatically convert the two-port data produced by ***em*** to the appropriate one-port data.

We need to modify another statement in the **Model** block:

```
MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);
```

This is meant to calculate the two-port *S* parameters in dB. Since now we have only one port, we can either change the array dimension or delete the statement entirely.

The easiest thing to do is to comment out the statement by inserting an exclamation mark at the beginning of the statement, as

```
! MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);
```

OSA90 treats any text following an exclamation mark as a comment (until the end of line).

Modifying the Sweep Block

The **Sweep** block of the **OSA90** input file controls the circuit simulation. For the resistor, the initial contents of the **Sweep** block are as follows.

```
Sweep
   AC: FREQ: 2GHz
       MS MS_DB PS
      {XSWEEP Title="MS11 and Spec"
       Y=MS11
       XMIN=1GHz XMAX=3GHz NXTICKS=2 X_title=FREQ
       SPEC=(at 2GHz, = 0.8)};
end
```

The keyword "AC" signals a small-signal AC simulation. The labels "MS", "MS_DB" and "PS" represent the *S*-parameter responses of interest.

For the resistor, it is much more convenient to interpret the *Z* parameters than the *S* parameters. The **OSA90** built-in label **RZ11** represents the real part of the Z_{11} parameter, which directly gives us the resistance of the resistor.

Modify the **Sweep** block to

```
Sweep
   AC: FREQ: 2GHz
       RZ11;
end
```

Modifying the Specification Block

The initial contents of the **Specification** block of the **OSA90** input file are as follows.

```
Spec
   AC: FREQ: 2GHz MS11 = 0.8;
end
```

Modify the **Specification** block to

```
Spec
   AC: FREQ: 2GHz RZ11 = 400;
end
```

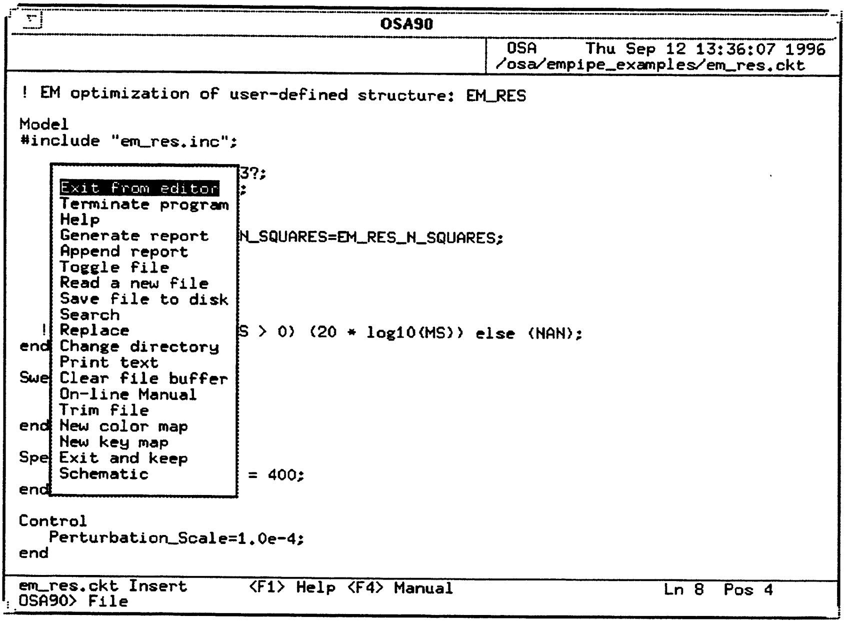
The Modified OSA90 Input File

The modified OSA90 input file is as follows.

```
Model
#include "em_res.inc";

   EM_RES_RDC: ?200.343?;
   EM_RES_N_SQUARES: 2;

   EM_RES  1 2 0
      RDC=EM_RES_RDC  N_SQUARES=EM_RES_N_SQUARES;

   PORT 1 2;

   CIRCUIT;

  ! MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);
end

Sweep
   AC: FREQ: 2GHz
       RZ11;
end

Spec
   AC: FREQ: 2GHz RZ11 = 400;
end

Control
   Perturbation_Scale=1.0e-4;
end
```

Simulation and Display of the Z Parameters

We have finished editing the OSA90 input file. To exit from the file editor, click the left-hand mouse button anywhere within the file editor. A menu box appears on the screen.

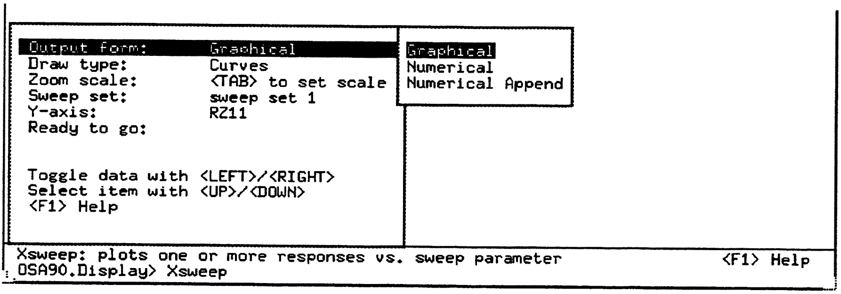


Click on the option "Exit from editor". You are back at the OSA90 main menu.

Click on the menu option "Display" and then click on "Xsweep". A pop-up window appears, showing the various display options.

Here we would like to demonstrate the OSA90 feature of numerical display. The selected circuit responses are printed on the screen as numerical outputs instead of graphical plots.

In the pop-up window, click on the option "Output form". You will see a list of choices.



The default choice is "Graphical". Change it by clicking on "Numerical". Then click on the "Ready to go" line.

The response RZ11 is displayed numerically on the screen:

```
! Parameter Sweep
 PARAMETER FREQ=2;
 FORMAT RZ11;
        400.4
```

It shows that the resistance is very close to the desired value of 400 ohms. The reason that it is not exactly 400 ohms is that we did not directly specify the *Z*-parameter response for optimization but reformulated it into a specification on the *S*-parameter response. The specification of 0.8 on MS11 was determined ignoring the structure reactance at 2 GHz, so it does not translate precisely to a 400 ohms resistance. Now we can re-optimize the circuit with the explicit specification on RZ11.

Optimization of the Z-Parameter Response

Exit from the numerical display window by pressing <Esc>. Then, exit from the "Display" menu by pressing <Esc>. At the OSA90 main menu, click on "Optimize". Press <Enter> to accept the default setting.

The progress of optimization is reported on the screen:

```
Iteration   1/30 L1 Objective=0.372966
Iteration   2/30 L1 Objective=0.000610983
Iteration   3/30 L1 Objective=2.25717e-05
Solution  L1 Objective=2.25717e-05
```

After the optimization is completed, click on the menu option "Display" and then click on "Xsweep". In the pop-up window, click on the option "Output form". From the list of choices, click on "Numerical". Then click on the "Ready to go" line.

The response RZ11 is displayed numerically on the screen:

```
! Parameter Sweep
 PARAMETER FREQ=2;
 FORMAT RZ11;
        400
```

As expected, we have obtained a more precise solution by defining the specification for optimization directly on the *Z*-parameter response.

6.5 Defining a Parameter Sweep

This section demonstrates parameter sweeps. We are already familiar with frequency sweep. In a similar manner, we define a range in which a selected parameter is swept with a given step size. We will be able then to plot the responses versus the swept parameter.

Using the resistor example, we will create two parameter sweeps, involving RDC and N_SQUARES, respectively.

Defining the Parameter Sweep of RDC

Exit from the numerical display window by pressing the <Esc> key. Then exit from the "Display" menu by pressing <Esc>. At the OSA90 main menu, click on the menu option "File". Now you are in the OSA90 input file editor.

Following the modification of the preceding section, the **Sweep** block contains:

```
Sweep
   AC: FREQ: 2GHz
       RZ11;
end
```

Insert one line of text into the **Sweep** block as follows.

```
Sweep
   AC: FREQ: 2GHz
       EM_RES_RDC: 150 200 250
       RZ11;
end
```

The added line defines a sweep of the label EM_RES_RDC which represents the parameter RDC of the element EM_RES.

In OSA90, the range of a parameter sweep can be defined by either

label: from *x1* to *x2* step=*x3*

or

label: *x1 x2 ... xn*

In the first case, the parameter represented by *label* is swept from the starting value *x1* to the stop value *x2* with the uniform step given by *x3*. In the second case, the parameter sweep is defined over *n* discrete values, given by *x1, x2, ..., xn*, which are not necessarily uniformly spaced.

Hence, the sweep of EM_RES_RDC can also be defined as

```
EM_RES_RDC: from 150 to 250 step=50
```

Displaying the Parameter Sweep of RDC

Exit from the file editor by clicking the left-hand mouse button in the editor window. As the menu box appears, click on the option "Exit from editor". At the OSA90 main menu, click on "Display". Wait for the simulation to finish and then click on "Xsweep". In the pop-up window press <Enter> to accept the default setting.

The graphical display of the response RZ11 versus the parameter RDC is shown in Fig. 6.10.

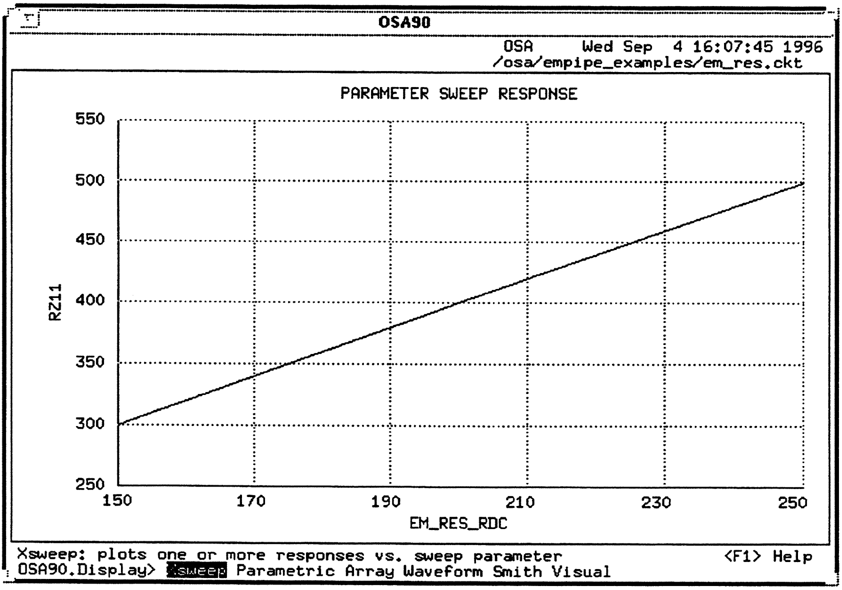


Fig. 6.10 Parameter sweep of RDC.

Defining the Parameter Sweep of N_SQUARES

Exit from the display menu by pressing the <Esc> key. At the OSA90 main menu, click on the menu option "File". You are back in the file editor.

We wish to define an additional sweep of the parameter N_SQUARES.

Insert a new statement to the Sweep block as follows.

```
Sweep
   AC: FREQ: 2GHz
       EM_RES_RDC: 150 200 250
       EM_RES_N_SQUARES: 1 2 3
       RZ11;
end
```

Exit from the file editor by clicking the left-hand mouse button in the editor window. As the menu box appears, click on the option "Exit from editor". At the OSA90 main menu, click on "Display". Wait for the simulation to finish and then click on "Xsweep".

A pop-up window appears as

```
Output form:        Graphical
Draw type:          Curves
Zoom scale:         <TAB> to set scale
Sweep set:          sweep set 1
Y-axis:             RZ11
X-axis:             EM_RES_N_SQUARES
EM_RES_RDC:         150
Ready to go:

<ENTER> = go or <ESC> = cancel
Select item with <UP>/<DOWN>
<F1> Help
Xsweep: plots one or more responses vs. sweep parameter          <F1> Help
OSA90.Display> Xsweep
```

Since we have defined two sweep parameters, we can choose either one of them to be displayed as the X-axis. By default, the second sweep parameter is chosen as the X-axis, which in this case is "EM_RES_N_SQUARES". In other words, the display will show the response RZ11 versus the parameter sweep of "EM_RES_N_SQUARES". For the other sweep parameter, namely "EM_RES_RDC", we can select one of the available values (in this case 150, 200 and 250). The default setting shows the value for "EM_RES_RDC" as 150.

Press <Enter> to accept the default setting. The graphical display is shown in Fig. 6.11.

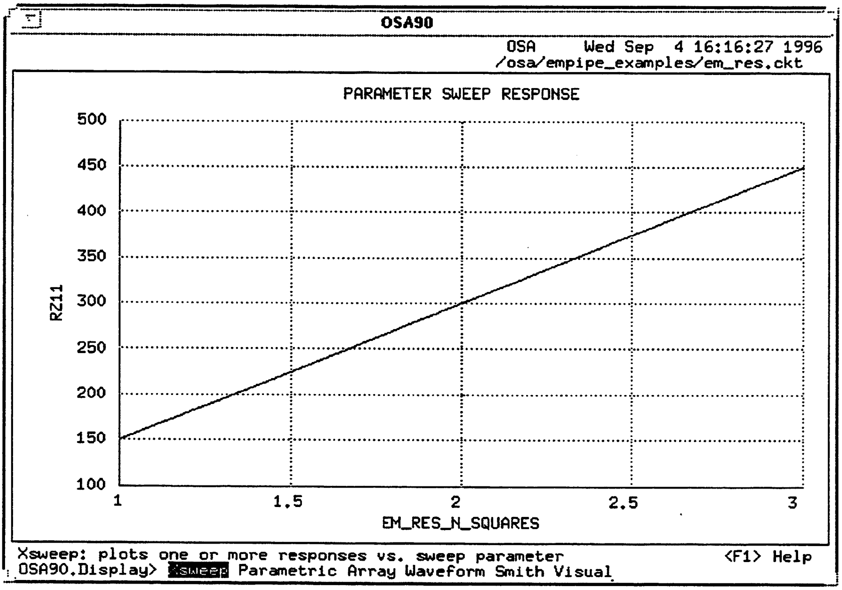
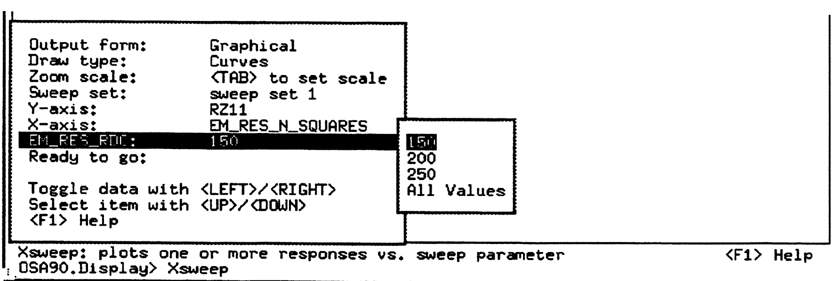


Fig. 6.11 Parameter sweep of N_SQUARES.

Displaying Both Sweep Parameters

Click on the menu option "Xsweep". After the pop-up window appears, click on the line "EM_RES_RDC:". A list of available choices appears as



Click on the last choice "All Values". Then click on the line "Ready to go". The graphical display is shown in Fig. 6.12.

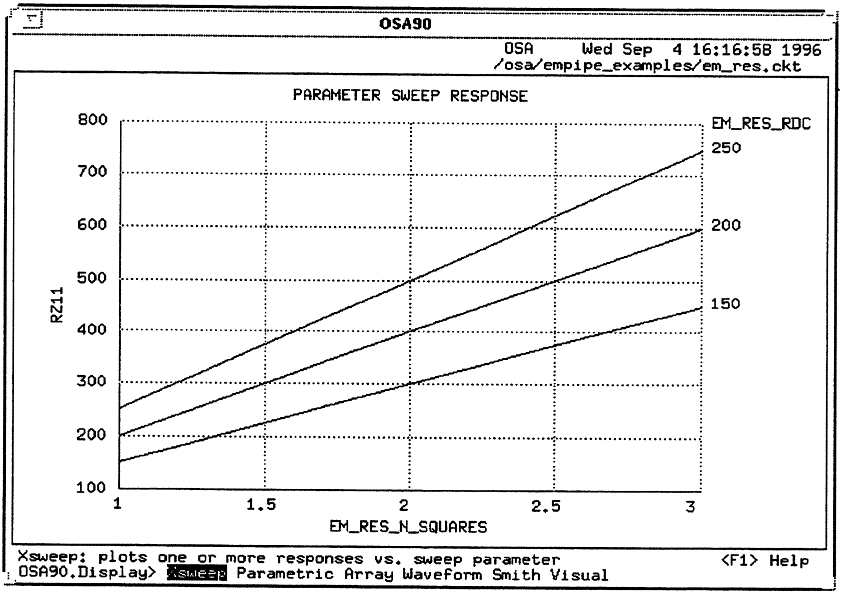


Fig. 6.12 Parameter sweep of both RDC and N_SQUARES.

Concluding the Tutorial

Exit from the display menu by pressing the <Esc> key. Exit from OSA90 by pressing <Esc> again and pressing <Y> when prompted for confirmation.

Upon exit from OSA90, the Empipe windows reappear on the screen. A dialogue box appears, allowing you to specify the file name for saving the optimized geometry. Click on the "OK" button to accept the default file name. Then, click on the "Cancel" button in the "Select Variables" window. Click on the "Quit" button in the Empipe main window. You will be prompted whether you wish to save the project file. Click "No" and Empipe will exit.

7

Tutorial: An MIM Capacitor

7

Tutorial: An MIM Capacitor

7.1 Introduction

This chapter is the last segment of the series of tutorials which systematically introduces you to the various features of Empipe.

We recommend that you follow the tutorials in the order that they are presented. At least you should study the introductory tutorial in Chapter 3 before any other chapters.

What You Will Learn From This Tutorial

1. How to capture parameters for the dielectric layer.
2. How to use OSA90 to convert the S parameters calculated by ***em*** to Y parameters.
3. How to formulate user-defined responses in OSA90.
4. How to display parametric plots.

The ***em*** simulation involved in this tutorial takes less than 10 seconds per frequency on a SPARCstation 10.

Are You in the Right Subdirectory?

All the operations in this tutorial should take place within the subdirectory which contains your copy of the Empipe examples (the instructions for copying the Empipe examples are found in Chapter 1). If you are not already in that subdirectory, type

`cd` *<Your Empipe Example Subdirectory>*

When this symbol appears on the left-hand side column, it highlights text that describes hands-on actions. You can take a "short-cut" through the tutorial by following this symbol and skip over the commentaries.

Description of the Example

The structure considered in this tutorial is based on one of the standard examples provided for *em* ("cap.geo" in the Sonnet example subdirectory). It is an MIM capacitor with a nominal capacitance of 1.4 pF.

The modified ".geo" file for this tutorial is named "em_cap0.geo" in the Empipe example subdirectory.

You can use ***xgeom*** to view the geometry by typing

```
xgeom em_cap0.geo
```

The ***xgeom*** display is shown in Fig. 7.1. The substrate is 50 micron thick with a relative dielectric constant of 12.88. The cell size is 10 × 10 microns.

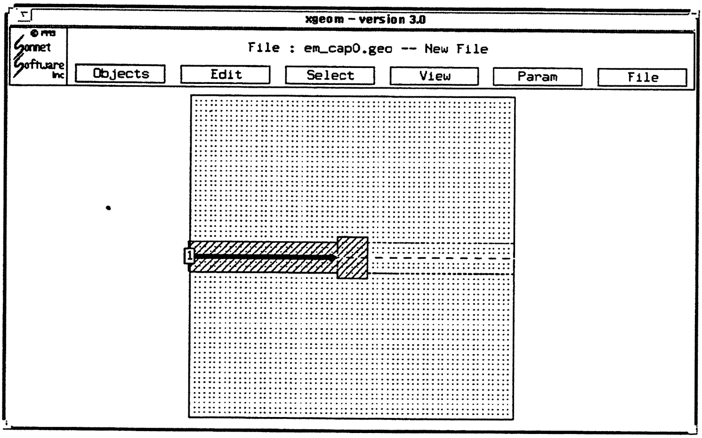


Fig. 7.1 The nominal geometry of the capacitor.

7.2 Parameterizing the Dielectric Layer

For the MIM capacitor, the capacitance depends on both the dimension of the metal patches and the characteristics of the dielectric layer between the patches.

The preceding tutorials have provided ample examples of parameterizing geometrical dimensions. Therefore, we turn our attention here to the dielectric layer. We wish to define two parameters: the dielectric constant and the thickness of the dielectric layer (i.e., the spacing between the two metallic patches).

Editing the Dielectric Layers Using *xgeom*

We can use ***xgeom*** to edit the dielectric layers with the command

```
Param→Dielectrics
```

The ***xgeom*** window is depicted in Fig. 7.2. The top layer represents air (open cover). The bottom layer represents the substrate. The middle layer is the one that we are interested in, namely the dielectric layer between the metallic patches.

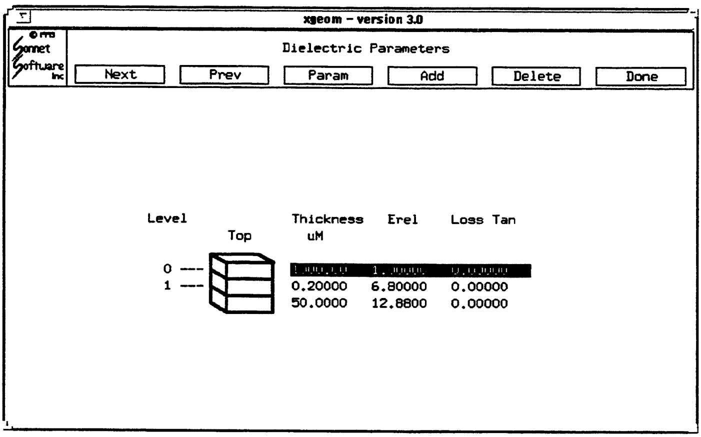


Fig. 7.2 xgeom window for editing the dielectric layers.

Changing the Dielectric Constant

In the *xgeom* window, the dielectric constant is listed under the heading "Erel". In Fig. 7.2, the dielectric constant of the middle layer is shown as 6.8. This is the value recorded in the nominal ".geo" file em_cap0.geo.

To change the value, you would simply click the mouse button on the number "6.8" and *xgeom* will prompt you for a new value.

To create an incremental change for Geometry Capture, we have changed the dielectric constant from 6.8 to 6.9. The change is saved in the file "em_cap1.geo" (included in the Empipe example files).

Changing the Dielectric Layer Thickness

In Fig. 7.2, the thickness of the middle layer is shown as 0.2 micron. To change the value, you simply click the mouse button on the number "0.2" and *xgeom* will prompt you for a new value.

To create an incremental change for Geometry Capture, we have changed the dielectric layer thickness from 0.2 to 0.3 micron. The change is saved in the file "em_cap2.geo" (included in the Empipe example files).

Geometry Capture by Empipe

Type

```
empipe em_cap
```

The Empipe Geometry Capture form editor window is depicted in Fig. 7.3. The data shown in the window is retrieved from the file em_cap.inc which is provided with the Empipe example files.

The *em* analysis control file "em_cap.an" contains one single frequency of 800 MHz.

We choose the names EPSR and T to identify the dielectric constant and dielectric layer thickness parameters, respectively.

The incremental change we made to EPSR is from 6.8 to 6.9. Although *em* allows arbitrary values for the dielectric constant, we treat EPSR as a discretized variable with a grid of 0.1. The reason for doing so is to reduce the number of potential *em* simulations. Empipe will invoke *em* only if EPSR changes by a significant amount (more than one grid, or 0.1). For smaller variations of EPSR, Empipe will utilize interpolation to provide results. So long as the grid selected is reasonably small, this scheme will improve the efficiency without excessive loss of accuracy.

Similarly, we impose a discretization grid of 0.1 micron on the parameter T, not because it is necessary, but in order to improve the efficiency.

Em Run-Time Options

Notice that the *em* run-time options for this example are defined as "-vd". The memory saver option "-m" is not included. The reason is that the simulation frequency is below the threshold recommended by *em*. If you include the "-m" option, you will get a warning message: "use of Memory Saver (-m) may result in error".

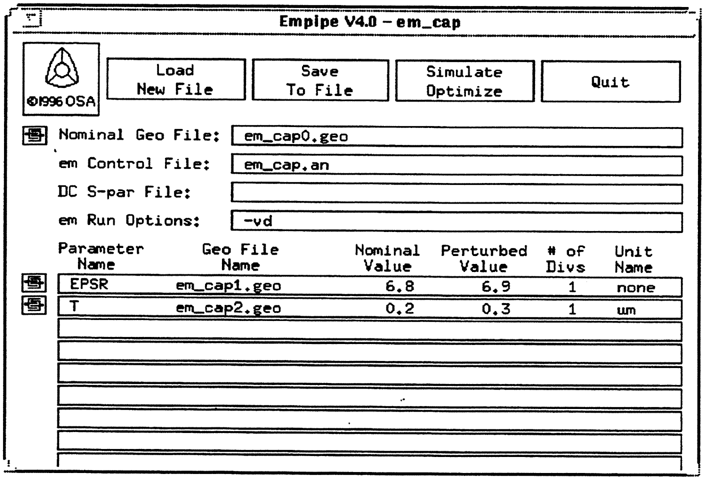


Fig. 7.3 Empipe Geometry Capture form editor.

