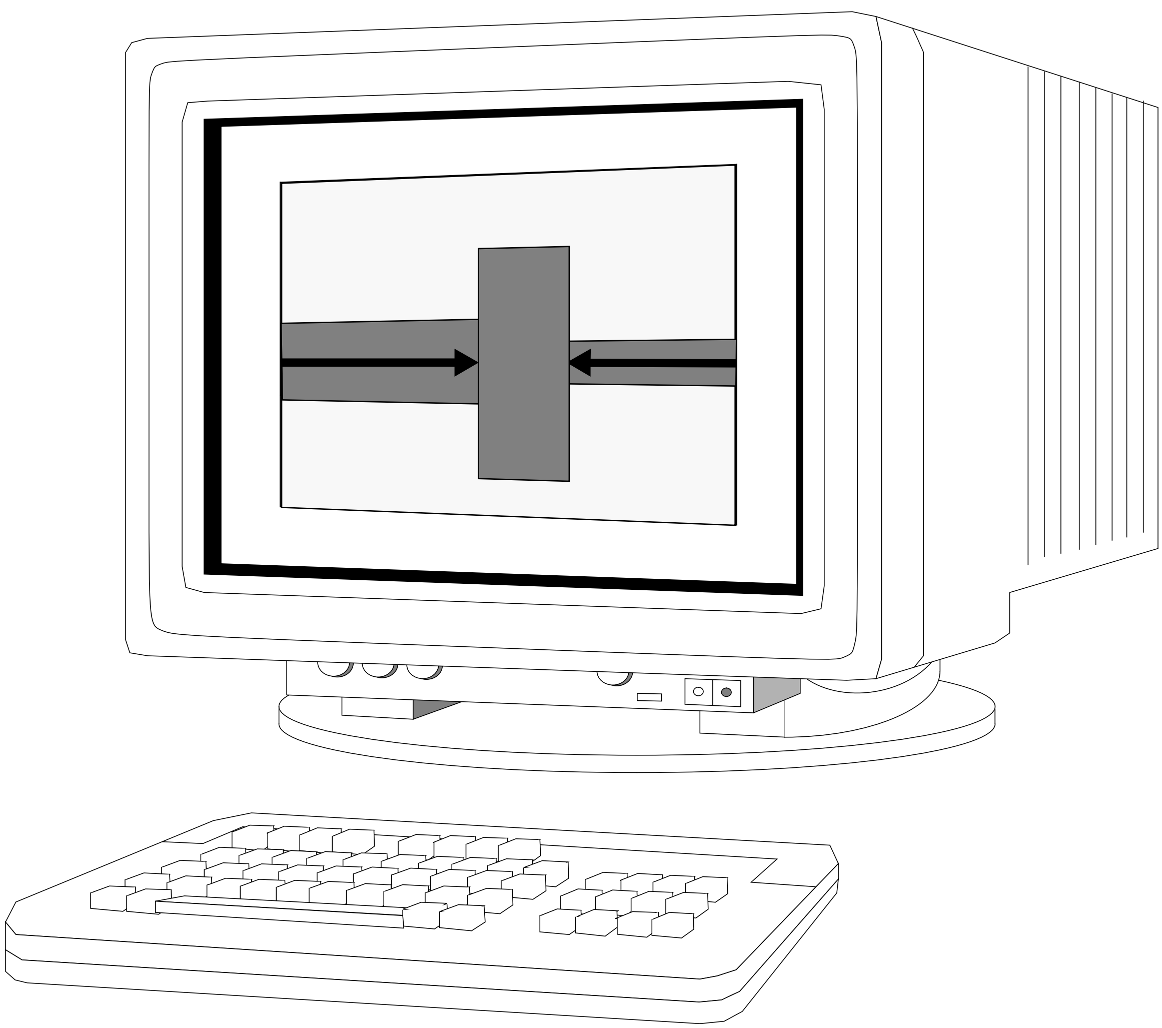
Em User's Manual



Em User's Manual

Printed: September 1, 1993

Release 2.4

Sonnet Software, Inc.
135 Old Cove Road, Suite 203
Liverpool, NY 13090-3774
(315) 453-3096

This publication is subject to change at any time and without notice. Any suggestions for improvements in this publication or in the software it describes are welcome.

Disclaimer of Warranties And Limitation of Liabilities

TABLE OF CONTENTS

Chapter 1 Introduction

Em is an electromagnetic analysis [1], [14], [16] which calculates S-parameters for arbitrary 3-D planar [5] (e.g., microstrip, coplanar, stripline, etc.) geometries, maintaining full accuracy at all microwave frequencies. ***Em*** is a “full-wave” analysis in that it takes into account all possible coupling mechanisms. The analysis inherently includes dispersion, stray coupling, discontinuities, surface waves, moding, metallization loss, dielectric loss, radiation loss; in short, it is a complete electromagnetic analysis. Since ***em*** uses a surface meshing technique (it meshes only the surface of the circuit metallization), ***em*** can analyze predominately planar circuits much faster than volume meshing techniques.

Em is a full three dimensional analysis in that both 3-D fields and 3-D currents are included in the analysis. This is in contrast to 2.5-D analyses which, while including full 3-D fields, allow only 2-D current. Thus, a 2.5-D analysis does not allow vias or any other vertical current. ***Em*** analyzes 3-D structures embedded in planar multilayered dielectric, thus ***em*** is appropriate for “3-D planar” circuits. For this class of circuits, ***em*** can use the FFT (Fast Fourier Transform) to efficiently calculate the electromagnetic coupling on and between each dielectric surface. This provides ***em*** with its several orders of magnitude of speed increase over volume meshing and other non-FFT based techniques.

Em performs an electromagnetic analysis of a microstrip, stripline, coplanar waveguide, or any other 3-D planar circuit (see Figure 1) by solving for the current distribution in the circuit metallization using the Method of Moments.

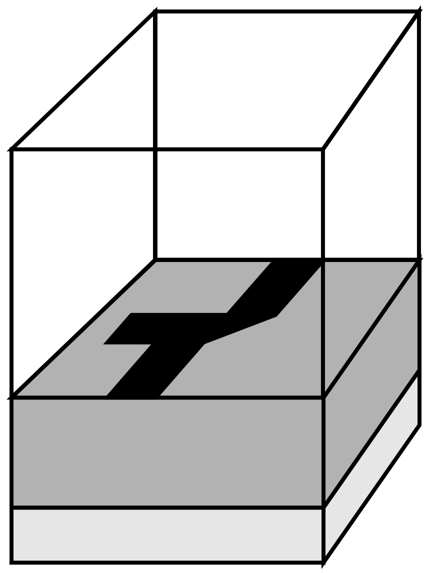


Figure 1 ***Em*** analyzes planar structures inside a shielding box. Port connections are usually made at the box sidewalls.