7.3 S-Parameter Optimization

Our design objective is to determine, by automated optimization, the thickness of the dielectric layer (parameter T) such that the capacitance is 2 pF (with EPSR kept constant).

By default, Empipe deals with the *S* parameters. We have to reformulate the specification on the capacitance into a specification on the *S* parameters. The desired capacitance of 2 pF roughly translates into a value of 0.7 for MS11 (with respect to the reference impedance of 50 ohms). In Section 7.4, we will show how to define the capacitance as a user-defined response in the OSA90 environment.

In the Empipe main window, click on the button

```
Simulate
Optimize
```

In the "Select Variables" window, click on the check box beside the parameter T to make it an optimization variable. We will keep the parameter EPSR constant, as shown in Fig. 7.4.

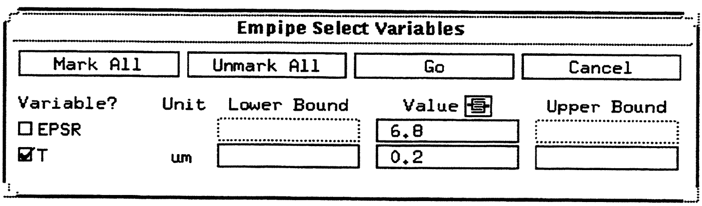


Fig. 7.4 The "Select Variables" window.

In the "Specifications" window, leave the response at the default (MS11). Change the type of specification from the default "<" to "=". Click on the goal box and type "0.7". Then click on the button

```
Add a new specification defined as follows
```

The "Specification" window should now contain

```
Specifications Currently Defined                    Delete
FREQ: 800MHz MS11 = 0.7
```

Simulation Before Optimization

In the "Select Variables" window, click on the button "Go". In the OSA90 window, click on "Display", and then click on "Xsweep". When the pop-up window appears, press <Enter> to accept the default setting. The simulation result is shown in Fig. 7.5.

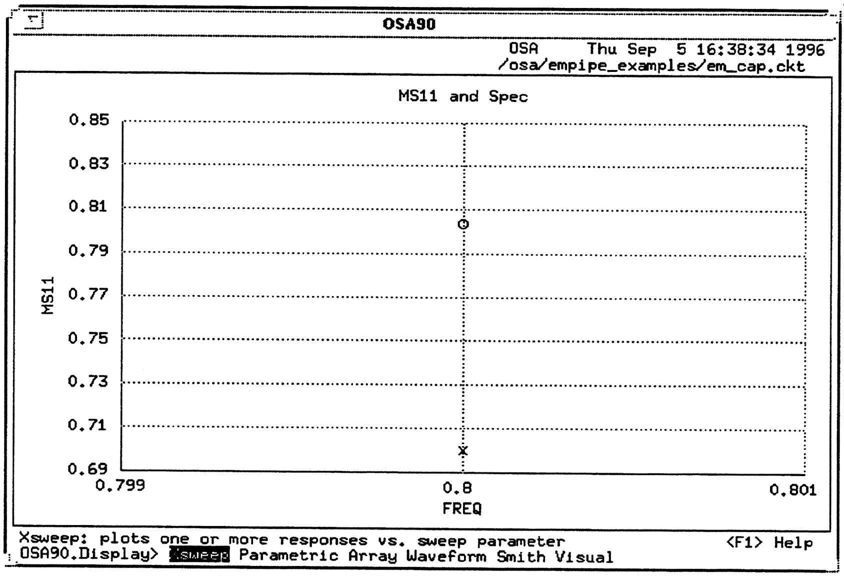


Fig. 7.5 MS11 and specification before optimization.

L1 Optimization

Click on the menu option "Optimize". Press <Enter> to accept the default setting.

The progress of optimization is reported on the screen:

```
Iteration   1/30 L1 Objective=0.103289
Iteration   2/30 L1 Objective=0.0984599
Iteration   3/30 L1 Objective=0.0889454
Iteration   4/30 L1 Objective=0.0704778
Iteration   5/30 L1 Objective=0.0356867
Iteration   6/30 L1 Objective=0.00149159
Iteration   7/30 L1 Objective=4.22001e-06
Iteration   8/30 L1 Objective=1.19209e-08
Solution  L1 Objective=1.19209e-08
```

Displaying the Optimized Responses

Click on the menu option "Display" and then click on "Xsweep". Press <Enter> to accept the default setting. As shown in Fig. 7.6, the specification is met.

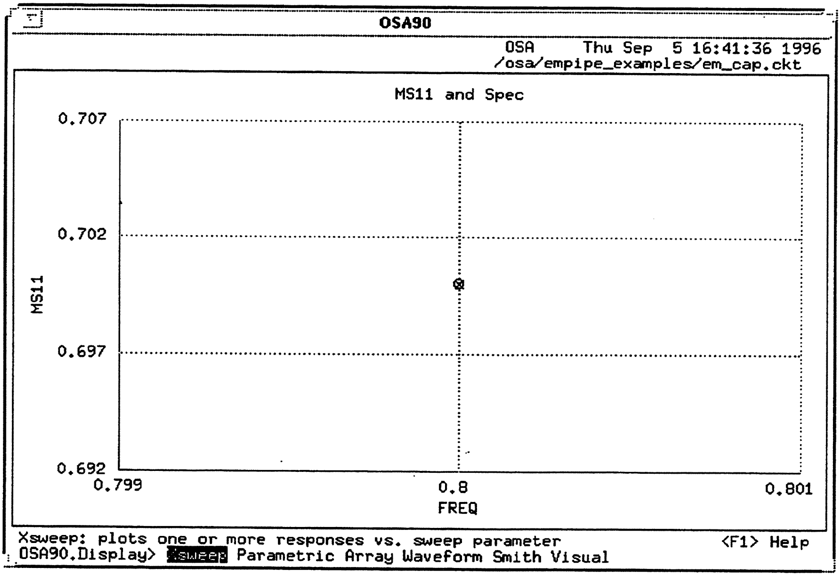


Fig. 7.6 MS11 meets specification after optimization.

7.4 Formulating User-Defined Responses

This section illustrates the feature of user-defined responses. Using the capacitor example, instead of showing the *S* parameters, we calculate and display the capacitance versus parameter sweeps of the dielectric constant EPSR and the dielectric layer thickness T.

Exit from the "Display" menu by pressing <Esc>. At the OSA90 main menu, click on the "File" option.

You are now in the OSA90 input file editor. You can use the cursor keys to move around the file; use the <Back Space>, <Delete> and the alphanumeric keys to modify the text.

The Empipe element is defined in the Model block:

```
Model
#include "em_cap.inc";

   EM_CAP_EPSR: 6.8;
   EM_CAP_T: ?0.157435?;

   EM_CAP  1 2 0
      EPSR=EM_CAP_EPSR  T=(EM_CAP_T * 1um);

   PORTS 1 0 2 0;

   CIRCUIT;

   MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);
end
```

The labels "EM_CAP_EPSR" and "EM_CAP_T" represent the two parameters of the Empipe element "EM_CAP". The pair of question marks following the label "EM_CAP_T" indicates that the "T" parameter is defined as an optimization variable.

Redefining the Ports

The Model block contains this statement:

```
PORTS 1 0 2 0;
```

It reflects the fact that in the ".geo" file the capacitor is defined as a two-port, as depicted in Fig. 7.7, where nodes 1 and 2 represent the two electrodes and node 0 represents the substrate.

It will be more convenient to analyze the characteristics of the capacitor if we redefine the circuit as a one-port, as shown in Fig. 7.8.

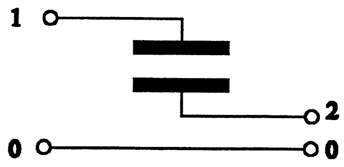


Fig. 7.7 The em port configuration for the capacitor.

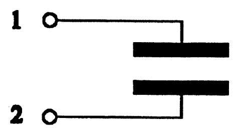


Fig. 7.8 The capacitor redefined as a one-port.

We can redefine the ports by editing the **Model** block. Change this statement

```
PORTS 1 0 2 0;
```

to

```
PORT 1 2;
```

OSA90 will automatically convert the two-port data produced by ***em*** to the appropriate one-port data.

We also need to modify another statement in the **Model** block:

```
MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);
```

This formula for the two-port *S* parameters in dB is no longer applicable to the redefined one-port circuit.

Comment out the statement by inserting an exclamation mark at the beginning of the line, as

```
! MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);
```

OSA90 treats any text following an exclamation mark as a comment (until the end of line).

Formulating Capacitance as User-Defined Response

Insert the following statement into the **Model** block just before the "**end**" line:

```
Capacitance = IY11 / (2 * PI * FREQ) * 1000;
```

The label IY11 represents the imaginary part of Y_{11}. OSA90 provides built-in labels for identifying the Y parameters: RY for the real parts and IY for the imaginary parts.

PI and FREQ are also built-in labels of OSA90, representing the constant π and the analysis frequency, respectively.

The default unit of FREQ is GHz. The multiplication by 1000 in the above formula converts the unit of **Capacitance** to pF.

OSA90 provides a comprehensive set of algebraic operations and mathematical functions to facilitate preprocessing of variables and postprocessing of responses. For further details please consult the *OSA90/hope User's Manual.*

The Modified Model Block

The **Model** block after the modifications should be as the following.

```
Model
#include "em_cap.inc";

   EM_CAP_EPSR: 6.8;
   EM_CAP_T: ?0.157435?;

   EM_CAP  1 2 0
      EPSR=EM_CAP_EPSR  T=(EM_CAP_T * 1um);

   PORT 1 2;

   CIRCUIT;

!   MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);

   Capacitance = IY11 / (2 * PI * FREQ) * 1000;
end
```

Parameter Sweeps of EPSR and T

In order to display the user-defined response Capacitance, we need to include it in the Sweep block (which controls the simulation ranges and outputs). We also wish to examine the effect of varying the dielectric parameters on the capacitance. To do that we need to define parameter sweeps of EPSR and T. (You should follow the tutorial in Chapter 6, where we discussed parameter sweeps, if you have not already done so.)

The original Sweep block contains

```
Sweep
   AC: FREQ: 800MHz
        MS MS_DB PS
       {XSWEEP Title="MS11 and Spec"
        Y=MS11
        XMIN=799MHz XMAX=801MHz NXTICKS=2 X_title=FREQ
        SPEC=(at 800MHz, = 0.7)};
end
```

where the labels "MS", "MS_DB" and "PS" represent the *S*-parameter responses.

Modify the Sweep block as follows.

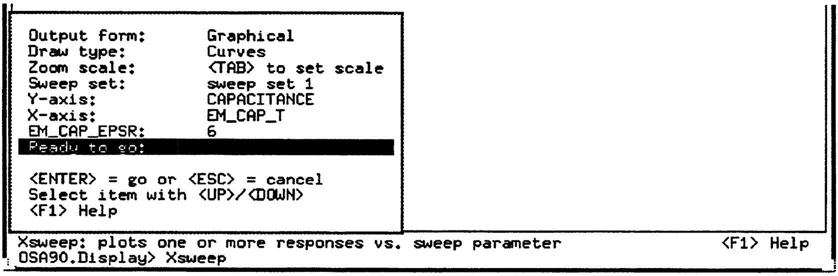
```
Sweep
   AC: FREQ: 800MHz
        EM_CAP_EPSR: 6 6.8 7.6
        EM_CAP_T: 0.1 0.2 0.3 0.4
        Capacitance;
end
```

The label EM_CAP_EPSR represents the parameter EPSR of the element EM_CAP and the label EM_CAP_T represents the parameter T. The user-defined response Capacitance is specified as the output.

Simulation and Display

Exit from the input file editor by clicking the left-hand mouse button. As the menu box appears, click on the option "Exit from editor". At the OSA90 main menu click on the menu option "Display". Wait for the simulation to finish and then click on the display menu option "Xsweep".

A pop-up window appears:



Click on the line "EM_CAP_EPSR". From the list of available choices, click on the line "All Values". Then, click on the line "Ready to go".

The graphical display is shown in Fig. 7.9.

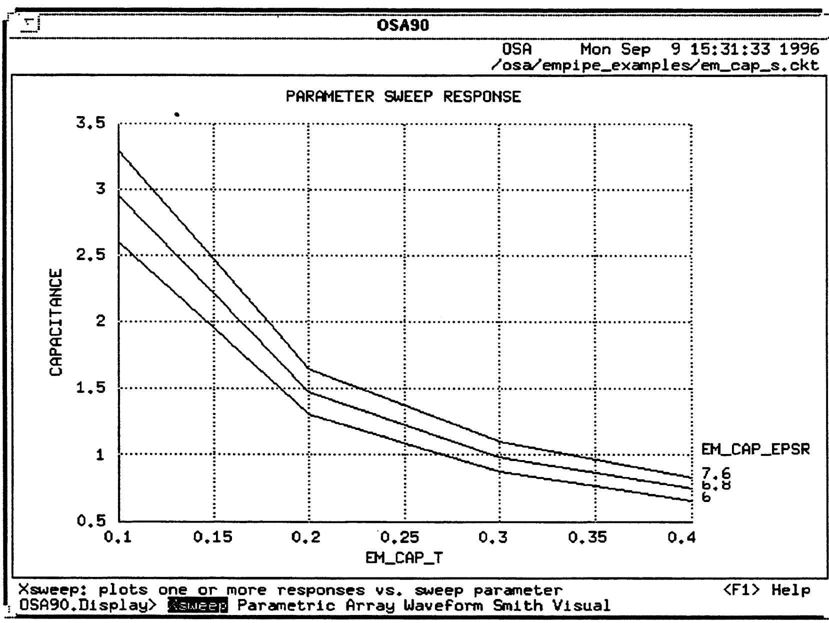


Fig. 7.9 Capacitance versus the dielectric layer thickness.

Capacitance Versus the Dielectric Constant

Click on the option "Xsweep". After the pop-up window appears, click on the line "X-axis". From the list of choices, click on the line "EM_CAP_EPSR". Click on the line "EM_CAP_T" and from the list of available choices click on the line "All Values". Then, click on the line "Ready to go".

The graphical display is shown in Fig. 7.10.

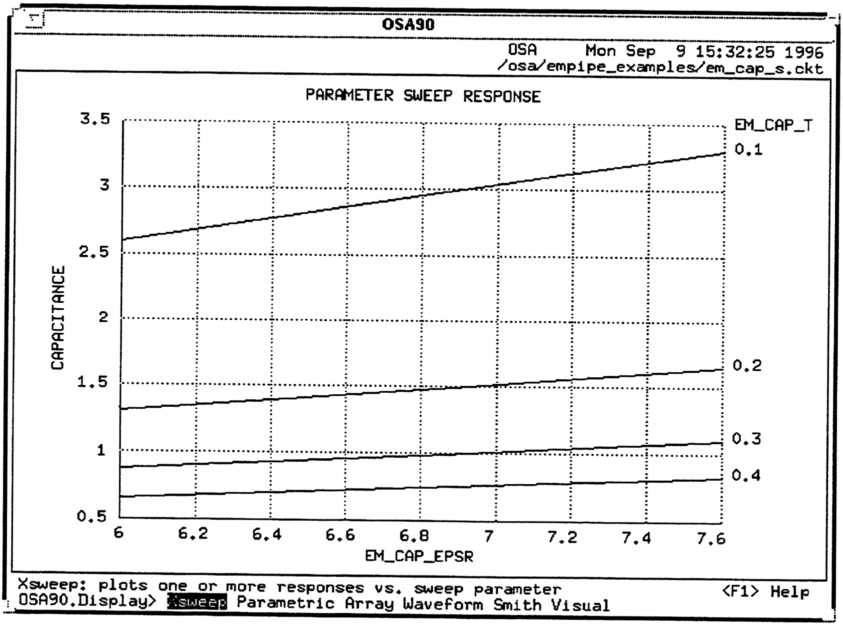


Fig. 7.10 Capacitance versus the dielectric constant.

7.5 Parametric Plots

The capacitance of an MIM capacitor is given by

$$C = A \epsilon / t$$

where A is the area of the metal (electrodes), ϵ is the dielectric constant (of the center layer) and t is the dielectric layer thickness. The results shown in Figs. 7.9 and 7.10 are consistent with this formula.

We can also use OSA90 to plot the capacitance versus $1/t$ instead of t. We can accomplish this using the parametric plot feature of OSA90. A parametric plot involves two responses, both of which are functions of the same sweep parameter. Instead of plotting a response versus the sweep parameter, we plot one response versus the other response.

For the example on hand, we will define $1/t$ as a user-defined response and then plot the capacitance (one response) versus $1/t$ (the other response), with t as the sweep parameter.

Editing the Input File

Exit from the display menu by pressing the <Esc> key. At the OSA90 main menu, click on the menu option "File".

In the Model block, the parameter t is represented by the label EM_CAP_T. Add to the Model block (just before the "end" line) this statement:

```
One_Over_T = 1 / EM_CAP_T;
```

The Model block after the modifications should contain the following.

```
Model
   ...
   ...

   Capacitance = IY11 / (2 * PI * FREQ) * 1000;
   One_Over_T = 1 / EM_CAP_T;
end
```

Then, include the label One_Over_T in the Sweep block as an additional response for the second sweep set, as

```
Sweep
   AC: FREQ: 800MHz
       EM_CAP_EPSR: 6 6.8 7.6
       EM_CAP_T: 0.1 0.2 0.3 0.4
       Capacitance One_Over_T;
end
```

Exit from the file editor by clicking the left-hand mouse button in the editor window. As the menu box appears, click on the option "Exit from editor". At the OSA90 main menu, click on "Display". Then, click on the option "Parametric". As the pop-up window appears, press <Enter> to accept the default setting.

The parametric plot of CAPACITANCE versus ONE_OVER_T is shown in Fig. 7.11. It clearly illustrates the linear dependence of the capacitance on $1/t$.

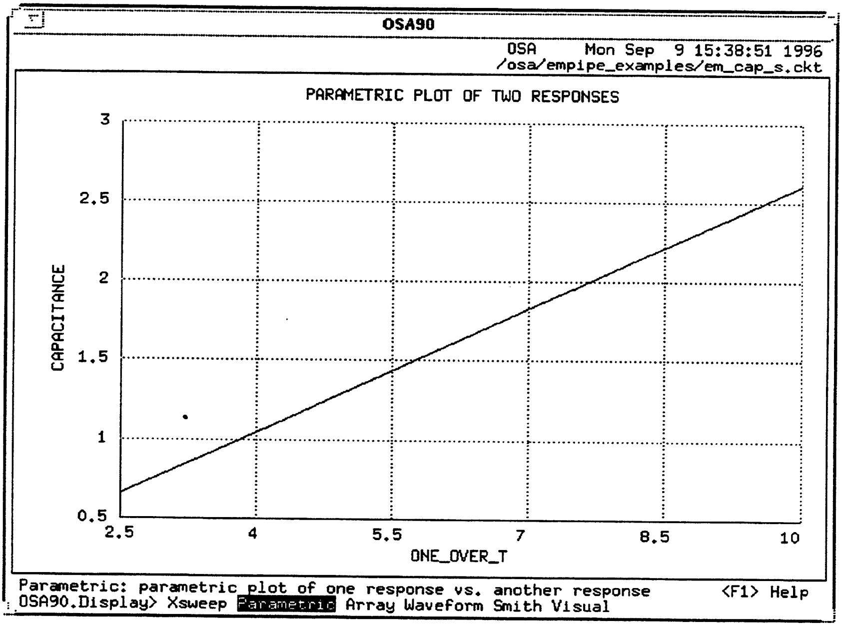


Fig. 7.11 Parametric plot of capacitance versus 1/t.

Concluding the Tutorial

Exit from the display menu by pressing the <Esc> key. Exit from OSA90 by pressing <Esc> again and pressing <Y> when prompted for confirmation.

Upon exit from OSA90, the Empipe windows reappear on the screen. By now, you should be able to exit from the Empipe program without requiring explicit instructions.

8

Empipe Geometry Capture

8

Empipe Geometry Capture

8.1 Introduction

We hope that the series of tutorials in Chapters 3 to 7 has acquainted you with the features and operations of Empipe.

The steps of using Empipe for EM optimization can be summarized as follows.

1. Start with a ".geo" file which represents the nominal structure.
2. Generate a set of ".geo" files which logically describes how the structure will evolve in relation to incremental changes in the designable parameters.
3. Invoke Empipe to process the set of ".geo" files by Geometry Capture.
4. Define optimization variables and specifications.
5. Perform simulation and optimization within the OSA90 environment.
6. Save the optimized geometry.

This chapter provides a formal description of the Geometry Capture procedure, focusing on steps 1 to 4.

8.2 Describing the Nominal Structure

The first step is to describe the nominal structure. You do this by creating a ".geo" file which contains, in accordance with the syntax specified by Sonnet Software, geometrical coordinates, substrate data, port definitions, etc.

Sonnet Software provides the *xgeom* program as the primary tool for manipulating ".geo" files. It also supply a utility program to convert GDSII files to ".geo" files.

Typically, the nominal geometry represents an initial design. It may be the result of a synthesis procedure, or an optimization using a circuit simulator, or an empirical design based on past experiences. As far as Empipe is concerned, it simply serves as a reference point.

em Analysis Control File

You also need an *em* analysis control file which defines the frequency points according to the syntax specified by Sonnet Software. By default, *em* analysis control files are given the extension ".an".

Testing the Files Before Using Empipe

We strongly recommend that you test the nominal ".geo" file and the analysis control file to make sure that they are acceptable to *em*. Empipe assumes that the ".geo" files provided to it are already validated. It does not perform a full-scale independent verification on the ".geo" file syntax.

Empipe Element Name

Each structure you wish to optimize using Empipe should be given a name. This name is also incorporated into the names of all the related files, serving as a common thread.

For example, the structure we considered in the tutorial of Chapter 4 is named "dfstub" (for double folded stub). The nominal ".geo" file is named dfstub0.geo, and the incremental ".geo" files are named dfstub1.geo, dfstub2.geo and dfstub3.geo. The *em* analysis control file is named dfstub.an. The Empipe element definition file is named dfstub.inc, the OSA90 input file is named dfstub.ckt, and the database file is named dfstub_1.dbs.

8.3 Creating Incremental Geo Files

You need to define a set of designable parameters. Typically they represent geometrical dimensions, but you can also include material parameters for the dielectric and metallization layers.

We use the terms "designable parameters" and "optimization variables" as two related but separate concepts. The designable parameters are candidates for optimization variables, but not all the designable parameters are necessarily optimized at the same time. Before an optimization starts, Empipe lets you choose which designable parameters are going to be optimization variables.

Empipe lets you define the designable parameters using the concept of Geometry Capture. It is general enough to accommodate arbitrary geometries and yet simple to understand and use.

Consider the example shown in Fig. 8.1. It has two designable parameters: W and L.

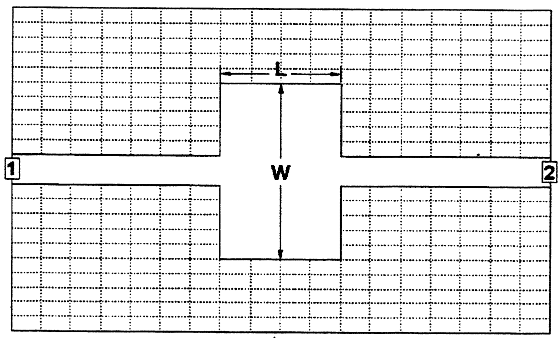


Fig. 8.1 A structure with two designable parameters.

Representing a Parameter by an Incremental Change

For each designable parameter, you need to create an additional ".geo" file. You make an incremental change in one of the designable parameters and show the corresponding change in the geometry with respect to the nominal structure.

Consider the example shown in Fig. 8.1. The incremental changes in the two designable parameters are illustrated in Fig. 8.2.

By comparing the nominal and incremental ".geo" files, Empipe automatically extracts the information necessary for translating a given set of parameter values to the corresponding geometrical coordinates.