♦ What Can *Em* Do?

As we mentioned, ***em*** is an analysis of arbitrary 3-D structures embedded in planar dielectrics. Since ***em*** is appropriate for 3-D planar structures, vias (to ground and between layers) are allowed. The via capability allows analysis of air bridges, wire bonds, spiral inductors, wafer probes and internal ports as well as the more usual ground vias.

It is appropriate to use ***em*** for:

- Evaluation of specific discontinuities or groups of interacting discontinuities to assist in the design of a 3-D planar circuit. ***Em*** provides ultra-precise S-parameters for discontinuities allowing designers to work with confidence. ***Em*** can also quickly synthesize an equivalent lumped model for discontinuities. The lumped model can be used directly in circuit theory programs.

- Design validation. This effectively eliminates expensive design iterations (i.e., “tweak”, refabricate, etc.) of the passive, planar portion of a circuit. At present, because ***em*** makes no compromise in accuracy, performing an analysis of, say, an entire amplifier with a single electromagnetic analysis on a mid-range workstation is difficult. However, validation of large portions of an amplifier design is reasonable. If a circuit is designed from the start with electromagnetic analysis in mind, much larger circuits can be done. For example, the analysis works best with tightly packed, rectangular circuits.
- Microwave package evaluation. What happens when the lid goes on? ***Em*** analyzes a circuit inside a conducting box. If the box (acting as a dielectric loaded resonator) is resonant at a frequency where the circuits still have gain, bad things happen. ***Em***, under the assumption of uniform dielectric within each layer of dielectric, provides an analysis of a package prior to fabrication. Resonances can then be dealt with on the computer rather than in a machine shop.
- Certain microstrip antennas. The “top” of ***em***’s box can be effectively removed by setting the cover resistivity to 377 ohms/square. While radiation is outside of ***em***’s primary thrust, a wide variety of microstrip antennas and radiating discontinuities can be evaluated.
- High speed digital interconnect. When an approximate model is not good enough, ***em*** can synthesize a SPICE lumped model including all delays and couplings. The lumped model is synthesized directly from electromagnetic data.

Em is not appropriate for doing an initial design. Rather, the faster circuit theory simulators (which do not typically include stray coupling) should be used for the first cut. ***Em*** can then enhance the simulator performance by providing custom, ultra-precise discontinuity data and for validating large portions of the final circuit, including all stray interactions. Consistent with our philosophy at Sonnet Software, ***em*** is designed to work with your existing CAE software, enhancing your capability, in a “win-win” relationship.

In addition, since the output data is in Touchstone or Compact format (at the user’s discretion), you can use the S-parameter plotting capability of whatever microwave CAE tool you are already familiar with. For time-domain users, SPICE lumped model output is another option. In addition, both EEsof and Cadence have developed their own interfaces to Sonnet.

Figure 2 shows how ***em*** works in conjunction with other Sonnet programs as well as with programs from third parties.

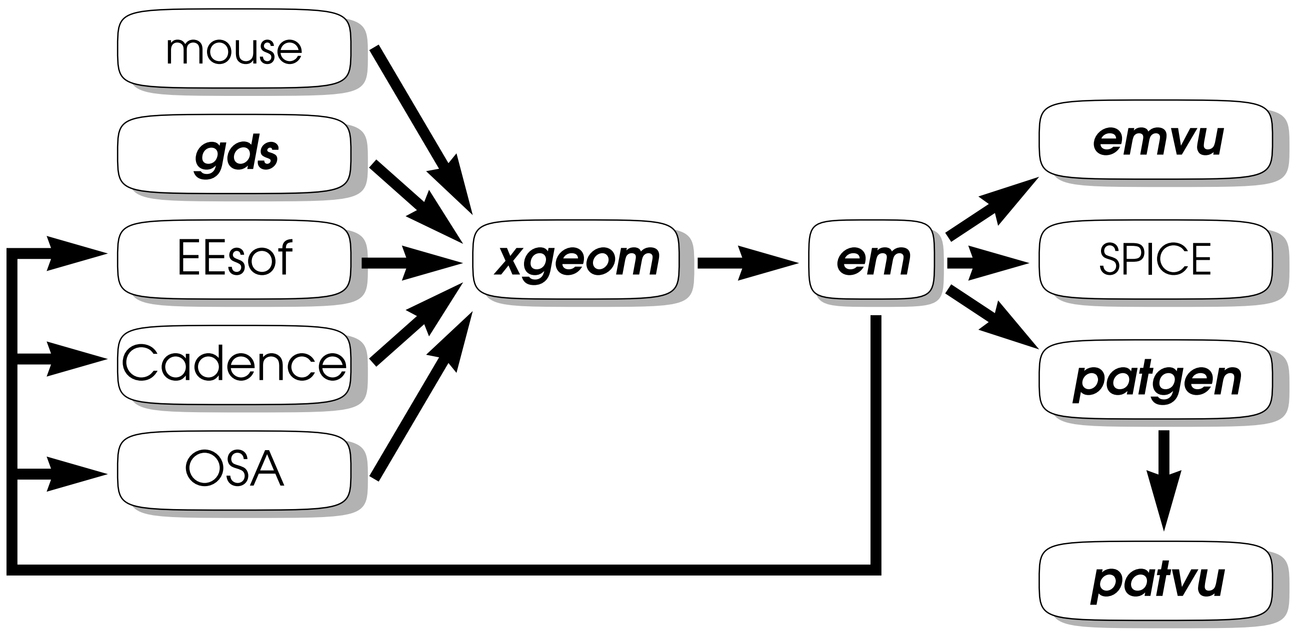


Figure 2 Relationship of Sonnet programs. "Mouse" indicates manual capture or modification of circuit geometry. "SPICE" is any SPICE software. Other non-bold text indicates software developed by other companies which interfaces with Sonnet. Sonnet products are listed in bold. Arrows indicate data flow.

♦ Software Un-Limits

Another part of our philosophy at Sonnet Software is to develop software which does not artificially restrict creative abilities, that is, software which knows no limits.

For example, the full version of ***em*** can handle any number of ports, any number of layers, any number of subsections, any number of frequencies over any frequency range. The list of limitless parameters is, itself, almost limitless. Such software is more difficult to develop, but it is worth it, because it sets you free to explore with your creativity unrestricted by any arbitrary, programmer specified limits.

Em is limited only by the hardware on which it is running. If you need better performance, just get a faster computer or more memory. The full version of ***em*** never limits you to, for example, "No More Than 4 ports".

♦ Where Did *Em* Come From?