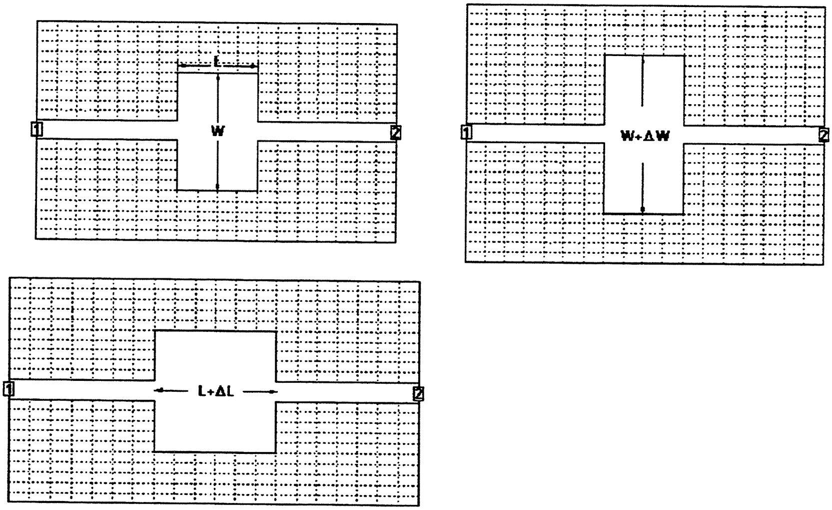


Fig. 8.2 Incremental changes in the parameters W and L.

Implicit Constraints On the Geometry

Geometry Capture allows you to express graphically subtle constraints on the geometry which are otherwise difficult to define.

For instance, the difference between Fig. 8.1 and Fig. 8.2 implies that the geometry remains symmetrical after changes in the parameter W. If we use Fig. 8.3 instead of Fig. 8.2 to represent an incremental change in the parameter W, then Empipe will interpret this as to imply that the parameter W has a one-sided effect on the geometry.

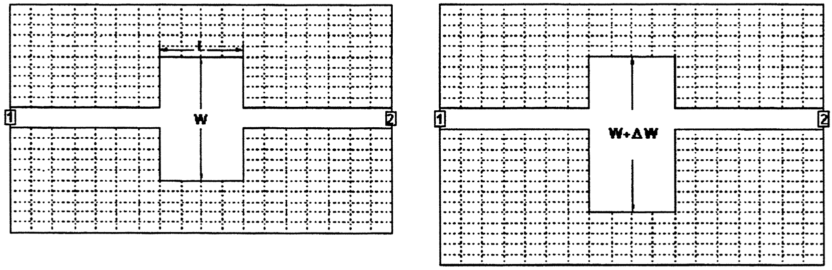


Fig. 8.3 One-sided incremental change in the parameter W.

Also, according to Fig. 8.2, the size of the substrate box varies with the parameter W to keep a constant margin between the metal and the substrate edges. If Fig. 8.4 is used instead of Fig. 8.2, then the substrate box size remains independent of the parameter W. (This would require, however, that the parameter W be bounded so that, as W may be increased during optimization, the metal will not grow beyond the substrate box. See the tutorial in Chapter 4 for an example on specifying bounds on variables.)

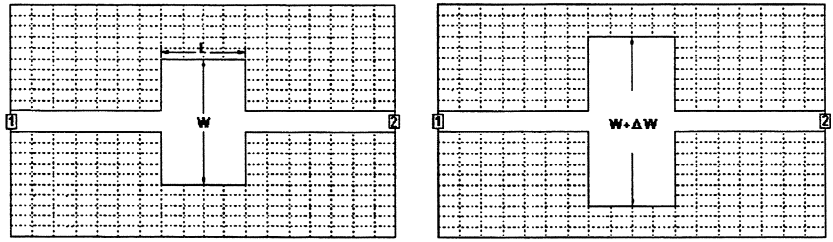


Fig. 8.4 The substrate box size is independent of the parameter W.

As another example, if Fig. 8.5 is used to represent an incremental change in the parameter L, the implication is that when L changes, the overall length of the structure remains the same. As the parameter L increases, the center polygon becomes longer and the feed lines become shorter, and vice versa.

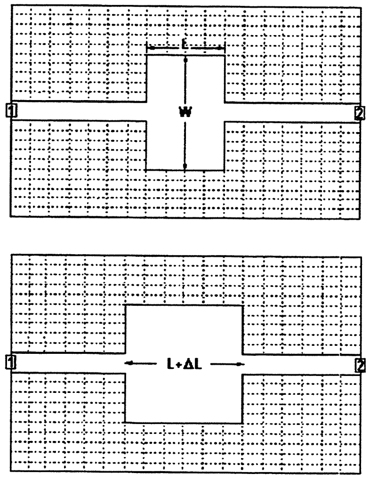


Fig. 8.5 A different definition of the parameter L.

Defining a Scaling Parameter

You can define parameters which scale the whole or part of the structure. For example, Fig. 8.6 represents an incremental change in a parameter which scales the center polygon in both the X- and Y-dimensions.

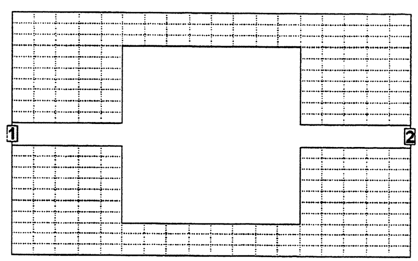


Fig. 8.6 Incremental change in a scaling parameter.

Rules for Creating Incremental Geo Files

The Cell Size Must Remain the Same

The *em* cell size in the incremental ".geo" files must remain exactly the same as that of the nominal ".geo" file. Parameter-dependent cell size is not permitted.

Be careful if you need to change the substrate box size. Be sure to choose the option of keeping the cell size constant when prompted by *xgeom*.

The Geometry Must Remain On-Grid

The nominal and incremental geometries must be on-grid. It means that the geometrical increments must be a multiple of the cell size.

You Cannot Add or Delete Polygons

The total number of polygons in the incremental ".geo" files must be identical to that of the nominal ".geo" file. Furthermore, the order in which the polygons appear in the ".geo" file must remain the same. When you need to stretch or shrink a polygon, utilize *xgeom*'s ability to move the individual vertices of the polygon. Sometimes you may think that instead of manipulating the vertices, it is easier just to delete the existing polygon and draw a new one. Do not do this, for *xgeom* may reshuffle the order of appearance of the polygons. This would confuse Empipe when it attempts to compare the ".geo" files.

You Cannot Change the Basic Shape of a Polygon

The basic shapes of the polygons in the incremental ".geo" files must remain the same as those of the nominal ".geo" file. For instance, you cannot change a polygon from a rectangle to a triangle.

8.4 Processing Geo Files by Empipe

Once you have prepared a complete set of ".geo" files, start Empipe by

```
empipe name
```

where *name* is the name of the structure, as discussed in Section 8.2.

Empipe first looks for the file *name*.inc in the current directory as an indication of whether or not the structure has been processed before. If the file exists, Empipe will retrieve the data from the file. If the file does not exist, then Empipe assumes that *name* represents a new element to be defined. In this case, the Empipe main window will be mostly blank, as depicted in Fig. 8.7.

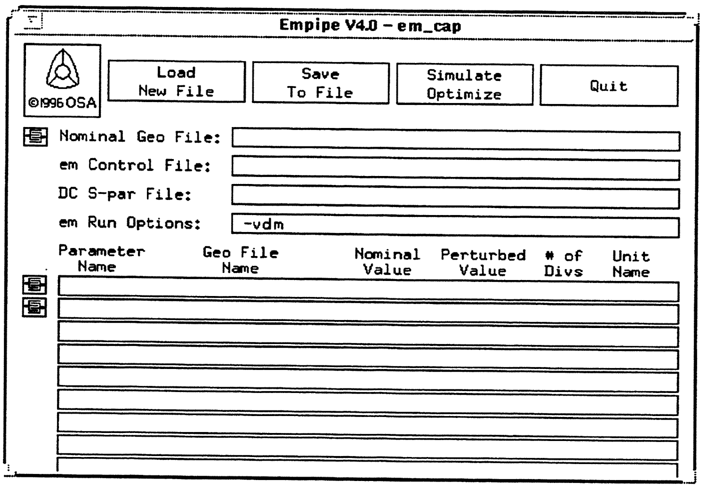


Fig. 8.7 Empipe Geometry Capture form editor.

Nominal Geo File

This entry identifies the nominal ".geo" file.

To change any entry in the form editor, click the left-hand mouse button on the box, and then type in the appropriate entry.

This button appears adjacent to the entry box. If you click on it, ***xgeom*** will be invoked to display the nominal ".geo" file.

em Control File

This entry identifies the *em* analysis control file. An analysis control file is required by *em*. It defines the frequencies for the *em* analysis.

DC S-par File

This is an optional entry. You can specify a file which contains the *S* parameters of the structure at DC. This is needed only if you wish to involve the Empipe element in DC or harmonic balance simulation (available through the OSA90/hope program).

The *em* analysis produces *S* parameters for AC only. The DC *S* parameters have to be obtained by other means (using a circuit simulator, for instance).

In the DC *S*-parameter file, if given, the data must be arranged in the following order:

```
S11  S12  ...  S1n
S21  S22  ...  S2n

...

Sn1  Sn2  ...  Snn
```

where *n* is the number of ports and `S11 ... Snn` are real numbers (DC *S* parameters cannot have nonzero imaginary parts).

If DC *S* parameters are needed during a nonlinear circuit simulation but are not supplied, they will be extrapolated from the AC *S* parameters computed by *em*. The impact on the accuracy of the results is unpredictable.

em Run Options

This entry allows you to specify the run-time options for *em*. For a complete list of the available options, please check the *em* User's Manual from Sonnet Software. The default options are -vdm (verbose, automatic de-embedding and memory saver).

Empipe operates on the *S* parameters calculated by *em*. The *em* run-time options "y" and "z" are incompatible with this requirement and therefore cannot be used.

Parameter Definition Data

The incremental changes for the designable parameters are described on the lines under the heading

Parameter Name	Geo File Name	Nominal Value	Perturbed Value	# of Divs	Unit Name

Each line corresponds to one parameter. We will discuss the individual fields (columns) in the following.

Parameter Name

This can be an arbitrary ASCII string of no more than 32 characters, such as "W", "Width", "L1", "EPSR" or "Dielectric_Constant".

Geo File Name

This entry identifies the ".geo" file which describes the geometry of the structure after an incremental change in the parameter value is made.

 Click on this button to invoke *xgeom* to view the ".geo" file.

Nominal Value

This refers to the value of the parameter represented by the nominal ".geo" file. It should be entered as a plain number, for the physical unit, if any, is entered as a separate item.

Perturbed Value

This refers to the parameter value after the incremental change.

Number of Divs

This entry measures the incremental change in terms of the *em* grid size. It is obtained by dividing the difference between the nominal value and the perturbed value by the *em* cell size along the appropriate dimension.

For example, suppose that the parameter is changed from 100 to 110 (we leave out the physical unit, whatever it may be). The incremental change is 10. If the geometrical change is along the X dimension of the layout and the grid size for the X dimension is 5, then the number of divisions is 2.

However, sometimes in order to preserve the geometrical symmetry, you may need to make symmetrical perturbations with respect to the plane of symmetry. In this case, the "grid" for the Empipe Geometry Capture may be different from the *em* grid. Typically, the smallest symmetrical perturbation is two times the *em* cell size, and hence the correct entry for "# of Divs" is half the number of *em* cells covered by the perturbation. The tutorial example in Chapter 5 illustrates this concept in detail.

Unit Name

This entry identifies the physical unit of the parameter. Permissible unit names include IN (inch), MIL (milli-inch), M (meter), CM (centimeter), MM (millimeter), UM (micron) and NONE (without unit).

Defining Dielectric and Metallization Parameters

In addition to geometrical dimensions, you can also define designable parameters for the dielectric and metallization layers of the structure.

In a ".geo" file, each dielectric layer has five parameters, namely thickness of the layer, relative dielectric constant, dielectric loss tangent, relative permeability and magnetic loss tangent. Each type of metallization has three parameters, namely DC resistivity, skin effect coefficient and surface reactance.

You can select any of these parameters for optimization by providing Empipe with an incremental ".geo" file in which the dielectric or metallization parameter value of interest is different from that of the nominal ".geo" file.

em does not impose a grid on dielectric and metallization parameters, i.e., *em* can analyze structures with arbitrary dielectric and metallization parameter values. However, if we define a designable dielectric or metallization parameter without discretization, then during optimization whenever there is a change in that parameter value, no matter how small, *em* will have to be invoked to re-analyze the structure.

We can reduce the number of *em* simulations by imposing a discretization grid on each designable dielectric or metallization parameter. *em* will be invoked only if the parameter value has changed by more than one grid. For variations of the parameter value within one grid, Empipe will utilize interpolation to obtain simulation results. So long as we choose a grid that is reasonably small, this scheme will improve the efficiency without excessive loss of accuracy.

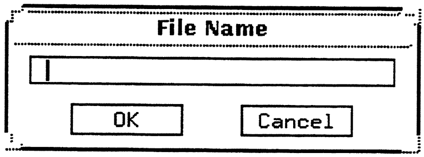
From the information you entered into the form editor, Empipe determines the size of the interpolation interval by dividing the difference between the nominal parameter value and the perturbed value by the number of divisions.

The tutorials in Chapters 6 and 7 demonstrate in detail the parameterization of dielectric and metallization layers.

Menu Buttons in the Empipe Main Window

Load New File

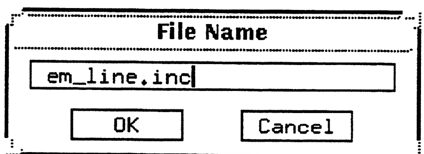
Click on this button to load an Empipe element definition file into the Empipe window. You will be prompted for the new file name:



Empipe then looks for the file *name*.inc. If it is found, the data contained in the file will be retrieved and displayed in the window. Otherwise, Empipe will assume that *name* identifies a new structure to be created. In either case, the data previously contained in the window is overwritten.

Save To File

Click on this button to save the current data in the Empipe window to a disk file. You will see a prompt similar to this one:



.The current name is offered as the default, but you have the opportunity to change it to a new name. Utilizing this feature, you can load an existing structure, make some modifications and then save it under a new name.

Simulate Optimize

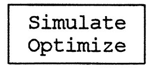
Click on this button to start simulation and optimization. It leads to separate Empipe windows for selecting optimization variables and formulating design specifications, which is the subject of the following section (Section 8.5).

Quit

Click on this button to exit from Empipe.

8.5 Optimization Variables and Specifications

To initiate simulation and optimization, click on this button in the Empipe form editor:



Two new windows entitled "Empipe Select Variables" and "Empipe Specifications" will appear, as depicted in Fig. 8.8.

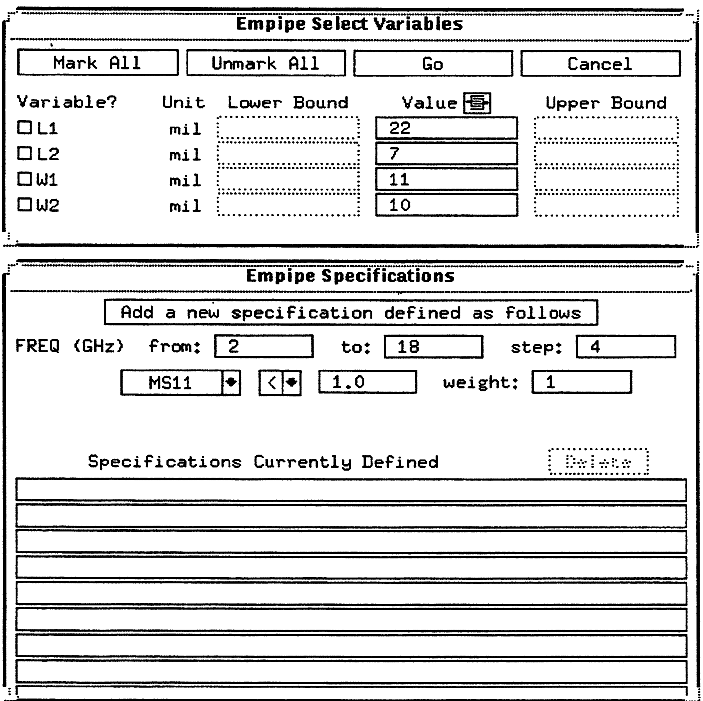


Fig. 8.8 The "Select Variables" and "Specifications" windows.

Selecting Optimization Variables

All the designable parameters are listed as candidates for optimization variables. For each parameter, there is a check box under the heading "Variable?" and adjacent to the parameter name. If you click on this box, a check mark will appear in the box to indicate that the associated parameter is selected as an optimization variable. If you click on the box a second time, the check mark will disappear. In other words, the check box acts as a toggle switch, turning on and off the status of selection.

Alternatively, you can click on the <Mark All> button to select all the parameters or the <Unmark All> button to undo the selection of all the parameters.

The <Go> and <Cancel> buttons apply to both the "Select Variables" and "Specifications" windows. Click on the <Go> button only after you have completed both windows. The <Cancel> button cancels both windows.

Value

The starting point (initial value) for each optimizable parameter is shown under the heading "Value". By default, it is set to the parameter's nominal value. If you wish to choose a different starting point, click on the entry box under "Value" and enter the desired value.

After Simulation and Optimization, the Optimized Solution is displayed under "Value".

You can click on the button to view the geometry of the starting point. This feature can also be used to generate new projects for arbitrary parameter values, as demonstrated in the tutorials of Chapters 4 and 5.

Upper and Lower Bounds

You can impose bounds on the optimization variables using the entry boxes under the headings "Lower Bound" and "Upper Bound".

During optimization, the parameter values are constrained within the lower and upper bounds if they are specified.

If the bounds are not given explicitly, they will be assigned automatically. Suppose that the starting point of a variable is x.

If the lower bound is not given explicitly, it is set to 0 if $x \geq 0$, or $-\infty$ if $x < 0$.

If the upper bound is not given explicitly, it is set to $+\infty$ if $x \geq 0$, or 0 if $x < 0$.

Specifications for Optimization

In general, the definition of a specification involves the following steps:

1. Select a frequency range.
2. Select an *S*-parameter response.
3. Select a specification type (upper, lower or equality specification).
4. Enter a numerical value as the goal.
5. Optionally, enter a weighting factor.

Selecting a Frequency Range

The frequency range definition line appears as

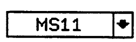
```
FREQ (GHz)   from: ...   to: ...   step: ....
```

The frequency range defined in the *em* analysis control file is offered as the default (if the analysis control file contains more than one frequency range, the first one is used).

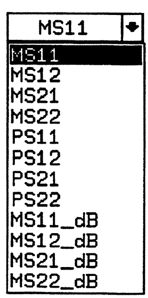
You can modify the frequency range as needed.

Selecting an S-Parameter Response

This is the entry box for selecting an *S*-parameter response:



To select a different *S*-parameter response, click on the arrow adjacent to the box. A list of available responses is displayed, similar to this one:



The label MS*ij* represents the magnitude of S_{ij}, PS*ij* represents the phase of S_{ij} in degrees, and MS*ij*_dB represents the magnitude of S_{ij} in decibels.

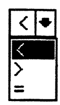
If the number of ports is more than 2, then the list will not show all the available responses at the same time due to the limited display area. A scroll bar will appear alongside the list to enable you to browse through different subsets of the available responses.

To select a response from the list, simply click on it. If you wish to cancel the list while it is shown, click the mouse button outside the list window.

Selecting the Type of Specification

The entry box for selecting the type of specification appears as

Click on the arrow and you will see the list of available choices:



The symbols "<", ">" and "=" represent upper, lower and equality specifications, respectively. To select a specification type, click on the appropriate symbol.

Entering a Numerical Value as the Goal

The third field on the specification line represents the numerical goal. To enter the goal, first click on the box and then type in the appropriate number. You may use the cursor keys, the <Back Space> key and the <Delete> key for editing, if necessary.

Optional Weighting Factor

You can enter an optional weighting factor. The default value is 1. To enter the weighting factor, first click on the box and then type in the appropriate number (the weighting factor must be a positive number). You may use the cursor keys, the <Back Space> key and the <Delete> key for editing, if necessary.

Completing the Definition of a Specification

Once you have selected the frequency range, the *S*-parameter response, the specification type, the numerical goal and, optionally, the weighting factor, click on the button

`Add a new specification defined as follows`

The specification you have just defined is added to the next available line under the heading "Specifications Currently Defined".

You can define up to 16 specifications with different frequency ranges, responses or goals.

Deleting a Specification

If you wish to change a specification which is already defined, you will have to delete the existing specification line and redefine it.

To delete a specification, click on the appropriate line under the heading "Specifications Currently Defined". You will see the button labelled "Delete", which was dimmed, changes to the normal color. Click on this "Delete" button to delete the selected specification line.

9

OSA90 Environment

9

OSA90 Environment

9.1 Introduction

OSA90 is a general-purpose simulation and optimization environment. It is included in the Empipe package to serve as the optimization engine. OSA90 is also marketed separately as a linear and nonlinear circuit CAD system.

OSA90 offers an impressive collection of state-of-the-art optimizers, including minimax, ℓ_1, ℓ_2 (least squares), quasi-Newton, conjugate gradient, Huber, simplex, random, simulated annealing and yield optimization algorithms.

OSA90 also offers a comprehensive set of algebraic operators and mathematical functions to facilitate user-defined pre- and post-processing of variables and responses. You can define labels, equations, conditional expressions (if and else), vectors and matrices. You can utilize built-in functions for matrix algebra, LU factorization, eigenvalues, eigenvectors, discrete Fourier transform, piece-wise linear and cubic spline interpolations, and so on.

You can plot responses and functions in a variety of formats: parameter sweeps, parametric plots, Smith chart and polar plots, even 3D visualization and contours.

Tools for statistical analysis are also at your finger tip: uniform, normal, exponential, lognormal and sample distributions, absolute and relative tolerances, correlation matrices, Monte Carlo analysis, histograms, run charts and scattering diagrams.

You can also license the OSA90/hope option, which offers additional circuit simulation and optimization capabilities, including nonlinear DC, small-signal AC and nonlinear large-signal harmonic balance analyses, comprehensive libraries of nonlinear active device and linear passive element models, user-definable linear subcircuits, user-definable nonlinear device models. You can simulate, display and optimize DC and AC voltages, currents, *S*, *Y* and *Z* parameters, insertion loss, stability factor, group delay, large-signal harmonic distortion, compression, intermodulation products, intercept points, and more. You can optimize designs of small-signal amplifiers, power amplifiers, filters, mixers, frequency multipliers and oscillators. You can take advantage of the unique Datapipe technology to connect external software for functionally integrated simulation and optimization.

The wealth of features of OSA90 is covered in detail in a separate manual: *OSA90/hope User's Manual*. In this chapter, we provide a summary of the input file structure and menu hierarchy, confined to the scope of Empipe applications.

9.2 OSA90 Window

When invoked by Empipe, the OSA90 window appears initially as illustrated in Fig. 9.1.

```
                                   OSA90
File Parsing Completed                        OSA      Fri Sep  6 15:50:59 1996
                                             /osa/empipe_examples/dfstub.ckt

! EM optimization of user-defined structure: DFSTUB

Model
#include "dfstub.inc";

   DFSTUB_L1: ?86.4?;
   DFSTUB_L2: ?81.6?;
   DFSTUB_S: ?0 4.8 16?;

   DFSTUB  1 2 0
      L1=(DFSTUB_L1 * 1mil)  L2=(DFSTUB_L2 * 1mil)
      S=(DFSTUB_S * 1mil);

   PORTS 1 0 2 0;

   CIRCUIT;

   MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);
   MS21_DB = MS_DB[2,1];
end

Sweep
   AC: FREQ: from 5GHz to 20GHz step=0.25GHz
       MS MS_DB PS MS21_dB
      {XSWEEP Title="MS21_dB and Spec"
       Y=MS21_dB X=FREQ
       SPEC=(from 12GHz to 14GHz, < -30) &
            (from 5GHz to 9.5GHz, > -3) &
            (from 16.5GHz to 20GHz, > -3)};
end

Spec
   AC: FREQ: from 12GHz to 14GHz step=0.25GHz MS21_dB < -30;
   AC: FREQ: from 5GHz to 9.5GHz step=0.25GHz MS21_dB > -3;
   AC: FREQ: from 16.5GHz to 20GHz step=0.25GHz MS21_dB > -3;
end

File: reads, edits, parses and saves files                          <F1> Help
OSA90> File Display Optimize Macro Sensitivity monteCarlo Learn
```

Fig. 9.1 OSA90 window.

At the top of the window is the message area, where the date, time and file name are shown. At the bottom of the window is the menu area, where the menu options are presented. In the middle is the display area, which contains either the text of the input file (netlist) or the graphics of the calculated responses.

9.3 OSA90 Input File

An OSA90 input file is like a netlist. It is an ASCII text file with the extension ".ckt". The input file contains definitions of parameters, variables, labels, models, responses, equations, frequency ranges, parameter sweeps, simulation outputs, optimization specifications, operation control options, statistical tolerances and distributions, etc.

The contents of an input file are divided into sections called file blocks. Each file block is designated to supply a particular type of information. Table 9.1 lists the input blocks that are relevant in the context of Empipe. More details are available in the *OSA90/hope User's Manual*.