The technique used in ***em*** was developed at Syracuse University in 1986 by Rautio and Harrington [1], [14], [16]. It was originally developed as an extension of an analysis of planar waveguide probes [2]. The technique expresses the fields inside the box as a sum of waveguide modes and is thus closely related to the spectral domain approach.

The complete theory has been published in detail in peer reviewed journals. We feel this is important and very much in the best interest of our customers. A full bibliography of relevant papers is presented at the end of this chapter.

♦ The *Em* User

The ***em*** user should have a good working knowledge of microwave design. In order to make use of ***em***'s results, access to a microwave circuit theory based analysis program is desirable. The user should also have at least some familiarity with computers including a minimal working knowledge of a text editor (which can be acquired in 15 minutes, Appendix V, "A Very Brief Introduction to Vi"). If you are working with the EEsof or Cadence interface to Sonnet, no knowledge of a text editor is needed.

♦ Enhancements with Release 2.4

This section summarizes new capability in detail. If you are not yet familiar with ***em***, you may want to just skim this section, skipping any terms that are unfamiliar. If you are an experienced user, this section merits detailed reading.

- ***Em*** now has a simplified command line. For example, if you type "em -vdj filter", input is assumed from "filter.geo", de-embedded S-Parameters go to "filter.d", and current distribution goes to "filter.jxy". In addition, output from one run can be directed to both de-embedded, "*.d", and non-de-embedded, "*.nd", files simultaneously. Frequencies are read from "ctl.an", as before, or from the keyboard if "ctl.an" does not exist. See Appendix I, "The Em Command Line" for more details.
- In prior releases, new current distributions output to an existing *.jxy file were merely appended to that file. Since ***emvu*** only looks at the first current distribution stored in a file, each current distribution had to be sent to different file names. With the new release, if the current distribution is about to be sent to an existing ".jxy" file, the existing file is renamed "*.jxb" and a new "*.jxy" file is created to hold the new data. Any previously existing "*.jxb" file is lost. In this way, only one current distribution is stored per file.

- In another enhancement to data output, when running with the verbose option, “-v”, S-parameter data is sent to the screen as it is calculated. In addition, timing information for the first frequency is both displayed and stored in the data file.
- A newly enhanced and documented facility is the SPICE synthesis option. ***Em*** can take any structure which is small with respect to wavelength, perform an electromagnetic analysis, and use the resulting data to synthesize a lumped network. The output is an ASCII net list in a “.subckt” format for incorporation directly in a SPICE deck. This capability has two important applications. First, in microwave work, obtaining lumped models of complex discontinuities is now almost trivial. Second, high speed digital applications often need an accurate lumped model for evaluation of cross-talk, distortion, and pulse delay. The SPICE option can be used for circuits with hundreds of ports. See Chapter 17, "SPICE Lumped Model Synthesis".
- Although in development for over 10 years now, we still keep finding more sophisticated ways of improving accuracy. As just one example, we found that if very short vias (1.5 microns long) are analyzed at low frequency (less than 2 GHz), residual error starts increasing unexpectedly. We found that this was due to numerical precision in the routines which calculate the coupling between subsections, i.e., the matrix fill. Such problems are inherent in many electromagnetic analyses. To answer this problem we added a “-q” option. This option changes the matrix fill from the usual double precision to quadruple precision. The matrix itself is still solved in the usual single or double precision. This option is important for low frequency analysis of small integrated circuit structures. ***Em*** invokes the option automatically in most cases when it becomes clearly needed.
- As another enhancement to accuracy, we found that when two polygons on different levels overlap, the residual error could again, under certain circumstances, become unexpectedly large. This happens when, as viewed from above, subsections on one polygon extend over the edge of the other polygon. To solve this problem, ***em*** now checks one level down and one level up for any polygon edges. If a polygon edge is found, the subsectioning is restarted at that edge. This high level of subsectioning sophistication is invoked automatically. The user may disable it, if desired, by using the “-e” option (see Appendix I, "The Em Command Line").
- Any vias captured (by using ***xgeom***) in the new release are automatically positioned on the inside of the base polygon. In prior releases, vias could be on either side if the polygon edge falls exactly on a cell boundary. The position of vias in files created prior to this release remain unchanged, even when viewed or analyzed using the present release.
- For the really esoteric minded, we have modified ***em*** so that dielectric constants less than unity can be specified. Dielectric constants must still be greater than zero. Such capability is useful in plasma physics.

- To speed analysis when there are a large number of subsections, we have added a parallel subsections removal capability. This is accessed in ***xgeom*** under the “Param ⇒ Parallel Subs...” option. The option removes all transverse current subsections within a user specified distance of each box edge. Transverse subsections are parallel to the box sidewall and, in most cases, carry no current anyway. All the current usually flows in the longitudinal direction, perpendicular to the box sidewall. See the section "Parallel Subsection Removal Option" on page 28.
- The memory estimation routine has been modified to give a more accurate estimate of the memory required for a particular analysis.
- Whenever de-embedding is invoked, a self-consistency check is now made of the de-embedding results. If the self consistency check provides anything but excellent results, a warning message is output. Box resonances (which can be detected and/or removed with the -b and -B options), multi-mode port line propagation, and use of the single precision matrix solution (-m option) are typical causes.
- Several new commands for the ctl.an file have been added. For example, FMAX followed by a frequency in the units previously specified in ctl.an, sets the frequency used to calculate the actual length for the subsections per wavelength parameter. If not present, the largest frequency in the ctl.an file determines FMAX. Also, open circuit limits for the SPICE file synthesis may be specified using LMAX, CMIN, RMAX, KMIN (see Appendix II, "The Analysis Control File Format").
- And for solving those really big problems ***em*** now allows a special kind of subsectioning. Previous releases already include the ability to vary how quickly subsections become large as their distance from polygon corners and edges increases. This is available in the XMIN, YMIN, XMAX, YMAX parameters in “*Objects ⇒Polygons ⇒Parameters ⇒ [Select a polygon] ⇒Subsections*” option. However, in previous releases, the edge subsections were always 1 cell wide regardless of the value of XMIN and YMIN. XMIN and YMIN were used only on interior subsections.

  With the new release, XMIN and YMIN are also used on the edge of the polygon if the polygon has nothing but right angle edges. We call this a Manhattan polygon, after the street layout of New York City. Manhattan polygons are automatically detected and the new subsectioning is automatically used. This means that, for example, a spiral inductor with 9 micron gaps and 10 micron wide lines can be analyzed easily and quickly. Just set XMIN and YMIN to 10 cells (assuming a 1 micron cell size) for all polygons and the subsections become exactly 10 microns (one subsection wide) with exactly 9 micron gaps. To access XMIN, and YMIN, use *Objects ⇒ Polygons ⇒ Param ⇒ [Select Polygon] ⇒ Subsections* (see also page 27).
- A new utility is provided which checks how much memory is available for use by ***em*** on your system. Available memory can be decreased by the window system, extra open windows, and other users. See the **Utilities Manual** for more details.