TABLE 9.1 INPUT FILE BLOCKS

Block Name	Contents
MODEL	variables, labels, equations, models
SWEEP	simulation types, ranges, outputs
SPECIFICATION	optimization specifications
CONTROL	operation control options
MONTECARLO	statistical analysis ranges, outputs
STATISTICS	statistical correlation matrices
TRACE	record of optimization variables

Each input file block begins with a block name and ends with the keyword "end". Block names and keywords are case insensitive. For example, MODEL, Model and model are all treated as identical.

Contained within an input file block are statements. A statement consists of one or more lines of text and must always be terminated by a semicolon.

Example:

```
DFSTUB_L1: 86.4;
```

Here is an example of a multi-line statement:

```
DFSTUB  1 2 0
   L1=(DFSTUB_L1 * 1mil)  L2=(DFSTUB_L2 * 1mil)
   S=(DFSTUB_S * 1mil);
```

Example of OSA90 Input File: "dfstub.ckt"

The following is the OSA90 input file for the double folded stub filter example used in the tutorial of Chapter 4. The file name is "dfstub.ckt". We will dissect this file to illustrate the structure and syntax of the OSA90 input files.

```
! EM optimization of user-defined structure: DFSTUB

Model
#include "dfstub.inc";

   DFSTUB_L1: ?86.4?;
   DFSTUB_L2: ?81.6?;
   DFSTUB_S: ?0 4.8 16?;

   DFSTUB  1 2 0
      L1=(DFSTUB_L1 * 1mil)  L2=(DFSTUB_L2 * 1mil)
      S=(DFSTUB_S * 1mil);

   PORTS 1 0 2 0;

   CIRCUIT;

   MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);
   MS21_DB = MS_DB[2,1];
end

Sweep
   AC: FREQ: from 5GHz to 20GHz step=0.25GHz
       MS MS_DB PS MS21_dB
      {XSWEEP Title="MS21_dB and Spec"
       Y=MS21_dB X=FREQ
       SPEC=(from 12GHz to 14GHz, < -30) &
            (from 5GHz to 9.5GHz, > -3) &
            (from 16.5GHz to 20GHz, > -3)};
end

Spec
   AC: FREQ: from 12GHz to 14GHz step=0.25GHz MS21_dB < -30;
   AC: FREQ: from 5GHz to 9.5GHz step=0.25GHz MS21_dB > -3;
   AC: FREQ: from 16.5GHz to 20GHz step=0.25GHz MS21_dB > -3;
end

Control
   Perturbation_Scale=1.0e-4;
   Optimizer=Minimax;
end
```

Comments

The first line of "dfstub.ckt" is a comment:

```
! EM optimization of user-defined structure: DFSTUB
```

OSA90 treats any text following an exclamation mark until the end of line as a comment.

Include Files

The first statement following the **Model** block header is

```
#include "dfstub.inc";
```

It instructs OSA90 to merge the contents of the file "dfstub.inc" into the body of the input file at the location where the **#include** directive appears.

The file "dfstub.inc" is generated by Empipe. It contains the parameters and ".geo" file template for the double folded stub filter. The use of include files is an efficient and convenient way of merging information stored in separate files.

Labels and Optimization Variables

The next three statements in the **Model** block are

```
DFSTUB_L1: ?86.4?;
DFSTUB_L2: ?81.6?;
DFSTUB_S: ?0 4.8 16?;
```

These statements define labels representing optimization variables.

In the OSA90 input file, optimization variables are delimited by a pair of question marks.

If there is only one number between the question marks, as

label: ?*x*?;

then x represents the starting point of the variable. The lower bound is implicitly set to 0 if $x \geq 0$, or $-\infty$ if $x < 0$. The upper bound is set to $+\infty$ if $x \geq 0$, or 0 if $x < 0$.

If there are three numbers between the question marks, then they represent the lower bound, the starting point and the upper bound of the variable, respectively.

You can also define labels to represent numerical constants, vectors, matrices and formulas. For details see *OSA90/hope User's Manual*.

Element Model

The next statement in the Model block is

```
DFSTUB  1 2 0
      L1=(DFSTUB_L1 * 1mil)  L2=(DFSTUB_L2 * 1mil)
      S=(DFSTUB_S * 1mil);
```

It refers to the Empipe element named DFSTUB. Following the element name are three integers representing the connection nodes. Following the nodes are the parameters of the element. In this case the parameter values are assigned through the use of labels. You can also assign parameter values directly using numerical constants, optimization variables and formulas (expressions).

Example:

```
DFSTUB  1 2 0
   L1=86.4mil  L2=?81.6mil?
   S=(3 * 1.6mil);
```

If you have licensed the OSA90/hope option then you can use the circuit model libraries, including nonlinear active device models and a large selections of linear elements.

Port Definition

Ports of the circuit are defined by this statement in the Model block:

```
PORTS 1 0 2 0;
```

The double folded stub filter is defined as a two-port. In general, for an *n*-port, the keyword PORTS is followed *n* pair of nodes, each pair (two consecutive nodes) defines one of the ports.

A related keyword PORT can be used to define a single port, followed by a pair of nodes.

The CIRCUIT Statement

The statement

```
CIRCUIT;
```

indicates the completion of the circuit definition. When the file parser encounters this statement, it checks the connections of the whole circuit and, if no error is found, generates the circuit response labels.

Defining Responses

OSA90 has a set of predefined circuit response labels, including those listed in Table 9.2.

TABLE 9.2 RESPONSE LABELS

Label	Response
MS*ij*	magnitude of S_{ij}
PS*ij*	phase of S_{ij}
RS*ij*	real part of S_{ij}
IS*ij*	imaginary part of S_{ij}
RY*ij*	real part of Y_{ij}
IY*ij*	imaginary part of Y_{ij}
RZ*ij*	real part of Z_{ij}
IZ*ij*	imaginary part of Z_{ij}
GD*ij*	group delay

In addition to the scalar labels listed in Table 9.2, you can also use matrix labels by omitting the indices. For instance, MS represents the matrix of all MS*ij*, i.e., the magnitude of all *S* parameters. The dimension of such matrices is $n \times n$, where n is the number of ports.

From these predefined labels, other responses can be derived using expressions. For instance, the last two statements in the **Model** block of the file "dfstub.ckt" define the *S*-parameter responses in dB:

```
MS_DB[2,2] = if (MS > 0) (20 * log10(MS)) else (NAN);
MS21_DB = MS_DB[2,1];
```

The first statement defines a 2 by 2 matrix MS_DB to be calculated from the predefined response label MS. The second statement establishes an alias for the matrix element MS_DB[2,1].

These statements are generated by Empipe according to the responses involved in the specifications you have defined for optimization.

Sweep Block

The Sweep block defines frequency range and simulation outputs. In the file "dfstub.ckt", the Sweep block contains

```
AC: FREQ: from 5GHz to 20GHz step=0.25GHz
    MS MS_DB PS MS21_dB
   {XSWEEP Title="MS21_dB and Spec"
    Y=MS21_dB X=FREQ
    SPEC=(from 12GHz to 14GHz, < -30) &
         (from 5GHz to 9.5GHz, > -3) &
         (from 16.5GHz to 20GHz, > -3)};
```

The leading keyword AC indicates the simulation type as small-signal AC (the OSA90/hope option, if licensed, supports two additional types of simulation: DC and harmonic balance).

Following the keyword AC is the frequency range definition. After that is the list of response labels. The next five lines define a graphical view, which is used to tailor the format of the graphical display.

In generating the OSA90 input file, Empipe automatically selects the response(s) of interest (primarily the ones for which specifications are given) to be included in the Sweep block. For the double folded stub filter, the specifications are imposed on the response MS21_dB, therefore MS21_dB is included in the response labels. The specifications on MS21_dB are included in the graphical view definition, so that they will be superimposed on the plot.

Specification Block

The statements in the Spec block represent the specifications you have defined in Empipe. The syntax of these statements are quite self-explanatory. The Spec block in the file "dfstub.ckt" contains

```
AC: FREQ: from 12GHz to 14GHz step=0.25GHz MS21_dB < -30;
AC: FREQ: from 5GHz to 9.5GHz step=0.25GHz MS21_dB > -3;
AC: FREQ: from 16.5GHz to 20GHz step=0.25GHz MS21_dB > -3;
```

Control Block

The Control block allows you to modify the default setting of a number of options of OSA90. The Control block in the file "dfstub.ckt" contains

```
Perturbation_Scale=1.0e-4;
Optimizer=Minimax;
```

It defines the perturbation scale for estimating gradients. This particular value is found to be a good choice for Empipe applications. It also selects the minimax optimizer according to the type of specifications involved.

9.4 OSA90 Menus

Near the bottom of the OSA90 window is the menu area, where the menu options are presented. The main menu looks like this:

```
File: reads, edits, parses and saves files                    <F1> Help
OSA90> File Display Optimize Macro Sensitivity monteCarlo Learn
```

You can use the mouse or the cursor keys to move the cursor to highlight the different menu options. As you do so, the line immediately above the menu line displays a brief comment on the function of the highlighted option.

To select a menu option, you can click the left-hand mouse button on the desired option, or you can move the cursor to highlight the option, and then press the <Enter> key.

Some menu options lead to another level of menu options. The OSA90 menu hierarchy is summarized in Table 9.3.

TABLE 9.3 OSA90 MENU OPTIONS

Menu Option	Brief Description
OSA90.File	reads, edits, parses and saves files
OSA90.Display	calculates and displays responses and functions
OSA90.Optimize	initiates optimization
OSA90.Macro	operates OSA90 from a macro command file
OSA90.Sensitivity	calculates and displays parameter sensitivities
OSA90.MonteCarlo	performs statistical (Monte Carlo) analysis
OSA90.Learn	learns user inputs to create a macro command file
OSA90.Display.Xsweep	displays responses versus parameter sweeps
OSA90.Display.Parametric	displays parametric plots of two responses
OSA90.Display.Array	displays elements of arrays
OSA90.Display.Waveform	displays time-domain waveforms
OSA90.Display.Smith	displays Smith charts and polar plots
OSA90.Display.Visual	displays 3D visualization and contours
OSA90.MonteCarlo.Xsweep	displays statistical sweep responses
OSA90.MonteCarlo.Parametric	displays statistical parametric plots
OSA90.MonteCarlo.Histogram	displays histogram of individual responses
OSA90.MonteCarlo.RunChart	displays run chart of individual responses
OSA90.MonteCarlo.Yield	displays yield estimated by Monte Carlo analysis
OSA90.MonteCarlo.Sensitivity	displays yield versus parameter sweeps
OSA90.MonteCarlo.Max	displays histogram of the maximum errors
OSA90.MonteCarlo.Scatter	displays scatter diagram between two responses

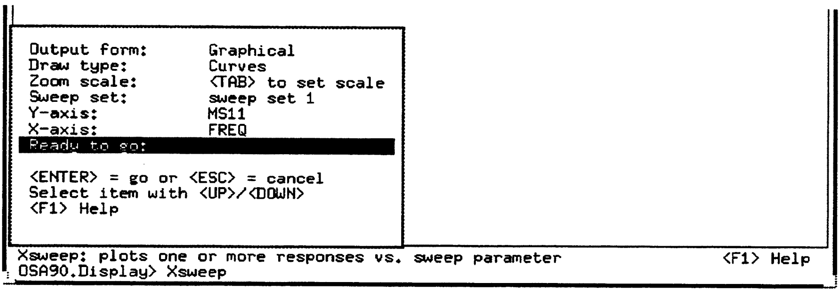
Exit and Cancelation

In OSA90, the right-hand mouse button and the <Esc> key are used to exit from a menu or to cancel an operation.

If you click the right-hand mouse button in the menu area or press the <Esc> key, you will exit from the current menu and return to the next higher-up menu level, if one exists. If you do so at the main menu, you will exit from OSA90 (you will be prompted first to confirm your intention).

Pop-Up Windows

Some OSA90 menu options lead to a pop-up window like the one illustrated here:



You can accept the default setting by clicking on the "Ready to go" line or pressing the <Enter> key. If you wish to change an option, move the cursor to highlight that option and a message is shown near the bottom of the pop-up window indicating the type of action that you may take. Some are multiple-choice options, for which you can click the left-hand mouse button to see a list of the available choices or you can use the <Left> and <Right> cursor keys to toggle through the choices. Some options expect a numerical entry, in which case you simply type in an appropriate value.

On-Line Help

In many cases OSA90 provides on-line help for menu and window options. Try the <F1> key if you need on-line help.

9.5 OSA90 File Editor

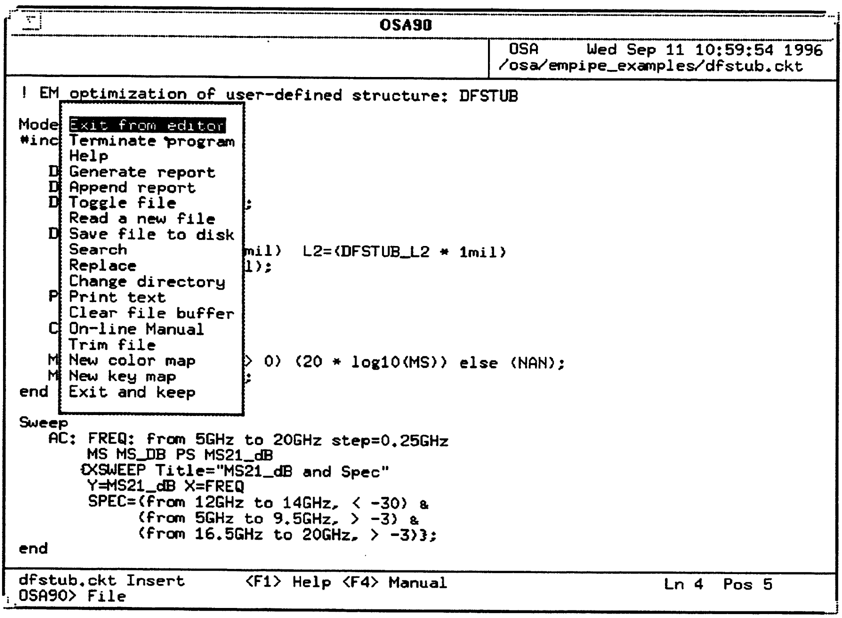
OSA90 has a built-in full screen ASCII text file editor. From the OSA90 main menu, you can invoke the editor by selecting the menu option File.

The editor is integrated with the OSA90 input file parser. So, after you finish modifying the file and exit from the editor, the file parser is automatically activated to check the syntax. If any error is detected, a message will be displayed and you will be returned to the editor to make the correction immediately.

The editor features search and replace, cut and paste, undo, macros and more.

Editor Function Menu

In the editor, you can click the left-hand mouse button to activate a menu of the editor functions, as shown in the following illustration.



The editor functions are summarized in Table 9.4. Click the left-hand mouse button on a function to select it. Click the right-hand mouse button to cancel the menu.

TABLE 9.4 EDITOR FUNCTION MENU

Function	Description
Exit from editor	exit the editor and invoke the input file parser
Terminate program	terminate the program and exit from OSA90
Help	request help messages
Generate report	generate report of simulation results
Append report	append simulation results to the report
Toggle file	toggle between the primary and secondary windows
Read a new file	read (load) a new file to the editor
Save file to disk	save the file or the marked block to a disk file
Search	search for a specified string
Replace	replace a string and with another string
Change directory	change the default path (working directory)
Print text	print the file or the marked block of text
Clear file buffer	discard the current file to start a new file
On-line Manual	request for on-line User's Manual
Trim file	delete trailing blanks throughout the file
New color map	change the display color map
New key map	change the key map
Exit and keep	exit the editor and keep the simulation results

Editor Function Keys

The default set of function keys defined for the OSA90 file editor is listed in Table 9.5. It is possible to re-map the function keys (see the *OSA90/hope User's Manual*).

TABLE 9.5 EDITOR FUNCTION KEYS

Key	Function
<Back Space>	delete the character to the left of the cursor
<F5>	change the default path (working directory)
<Ctrl-N>	discard the current file to start a new (empty) file
<Ctrl-Y>	change the display color map
<Del>	delete the character under cursor or the marked block
<Ctrl-W>	delete the word under cursor
<F6>	toggle between edit and read-only modes
<Ctrl-Home>	move cursor to the end of the line
<Esc>	terminate the program or cancel a command
<F7>	exit the editor and invoke the input file parser
<Ctrl-F7>	exit the editor and keep the simulation results
<Ctrl-Down>	move cursor to the end of the file
<Ctrl-Up>	move cursor to the beginning of the file
<F1>	request help messages
<Home>	move cursor to the beginning of the line
<Ins>	toggle between insert and typeover modes
<Ctrl-K>	change the key map
<Ctrl-D>	delete from cursor to the end of line
<F9>	begin/end the definition of an editing macro
<F10>	execute the macro defined
<F4>	request for on-line User's Manual
<F2>	mark/unmark a block of text for copying or moving
<F3>	copy the marked block, or paste the copied text
<Pg Dn>	move cursor down by one page
<Pg Up>	move cursor up by one page
<Ctrl-P>	print the file or the marked block of text
<Ctrl-I>	input (read in) a new file to the editor
<Ctrl-R>	replace a string and with another string
<Ctrl-G>	generate report of simulation results
<Ctrl-A>	append simulation results to the report
<Ctrl-S>	save the file or the marked block to a disk file
<Ctrl-B>	search backward for a specified string
<Ctrl-F>	search forward for a specified string
<F8>	toggle between the primary and secondary windows
<Ctrl-X>	delete trailing blanks throughout the file (trimming)
<Ctrl-U>	undo the last sequence of consecutive deletions
<Ctrl-Left>	move cursor to the left by one word
<Ctrl-Right>	move cursor to the right by one word

9.6 OSA90 Display

The OSA90 menu option "Display" allows you to view the responses calculated by *em* in a variety of formats. When you click on the "Display" menu option, it leads to another level of menu which appears as

```
Xsweep: plots one or more responses vs. sweep parameter   <F1> Help
OSA90.Display> Xsweep Parametric Array Waveform Smith Visual
```

The display formats represented in this menu are summarized in Table 9.6. The Xsweep display option provides the most commonly used format of presentation: a rectangular plot of one or more response labels versus the frequency or another parameter.

TABLE 9.6 DISPLAY FORMATS

Option	Brief Description
Xsweep	displays one or more responses versus frequency or a parameter
Parametric	displays one or more responses versus another response
Array	displays elements of an array
Waveform	displays time-domain waveforms (for the OSA90/hope option)
Smith	displays Smith charts and polar plots
Visual	displays 3D visualization and contours

Defining a Parameter Sweep

A very useful feature is to display responses versus model parameters. For instance, you may ask OSA90 to sweep the parameter "RDC" of the resistor defined in the tutorial of Chapter 6 and then display the *S*-parameter responses versus the parameter "RDC".

First, you have to define the parameter sweep in the Sweep block of the OSA90 input file. For example:

```
Sweep
   AC: FREQ: 2GHz
       EM_RES_RDC: 150 200 250
       MS11;
end
```

The parameter label ("EM_RES_RDC" in the example) must be followed by a colon ":".

You can assign a set of discrete values for the sweep parameter, as shown in the above example. You can also assign a set of uniformly spaced points with an interval:

```
EM_RES_RDC: from 100 to 300 step=50
```

This translates into a set of values for EM_RES_RDC as 100, 150, ..., 300.

You can also specify the number of divisions within the sweep interval:

```
EM_RES_RDC: from 100 to 300 N=4
```

This translates into a step size of (300 - 100) / 4 = 50. Consequently, the parameter EM_RES_RDC will be swept for these values: 100, 150, ..., 300.

Parameter sweeps with exponential step size are also possible. See the *OSA90/hope User's Manual* for detail.

Selecting a Sweep Parameter for the X-axis

If you have defined a parameter sweep in addition to a frequency sweep, then you can select to display the response(s) versus either the frequency or the sweep parameter.

When invoking the Xsweep display option, you will see a pop-up window such as

```
Output form:       Graphical
Draw type:         Curves
Zoom scale:        <TAB> to set scale
Sweep set:         sweep set 1
Y-axis:            MS11
X-axis:            FREQ
Ready to go:

<ENTER> = go or <ESC> = cancel
Select item with <UP>/<DOWN>
<F1> Help

Xsweep: plots one or more responses vs. sweep parameter          <F1> Help
OSA90.Display> Xsweep
```

To select a sweep parameter, click on the "X-axis:" option. The available choices will be listed from which you can make a selection.

Selecting Response Labels for the Y-axis

For sweep sets which contain multiple response labels, you can choose to display one of the responses or all of the responses. In the pop-up window for the Xsweep display, click on the "Y-axis:" option. The available choices will be listed.

9.7 OSA90 Optimization

When you invoke the "OSA90.Optimize" menu option to start optimization, a pop-up window will appear, such as

```
Optimizer:                  L1
objective Function:         L1
Number of iterations:       30
Accuracy of solution:       0.0001
Display option:             every iteration
Ready to go:

<ENTER> = go or <ESC> = cancel
Select item with <UP>/<DOWN>
<F1> Help

Optimize: optimizes variables according to specifications              <F1> Help
OSA90>                  Optimize
```

The options in the pop-up window are summarized in Table 9.7.

TABLE 9.7 OPTIONS FOR OSA90.Optimize

Option	Available Choices
Optimizer	L1, L2, Minimax, Quasi-Newton, Random, Simplex, Conjugate Gradient, Huber, One-Sided L1, Simulated Annealing, Yield, Yield Huber
Objective Function	*depending on the choice of Optimizer*
Number of iterations	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 30, 50, 100, 999, 9999
Accuracy of solution	0.01, 0.001, 1.0E-4, 1.0E-5, 1.0E-6
Display option	every iteration, best iterations

Optimizer and Objective Function

The choices available for the Optimizer option and the corresponding choices of objective functions are listed in Table 9.8. By default, Empipe selects a suitable optimizer based on the type of specifications you have defined.

TABLE 9.8 OPTIMIZERS AND OBJECTIVE FUNCTIONS

Optimizer	Gradient-Based	Objective Function
L1	Yes	L1
L2	Yes	L2
Minimax	Yes	Minimax
Quasi-Newton	Yes	Generalized L2, Sum of Errors
Random	No	Generalized L2, Sum of Errors
Simplex	No	Generalized L2, Sum of Errors
Conjugate Gradient	Yes	Generalized L2, Sum of Errors
Huber	Yes	Huber, Huber+
One-Sided L1	Yes	One-Sided L1
Simulated Annealing	No	Generalized L2, Sum of Errors

Number of Iterations

This option in the pop-up window allows you to limit the maximum number of iterations. The optimizer will stop once this limit is reached even if a solution within the desired accuracy has not been found. If this happens, the best set of values of the variables (in terms of minimizing the objective function) is presented as the solution.

Accuracy of Solution

This option allows you to specify the desired accuracy of the solution. The optimization will stop when the step size separating the current step and the next step is smaller than the specified accuracy. The step size is relative to the norm of the variable vector.

Display Option

During optimization, OSA90 displays the iteration count and current value of the objective function on the screen. By default, this information is displayed at every iteration.

For example:

```
Iteration   1/30 Max Error=8.14511
Iteration   2/30 Max Error=6.34936
Iteration   3/30 Max Error=1.53442
Iteration   4/30 Max Error=0.553539
Iteration   5/30 Max Error=-0.19606
Iteration   6/30 Max Error=-0.213403
Iteration   7/30 Max Error=-0.213773
Solution  Max Error=-0.213773
```

Of the iteration count, the first number is the current iteration and the second one is the maximum number of iterations specified. For example, 3/30 means that the current iteration is the 3rd out of a maximum of 30.

Sometimes it is possible to observe the objective function value increase rather than decrease from the previous iteration. This is not an error. It is a part of the optimization algorithm as different trial points are tested. The final solution will always correspond to the smallest objective function value.

You can instruct the program to display only those iterations that lead to a better (smaller) objective function value than the previous ones. You can do so by changing the "Display option" to "best iterations". You may find this especially desirable for the random optimizer, since it usually requires a large number of random explorations which result in fluctuating objective function values.

9.8 Producing Plotter Files

OSA90 can produce HPGL (Hewlett-Packard Graphics Language) and PostScript plotter files for the graphical displays.

HPGL and PostScript files are ASCII files which can be directly sent to a compatible printer or plotter or imported into word processing programs.

You can request an HPGL or PostScript file only when the OSA90 window is displaying graphics. Press the key <Ctrl-P>. A pop-up window will appear, as shown in Fig. 9.2.