- Another utility allows new users to easily get their own private copies of the example files. For example, to get the example file, “steps.geo”, the user just types “copyex steps.geo”. The file is copied from the Sonnet directory into the user’s present directory.
- This manual includes several new and significantly revised sections. Be sure to read Chapter 18, "Accuracy Benchmarking". We have performed detailed residual error evaluations so you can answer that difficult question, “Just how good is it?” very precisely. A section has been added on viewing tangential electric fields. If these are of interest, or if you just want to see some snazzy images, check out the section "Viewing Tangential Electric Fields" on page 56. Finally, the chapter on antenna analysis has been updated. If any of the conditions for valid analysis are violated, it is easy to get incorrect results. If antenna analysis is part of your work, be sure to read it carefully.

We put a lot of work into this release, and there is quite a bit of new capability here. We also put strong emphasis on quality. Anyone who has used a computer knows the frustration of working with poorly designed or tested software. We hope you never experience that frustration with our software and we have done our best to make sure that everything goes smoothly with your use of Sonnet. But if there is ever a moment of frustration or uncertainty, we want to hear about it. Give us a call!

♦ Describing *Xgeom* Menu Button Presses

In the rest of this manual, we describe accessing the menu items of ***xgeom*** using a “pointer” description to illustrate selecting the desired menu buttons.

For example, *Param* ⇒ *Box* ⇒ *Area* means to move the cursor to the *Param* menu button in the main button menu, press (and hold down) the left mouse button, drag the cursor down the menu which appears until *Box* is highlighted, and release the mouse button. When the new button menu appears, click on *Area*.

♦ Organization of This Manual

In the next section, a brief outline of ***em***’s theory provides background for using ***em***. A description of the environment expected by ***em*** follows. Then we dive into a series of examples illustrating ***em***’s capabilities. A detailed reference for ***em***’s command line and file formats provides the appendices.

♦ Using the Example Files

In the following chapters, we refer to several example files which are supplied with ***em***. To use these files, you must copy them into your directory. A utility called ***copyex*** has been provided with ***em*** which copies an example file into your directory. To copy an example file, type in "copyex" followed by the file name. For example, to copy the example file "steps.geo", type:

copyex steps.geo

Copyex copies the example file from the Sonnet example directory into your directory. If a file in your directory has the same name, ***copyex*** overwrites it with the original example file. This feature is useful if you modify an example file and want to restore the file to its original configuration.

♦ References

[1] J. C. Rautio and R. F. Harrington, "An Electromagnetic Time-Harmonic Analysis of Shielded Microstrip Circuits", IEEE MTT Transactions, Aug. 1987, Vol. MTT-35, No. 8, pp. 726-730.

[2] R. F. Harrington, **Time-Harmonic Electromagnetic Fields**, New York: McGraw-Hill, 1961, section 8-11.

[3] J. C. Rautio, "Synthesis of Lumped Models from N-Port Scattering Parameter Data," IEEE Trans. Microwave Theory Tech., accepted for publication.

[4] J. C. Rautio, "Some Comments on Electromagnetic De-Embedding and Microstrip Characteristic Impedance" International Journal of Microwave & Millimeter-Wave Computer-Aided Engineering, Vol. 3, No. 2, April 1993, pp. 151-153.

[5] J. C. Rautio, "Some Comments on Electromagnetic Dimensionality," IEEE MTT-S Newsletter, Winter 1992, pg. 23.

[6] J. C. Rautio, "Experimental Validation of Electromagnetic Software," International Journal of Microwave & Millimeter-Wave Computer-Aided Engineering, Vol. 1, No. 4, Oct. 1991, pp. 379-385.

[7] J. C. Rautio, "A New Definition of Characteristic Impedance", MTT International Symposium Digest, June 1991, Boston, pp. 761-764.

[8] J. C. Rautio, "A De-Embedding Algorithm for Electromagnetics," International Journal of Microwave & Millimeter-Wave Computer-Aided Engineering, Vol.1, No. 3, July 1991, pp. 282-287

[9] J. C. Rautio, "Triangle Cells in an Electromagnetic Analysis of Arbitrary Microstrip Circuits," MTT International Microwave Symposium Digest, Dallas, June 1990.

[10] J. C. Rautio, "Experimental Validation of Microwave Software," ARFTG Conference Digest, Dallas, June 1990. (This paper was voted best paper at the conference.)

[11] J. C. Rautio, "A Possible Source of Error in On-Wafer Calibration," ARFTG Conference Digest, Ft. Lauderdale, Nov. 1989, pp. 118-126.

[12] J. C. Rautio, "An Experimental Investigation of the Microstrip Step Discontinuity," IEEE Trans. Microwave Theory Tech., Vol. MTT-37, pp. 1816-1818, Nov. 1989.

[13] J. C. Rautio, "Microstrip Program Improves Accuracy of Circuit Models," Microwaves & RF, Vol. 27, No. 12, pp. 89-96, Nov. 1988.

[14] J. C. Rautio and R. F. Harrington, "An Efficient Electromagnetic Analysis of Arbitrary Microstrip Circuits," MTT International Microwave Symposium Digest, Las Vegas, June 1987, pp. 295-298.

[15] J. C. Rautio and R. F. Harrington, "Results and Experimental Verification of an Electromagnetic Analysis of Microstrip Circuits," Trans. of The Society for Computer Simulation, Vol. 4, No. 2, pp. 125-156, Apr. 1987.

[16] J. C. Rautio, "A Time-Harmonic Electromagnetic Analysis of Shielded Microstrip Circuits," Ph. D. Dissertation, Syracuse University, Syracuse, NY, 1986.

[17] J. C. Rautio, "Preliminary Results of a Time-Harmonic Electromagnetic Analysis of Shielded Microstrip Circuits," ARFTG Conference Digest, Baltimore, pp. 121-134, June 1986. (Voted best paper at the conference.)

[18] J. C. Rautio, "Techniques for Correcting Scattering Parameter Data of an Imperfectly Terminated Multiport When Measured with a Two-Port Network Analyzer," IEEE Trans. Microwave Theory Tech., Vol. MTT-31, pp. 407-412, May 1983.

[19] J. C. Rautio, "Educational Use of a Microwave Electromagnetic Analysis of 3-D Planar Structures," Computer Applications in Engineering Education, Vol. 1, No. 3, 1993, pp. 243-254

[20] J. C. Rautio, "Modeling Radiation Using the ***em*** Electromagnetic Analysis," Sonnet Application Note 93-01, March 1993.