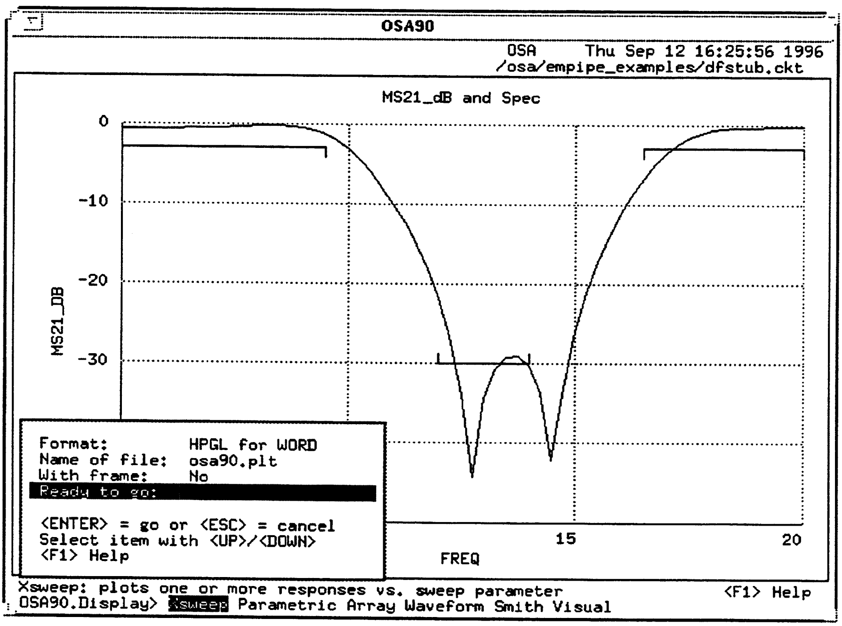


Fig. 9.2 OSA90 window for producing a plotter file.

Selecting the File Format

The "Format" option in the pop-up window allows you to select from the available formats listed in Table 9.9. The differences between the HPGL formats are their interpretation of colors and font sizes.

TABLE 9.9 PLOTTER FILE FORMATS

Format	Description
HPGL for WORD	HPGL file to be imported into Microsoft Word
HPGL for WP	HPGL file to be imported into WordPerfect
HPGL 6-Pen Plotter	HPGL file for HP 6-pen plotter
PS Portrait	PostScript file, portrait orientation
PS Landscape	PostScript file, landscape orientation

Plotter File Name

The option "Name of file" in the pop-up window allows you to specify a name for the plotter file. The default file name is "osa90.plt". To change it, point the cursor to the option "Name of file" and type in the desired file name.

Including or Excluding the Frame of the Graphical Display

The graphical display on the screen has a frame (white border lines). You can include or exclude this frame from the plotter file by choosing "Yes" or "No" for the "With frame" option. The default is "No", i.e., the frame is excluded from the plotter file.

HPGL for 6-Pen Plotter

For HP 6-pen plotters, if you wish the colors to closely match the colors displayed on the screen, you should load the plotter pen carousel in the following color sequence (pen 1 to pen 6): black, red, green, blue, yellow and pink.

9.9 Starting OSA90 Directly

Normally, with the Empipe package, you start the operations from the Empipe program. However, in some situations it may be more convenient to bypass the Empipe user-interface and start the OSA90 program directly.

To illustrate, we use the double folded stub filter example. In the tutorial of Chapter 4, we started with the Empipe program by

```
empipe dfstub
```

After we have defined parameters, optimization variables and specifications, Empipe invokes OSA90. At this point, Empipe created an OSA90 input file named dfstub.ckt.

The file dfstub.ckt contains all the necessary information for the simulation and optimization of the double folded stub filter, including the definitions of variables, responses, frequency ranges and specifications.

After this first session, the next time when you want to simulate or optimize the same double folded stub filter, it is not necessary to start from the Empipe user-interface. You can start OSA90 directly by

```
osa90 dfstub.ckt
```

OSA90 will load the input file dfstub.ckt and place you in the file editor. You can then make modifications to the file if necessary; exit from the editor; and simulate or optimize the filter.

You must include the extension ".ckt" in the input file name when you start OSA90 directly.

It is advantageous to start OSA90 directly if you have made significant modifications to the input file originally created by Empipe. For instance, in the tutorials in Chapters 6 and 7 we have redefined the ports and used the *Y* and *Z* parameter responses instead of the *S* parameters. In such cases, if you restart from Empipe, the modifications you made to the OSA90 input file may be overwritten.

An alternative way of making sure that any modifications you have made will not be overwritten is to save the modified input file to the disk under a new name. You can do that using the OSA90 file editor functions (see Section 9.5). In the editor, click the left-hand mouse button to activate the editor function menu. Then, click on the option "Save file to disk". You will be prompted for the file name.

10

Empipe Database

10

Empipe Database

10.1 Database Index

Empipe employs a database system to avoid duplicate ***em*** analyses. Every time Empipe invokes ***em***, the simulation results are stored in a database, together with the corresponding parameter values. Subsequently, when ***em*** analysis results are needed, Empipe checks the database first.

Empipe database file names have the form of

element_i.dbs

where *element* identifies the Empipe element and *i* is an integer index. For example, the database file name for the double folded stub filter tutorial in Chapter 4 is dfstub_1.dbs.

The database index is used to distinguish different versions of the same structure. The default index is 1.

Normally, there are few reasons to create a new database index. One reason may be to compare the results of different versions of ***em*** by storing them in separate database files.

The only way to change the database index is to modify the OSA90 input file. Locate within the input file the Empipe element reference and add the INDEX parameter.

Example:

```
DFSTUB  1 2 0
   INDEX=2
   ...
```

The database created after this modification will be named dfstub_2.dbs.

Note, however, that dfstub_2.dbs will start out as an empty file. It will not contain the data stored in dfstub_1.dbs, which means that all the ***em*** simulations saved in dfstub_1.dbs will have to be repeated when they are needed.

You can disable the database mechanism entirely by assigning INDEX = 0. This instructs Empipe to ignore the database and always invoke ***em*** for EM simulation. Furthermore, the EM simulation results will not be saved in any database files. We did this in Chapter 4 intentionally in order to track the ***em*** simulation in real time.

10.2 Converting Database to ASCII Data File

As part of the Empipe package, a utility program called dbs2dat is provided to facilitate converting Empipe database files to ASCII data files.

Usage:

```
dbs2dat database datafile
```

where *database* denotes the name of an existing Empipe database file and *datafile* is the name of the output file which will contain the converted data.

Example:

```
dbs2dat tpad_1.dbs tpad.dat
```

where tpad_1.dbs is the database for the distributed attenuator example used in Chapter 5. The converted data written to tpad.dat is partially shown as follows.

```
Data From File tpad_1.dbs (Last Modified on  Aug  8 18:46:35 1995)

N_PARS 4
N_FREQS 5
N_PORTS 2

 22 7 11 10
   2 0.232043 -9.294 0.254789 -8.861 0.254789 -8.861 0.219005 -8.365
   6 0.228578 -26.21 0.250072 -25.69 0.250072 -25.69 0.219694 -24.25
   10 0.23058 -41.3 0.246908 -41.57 0.246908 -41.57 0.225596 -39.17
   14 0.240537 -55.55 0.247087 -57.32 0.247087 -57.32 0.238017 -53.63
   18 0.257626 -69.58 0.250045 -73.42 0.250045 -73.42 0.256298 -67.93

 24 7 11 10
   2 0.260873 -8.773 0.237937 -8.83 0.237937 -8.83 0.24862 -8.102
   6 0.25735 -25.16 0.233402 -25.62 0.233402 -25.62 0.249128 -23.54
   10 0.259171 -39.97 0.230408 -41.43 0.230408 -41.43 0.254695 -38.2
   14 0.268684 -54.11 0.230634 -57.1 0.230634 -57.1 0.266466 -52.52
   18 0.284987 -68.12 0.233517 -73.13 0.233517 -73.13 0.283883 -66.78

  ......
  ......

 12 6 9 10
   2 0.076526 -15.09 0.33132 -8.934 0.33132 -8.934 0.053164 -11.27
   6 0.07177 -39.58 0.324903 -25.96 0.324903 -25.96 0.054182 -30.2
   10 0.071289 -54.6 0.320292 -41.99 0.320292 -41.99 0.061107 -43.4
   14 0.079908 -64 0.319902 -57.81 0.319902 -57.81 0.075948 -54.69
   18 0.098237 -72.76 0.322978 -73.92 0.322978 -73.92 0.098448 -66.38
```

In the data file, a header line identifies the source database, followed by three lines listing the number of parameters, the number of frequencies and the number of ports, respectively.

The data is divided into data sets, separated by a blank line. Each data set begins with a list of parameter values, followed by the *S* parameters. For example, the first data set in tpad.dat is

```
22 7 11 10
  2 0.232043 -9.294 0.254789 -8.861 0.254789 -8.861 0.219005 -8.365
  6 0.228578 -26.21 0.250072 -25.69 0.250072 -25.69 0.219694 -24.25
  10 0.23058 -41.3 0.246908 -41.57 0.246908 -41.57 0.225596 -39.17
  14 0.240537 -55.55 0.247087 -57.32 0.247087 -57.32 0.238017 -53.63
  18 0.257626 -69.58 0.250045 -73.42 0.250045 -73.42 0.256298 -67.93
```

The first line shows the values of the four parameters, in the same order as they appear in the Empipe form editor (in this case they are L1, L2, W1 and W2, see Chapter 5).

The next five lines correspond to the five frequencies. The data on each line represents

```
FREQ MS11 PS11 MS21 PS21 MS12 PS12 MS22 PS22
```

where FREQ denotes the frequency (in GHz), MS*ij* denotes the magnitude of S_{ij} and PS*ij* denotes the phase of S_{ij} (in degrees).

Database Files from Earlier Versions of Empipe

The conversion of a database created by Empipe Version 3.1 (or later) to an ASCII data file is fully automated.

The utility can also be applied to database files created by earlier versions of Empipe. If such a database represents an Empipe library element (see Chapter 12), then the process is also fully automated. If the database represents an arbitrary structure defined by Geometry Capture, then you will be prompted to specify the number of parameters, the number of frequencies and the number of ports, for this information was not stored in the earlier versions of Empipe database.

Note that the Empipe program itself has no problem reading database files of an earlier version, because it already has all the necessary information.

10.3 Creating Database from Data File

A utility program called dat2dbs is included in the Empipe package. It can be used to create an Empipe database file from an ASCII data file.

Usage:

```
dat2dbs datafile database
```

where *datafile* denotes the name of an existing ASCII data and *database* is the name of the database file to be created.

Example:

```
dat2dbs tpad.dat tpad_2.dbs
```

where tpad.dat is produced by dbs2dat from tpad_1.dbs, as described in Section 10.2. The resulting database tpad_2.dbs should contain the same data as in tpad_1.dbs.

In general, the data file must contain a header like this

```
N_PARS  n1
N_FREQS n2
N_PORTS n3
```

where *n1*, *n2* and *n3* denote the number of parameters, the number of frequencies and the number of ports, respectively.

Editing Empipe Database

By combining the two utility programs dbs2dat and dat2dbs, you will be able to edit the contents of an Empipe database in three steps.

1. Use dbs2dat to convert an existing database to a data file.

2. Edit the data file using a text editor. You can delete unwanted data sets to trim down an oversized database. You can manually expand the database by adding new sets of *S* parameters produced by ***em*** (the ***em*** outputs can be included directly without editing). You can even modify the numbers, but that is cheating.

3. Use dat2dbs to convert the edited data file to a new database, perhaps using an index different from the original database.

11

Response Interpolation

11

Response Interpolation

11.1 Linear and Quadratic Interpolations

Empipe utilizes response interpolation in order to accommodate parameter values which do not exactly fall on the predefined discretization grid required by ***em***.

The type of interpolation is user-selectable by means of the Empipe element parameter MODEL. For example:

```
DFSTUB  1 2 0
   MODEL=2
   ...
```

Possible values for MODEL are listed in Table 11.1.

TABLE 11.1   MODEL CHOICES

Value	Description
0	interpolation disabled
1	linear interpolation (this is the default value)
2	quadratic interpolation
3	linear interpolation based on *S* parameters, magnitude and phase
4	quadratic interpolation based on *S* parameters, magnitude and phase
5	linear interpolation based on *Y* parameters, real and imaginary parts
6	quadratic interpolation based on *Y* parameters, real and imaginary parts
7	linear interpolation based on *S* parameters, real and imaginary parts
8	quadratic interpolation based on *S* parameters, real and imaginary parts
9	linear interpolation based on *Z* parameters, real and imaginary parts
10	quadratic interpolation based on *Z* parameters, real and imaginary parts

The choices 1 and 2 represent the default schemes for linear and quadratic interpolations, respectively. Currently, the choices 1 and 2 are mapped to 3 and 4, respectively. This may be replaced in future releases by more sophisticated schemes.

Disabling Response Interpolation

The response interpolation feature can be entirely disabled by setting MODEL=0. In this case, any off-grid parameters are simply snapped to the nearest on-grid values, as illustrated in Fig. 11.1.

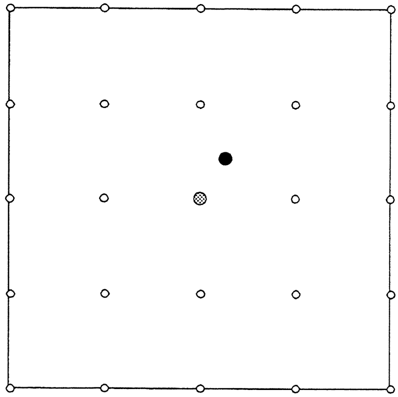


Fig. 11.1 Interpolation disabled: the off-grid point (solid) is snapped to the nearest on-grid point (shaded).

Linear Interpolation

Linear interpolation of an off-grid point requires ***em*** analyses at $n + 1$ base points, where n is the number of parameters whose values are off-grid. The placement of the base points is illustrated in Fig. 11.2.

Linear interpolation is adequate when the grid size is sufficiently small that the variation of the response of interest within one cell can be approximated with reasonable accuracy by a linear function.

Quadratic Interpolation

Quadratic interpolation generally provides more accurate results than linear interpolation, at the expense of increased computational effort. Quadratic interpolation of an off-grid point requires ***em*** analyses at $2 \times n + 1$ base points, where n is the number of parameters whose values are off-grid. The placement of the base points for quadratic interpolation is illustrated in Fig. 11.3.

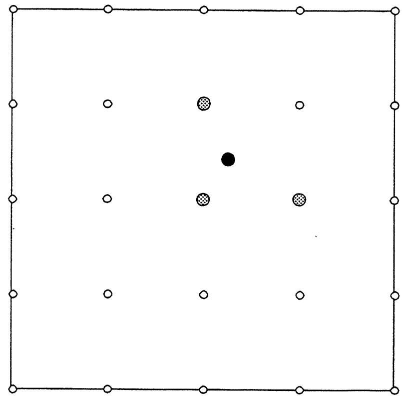


Fig. 11.2 *The base points (shaded) needed for linear interpolation of an off-grid point (solid).*

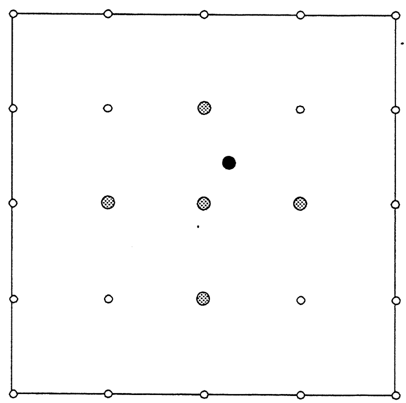


Fig. 11.3 *The base points (shaded) needed for quadratic interpolation of an off-grid point (solid).*

11.2 Choosing S, Y or Z Parameters

In using interpolation, a problem may arise when the response function to be interpolated has a local minimum or maximum with respect to the frequency and its location on the frequency axis shifts significantly between two base points.

Consider the illustration in Fig. 11.4. The two solid lines represent the response MS11 at two base points. The response has a local minimum with respect to the frequency and its location changes from 8.53 GHz for one base point to 8.72 GHz for the other base point. A linear interpolation applied to these two base points leads to the erroneous result depicted by the dashed line in Fig. 11.4.

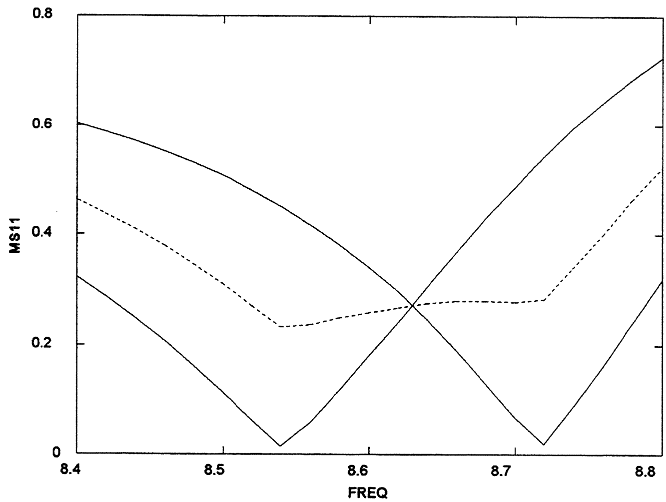


Fig. 11.4 A problem with interpolation.

If you encounter this problem, you can reduce its severity by using a finer grid or a set of more densely spaced frequencies. Unfortunately, this means you will have to repeat the ***em*** simulation.

A more efficient remedy is to choose the *Y* or *Z* parameters instead of the *S* parameters as the response functions used in the interpolation. This can solve the problem if the *Y* or *Z* parameters do not contain any shifting minima or maxima. The most significant advantage of this solution is that it does not require any new ***em*** simulation. Empipe first converts the *S* parameters to the *Y* or *Z* parameters, performs the interpolation and then converts the results back to *S* parameters.

Currently, you have to choose the *Y* or *Z* parameters manually on a trial and error basis. An automated scheme is under development.

12

Empipe Library Elements

12

Empipe Library Elements

12.1 Introduction

In addition to the Geometry Capture capability for user-defined elements, Empipe provides a built-in library of preprogrammed elements. These elementary microstrip structures are commonly included in the empirical model libraries of circuit simulators.

The main advantage of the Empipe library elements is that you can avoid entirely the overhead in creating a Geometry Captured element. Typically, a library element includes all the substrate parameters and geometrical dimensions as designable parameters, with the names that are conventionally used by popular circuit simulators. You can transfer a design from a circuit simulator to Empipe with the smallest amount of effort.

The main drawback of this approach is the limited number of elements contained in the library. You may be able to use them as building blocks to construct a more complex structure, but the results may be less accurate than those obtained by simulating the structure as a whole. The building blocks are treated as linear subcircuits by OSA90 and connected by circuit theory without taking into account any EM couplings between them.

To use the Empipe library elements, you start the OSA90 program directly. The Empipe Geometry Capture form editor is not needed. The operations in the OSA90 environment are basically the same as those described in Chapter 9 for Geometry Captured elements. But now you have to create the OSA90 input file yourself, instead of having it created for you by Empipe. Review Chapter 9 for a description of the OSA90 input file structure and syntax. A template for the Model block is provided as follows.

Syntax:

```
Model
   #include "em_xxxxx.inc"

   EM_xxxxx nodes parameters;

   ...
end
```

where EM_*xxxxx* is the name of an Empipe library element.

Table 12.1 lists the Empipe library elements. Further details, including the schematics and parameters, are provided in the catalog in Section 12.5.

TABLE 12.1 EMPIPE LIBRARY ELEMENT NAMES

Name	Brief Description
EM_AGAP	microstrip asymmetrical gap
EM_BEND1	microstrip bend
EM_BEND2	mitered microstrip bend
EM_CPLF	microstrip coupled line filter
EM_CROSS	microstrip cross junction
EM_DCAP1	microstrip interdigital capacitor, model 1
EM_DCAP2	microstrip interdigital capacitor, model 2
EM_DPCAP	microstrip double patch capacitor
EM_DSTB1	microstrip double stub, symmetrical
EM_DSTB2	microstrip double stub, asymmetrical
EM_FDST1	microstrip folded double stub, symmetrical
EM_FDST2	microstrip folded double stub, asymmetrical
EM_GAP	microstrip symmetrical gap
EM_MSL	microstrip line
EM_ODPC1	microstrip overlay double patch capacitor, symmetrical
EM_ODPC2	microstrip overlay double patch capacitor, symmetrical
EM_ODPC3	microstrip overlay double patch capacitor, asymmetrical
EM_ODPC4	microstrip overlay double patch capacitor, asymmetrical
EM_OPEN	microstrip open stub
EM_RECT	microstrip rectangular structure
EM_SPIND	microstrip spiral inductor
EM_STEP	microstrip step junction
EM_TEE	microstrip T-junction

Empipe Include Files

The definitions of the Empipe library elements are contained a set of include files located in the subdirectory

<OSA Installation Directory>/`include`

It is not necessary to copy the include files to your working directory. When they are needed, OSA90 will automatically search the installation directory.

Each Empipe library element has a unique include file. The name of the include file is created by appending the extension ".inc" to the name of the element (in lower case letters, following the UNIX convention). For example, the include file for the element EM_MSL is em_msl.inc.

12.2 Empipe Element Parameters

The parameters for the Empipe library elements can be divided into three categories, namely geometrical parameters, substrate parameters and control parameters.

The geometrical parameters define the physical dimensions, such as width and length. The set of geometrical parameters available for each element is described in the Empipe library catalog in Section 12.5.

The substrate parameters for the Empipe library elements are listed in Table 12.2. Two-layer structures, such as EM_ODPC1 (microstrip overlay double patch capacitor), have two sets of substrate parameters: one for the top layer and one for the bottom layer.

TABLE 12.2 SUBSTRATE PARAMETERS

Name	Description
H	substrate thickness
H2	shielding height
EPSR	relative dielectric constant
UR	relative permeability
TAND	dielectric loss tangent
MTAND	magnetic loss tangent
T	conducting metal strip thickness
RHO	conducting metal strip resistivity
SR	surface reactance

The control parameters for the Empipe library elements are listed in Table 12.3.

TABLE 12.3 EMPIPE ELEMENT CONTROL PARAMETERS

Name	Description
XCELL, YCELL	cell size for ***em*** analysis
FMIN, FMAX, FSTEP	frequency range for ***em*** analysis
INDEX	Empipe database index
MODEL	type of response interpolation
ACCU	single or double precision in ***em*** analysis
LEFT, RIGHT, TOP, BOTTOM	substrate box size
MIN_SUB	minimum number of subsections per wavelength
G*i*MIN, G*i*MAX	limits on subsection size for the *i*th polygon
UNIT	unit of geometrical parameters used by ***em***
UPDATE	control over when to update the database

Example:

```
EM_MSL  1 4 0  W=2mil L=15mil
               H=5mil EPSR=2.9
               XCELL=1mil YCELL=0.5mil
               FMIN=2GHz FMAX=10GHz FSTEP=1GHz;
```

The element EM_MSL has two geometrical parameters, namely W for the width and L for the length. The substrate parameters are H for the substrate thickness and EPSR for the dielectric constant (the other substrate parameters are not explicitly specified). The control parameters XCELL and YCELL define the ***em*** cell size as 1 mil × 0.5 mil. FMIN, FMAX and FSTEP are also control parameters and they define the frequency range for the ***em*** analysis as from 2 GHz to 10 GHz with a step of 1 GHz.

Empipe Database

A general description of the Empipe database is provided in Chapter 10. The same concept applies to both the library elements and Geometry Captured elements. The main difference is in the database names.

The name of an Empipe library element database is of the form

em*k*_*i*.dbs

where *k* is an integer identifer of the element and *i* is the database index specified by the parameter INDEX.

In comparison, the database name for a Geometry Captured element has the form

*name*_*i*.dbs

where *name* is the user-defined name for the element.

Each library element is assigned a unique integer identifier which is listed in the Empipe library catalog in Section 12.5. For example, the element identifier for EM_STEP is 16.

Another parameter which controls the database management is UPDATE. Setting UPDATE=1 instructs Empipe to update the database whenever a new ***em*** analysis is performed. This is the default. Setting UPDATE=0 instructs Empipe to wait until the optimization is completed before updating the database.

Response Interpolation

Empipe uses interpolation to obtain responses for off-grid points (see Chapter 11 for a general discussion on response interpolation).

You can select linear interpolation by setting the control parameter MODEL=1 or select quadratic interpolation by setting MODEL=2. The default choice is linear interpolation.

Single or Double Precision for *em* Analysis

The Empipe library element control parameter ACCU allows you can choose single or double precision for ***em*** analysis by setting ACCU to 0 or 1, respectively.

The default value for ACCU is 0, i.e., single precision.

According to the ***em*** User's Manual, using single precision can reduce the simulation time significantly.

Specifying Substrate Box Size

For Empipe library elements, the substrate box size is automatically defined. When the parameter values change during optimization, the substrate box size is adjusted accordingly, keeping the margins between the metal and the edges of the box constant.

Fig. 12.1 illustrates this concept, where X1, X2, Y1 and Y2 represent the margins between the metal and the edges of the substrate box.

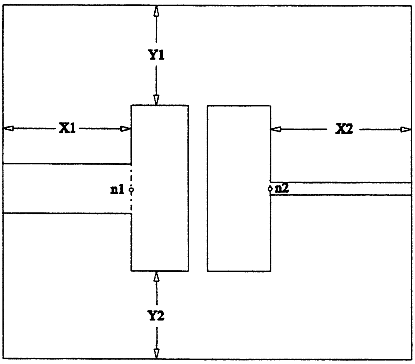


Fig. 12.1 The substrate box of a double patch capacitor.