[21] J. C. Rautio, "Comparison of Strategies for Analysis of Diagonal Structures," Sonnet Application Note 93-02, May 1993.

Chapter 2 A Simple Outline of the Theory

Another part of the Sonnet Software philosophy is to inform, rather than merely impress. There has been much written (especially in electromagnetics) with the primary intent of impressing the reader, while conveying minimal useful information. We feel that such "content free" text has no place in engineering. Thus, we present a description of ***em***'s theory with the intent to inform. A full set of references is at the end of Chapter 1.

♦ Subsectioning the Circuit

The analysis starts by subdividing the circuit metallization into small rectangular subsections. In an actual subsectioning, (see Figure 3), small subsections are used where needed and larger subsections where the analysis does not need small subsections. Triangular subsections can be used to fill in the diagonal "staircase", at the user's discretion.

Em then evaluates the electric field everywhere due to the current in a single subsection.

Next, ***em*** repeats the calculation for every subsection in the circuit, one at a time. In so doing, ***em*** effectively calculates the “coupling” between each possible pair of subsections in the circuit.

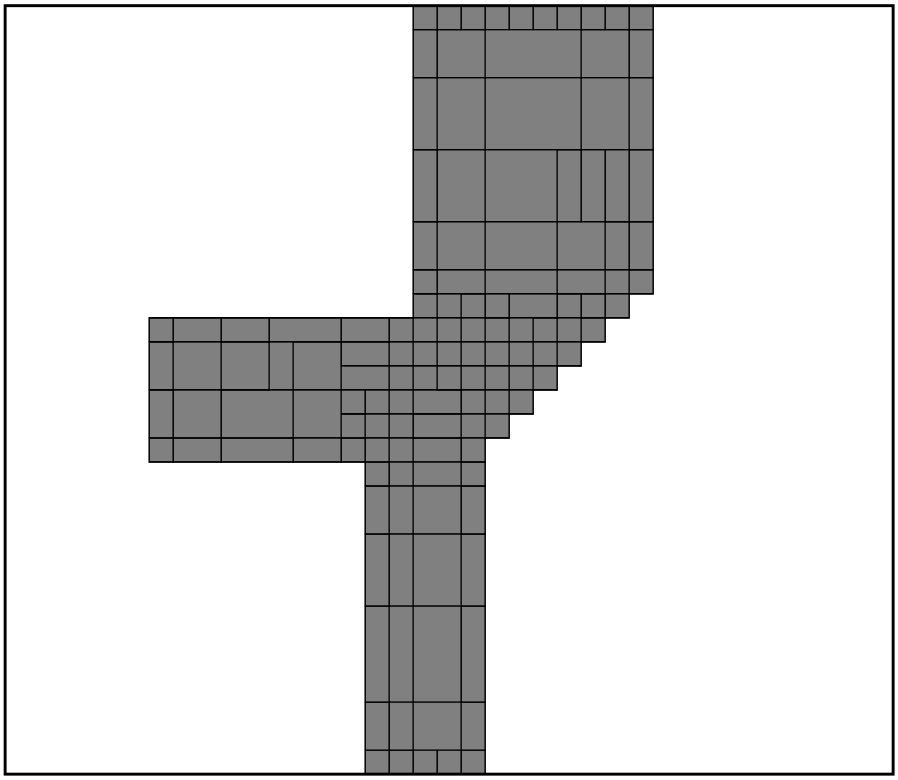


Figure 3 ***Em*** calculates the tangential electric field on all subsections, given current on one subsection. This figure shows the actual subsectioning for an example circuit.

♦ Voltage Across a Conductor is Zero

Each subsection generates electric field everywhere on the surface of the substrate, but we know that the total tangential electric field must be zero on the surface of any conductor (this is the boundary condition, no voltage allowed across a perfect conductor).

The problem is solved by assuming current on all subsections simultaneously. ***Em*** adjusts these currents so that the total tangential electric field (i.e., the sum of all the individual electric fields just calculated) goes to zero everywhere that there is conductor. The currents that do this form the current distribution on the metallization. Once we have the currents, the S-parameters (or Y, or Z, etc.) follow immediately.

If there is metallization loss, we modify the boundary condition. Rather than zero tangential electric field (zero voltage), we make the tangential electric field (the voltage on each subsection) proportional to the current in the subsection. The constant of proportionality is the metallization surface resistivity (in ohms/square). Sound familiar? It's Ohm's Law!

♦ An Alternate Description

We hope our intent to inform has been successful. In contrast, if our intent had been only to impress, the analysis would have been described in the following "content free" format:

> *This analytical full-wave numerical approach invokes a Galerkin method of moments technique which develops the dyadic Green's function operator as a bi-dimensional infinite vectorial summation of homogeneous rectangular waveguide eigenfunctions resulting in a subsectional technique using roof-top expansion and testing functions closely related to spectral domain techniques.*

Both descriptions, by the way, are accurate.

Chapter 3 The Environment Expected by *Em*

If ***em*** has not yet been installed on your system, refer to the **Sonnet Installation Manual** for installation information.

Em is computation and memory intensive. As much real memory as possible is recommended. If more real memory is required than is installed, the program starts swapping out to disk, increasing execution time substantially. We recommend a minimum of 8 Mbytes, up to 64 Mbytes (and occasionally even more) can be useful.

There are no special display or peripheral requirements. Unlike ***xgeom***, the X Window System is not required.

♦ The Input and Output Files

Em does four things:

- Reads an ASCII ".geo" file describing the circuit.
- Reads an ASCII ".an" file specifying the analysis frequencies.
- Number crunches.
- Outputs an ASCII file with the result.

There are three files associated with ***em***: a file which controls the analysis (default name “ctl.an”), a file which specifies the circuit geometry (from ***xgeom***, default name “cir.geo”), and a circuit response file (default name “cir.nd”). The default file names are used if there is no file name with the appropriate extension (the characters following the period in the file name) in the command line.

The analysis control file determines the frequencies for analysis. Its other uses are detailed in Appendix II, "The Analysis Control File Format". An example file is given in the next section. If a name other than “ctl.an” is selected, a name ending with “.an” is recommended.

The circuit geometry file is generated with a geometry capture program such as ***xgeom***. Geometry files can also be generated by means of the interfaces provided by EEsof and Cadence, as well as by means of Sonnet utilities operating on GDSII Stream files.

The geometry file specifies the size of the box containing the circuit, the number of layers and the dielectric constant of each layer as well as a polygonal description of the metallization outline. The format of the file is specified in Appendix III, "The Circuit Geometry File Format". Experienced users may wish to make minor changes directly to the file rather than going through the geometry capture program. A geometry file name ending with “.geo” is recommended.

The format of the geometry file is simple. In fact several users have written simple programs to convert output from their favorite CAD program (for example, AutoCad) into “.geo” format.

The circuit response file format can be selected on the command line for compatibility with popular microwave circuit analysis programs. The circuit response file contains automatically generated comment lines as well as copies of any comment lines in the analysis control and geometry files. A file name ending with “.d” is recommended for de-embedded data and a file name ending with “.nd” is recommended for response files in which the data is not de-embedded.

Chapter 4 The First Example -- Cascaded Steps

The circuit for the first example is described in the ***Xgeom* User's Manual**, starting in the Chapter 4, "Specifying Box Parameters". If you followed the tutorial example there, the circuit should now be stored in the file “steps.geo”. If not, copy the file “steps.geo” from the examples directory using the command:

copyex steps.geo

♦ Creating the Analysis Control File

Before doing an analysis, we usually create an analysis control file. Unless you are using the EEsof or Cadence interfaces, this requires editing a file. If you prefer not to take the time to create this file, there is no problem. If the file does not exist, ***em*** prompts you for the required analysis frequencies interactively.

If you are not familiar with an editor and wish to learn one, we recommend the vi editor for UNIX systems. It takes about 15 minutes to learn the four or five commands needed for this example as described in Appendix V, "A Very Brief Introduction to

Vi". While no longer considered a top line editor, vi is useful in that it is always available on all UNIX systems. For MS-DOS systems, use any text editor capable of saving an ASCII file to disk, such as the MS-DOS Edit command.

If you choose to do so, edit a new file named “ctl.an”. Enter the following text using upper or lower case letters.

```
VER 2.4
GHZ
FRE 10.0
```

The first line is the ***em*** version number. It is not needed now, but we strongly recommend including it for compatibility with future versions.

The second line indicates the units for frequency. The only other choice is “MHZ” which is the default. Switching between MHZ and GHZ is permitted at any point in the file.

The third line indicates that we do the analysis at 10 GHz only. A line like

```
FRE 1 10 2
```

starts the analysis at 1 GHz and ends it at 10 GHz with a step of 2 GHz. You can enter in as many FRE lines as you like in any order.

Other than the amount of time you are willing to wait for an analysis to complete, there is no limit on the number of frequencies.

The “ctl.an” file and the “steps.geo” file are used for the example runs of ***em*** in this section.

♦ Baseline Run of Steps.geo

To run ***em*** for the first example, type:

em -v steps

and press the *Return* key.

The "steps.geo" file, as displayed by ***xgeom***, is shown in Figure 4.

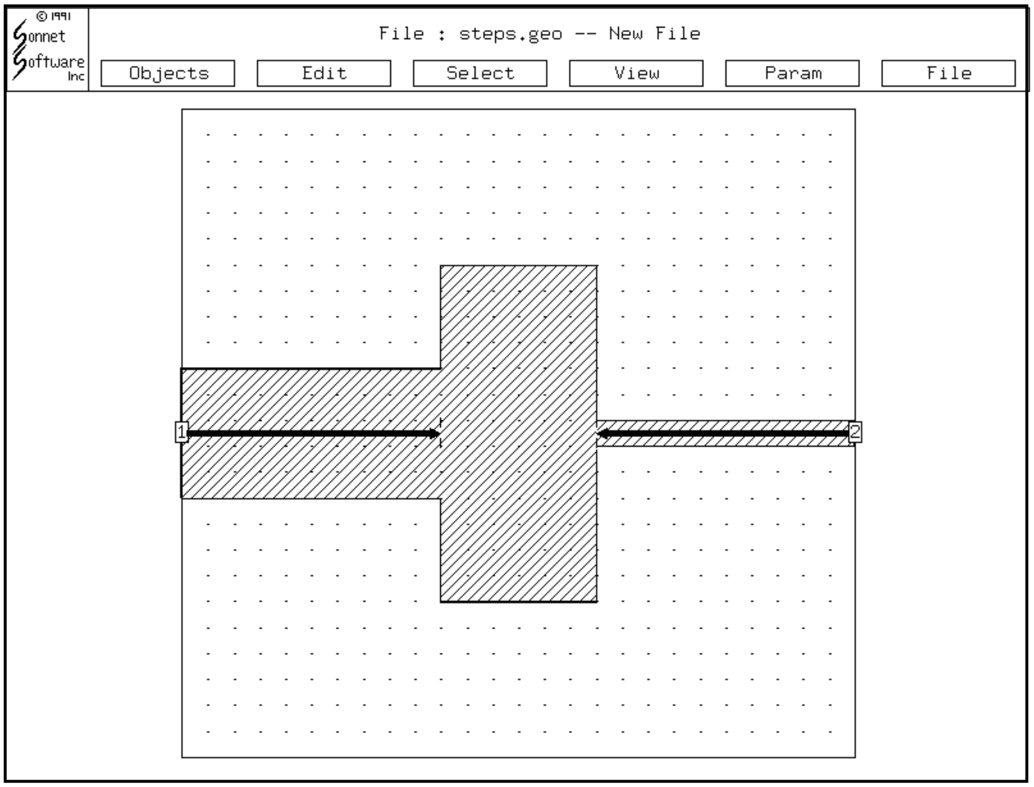


Figure 4 Our first example, in the file "steps.geo", is a cascade of two steps. The arrows indicate reference planes. Reference planes are ignored until de-embedding is used in a subsequent chapter.

If you did not create the ctl.an file, ***em*** asks for a start, stop and step frequency in MHz. Unlike using an analysis control file, all three numbers must be given, they must be in MHz and only one set of start, stop and step frequencies may be specified. For this example, specify 10000 for each start, stop, and step.

♦ *Em*'s Output to the Screen

The "-v" (verbose mode) in the command line causes the program to print out its status as the analysis proceeds:

```
Circuit geometry file -- steps.geo --
Analysis control file -- ctl.an --
Non-de-embedded response file -- steps.nd --

                  ELECTROMAGNETIC ANALYSIS OF 3-D PLANAR CIRCUITS
                                    Version 2.4

                  (C) Copyright 1987 - 1993 Sonnet Software, Inc.
                               All rights reserved.

Starting subsectioning...done.
Circuit uses 70 subsections, 1 Mbyte.
Analyzing 10.000000 GHz.
 Loading matrix, source level 0, field level 0.
 Reducing matrix.
 Analysis time per frequency (mm:ss) -- 0:00 user, 0:00 system, 0:00 real
10.0000000 0.515668 152.64 0.856789 -163.5 0.856789 -163.5 0.515668 60.400

em: finished.
```

The first three lines indicate the files which are being used for the analysis. For example, the circuit response file is "steps.nd". Since the command line ended with "steps" with no extension, ***em*** assumes that input comes from steps.geo and output goes to steps.nd. The ".nd" stands for "not de-embedded". De-embedding is covered in subsequent chapters. Frequencies are specified by "ctl.an". The "steps.nd" file may be used just like measured data in the usual microwave circuit theory programs.