You can explicitly specify the margins by the parameters LEFT, RIGHT, TOP and BOTTOM:

$$X1 = h \times LEFT$$
$$X2 = h \times RIGHT$$
$$Y1 = h \times TOP$$
$$Y2 = h \times BOTTOM$$

where h is the substrate thickness. In other words, the margins are normalized with respect to the substrate thickness. The default values for LEFT, RIGHT, TOP and BOTTOM are given in the Empipe library element catalog in Section 12.5.

Specifying the Minimum Number of Subsections per Wavelength

The parameter MIN_SUB specifies the minimum number of subsections per wavelength, which controls the absolute maximum subsection size used by ***em***.

As suggested in the ***em*** User's Manual, the default value for MIN_SUB is 20, i.e., a minimum of 20 subsections per wavelength.

Specifying the Subsection Size for Individual Polygon Groups

For ***em*** analysis, a microstrip structure is divided into "polygon groups". For example, a step junction is divided into two polygon groups, as illustrated in Fig. 12.2.

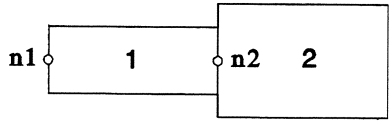


Fig. 12.2 Two polygon groups of a microstrip step junction.

You can specify the minimum and maximum of the subsection size for each polygon group using the parameters G*i*MIN and G*i*MAX, respectively, where $i = 1, 2, ..., n$, and n is the total number of polygon groups. For example, G1MIN defines the minimum subsection size for the first polygon group and G2MAX defines the maximum subsection size for the second polygon group.

The values of G*i*MIN and G*i*MAX are normalized w.r.t. the cell size. For instance, a value of 100 for G1MAX means that the maximum subsection size for the first polygon group is 100 times the cell size.

The default value for G*i*MIN is 1. The default value for G*i*MAX is 100.

Unit of Geometrical Parameters Used by *em*

The parameter UNIT indicates the length unit to be used by in generating the ".geo" file. The set of units allowed is represented by a set of integer values for the parameter UNIT, as listed in Table 12.4.

Before Empipe invokes ***em***, it creates a ".geo" file to describe the structure to be analyzed. The ".geo" file contains geometrical coordinates which are translated by Empipe from the parameter values. The geometrical coordinates in the ".geo" file will be scaled according to the value you assigned to UNIT.

The default value for UNIT is 3, which represents mm.

TABLE 12.4 UNIT PARAMETER

Parameter	Unit Used in ".geo" File
0	mil (milli-inch)
1	in (inch)
2	um (micron)
3	mm (millimeter)
4	cm (centimeter)
5	m (meter)

12.3 Example

Among the examples supplied with Empipe is the file **emp01.ckt** which demonstrates the simulation and optimization of a three section 3:1 microstrip impedance transformer, as depicted in Fig. 12.3. The structure is modeled by three Empipe elements: two steps (EM_STEP) and a line (EM_MSL).

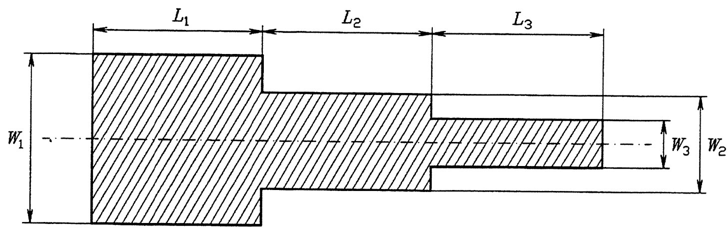


Fig. 2.3 Three section 3:1 microstrip impedance transformer.

Simulation and Optimization of the Transformer

We now describe the steps to simulate and optimize the microstrip impedance transformer using OSA90 and Empipe.

Go to the subdirectory which contains your copy of the Empipe examples (the instructions for copying the Empipe examples are found in Chapter 1):

```
cd <Your Empipe Example Subdirectory>
```

Start the OSA90 program:

```
osa90 emp01.ckt
```

As stated in Section 12.1, for Empipe library elements, we do not need the Geometry Capture form editor. Instead, we start directly with the OSA90 program.

The input file emp01.ckt will be loaded into the OSA90 file editor. This file illustrates the basic file syntax for using the Empipe library elements. You can use the cursor keys to move around the editor and browse through the file.

The entire contents of the file emp01.ckt are listed as follows.

```
!  Example: emp01.ckt
!  Three-section impedance transformer optimization
!  demonstrates Empipe library elements

Model
#include "em_msl.inc"
#include "em_step.inc"

   W1 = ?0.65mm?;
   W2 = ?0.35mm?;
   W3 = ?0.07mm 0.15mm 1mm?;

   EM_STEP  1 2 0  INDEX=1 W1=W1 W2=W2 L=3mm H=0.635mm EPSR=9.7
                   XCELL=75um YCELL=50um FMIN=5 FMAX=15 FSTEP=1;

   EM_STEP  2 3 0  INDEX=2 W1=W2 W2=W3 L=3mm H=0.635mm EPSR=9.7
                   XCELL=50um YCELL=35um FMIN=5 FMAX=15 FSTEP=1;

   EM_MSL   3 4 0  INDEX=1 W=W3 L=3mm H=0.635mm EPSR=9.7
                   XCELL=50um YCELL=35um FMIN=5 FMAX=15 FSTEP=1;

   RES   4 0  R=150;
   PORT  1 0;

   CIRCUIT;
end

Sweep
   AC: freq: from 5ghz to 15ghz step=0.5ghz  MS11 PS11
   {Xsweep  Y=MS11 Ymin=0 Ymax=0.5 NYticks=5 NXticks=5};
end

Specification
   AC: freq: from 5ghz to 15ghz step 0.5ghz MS11;
end

Control
   Optimizer=Minimax;
end
```

Simulation of the Transformer Before Optimization

Exit from the file editor by clicking the left-hand mouse button within the editor. When the menu box appears on the screen, click on the option "Exit from editor". At the OSA90 main menu, click on the option "Display" to start the simulation.

The simulation finishes very quickly, since all the necessary *em* analysis results are already stored in the database. At the display menu, click on the option "Xsweep". When a pop-up window appears, press <Enter> to accept the default setting. The graphical display is depicted in Fig. 12.4.

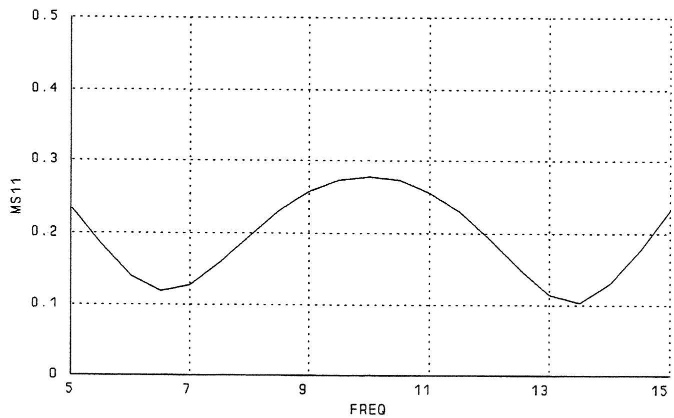


Fig. 12.4 Reflection coefficient of the impedance transformer before optimization.

Minimax Optimization

We wish to minimize the reflection coefficient of the impedance transformer by minimax optimization. In emp01.ckt, the widths of the microstrip lines are defined as optimization variables.

Exit from the display menu by clicking the right-hand mouse button or pressing <Esc>. At the OSA90 main menu, click on the option "Optimize". When a pop-up window appears, press <Enter> to accept the default setting.

The optimization proceeds very quickly. Again, this is because all the necessary *em* analysis results are already stored in the database.

Simulation of the Transformer After Optimization

After the optimization is completed, click on the option "Display" at the OSA90 main menu. Then click on the display menu option "Xsweep". When a pop-up window appears, press <Enter> to accept the default setting.

The graphical display depicted in Fig. 12.5.

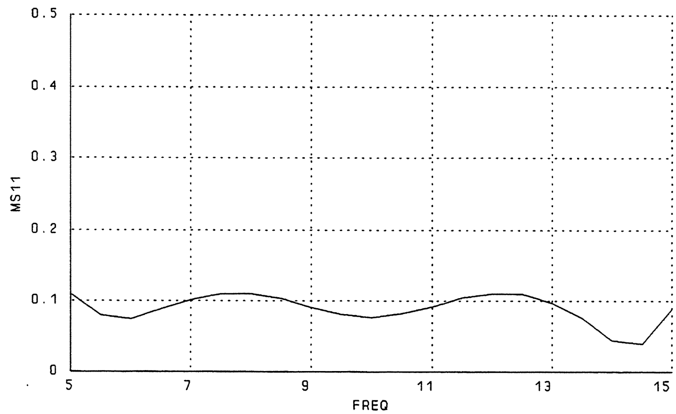


Fig. 12.5 Reflection coefficient of the impedance transformer after optimization.

Saving the Optimized Transformer

In this example, we decompose the impedance transformer into three sections, each modeled by a separate Empipe library element. These elements are connected through the circuit simulator in OSA90. This approach avoided the overhead involved in Geometry Capture, but, on the other hand, we cannot view the optimized transformer as a whole in ***xgeom*** without having to manually paste the pieces together.

We can still save the optimized dimensions which are recorded in the OSA90 input file. First, exit from the display menu by clicking the right-hand mouse button or pressing <Esc>. At the OSA90 main menu, click on the option "File". In the file editor click the left-hand mouse button and select from the editor menu box the option "Save file to disk". The editor will prompt you for a file name, type "emp01_o.ckt" and then press <Enter>. The optimized geometrical parameters of the impedance transformer are now saved in the file "emp01_o.ckt". Finally, exit from OSA90 by clicking the right-hand mouse button or pressing <Esc>.

12.4 Modifying Library Defaults

This section provides guidelines for modifying the default definitions of the Empipe library element. You must have sufficient experience with Empipe and a good reason in order to attempt such a modification.

You can skip this section entirely if you do not plan to alter the Empipe library.

The Empipe library elements are defined as parameterized subcircuits according to the requirements specified in the *OSA90/hope User's Manual*. The definitions of the Empipe library elements are contained in the Empipe include files (Section 12.1).

Include File of a Generic Empipe Library Element

```
ELEMENT  EM_xxxxx  nodes
                   INDEX=1  geo_parameter=1  sub_parameter=value
                   UNIT=3 XCELL=1 YCELL=1 ACCU=0 UPDATE=1
                   LEFT=3 RIGHT=3 TOP=3 BOTTOM=3 MIN_SUB=20
                   GiMIN=1 GiMAX=100
                   FMIN=0 FMAX=1 FSTEP=1 MODEL=1 {
   Datapipe: LINEAR  FILE="empipe"  NAME=em_xxxxx
             N_INPUT=n_input
             INPUT=(type, INDEX, n_geo_par, geo_parameters,
                    n_sub_par, sub_parameters
                    20, UNIT, XCELL, YCELL, ACCU, UPDATE,
                    LEFT, RIGHT, TOP, BOTTOM, MIN_SUB,
                    GiMIN, GiMAX,
                    0, FMIN, FMAX, FSTEP, FREQ, MODEL)
             FORMAT=S  N_PORT=n_port;

   DATAPORT  nodes  DATA=em_xxxxx;
};
```

where

EM_*xxxxx*	is the name of the element, e.g., EM_MSL.
nodes	are the connection nodes.
geo_parameter	represents the appropriate geometrical parameters.
sub_parameter	represents the appropriate substrate parameters.
value	represents the default value for a substrate parameter.
type	is the element identifier, e.g., 16 for EM_STEP.
n_geo_par	is the total number of geometrical parameters.
n_sub_par	is the total number of substrate parameters.

n_input is the total number of inputs to the Datapipe child empipe. Typically, *n_input* = 31 + *n_geo_par* + *n_sub_par*.

n_port is the number of ports of the element (subcircuit), e.g., EM_MSL is defined as a two-port element.

G*i*MIN, G*i*MAX define the limits on subsection size for the *i*th polygon group. The Datapipe inputs always include the limits for 5 polygon groups. If the actual number of polygon groups for the element is less than 5, constant values 1 and 100 are used as placeholders.

DATAPORT is used to incorporate an externally simulated linear subcircuit into the overall circuit definition in OSA90. In this case, the linear subcircuit is defined by the *S* parameters calculated by ***em*** and imported by Empipe. For more details, see *OSA90/hope User's Manual*.

nodes represent the connection nodes of DATAPORT.

Example: the Definition of EM_MSL

```
!  em_msl.inc
!  empipe element definition

ELEMENT  EM_MSL     1 2 0
                    INDEX=1 W=1 L=1
                    H=0.635mm H2=-1 EPSR=2.35 UR=1 TAND=0 MTAND=0
                    T=0 RHO=0 SR=0
                    UNIT=3 XCELL=1 YCELL=1 ACCU=0 UPDATE=1
                    LEFT=0 RIGHT=0 TOP=3 BOTTOM=3 MIN_SUB=20
                    G1MIN=1 G1MAX=100
                    FMIN=0 FMAX=1 FSTEP=1 MODEL=1 {
   Datapipe: LINEAR  FILE="empipe"  NAME=em_msl
             N_INPUT=42
             INPUT=(13, INDEX, 2, W, L,
                    9, H, H2, EPSR, UR, TAND, MTAND, T, RHO, SR,
                    20, UNIT, XCELL, YCELL, ACCU, UPDATE,
                    LEFT, RIGHT, TOP, BOTTOM, MIN_SUB,
                    G1MIN, 1, 1, 1, 1, G1MAX, 100, 100, 100, 100,
                    0, FMIN, FMAX, FSTEP, FREQ, MODEL)
             FORMAT=S  N_PORT=2;

   DATAPORT 1 2 0 DATA=em_msl;
};
```

Changing the Default Parameter Values

The default parameter values are defined in the Empipe include files.

For example, the file em_msl.inc contains

```
ELEMENT   EM_MSL   1 2 0
                   INDEX=1 W=1 L=1
                   H=0.635mm H2=-1 EPSR=2.35 UR=1
          ....
```

where the default values for the parameters INDEX, W, L, H, H2, EPSR, etc, are clearly listed. For instance, the default value for EPSR (substrate dielectric constant) is 2.35.

The default parameters values can be changed. For instance, you may find it convenient to customize the default substrate parameter values to represent a particular material used in your current project.

For example, you can change the default value for EPSR to 6.5 by modifying the EM_MSL definition as

```
ELEMENT   EM_MSL   1 2 0
                   INDEX=1 W=1 L=1
                   H=0.635mm H2=-1 EPSR=6.5 UR=1
          ....
```

However, this makes the Empipe element definition inconsistent with the Empipe library element catalog.

A better way is to rename the customized Empipe elements. For example, using em_msl.inc as a template, you can create a new include file named em_msl_a.inc which contains

```
ELEMENT   EM_MSL_a   1 2 0
                     INDEX=1 W=1 L=1
                     H=0.635mm H2=-1 EPSR=6.5 UR=1
          ....
```

where EM_MSL_a is the new name of the modified Empipe element.

You can even change the names of the parameters. For example, you can change W to WIDTH (you must also change the reference to that parameter in the Datapipe statement).

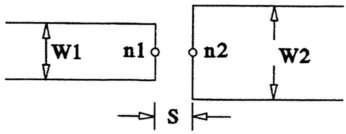
Do not attempt to add or delete parameters from the Empipe element definition. Such an attempt will lead to unpredictable results.

12.5 Empipe Library Element Catalog

This section contains the Empipe library element catalog.

The description of each element includes a schematic diagram, the element identifier, the number of nodes, the set of parameters and their default values, as well as a template of the syntax.

EM_AGAP microstrip asymmetrical gap EM_AGAP



Element Identifier

1

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W1	line width at node n1 (symmetrical)	1
W2	line width at node n2 (symmetrical)	1
S	gap spacing	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in ***em*** simulation	0 (single)
UNIT	geometrical parameter unit seen by ***em***	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
MODEL	interpolation model flag	1 (linear)

EM_AGAP microstrip asymmetrical gap EM_AGAP

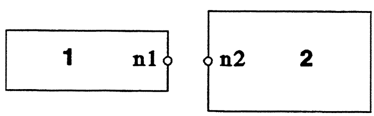
Form

```
EM_AGAP  n1 n2 0  INDEX=x1      W1=x2       W2=x3       S=x4        H=x5
                  H2=x6         EPSR=x7     UR=x8       TAND=x9     MTAND=x10
                  T=x11         RHO=x12     SR=x13      XCELL=x14   YCELL=x15
                  FMIN=x16      FMAX=x17    FSTEP=x18   ACCU=x19    UNIT=x20
                  UPDATE=x21    LEFT=x22    RIGHT=x23   TOP=x24     BOTTOM=x25
                  MIN_SUB=x26   G1MIN=x27   G1MAX=x28   G2MIN=x29   G2MAX=x30
                  MODEL=x31;
```

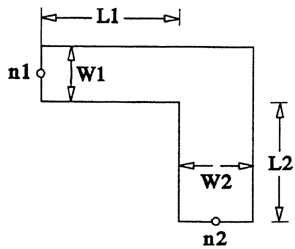
Example

```
EM_AGAP  1 2 0  INDEX=1      W1=2mil     W2=4mil      S=1mil         H=5mil
                H2=120mil    EPSR=1.9    TAND=0.001   T=3um          XCELL=1mil
                YCELL=1mil   FMIN=1GHZ   FMAX=10GHZ   FSTEP=1GHZ;
```

Polygon Groups



EM_BEND1 microstrip bend EM_BEND1



Element Identifier

2

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W1	line width at node n1	1
W2	line width at node n2	1
L1	line length at node n1	1
L2	line length at node n2	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in ***em*** simulation	0 (single)
UNIT	geometrical parameter unit seen by ***em***	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20

(*continued on the next page*)

EM_BEND1 microstrip bend EM_BEND1

Keywords

		Defaults
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
MODEL	interpolation model flag	1 (linear)

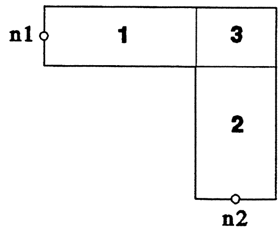
Form

```
EM_BEND1  n1 n2 0  INDEX=x1     W1=x2        W2=x3       L1=x4       L2=x5
                   H=x6         H2=x7        EPSR=x8     UR=x9       TAND=x10
                   MTAND=x11    T=x12        RHO=x13     SR=x14      XCELL=x15
                   YCELL=x16    FMIN=x17     FMAX=x18    FSTEP=x19   ACCU=x20
                   UNIT=x21     UPDATE=x22   LEFT=x23    RIGHT=x24   TOP=x25
                   BOTTOM=x26   MIN_SUB=x27  G1MIN=x28   G1MAX=x29   G2MIN=x30
                   G2MAX=x31    G3MIN=x32    G3MAX=x33   MODEL=x34;
```

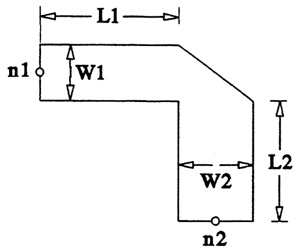
Example

```
EM_BEND1  1 2 0  INDEX=1      W1=2mil      W2=4mil     L1=3mil     L2=4mil
                 H=5mil       H2=120mil    EPSR=1.9    TAND=0.001  T=3um
                 XCELL=1mil   YCELL=1mil   FMIN=1GHZ   FMAX=10GHZ  FSTEP=1GHZ;
```

Polygon Groups



EM_BEND2 mitered microstrip bend EM_BEND2



Element Identifier

3

Keywords

		Defaults
INDEX	database index	1
W1	line width at node n1	1
W2	line width at node n2	1
L1	line length at node n1	1
L2	line length at node n2	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in ***em*** simulation	0 (single)
UNIT	geometrical parameter unit seen by ***em***	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20

(*continued on the next page*)

EM_BEND2 mitered microstrip bend EM_BEND2

Keywords

Keyword	Description	Defaults
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
MODEL	interpolation model flag	1 (linear)

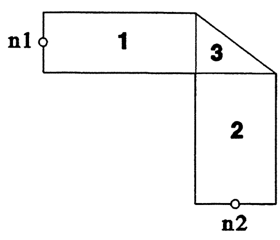
Form

```
EM_BEND2  n1 n2 0  INDEX=x1     W1=x2        W2=x3       L1=x4        L2=x5
                   H=x6         H2=x7        EPSR=x8     UR=x9        TAND=x10
                   MTAND=x11    T=x12        RHO=x13     SR=x14       XCELL=x15
                   YCELL=x16    FMIN=x17     FMAX=x18    FSTEP=x19    ACCU=x20
                   UNIT=x21     UPDATE=x22   LEFT=x23    RIGHT=x24    TOP=x25
                   BOTTOM=x26   MIN_SUB=x27  G1MIN=x28   G1MAX=x29    G2MIN=x30
                   G2MAX=x31    G3MIN=x32    G3MAX=x33   MODEL=x34;
```

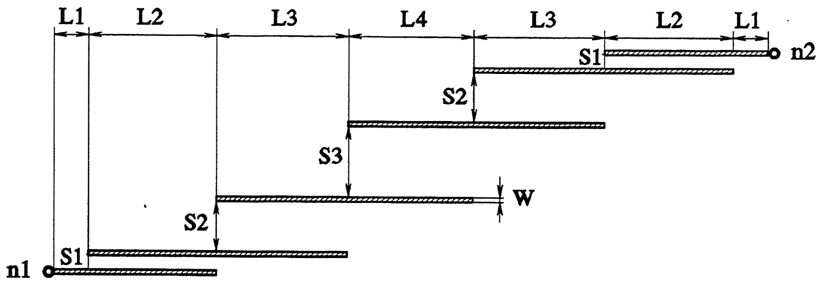
Example

```
EM_BEND2  1 2 0  INDEX=1      W1=2mil      W2=4mil     L1=3mil      L2=4mil
                 H=5mil       H2=120mil    EPSR=1.9    TAND=0.001   T=3um
                 XCELL=1mil   YCELL=1mil   FMIN=1GHZ   FMAX=10GHZ   FSTEP=1GHZ;
```

Polygon Groups



EM_CPLF microstrip coupled line filter EM_CPLF



Element Identifier

19

Keywords

Keyword	Description	Defaults
INDEX	database index	1
L1	length of input and output lines	1
L2	length of the 1st and 5th sections	1
L3	length of the 2nd and 4th sections	1
L4	length of the 3rd section	1
S1	spacing of the 1st and 5th sections	1
S2	spacing of the 2nd and 4th sections	1
S3	spacing of the 3rd section	1
W	width of lines	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1

(continued on the next page)

EM_CPLF microstrip coupled line filter EM_CPLF

Keywords

		Defaults
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	0
RIGHT	substrate box marginal spacing normalized w.r.t. H	0
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
MODEL	interpolation model flag	1 (linear)

Form

```
EM_CPLF  n1 n2 0 INDEX=x1    L1=x2       L2=x3        L3=x4       L4=x5
                 S1=x6       S2=x7       S3=x8        W=x9        H=x10
                 H2=x11      EPSR=x12    UR=x13       TAND=x14    MTAND=x15
                 T=x16       RHO=x17     SR=x18       UNIT=x19    XCELL=x20
                 YCELL=x21   ACCU=x22    UPDATE=x23   LEFT=x24    RIGHT=x25
                 TOP=x26     BOTTOM=x27  MIN_SUB=x28  G1MIN=x29   G1MAX=x30
                 FMIN=x31    FMAX=x32    FSTEP=x33    MODEL=x34;
```

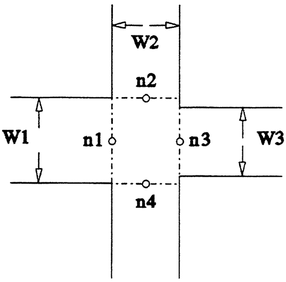
Example

```
EM_CPLF   1 2 0 INDEX=1     L1=50mil    L2=191mil    L3=198mil   L4=185mil
                S1=22mil    S2=99mil    S3=112mil    W=7mil      H=20mil
                H2=250mil   EPSR=24     TAND=3E-5    T=0mil      RHO=0
                UNIT=0      XCELL=1mil  YCELL=1mil   FMIN=3.895  FMAX=4.165
                FSTEP=0.002;
```

Polygon Group

The whole structure is considered as a single polygon group.

EM_CROSS microstrip cross junction EM_CROSS



Element Identifier

4

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W1	line width at node n1 (symmetrical)	1
W2	line width at nodes n2 and n4	1
W3	line width at node n3 (symmetrical)	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20

(*continued on the next page*)

EM_CROSS microstrip cross junction EM_CROSS

Keywords

Keyword	Description	Defaults
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
G4MIN	minimum subsection size for the 4th polygon group	1
G4MAX	maximum subsection size for the 4th polygon group	100
MODEL	interpolation model flag	1 (linear)

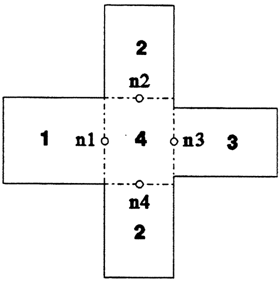
Form

```
EM_CROSS  n1 n2 n3 n4 0  INDEX=x1    W1=x2       W2=x3       W3=x4       H=x5
             H2=x6       EPSR=x7     UR=x8       TAND=x9     MTAND=x10
             T=x11       RHO=x12     SR=x13      XCELL=x14   YCELL=x15
             FMIN=x16    FMAX=x17    FSTEP=x18   ACCU=x19    UNIT=x20
             UPDATE=x21  LEFT=x22    RIGHT=x23   TOP=x24     BOTTOM=x25
             MIN_SUB=x26 G1MIN=x27   G1MAX=x28   G2MIN=x29   G2MAX=x30
             G3MIN=x31   G3MAX=x32   G4MIN=x33   G4MAX=x34   MODEL=x35;
```

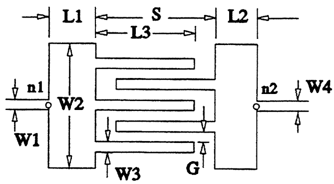
Example

```
EM_CROSS  1 2 3 4 0  INDEX=1    W1=2mil     W2=4mil      W3=3mil
                     H=5mil     H2=120mil   EPSR=1.9     TAND=0.001
                     T=3um      XCELL=1mil  YCELL=1mil
                     FMIN=1GHZ  FMAX=10GHZ  FSTEP=1GHZ;
```

Polygon Groups



EM_DCAP1 microstrip interdigital capacitor model 1 EM_DCAP1



Element Identifier

7

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W1	line width at node n1 (symmetrical)	1
W2	end patch width (symmetrical)	1
W3	finger width (symmetrical)	1
W4	line width at node n2 (symmetrical)	1
L1	end patch length at node n1	1
L2	end patch length at node n2	1
L3	finger length	1
S	patch spacing	1
G	finger spacing	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3

(*continued on the next page*)

EM_DCAP1 microstrip interdigital capacitor model 1 EM_DCAP1

Keywords

Keyword	Description	Defaults
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
G4MIN	minimum subsection size for the 4th polygon group	1
G4MAX	maximum subsection size for the 4th polygon group	100
G5MIN	minimum subsection size for the 5th polygon group	1
G5MAX	maximum subsection size for the 5th polygon group	100
MODEL	interpolation model flag	1 (linear)

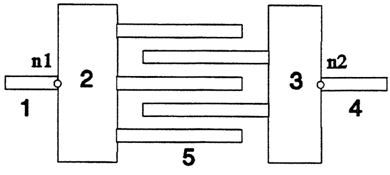
Form

```
EM_DCAP1   n1 n2 0   INDEX=x1     W1=x2         W2=x3        W3=x4        W4=x5
                     L1=x6        L2=x7         L3=x8        S=x9         G=x10
                     H=x11        H2=x12        EPSR=x13     UR=x14       TAND=x15
                     MTAND=x16    T=x17         RHO=x18      SR=x19       XCELL=x20
                     YCELL=x21    FMIN=x22      FMAX=x23     FSTEP=x24    ACCU=x25
                     UNIT=x26     UPDATE=x27    LEFT=x28     RIGHT=x29    TOP=x30
                     BOTTOM=x31   MIN_SUB=x32   G1MIN=x33    G1MAX=x34    G2MIN=x35
                     G2MAX=x36    G3MIN=x37     G3MAX=x38    G4MIN=x39    G4MAX=x40
                     G5MIN=x41    G5MAX=x42     MODEL=x43;
```

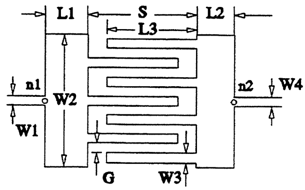
Example

```
EM_DCAP1   1 2 0   INDEX=1       W1=2mil       W2=18mil     W3=2mil       W4=2mil
                   L1=6mil       L2=7mil       L3=5mil      S=10mil       G=1mil
                   H=5mil        H2=120mil     EPSR=1.9     TAND=0.001    T=3um
                   XCELL=1mil    YCELL=1mil    FMIN=1GHZ    FMAX=10GHZ    FSTEP=1GHZ;
```

Polygon Groups



EM_DCAP2 microstrip interdigital capacitor model 2 EM_DCAP2



Element Identifier

8

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W1	line width at node n1 (symmetrical)	1
W2	end patch width (symmetrical)	1
W3	finger width (symmetrical)	1
W4	line width at node n2 (symmetrical)	1
L1	end patch length at node n1	1
L2	end patch length at node n2	1
L3	finger length	1
S	patch spacing	1
G	finger spacing	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3

(*continued on the next page*)

EM_DCAP2 microstrip interdigital capacitor model 2 EM_DCAP2

Keywords

Keyword	Description	Defaults
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
G4MIN	minimum subsection size for the 4th polygon group	1
G4MAX	maximum subsection size for the 4th polygon group	100
G5MIN	minimum subsection size for the 5th polygon group	1
G5MAX	maximum subsection size for the 5th polygon group	100
MODEL	interpolation model flag	1 (linear)

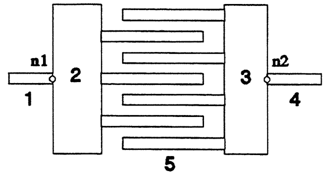
Form

```
EM_DCAP2   n1 n2 0  INDEX=x1     W1=x2         W2=x3        W3=x4       W4=x5
                    L1=x6        L2=x7         L3=x8        S=x9        G=x10
                    H=x11        H2=x12        EPSR=x13     UR=x14      TAND=x15
                    MTAND=x16    T=x17         RHO=x18      SR=x19      XCELL=x20
                    YCELL=x21    FMIN=x22      FMAX=x23     FSTEP=x24   ACCU=x25
                    UNIT=x26     UPDATE=x27    LEFT=x28     RIGHT=x29   TOP=x30
                    BOTTOM=x31   MIN_SUB=x32   G1MIN=x33    G1MAX=x34   G2MIN=x35
                    G2MAX=x36    G3MIN=x37     G3MAX=x38    G4MIN=x39   G4MAX=x40
                    G5MIN=x41    G5MAX=x42     MODEL=x43;
```

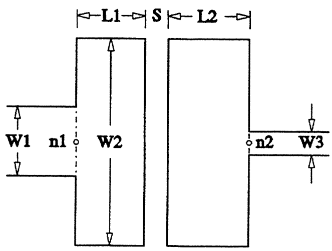
Example

```
EM_DCAP2   1 2 0  INDEX=1      W1=2mil       W2=18mil     W3=2mil      W4=2mil
                  L1=6mil      L2=7mil       L3=5mil      S=10mil      G=1mil
                  H=5mil       H2=120mil     EPSR=1.9     TAND=0.001   T=3um
                  XCELL=1mil   YCELL=1mil    FMIN=1GHZ    FMAX=10GHZ   FSTEP=1GHZ;
```

Polygon Groups



EM_DPCAP microstrip double patch capacitor EM_DPCAP



Element Identifier

9

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W1	line width at node n1 (symmetrical)	1
W2	patch width (symmetrical)	1
W3	line width at node n2 (symmetrical)	1
L1	patch length at node n1	1
L2	patch length at node n2	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3

(*continued on the next page*)

EM_DPCAP microstrip double patch capacitor EM_DPCAP

Keywords

		Defaults
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
MODEL	interpolation model flag	1 (linear)

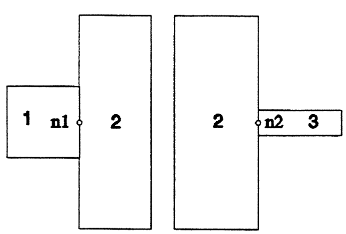
Form

```
EM_DPCAP  n1 n2 0  INDEX=x1     W1=x2       W2=x3        W3=x4         L1=x5
                   L2=x6        S=x7        H=x8         H2=x9         EPSR=x10
                   UR=x11       TAND=x12    MTAND=x13    T=x14         RHO=x15
                   SR=x16       XCELL=x17   YCELL=x18    FMIN=x19      FMAX=x20
                   FSTEP=x21    ACCU=x22    UNIT=x23     UPDATE=x24    LEFT=x25
                   RIGHT=x26    TOP=x27     BOTTOM=x28   MIN_SUB=x29   G1MIN=x30
                   G1MAX=x31    G2MIN=x32   G2MAX=x33    G3MIN=x34     G3MAX=x35
                   MODEL=x36;
```

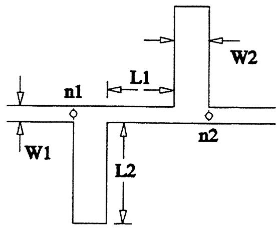
Example

```
EM_DPCAP  1 2 0  INDEX=1      W1=2mil     W2=8mil      W3=4mil      L1=3mil
                 L2=5mil      S=1mil      H=5mil       H2=120mil    EPSR=1.9
                 TAND=0.001   T=3um       XCELL=1mil   YCELL=1mil
                 FMIN=1GHZ    FMAX=10GHZ  FSTEP=1GHZ;
```

Polygon Groups



EM_DSTB1 symmetrical microstrip double stub EM_DSTB1



Element Identifier

5

Keywords

		Defaults
INDEX	database index	1
W1	line width	1
W2	stub width	1
L1	stub spacing	1
L2	stub length	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20

(continued on the next page)

EM_DSTB1 symmetrical microstrip double stub EM_DSTB1

Keywords

		Defaults
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
MODEL	interpolation model flag	1 (linear)

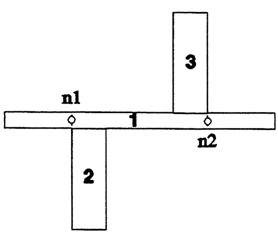
Form

```
EM_DSTB1   n1 n2 0  INDEX=x1     W1=x2        W2=x3       L1=x4        L2=x5
                    H=x6         H2=x7        EPSR=x8     UR=x9        TAND=x10
                    MTAND=x11    T=x12        RHO=x13     SR=x14       XCELL=x15
                    YCELL=x16    FMIN=x17     FMAX=x18    FSTEP=x19    ACCU=x20
                    UNIT=x21     UPDATE=x22   LEFT=x23    RIGHT=x24    TOP=x25
                    BOTTOM=x26   MIN_SUB=x27  G1MIN=x28   G1MAX=x29    G2MIN=x30
                    G2MAX=x31    G3MIN=x32    G3MAX=x33   MODEL=x34;
```

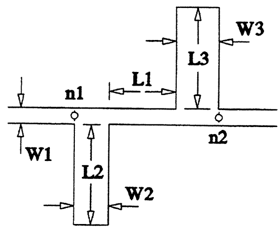
Example

```
EM_DSTB1   1 2 0  INDEX=1      W1=2mil      W2=3mil     L1=3mil      L2=5mil
                  H=5mil       H2=120mil    EPSR=1.9    TAND=0.001   T=3um
                  XCELL=1mil   YCELL=1mil   FMIN=1GHZ   FMAX=10GHZ   FSTEP=1GHZ;
```

Polygon Groups



EM_DSTB2 asymmetrical microstrip double stub EM_DSTB2



Element Identifier

6

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W1	line width	1
W2	stub width at node n1	1
W3	stub width at node n2	1
L1	stub spacing	1
L2	stub length at node n1	1
L3	stub length at node n2	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in ***em*** simulation	0 (single)
UNIT	geometrical parameter unit seen by ***em***	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3

(*continued on the next page*)

EM_DSTB2 asymmetrical microstrip double stub EM_DSTB2

Keywords

		Defaults
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
MODEL	interpolation model flag	1 (linear)

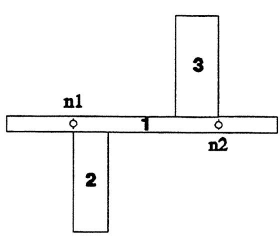
Form

```
EM_DSTB2  n1 n2 0  INDEX=x1     W1=x2       W2=x3        W3=x4        L1=x5
                   L2=x6        L3=x7       H=x8         H2=x9        EPSR=x10
                   UR=x11       TAND=x12    MTAND=x13    T=x14        RHO=x15
                   SR=x16       XCELL=x17   YCELL=x18    FMIN=x19     FMAX=x20
                   FSTEP=x21    ACCU=x22    UNIT=x23     UPDATE=x24   LEFT=x25
                   RIGHT=x26    TOP=x27     BOTTOM=x28   MIN_SUB=x29  G1MIN=x30
                   G1MAX=x31    G2MIN=x32   G2MAX=x33    G3MIN=x34    G3MAX=x35
                   MODEL=x36;
```

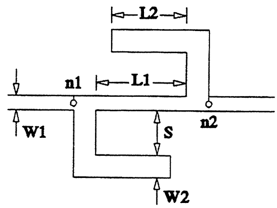
Example

```
EM_DSTB2  1 2 0  INDEX=1      W1=2mil      W2=3mil     L1=3mil      L2=5mil
                 H=5mil       H2=120mil    EPSR=1.9    TAND=0.001   T=3um
                 XCELL=1mil   YCELL=1mil   FMIN=1GHZ   FMAX=10GHZ   FSTEP=1GHZ;
```

Polygon Groups



EM_FDST1 symmetrical microstrip folded double stub EM_FDST1



Element Identifier

10

Keywords

		Defaults
INDEX	database index	1
W1	line width	1
W2	stub width	1
L1	stub spacing	1
L2	stub length	1
S	line-stub spacing	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size for both the X-axis and the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100

(*continued on the next page*)

EM_FDST1 symmetrical microstrip folded double stub EM_FDST1

Keywords

		Defaults
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
MODEL	interpolation model flag	1 (linear)

Notes

▷ The folded structure of this element requires a square cell, i.e., the cell size along the X-axis must be identical to the cell size along the Y-axis. Therefore, only the parameter XCELL is user-definable. YCELL is automatically made equal to XCELL.

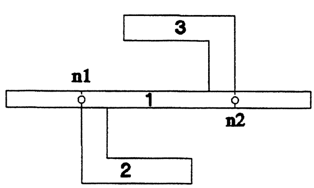
Form

```
EM_FDST1  n1 n2 0  INDEX=x1     W1=x2        W2=x3      L1=x4      L2=x5
                   S=x6         H=x7         H2=x8      EPSR=x9    UR=x10
                   TAND=x11     MTAND=x12    T=x13      RHO=x14    SR=x15
                   XCELL=x16    FMIN=x17     FMAX=x18   FSTEP=x19  ACCU=x20
                   UNIT=x21     UPDATE=x22   LEFT=x23   RIGHT=x24  TOP=x25
                   BOTTOM=x26   MIN_SUB=x27  G1MIN=x28  G1MAX=x29  G2MIN=x30
                   G2MAX=x31    G3MIN=x32    G3MAX=x33  MODEL=x34;
```

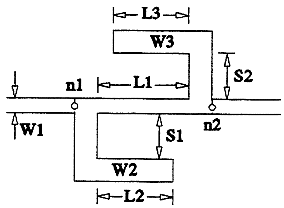
Example

```
EM_FDST1  1 2 0  INDEX=1  W1=2mil     W2=3mil    L1=6mil     L2=5mil
                 S=2mil   H=5mil      H2=120mil  EPSR=1.9    TAND=0.001
                 T=3um    XCELL=1mil  FMIN=1GHZ  FMAX=10GHZ  FSTEP=1GHZ;
```

Polygon Groups



EM_FDST2 asymmetrical microstrip folded double stub EM_FDST2



Element Identifier

11

Keywords

		Defaults
INDEX	database index	1
W1	line width	1
W2	stub width at node n1	1
W3	stub width at node n2	1
L1	stub spacing	1
L2	stub length at node n1	1
L3	stub length at node n2	1
S1	line-stub spacing of stub at node n1	1
S2	line-stub spacing of stub at node n2	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size for both the X-axis and the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20

(*continued on the next page*)

EM_FDST2 asymmetrical microstrip folded double stub EM_FDST2

Keywords

Keyword	Description	Defaults
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
MODEL	interpolation model flag	1 (linear)

Notes

▷ The folded structure of this element requires a square cell, i.e., the cell size along the X-axis must be identical to the cell size along the Y-axis. Therefore, only the parameter XCELL is user-definable. YCELL is automatically made equal to XCELL.

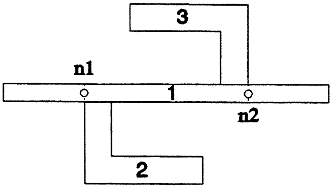
Form

```
EM_FDST2  n1 n2 0  INDEX=x1   W1=x2      W2=x3     W3=x4       L1=x5
                   L2=x6      L3=x7      S1=x8     S2=x9       H=x10
                   H2=x11     EPSR=x12   UR=x13    TAND=x14    MTAND=x15
                   T=x16      RHO=x17    SR=x18    XCELL=x19   FMIN=x20
                   FMAX=x21   FSTEP=x22  ACCU=x23  UNIT=x24    UPDATE=x25
                   LEFT=x26   RIGHT=x27  TOP=x28   BOTTOM=x29  MIN_SUB=x30
                   G1MIN=x31  G1MAX=x32  G2MIN=x33 G2MAX=x34   G3MIN=x35
                   G3MAX=x36  MODEL=x37;
```

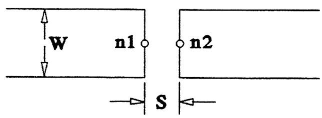
Example

```
EM_FDST2  1 2 0  INDEX=1     W1=2mil     W2=3mil      W3=2mil  L1=6mil
                 L2=5mil     L3=4mil     S1=2mil      S2=1mil  H=5mil
                 H2=120mil   EPSR=1.9    TAND=0.001   T=3um    XCELL=1mil
                 FMIN=1GHZ   FMAX=10GHZ  FSTEP=1GHZ;
```

Polygon Groups



EM_GAP microstrip symmetrical gap EM_GAP



Element Identifier

12

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W	line width (symmetrical)	1
S	gap spacing	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
MODEL	interpolation model flag	1 (linear)

EM_GAP microstrip symmetrical gap EM_GAP

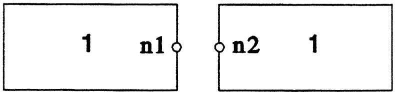
Form

```
EM_GAP   n1 n2 0  INDEX=x1    W=x2       S=x3       H=x4        H2=x5
                  EPSR=x6     UR=x7      TAND=x8    MTAND=x9    T=x10
                  RHO=x11     SR=x12     XCELL=x13  YCELL=x14   FMIN=x15
                  FMAX=x16    FSTEP=x17  ACCU=x18   UNIT=x19    UPDATE=x20
                  LEFT=x21    RIGHT=x22  TOP=x23    BOTTOM=x24  MIN_SUB=x25
                  G1MIN=x26   G1MAX=x27  MODEL=x28;
```

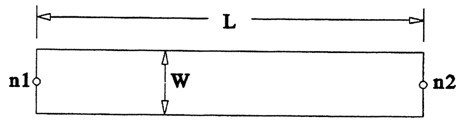
Example

```
EM_GAP   1 2 0  INDEX=1     W=4mil      S=1mil      H=5mil       H2=120mil
                EPSR=1.9    TAND=0.001  T=3um       XCELL=1mil   YCELL=1mil
                FMIN=1GHZ   FMAX=10GHZ  FSTEP=1GHZ;
```

Polygon Groups



EM_MSL microstrip line EM_MSL



Element Identifier

13

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W	line width (symmetrical)	1
L	line length	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in ***em*** simulation	0 (single)
UNIT	geometrical parameter unit seen by ***em***	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	0
RIGHT	substrate box marginal spacing normalized w.r.t. H	0
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
MODEL	interpolation model flag	1 (linear)

EM_MSL microstrip line EM_MSL

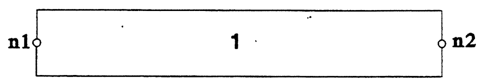
Form

```
EM_MSL   n1 n2 0   INDEX=x1    W=x2        L=x3         H=x4         H2=x5
                   EPSR=x6     UR=x7       TAND=x8      MTAND=x9     T=x10
                   RHO=x11     SR=x12      XCELL=x13    YCELL=x14    FMIN=x15
                   FMAX=x16    FSTEP=x17   ACCU=x18     UNIT=x19     UPDATE=x20
                   LEFT=x21    RIGHT=x22   TOP=x23      BOTTOM=x24   MIN_SUB=x25
                   G1MIN=x26   G1MAX=x27   MODEL=x28;
```

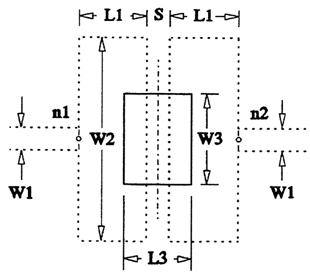
Example

```
EM_MSL   1 2 0   INDEX=1     W=2mil       L=5mil       H=5mil       H2=120mil
                 EPSR=1.9    TAND=0.001   T=3um        XCELL=1mil   YCELL=1mil
                 FMIN=1GHZ   FMAX=10GHZ   FSTEP=1GHZ;
```

Polygon Groups



EM_ODPC1 microstrip overlay double patch capacitor 1 EM_ODPC1



Element Identifier

101

Notes

▷ The elements EM_ODPC1 and EM_ODPC2 have symmetrical end patches. The only difference between them is the meaning of the parameter L3. The elements EM_ODPC3 and EM_ODPC4 have asymmetrical end patches.

▷ The top and bottom layers are represented in the schematic diagram by solid and dotted lines, respectively.

Keywords

		Defaults
INDEX	database index	1
W1	line width (symmetrical)	1
W2	end patch width (symmetrical)	1
W3	overlay patch width (symmetrical)	1
L1	end patch length at node n1 and node n2	1
L3	overlay patch length (symmetrical)	1
S	patch spacing (symmetrical)	1
H_1	top layer substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR_1	top layer relative dielectric constant of the substrate	2.35
UR_1	top layer relative permeability of the substrate	1
TAND_1	top layer dielectric loss tangent	0
MTAND_1	top layer magnetic loss tangent	0
T_1	top layer conducting metal strip thickness	0
RHO_1	top layer conducting metal strip resistivity Ωm	0
SR_1	top layer surface reactance in Ω/square	0

(*continued on the next page*)

EM_ODPC1 microstrip overlay double patch capacitor 1 EM_ODPC1

Keywords

Keyword	Description	Defaults
H_2	bottom layer substrate thickness	H_1
EPSR_2	bottom layer relative dielectric constant of the substrate	EPSR_1
UR_2	bottom layer relative permeability of the substrate	UR_1
TAND_2	bottom layer dielectric loss tangent	TAND_1
MTAND_2	bottom layer magnetic loss tangent	MTAND_1
T_2	bottom layer conducting metal strip thickness	T_1
RHO_2	bottom layer conducting metal strip resistivity Ωm	RHO_1
SR_2	bottom layer surface reactance in Ω/square	SR_1
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
MODEL	interpolation model flag	1 (linear)

Form

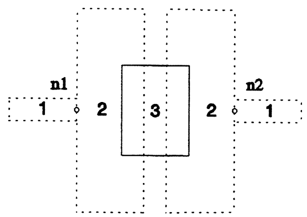
```
EM_ODPC1  n1 n2 0  INDEX=x1     W1=x2        W2=x3         W3=x4      L1=x5
                   L3=x6        S=x7         H_1=x8        H2=x9      EPSR_1=x10
                   UR_1=x11     TAND_1=x12   MTAND_1=x13   T_1=x14    RHO_1=x15
                   SR_1=x16     H_2=x17      EPSR_2=x18    UR_2=x19   TAND_2=x20
                   MTAND_2=x21  T_2=x22      RHO_2=x23     SR_2=x24   XCELL=x25
                   YCELL=x26    FMIN=x27     FMAX=x28      FSTEP=x29  ACCU=x30
                   UNIT=x31     UPDATE=x32   LEFT=x33      RIGHT=x34  TOP=x35
                   BOTTOM=x36   MIN_SUB=x37  G1MIN=x38     G1MAX=x39  G2MIN=x40
                   G2MAX=x41    G3MIN=x42    G3MAX=x43     MODEL=x44;
```

EM_ODPC1 microstrip overlay double patch capacitor 1 EM_ODPC1

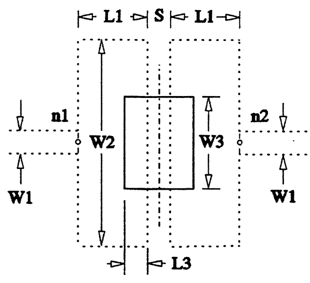
Example

```
EM_ODPC1  1 2 0  INDEX=1      W1=2mil      W2=14mil        W3=8mil        L1=10mil
                 L3=8mil      S=2mil       H_1=1mil        H_2=5mil       H2=120mil
                 EPSR_1=1.9   EPSR_2=2.9   TAND_1=0.001    T_1=3um        XCELL=1mil
                 YCELL=1mil   FMIN=1GHZ    FMAX=10GHZ      FSTEP=1GHZ;
```

Polygon Groups



EM_ODPC2 microstrip overlay double patch capacitor 2 EM_ODPC2



Element Identifier

102

Notes

- ▷ The elements EM_ODPC1 and EM_ODPC2 have symmetrical end patches. The only difference between them is the meaning of the parameter L3. The elements EM_ODPC3 and EM_ODPC4 have asymmetrical end patches.
- ▷ The top and bottom layers are represented in the schematic diagram by solid and dotted lines, respectively.