The file information is followed by the title page. Then ***em*** subsections the circuit. Large circuits can take several minutes to complete subsectioning.

When finished subsectioning, ***em*** prints out the number of subsections and estimated required memory for the circuit given the cell size specified in ***xgeom***. This gives an indication of how long the analysis will take. As you do different circuits, you will get a good feel for what is a reasonable number and what is not. On an HP-710 with 64 MBytes of RAM, the above example of about a hundred subsections is fast, less than one second. A circuit of 1,000 subsections requires 10 or 20 minutes per frequency and 4,000 subsections is about the upper limit.

This line indicates that 1 Mbyte of memory is estimated to be required. The estimate of memory required is given to the nearest Mbyte, but it can be in error by more than 1 Mbyte. To find out how much of your computer memory is available for use, use the "checkmem" command (see the **Utilities Manual**). Available memory is decreased by the window system, by the operating system, by other programs running, and by other users.

When finished with the automatic subsectioning, ***em*** displays the first frequency of analysis and then starts the analysis. At this time, ***em*** is setting parameters for the analysis.

When complete, ***em*** states that it is loading the matrix. This is where ***em*** calculates the electric field generated by current in each subsection. For this circuit there is only one metallization level, level 0. ***Em*** calculates the electric field induced in level 0 which is due to current in each level 0 subsection. With more levels, each level-to-level coupling is calculated separately.

Depending on your computer, there may be another line printed out next that is not shown here. This line gives the amount of time spent filling the matrix. The computer used for this example was fast enough that the time was less than 1 second. When this occurs, ***em*** does not print this line. You may also see this difference in other places in this manual.

The next line of ***em*** output indicates that it is reducing the matrix. ***Em*** is now effectively adjusting the current in each subsection for zero tangential electric field on all metallization. Matrix reduction is about one half of a matrix inversion. A custom LU decomposition algorithm (which takes advantage of the special characteristics of our problem) is used.

The next line of the output provides timing statistics for the first analysis frequency. The format is mm:ss, or minutes followed by seconds. In this case, the circuit is so small all times are less than one half second and they print out as 0:00.

The first time is “user” time. That’s the amount of time the computer spent doing useful work on your problem. “System” time is the amount of time spent doing less useful things, like swapping memory to disc and other file access. If the “system” time is large, it is an indication that there are too many programs running at the same time, or that the computer needs more memory. It is also frequently possible to reduce the complexity of the problem you are solving. See Chapter 5, "Tricks to Get a Faster Analysis".

“Real” time is the actual clock time from start to finish. For example, if there were a second CPU intensive program running, your “real” time might be twice your “user” time because the computer spent half of it’s time working on the other program. You can expect your timings to be different from those listed in this manual.

On longer analyses, the time required for various portions of the analysis are also displayed during analysis at the first frequency. Timing information is also stored in the output data files.

Finally, the resulting S-parameter data is displayed. Default is EEsof Touchstone format. In this case, the first number is the frequency, 10 GHz. The next number is S_{11} magnitude followed by S_{11} phase in degrees. This is followed by S_{21}, S_{12}, and S_{22}, both magnitude and phase. This data is also stored in “steps.nd” as described in the next section.

♦ The Circuit Response File

When complete, the resulting S-parameter data is displayed on the screen and also stored in the file “steps.nd”. Your file may look slightly different:

```
! **************************************************
! em -v steps -- Wed May 12 10:45:02 1993
! em 2.4, id sonnet.101, on 2009840242.
! Last time steps.geo was updated - Wed Dec 9 12:19:12 1992
!-> Example of 1) two cascaded steps, or
!-> 2) a cross junction and two short stubs.
!-> With em, there is no need to decide which way to model it.
! 20 subs/wavelength at 10.000000 GHz.
! All dimensions are in mils.
! A = 160.000000(16), B = 170.000000(17), C = 125.000000, with 2 layers.
! Layer  H( mils)     Erel    tan(d)     Murel    tan(d)  subs
!   0    100.0000 1.000000  0.000000  1.000000  0.000000  70
!   1    25.00000 9.800000  0.000000  1.000000  0.000000  0
! S-Parameters, Magnitude/Angle, Matrix order, Normalized to 50 Ohms.
# GHZ S MA
! Analysis time per frequency (mm:ss) -- 0:00 user, 0:00 system, 0:00 real
10.0000000 0.515668 152.64 0.856789 -163.5 0.856789 -163.5 0.515668 60.400
```

An easy way to view the file is by using the UNIX “cat” command (concatenate). For UNIX systems, type:

cat steps.nd

If you only want the end of the file listed out, Type:

tail steps.nd

If you want to list, say, the last 24 lines of a file, type:

tail -24 steps.nd

On DOS systems, type:

type steps.nd

The first automatically generated comment in the file contains the command line that was typed to start the run and the time and date that the run started:

```
! em -v steps -- Wed May 12 10:45:02 1993
```

The next two lines specify the version of ***em*** used, the license identification and the last time the “steps.geo” file was updated:

```
! em 2.4, id sonnet.101, on 2009840242.
! Last time steps.geo was updated - Wed Dec 9 12:19:12 1992
```

The next three lines are user provided comments entered when ***xgeom*** saved the “steps.geo” file to disk. Note that user provided comments are automatically highlighted with “->” at the beginning of each line.

```
!-> Example of 1) two cascaded steps, or
!-> 2) a cross junction and two short stubs.
!-> With em, there is no need to decide which way to model it.
```

The next line (see below) informs us that the maximum subsection size provides at least 20 subsections per wavelength. Not only does ***em*** automatically subsection the circuit for you, the subsectioning algorithm divides the circuit into variable sized subsections. Small subsections are used where needed with larger subsections filling less critical areas. The upper limit on subsection size was set in ***xgeom*** and is repeated here for future reference.

```
! 20 subs/wavelength at 10.000000 GHz.
```

The length units in use (mils) is then displayed, followed by box dimensions. The A dimension (160 mils) is along the X-axis (horizontal in the ***xgeom*** display) and the B dimension (170 mils) is along the Y axis (vertical in the ***xgeom*** display). The A dimension is divided into 16 cells while the B dimension is divided into 17 cells as is also indicated on the line. The C dimension is the height of the box, the sum of the thicknesses of each dielectric layer in the box.

```
! All dimensions are in mils.
! A = 160.000000(16), B = 170.000000(17), C = 125.000000, with 2 layers.
```