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W1	line width (symmetrical)	1
W2	end patch width (symmetrical)	1
W3	overlay patch width (symmetrical)	1
L1	end patch length at node n1 and node n2	1
L3	partial overlay patch length	1
S	patch spacing	1
H_1	top layer substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR_1	top layer relative dielectric constant of the substrate	2.35
UR_1	top layer relative permeability of the substrate	1
TAND_1	top layer dielectric loss tangent	0
MTAND_1	top layer magnetic loss tangent	0
T_1	top layer conducting metal strip thickness	0
RHO_1	top layer conducting metal strip resistivity Ωm	0

(*continued on the next page*)

EM_ODPC2 microstrip overlay double patch capacitor 2 EM_ODPC2

Keywords

Keyword	Description	Defaults
SR_1	top layer surface reactance in Ω/square	0
H_2	bottom layer substrate thickness	H_1
EPSR_2	bottom layer relative dielectric constant of the substrate	EPSR_1
UR_2	bottom layer relative permeability of the substrate	UR_1
TAND_2	bottom layer dielectric loss tangent	TAND_1
MTAND_2	bottom layer magnetic loss tangent	MTAND_1
T_2	bottom layer conducting metal strip thickness	T_1
RHO_2	bottom layer conducting metal strip resistivity Ωm	RHO_1
SR_2	bottom layer surface reactance in Ω/square	SR_1
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
MODEL	interpolation model flag	1 (linear)

Form

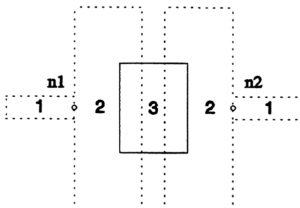
```
EM_ODPC2  n1 n2 0  INDEX=x1      W1=x2        W2=x3        W3=x4       L1=x5
                   L3=x6         S=x7         H_1=x8       H2=x9       EPSR_1=x10
                   UR_1=x11      TAND_1=x12   MTAND_1=x13  T_1=x14     RHO_1=x15
                   SR_1=x16      H_2=x17      EPSR_2=x18   UR_2=x19    TAND_2=x20
                   MTAND_2=x21   T_2=x22      RHO_2=x23    SR_2=x24    XCELL=x25
                   YCELL=x26     FMIN=x27     FMAX=x28     FSTEP=x29   ACCU=x30
                   UNIT=x31      UPDATE=x32   LEFT=x33     RIGHT=x34   TOP=x35
                   BOTTOM=x36    MIN_SUB=x37  G1MIN=x38    G1MAX=x39   G2MIN=x40
                   G2MAX=x41     G3MIN=x42    G3MAX=x43    MODEL=x44;
```

EM_ODPC2 microstrip overlay double patch capacitor 2 EM_ODPC2

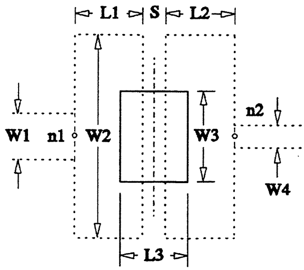
Example

```
EM_ODPC2  1 2 0  INDEX=1        W1=2mil       W2=14mil       W3=8mil       L1=10mil
                 L3=3mil        S=2mil        H_1=1mil       H_2=5mil      H2=120mil
                 EPSR_1=1.9     EPSR_2=2.9    TAND_1=0.001   T_1=3um       XCELL=1mil
                 YCELL=1mil     FMIN=1GHZ     FMAX=10GHZ     FSTEP=1GHZ;
```

Polygon Groups



EM_ODPC3 microstrip overlay double patch capacitor 3 EM_ODPC3



Element Identifier

103

Notes

- ▷ The elements EM_ODPC3 and EM_ODPC4 have asymmetrical end patches. The only difference between them is the meaning of the parameter L3. The elements EM_ODPC1 and EM_ODPC2 have symmetrical end patches.
- ▷ The top and bottom layers are represented in the schematic diagram by solid and dotted lines, respectively.

Keywords

		Defaults
INDEX	database index	1
W1	line width at node n1 (symmetrical)	1
W2	end patch width (symmetrical)	1
W3	overlay patch width (symmetrical)	1
W4	line width at node n2 (symmetrical)	1
L1	end patch length at node n1	1
L2	end patch length at node n2	1
L3	overlay patch length (symmetrical)	1
S	patch spacing (symmetrical)	1
H_1	top layer substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR_1	top layer relative dielectric constant of the substrate	2.35
UR_1	top layer relative permeability of the substrate	1
TAND_1	top layer dielectric loss tangent	0
MTAND_1	top layer magnetic loss tangent	0
T_1	top layer conducting metal strip thickness	0

(continued on the next page)

EM_ODPC3 microstrip overlay double patch capacitor 3 EM_ODPC3

Keywords

Keyword	Description	Defaults
RHO_1	top layer conducting metal strip resistivity Ωm	0
SR_1	top layer surface reactance in Ω/square	0
H_2	bottom layer substrate thickness	H_1
EPSR_2	bottom layer relative dielectric constant of the substrate	EPSR_1
UR_2	bottom layer relative permeability of the substrate	UR_1
TAND_2	bottom layer dielectric loss tangent	TAND_1
MTAND_2	bottom layer magnetic loss tangent	MTAND_1
T_2	bottom layer conducting metal strip thickness	T_1
RHO_2	bottom layer conducting metal strip resistivity Ωm	RHO_1
SR_2	bottom layer surface reactance in Ω/square	SR_1
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
G4MIN	minimum subsection size for the 4th polygon group	1
G4MAX	maximum subsection size for the 4th polygon group	100
MODEL	interpolation model flag	1 (linear)

Form

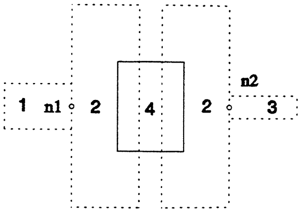
```
EM_ODPC3  n1 n2 0  INDEX=x1    W1=x2        W2=x3         W3=x4        W4=x5
                   L1=x6       L2=x7        L3=x8         S=x9         H_1=x10
                   H2=x11      EPSR_1=x12   UR_1=x13      TAND_1=x14   MTAND_1=x15
                   T_1=x16     RHO_1=x17    SR_1=x18      H_2=x19      EPSR_2=x20
                   UR_2=x21    TAND_2=x22   MTAND_2=x23   T_2=x24      RHO_2=x25
                   SR_2=x26    XCELL=x27    YCELL=x28     FMIN=x29     FMAX=x30
                   FSTEP=x31   ACCU=x32     UNIT=x33      UPDATE=x34   LEFT=x35
                   RIGHT=x36   TOP=x37      BOTTOM=x38    MIN_SUB=x39  G1MIN=x40
                   G1MAX=x41   G2MIN=x42    G2MAX=x43     G3MIN=x44    G3MAX=x45
                   G4MIN=x46   G4MAX=x47    MODEL=x48;
```

EM_ODPC3 microstrip overlay double patch capacitor 3 EM_ODPC3

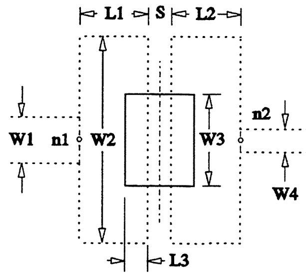
Example

```
EM_ODPC3  1 2 0  INDEX=1    W1=2mil      W2=14mil     W3=8mil      W4=4mil
                 L1=10mil   L2=5mil      L3=8mil      S=2mil       H_1=1mil
                 H_2=5mil   H2=120mil    EPSR_1=1.9   EPSR_2=2.9   TAND_1=0.001
                 T_1=3um    XCELL=1mil   YCELL=1mil   FMIN=1GHZ    FMAX=10GHZ
                 FSTEP=1GHZ;
```

Polygon Groups



EM_ODPC4 microstrip overlay double patch capacitor 4 EM_ODPC4



Element Identifier

104

Notes

▷ The elements EM_ODPC3 and EM_ODPC4 have asymmetrical end patches. The only difference between them is the meaning of the parameter L3. The elements EM_ODPC1 and EM_ODPC2 have symmetrical end patches.

▷ The top and bottom layers are represented in the schematic diagram by solid and dotted lines, respectively.

Keywords

		Defaults
INDEX	database index	1
W1	line width at node n1 (symmetrical)	1
W2	end patch width (symmetrical)	1
W3	overlay patch width (symmetrical)	1
W4	line width at node n2 (symmetrical)	1
L1	end patch length at node n1	1
L2	end patch length at node n2	1
L3	partial overlay patch length	1
S	patch spacing	1
H_1	top layer substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR_1	top layer relative dielectric constant of the substrate	2.35
UR_1	top layer relative permeability of the substrate	1
TAND_1	top layer dielectric loss tangent	0
MTAND_1	top layer magnetic loss tangent	0
T_1	top layer conducting metal strip thickness	0

(*continued on the next page*)

EM_ODPC4 microstrip overlay double patch capacitor 4 EM_ODPC4

Keywords

Keyword	Description	Defaults
RHO_1	top layer conducting metal strip resistivity Ωm	0
SR_1	top layer surface reactance in Ω/square	0
H_2	bottom layer substrate thickness	H_1
EPSR_2	bottom layer relative dielectric constant of the substrate	EPSR_1
UR_2	bottom layer relative permeability of the substrate	UR_1
TAND_2	bottom layer dielectric loss tangent	TAND_1
MTAND_2	bottom layer magnetic loss tangent	MTAND_1
T_2	bottom layer conducting metal strip thickness	T_1
RHO_2	bottom layer conducting metal strip resistivity Ωm	RHO_1
SR_2	bottom layer surface reactance in Ω/square	SR_1
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
G4MIN	minimum subsection size for the 4th polygon group	1
G4MAX	maximum subsection size for the 4th polygon group	100
MODEL	interpolation model flag	1 (linear)

Form

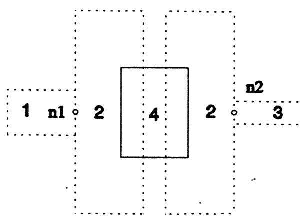
```
EM_ODPC4  n1 n2 0  INDEX=x1   W1=x2       W2=x3        W3=x4       W4=x5
                   L1=x6      L2=x7       L3=x8        S=x9        H_1=x10
                   H2=x11     EPSR_1=x12  UR_1=x13     TAND_1=x14  MTAND_1=x15
                   T_1=x16    RHO_1=x17   SR_1=x18     H_2=x19     EPSR_2=x20
                   UR_2=x21   TAND_2=x22  MTAND_2=x23  T_2=x24     RHO_2=x25
                   SR_2=x26   XCELL=x27   YCELL=x28    FMIN=x29    FMAX=x30
                   FSTEP=x31  ACCU=x32    UNIT=x33     UPDATE=x34  LEFT=x35
                   RIGHT=x36  TOP=x37     BOTTOM=x38   MIN_SUB=x39 G1MIN=x40
                   G1MAX=x41  G2MIN=x42   G2MAX=x43    G3MIN=x44   G3MAX=x45
                   G4MIN=x46  G4MAX=x47   MODEL=x48;
```

EM_ODPC4 microstrip overlay double patch capacitor 4 EM_ODPC4

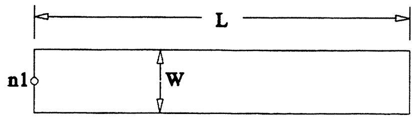
Example

```
EM_ODPC4  1 2 0  INDEX=1    W1=2mil     W2=14mil     W3=8mil      W4=4mil
                 L1=10mil   L2=5mil     L3=3mil      S=2mil       H_1=1mil
                 H_2=5mil   H2=120mil   EPSR_1=1.9   EPSR_2=2.9   TAND_1=0.001
                 T_1=3um    XCELL=1mil  YCELL=1mil   FMIN=1GHZ    FMAX=10GHZ
                 FSTEP=1GHZ;
```

Polygon Groups



EM_OPEN microstrip open stub EM_OPEN



Element Identifier

14

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W	line width (symmetrical)	1
L	line length	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	0
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
MODEL	interpolation model flag	1 (linear)

EM_OPEN microstrip open stub EM_OPEN

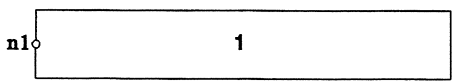
Form

```
EM_OPEN   n1 0   INDEX=x1    W=x2        L=x3         H=x4          H2=x5
                 EPSR=x6     UR=x7       TAND=x8      MTAND=x9      T=x10
                 RHO=x11     SR=x12      XCELL=x13    YCELL=x14     FMIN=x15
                 FMAX=x16    FSTEP=x17   ACCU=x18     UNIT=x19      UPDATE=x20
                 LEFT=x21    RIGHT=x22   TOP=x23      BOTTOM=x24    MIN_SUB=x25
                 G1MIN=x26   G1MAX=x27   MODEL=x28;
```

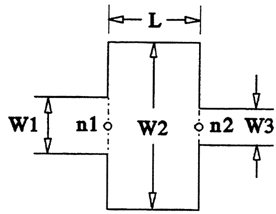
Example

```
EM_OPEN   1 0   INDEX=1     W=2mil      L=15mil      H=5mil       H2=120mil
                EPSR=1.9    TAND=0.001  T=3um        XCELL=1mil   YCELL=1mil
                FMIN=1GHZ   FMAX=10GHZ  FSTEP=1GHZ;
```

Polygon Groups



EM_RECT microstrip rectangular structure EM_RECT



Element Identifier

15

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W1	line width at node n1 (symmetrical)	1
W2	rectangle width (symmetrical)	1
W3	line width at node n2	1
L	rectangle length	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100

(*continued on the next page*)

EM_RECT microstrip rectangular structure EM_RECT

Keywords

		Defaults
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
MODEL	interpolation model flag	1 (linear)

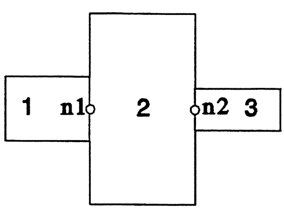
Form

```
EM_RECT  n1 n2 0  INDEX=x1     W1=x2         W2=x3       W3=x4       L=x5
                  H=x6         H2=x7         EPSR=x8     UR=x9       TAND=x10
                  MTAND=x11    T=x12         RHO=x13     SR=x14      XCELL=x15
                  YCELL=x16    FMIN=x17      FMAX=x18    FSTEP=x19   ACCU=x20
                  UNIT=x21     UPDATE=x22    LEFT=x23    RIGHT=x24   TOP=x25
                  BOTTOM=x26   MIN_SUB=x27   G1MIN=x28   G1MAX=x29   G2MIN=x30
                  G2MAX=x31    G3MIN=x32     G3MAX=x33   MODEL=x34;
```

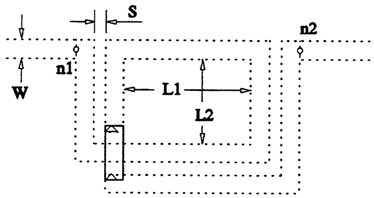
Example

```
EM_RECT  1 2 0  INDEX=1      W1=2mil       W2=24mil    W3=4mil      L=8mil
                H=5mil       H2=120mil     EPSR=1.9    TAND=0.001   T=3um
                XCELL=1mil   YCELL=1mil    FMIN=1GHZ   FMAX=10GHZ   FSTEP=1GHZ;
```

Polygon Groups



EM_SPIND microstrip spiral inductor EM_SPIND



Element Identifier

105

Notes

▷ The top and bottom layers are represented in the schematic diagram by solid and dotted lines, respectively.

Keywords

Keyword	Description	Defaults
INDEX	database index	1
N	number of spiral turns (must be an integer between 1 and 5)	1
W	line width	1
L1	interior spiral length	1
L2	interior spiral height	1
S	spiral line spacing	1
H_1	top layer substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR_1	top layer relative dielectric constant of the substrate	2.35
UR_1	top layer relative permeability of the substrate	1
TAND_1	top layer dielectric loss tangent	0
MTAND_1	top layer magnetic loss tangent	0
T_1	top layer conducting metal strip thickness	0
RHO_1	top layer conducting metal strip resistivity Ωm	0
SR_1	top layer surface reactance in Ω/square	0
H_2	bottom layer substrate thickness	H_1
EPSR_2	bottom layer relative dielectric constant of the substrate	EPSR_1
UR_2	bottom layer relative permeability of the substrate	UR_1
TAND_2	bottom layer dielectric loss tangent	TAND_1
MTAND_2	bottom layer magnetic loss tangent	MTAND_1
T_2	bottom layer conducting metal strip thickness	T_1
RHO_2	bottom layer conducting metal strip resistivity Ωm	RHO_1
SR_2	bottom layer surface reactance in Ω/square	SR_1

(*continued on the next page*)

EM_SPIND microstrip spiral inductor EM_SPIND

Keywords

Keyword	Description	Defaults
XCELL	cell size along the X-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
MODEL	interpolation model flag	1 (linear)

Form

```
EM_SPIND  n1 n2 0  INDEX=x1     N=x2          W=x3      L1=x4       L2=x5
                   S=x6         H_1=x7        H2=x8     EPSR_1=x9   UR_1=x10
                   TAND_1=x11   MTAND_1=x12   T_1=x13   RHO_1=x14   SR_1=x15
                   H_2=x16      EPSR_2=x17    UR_2=x18  TAND_2=x19  MTAND_2=x20
                   T_2=x21      RHO_2=x22     SR_2=x23  XCELL=x24   FMIN=x25
                   FMAX=x26     FSTEP=x27     ACCU=x28  UNIT=x29    UPDATE=x30
                   LEFT=x31     RIGHT=x32     TOP=x33   BOTTOM=x34  MIN_SUB=x35
                   G1MIN=x36    G1MAX=x37     MODEL=x38;
```

Example

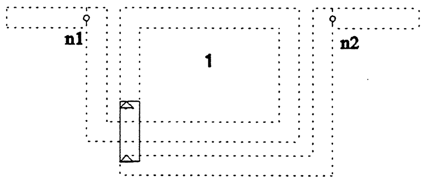
```
EM_SPIND  1 2 0  INDEX=1       N=2            W=2mil    L1=7mil     L2=5mil
                 S=1mil        H_1=1mil       H_2=5mil  H2=120mil   EPSR_1=1.9
                 EPSR_2=2.9    TAND_1=0.001   T_1=3um   XCELL=1mil  FMIN=1GHZ
                 FMAX=10GHZ    FSTEP=1GHZ;
```

Notes

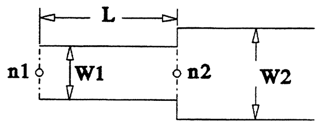
▷ The spiral structure of this element requires a square cell, i.e., the cell size along the X-axis must be identical to the cell size along the Y-axis. Therefore, only the parameter XCELL is user-definable. YCELL is automatically made equal to XCELL.

EM_SPIND microstrip spiral inductor EM_SPIND

Polygon Groups



EM_STEP microstrip step junction EM_STEP



Element Identifier

16

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W1	line width at node n1 (symmetrical)	1
W2	line width at node n2 (symmetrical)	1
L	line length at node n1	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	0
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
MODEL	interpolation model flag	1 (linear)

EM_STEP microstrip step junction EM_STEP

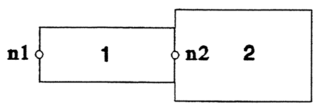
Form

```
EM_STEP  n1 n2 0  INDEX=x1     W1=x2      W2=x3       L=x4        H=x5
                  H2=x6        EPSR=x7    UR=x8       TAND=x9     MTAND=x10
                  T=x11        RHO=x12    SR=x13      XCELL=x14   YCELL=x15
                  FMIN=x16     FMAX=x17   FSTEP=x18   ACCU=x19    UNIT=x20
                  UPDATE=x21   LEFT=x22   RIGHT=x23   TOP=x24     BOTTOM=x25
                  MIN_SUB=x26  G1MIN=x27  G1MAX=x28   G2MIN=x29   G2MAX=x30
                  MODEL=x31;
```

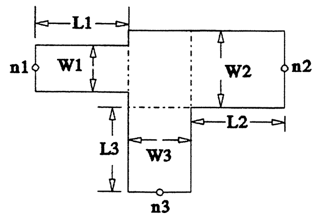
Example

```
EM_STEP  1 2 0  INDEX=1      W1=2mil    W2=4mil     L=8mil       H=5mil
                H2=120mil    EPSR=1.9   TAND=0.001  T=3um        XCELL=1mil
                YCELL=1mil   FMIN=1GHZ  FMAX=10GHZ  FSTEP=1GHZ;
```

Polygon Groups



EM_TEE microstrip T-junction EM_TEE



Element Identifier

17

Keywords

Keyword	Description	Defaults
INDEX	database index	1
W1	line width at node n1 (symmetrical)	1
W2	line width at node n2 (symmetrical)	1
W3	line width at node n3	1
L1	line length at node n1	1
L2	line length at node n2	1
L3	line length at node n3	1
H	substrate thickness	0.635mm
H2	shielding height	-1 (no shielding)
EPSR	relative dielectric constant of the substrate	2.35
UR	relative permeability of the substrate	1
TAND	dielectric loss tangent	0
MTAND	magnetic loss tangent	0
T	conducting metal strip thickness	0
RHO	conducting metal strip resistivity Ωm	0
SR	surface reactance in Ω/square	0
XCELL	cell size along the X-axis	1
YCELL	cell size along the Y-axis	1
FMIN	lower limit of the frequency range	0
FMAX	upper limit of the frequency range	1
FSTEP	size of increments in the frequency range	1
ACCU	precision in *em* simulation	0 (single)
UNIT	geometrical parameter unit seen by *em*	3 (mm)
UPDATE	database update option	1
LEFT	substrate box marginal spacing normalized w.r.t. H	3
RIGHT	substrate box marginal spacing normalized w.r.t. H	3
TOP	substrate box marginal spacing normalized w.r.t. H	3
BOTTOM	substrate box marginal spacing normalized w.r.t. H	3

(*continued on the next page*)

EM_TEE microstrip T-junction EM_TEE

Keywords

		Defaults
MIN_SUB	minimum number of subsections per wavelength	20
G1MIN	minimum subsection size for the 1st polygon group	1
G1MAX	maximum subsection size for the 1st polygon group	100
G2MIN	minimum subsection size for the 2nd polygon group	1
G2MAX	maximum subsection size for the 2nd polygon group	100
G3MIN	minimum subsection size for the 3rd polygon group	1
G3MAX	maximum subsection size for the 3rd polygon group	100
G4MIN	minimum subsection size for the 4th polygon group	1
G4MAX	maximum subsection size for the 4th polygon group	100
MODEL	interpolation model flag	1 (linear)

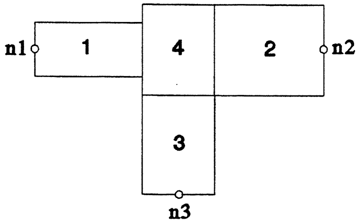
Form

```
EM_TEE  n1 n2 n3 0  INDEX=x1    W1=x2      W2=x3       W3=x4
         L1=x5      L2=x6       L3=x7      H=x8        H2=x9
         EPSR=x10   UR=x11      TAND=x12   MTAND=x13   T=x14
         RHO=x15    SR=x16      XCELL=x17  YCELL=x18   FMIN=x19
         FMAX=x20   FSTEP=x21   ACCU=x22   UNIT=x23    UPDATE=x24
         LEFT=x25   RIGHT=x26   TOP=x27    BOTTOM=x28  MIN_SUB=x29
         G1MIN=x30  G1MAX=x31   G2MIN=x32  G2MAX=x33   G3MIN=x34
         G3MAX=x35  MODEL=x36;
```

Example

```
EM_TEE  1 2 3 0  INDEX=1     W1=2mil      W2=4mil      W3=3mil      L1=8mil
                 L2=4mil     L3=0         H=5mil       H2=120mil    EPSR=1.9
                 TAND=0.001  T=3um        XCELL=1mil   YCELL=1mil   FMIN=1GHZ
                 FMAX=10GHZ  FSTEP=1GHZ;
```

Polygon Groups



Index

OSA

Dundas, Ontario, Canada