Following the box dimensions is one line for each dielectric layer in the box. The parameters of each layer are listed in labeled columns. At the end of each line, the number of subsections generated by ***em*** is listed for the metallization located between that dielectric layer and the next lower layer.

```
! Layer  H( mils)     Erel    tan(d)     Murel    tan(d)  subs
!   0    100.0000 1.000000  0.000000  1.000000  0.000000  70
!   1    25.00000 9.800000  0.000000  1.000000  0.000000  0
```

Next comes a comment stating the kind of data stored (S-Parameters), a vendor specific data line, timing statistics (see the previous section) and finally the data itself. The data consists of frequency followed by the four S-parameters in magnitude, angle format:

```
! S-Parameters, Magnitude/Angle, Matrix order, Normalized to 50 Ohms.
# GHZ S MA
! Analysis time per frequency (mm:ss) -- 0:00 user, 0:00 system, 0:00 real
10.0000000 0.515668 152.64 0.856789 -163.5 0.856789 -163.5 0.515668 60.400
```

The default S-Parameter format is Touchstone compatible.

If any portion of the analysis requires one second or more, additional timing statistics for the first frequency appear just prior to the S-Parameters.

Since we did not invoke ***em***'s automatic de-embedding (which is covered in a subsequent chapter), the S-parameters are for the entire structure, including port discontinuities, not just the step discontinuity all by itself. Also, the reference planes are ignored. Most circuits require de-embedding for accurate results.

Before we get to the automatic de-embedding example, the next example shows how to get the same results as this example, only faster.

Chapter 5 Tricks to Get a Faster Analysis

♦ Symmetry Provides a Faster Analysis

The microstrip circuit in “steps.geo” (see Figure 4 on page 19) is symmetric about the horizontal center line. ***Em*** can take advantage of this, provided all ports are also on the center line. This second condition is important. If there is a port off the center line, ***em*** creates the port’s image on the other side of the center line and shorts the two together, a result usually not desired.

If you want to short two symmetrical ports together (see section "The Coplanar Short" on page 68), make sure you remove the port below the line of symmetry. Whenever you see an error message stating that there are no subsections for a given port, check for ports below the axis of symmetry.

To permit ***em*** to take advantage of the symmetry, run ***xgeom*** and modify “steps.geo” using *Param* ⇒ *Box* ⇒ *Symm.* When you click on the *Symm*. menu button, it toggles from *Off* to *On* and a dashed center line is drawn on the circuit. Visually check to make sure the circuit really is centered on the center line. If not, move the circuit until it is centered.

In electromagnetic terms, the symmetry option places a magnetic wall along the center line. In terms of performance, the number of subsections is reduced by nearly half, reducing the size of the matrix to one quarter of its original memory requirement, and increasing matrix solve speed by eight times.

When finished, save the file under the new name “steps_sy.geo”. You may want to add a comment or two indicating the modification as you store the file.

If you are using the “steps.geo” file, just use “steps_sy.geo” file instead (likewise for all subsequent examples).

Run the analysis with “em -v steps_sy”. The output to the screen should be the same as the previous section, with one possible exception. If a non-zero user time was displayed in the previous example, the time should be about half that in this example. Note that the number of subsections is cut by almost half. While not particularly important here, the improvement in speed is very important for large circuits.

♦ Invoking Single Precision for More Speed

Em normally works with double precision. In many cases, a single precision matrix reduction is all that is needed. To invoke a single precision matrix, run ***em*** with the memory saver “-m” option. Keep in mind that the matrix elements are still calculated in double precision because of potential numerical difficulties. Once calculated, they are then truncated and stored in single precision format.

By invoking single precision, you cut the memory required by a factor of two. This has a significant effect if your problem is large enough to cause ***em*** to swap out to disk. For big problems, this can result in an order of magnitude increase in speed.

On some machines the floating point speed is also increased because single precision calculations can be done faster than double precision. The Sun SPARCstation machines are such. Unfortunately, compilers on certain other computers perform single precision arithmetic by first converting to double precision, performing the operation and then converting back to single precision. If such a machine is not memory swapping in double precision, there is no increase in speed in single precision. It may even be slightly slower.

In general, use the memory saver option on large circuits (1000 subsections or more). It is not needed on small circuits. On occasion, the memory saver option can result in slightly different results. At extremely low frequencies, when the subsections size is on the order of 0.0001 wavelength or smaller, single precision may not be enough to allow the matrix to be properly inverted. In this case, very strange results are generated. For example, S_{21} may be different from S_{12}. In such a case, eliminate the -m option and re-analyze.

♦ Adjusting the Subsectioning

The biggest way to get more speed is to reduce the number of subsections. A reduction in number of subsections by half can result in one, or even two, orders of magnitude increase in speed. The most direct means of reducing subsection count is to reduce the amount of metal. Any metal which is unlikely to carry current should be religiously removed. This includes distant ground plane regions in coplanar waveguide and long, unneeded lengths of connecting transmission line.

A second way to reduce the number of subsections is to cut the circuit into pieces and analyze each piece separately. For example, instead of analyzing an entire bandpass filter, analyze half of it. Then, if both halves are identical, just use a circuit theory program to cascade the half filter analysis with itself. Most filters, and other structures as well, can be broken down into a number of small circuits which can be easily analyzed. In fact, a pure circuit theory analysis takes this "breaking down" to the limit.

A third way to reduce subsection count is to simply use a larger cell size. It can be surprising how little a larger cell size changes results on many circuits. Be careful not to reject viable options because of the length units in use. For example, if you want to analyze a line 7.0 microns wide. The obvious potential cell dimensions are 1 micron (for a seven cell wide line) and 7 microns (for a one cell wide line). There are other options too! For example, the cell size can be specified to be 7/3 = 2.33333 microns for a three cell wide line.

Also, don't worry too much about the n-th decimal place on dimensions. If a 7.1 micron wide line is called for, and you use a 7 micron cell size, the residual error introduced is probably less than the manufacturing tolerances.

However, there are circuits which have very fine geometries combined with very large geometries. The fine geometry determines the cell size and the large geometry unnecessarily uses a large number of cells.

Addressing this problem, ***em*** allows a minimum and maximum subsection size, in terms of cells, to be specified for each polygon. The parameters XMIN, YMIN, XMAX, and YMAX are used to do this. For example, in a large polygon, one could specify, say, the smallest subsection to be 10 cells in the x direction (XMIN=10) and 5 cells in the y direction (YMIN=5). For an arbitrary polygon, ***em*** still uses 1 cell wide subsections on the edge, but uses larger subsections on the interior of the polygon. The large polygon no longer uses such a large number of subsections and you can still use the small subsections in any region of fine geometry.

XMIN and YMIN can be extremely useful on polygons that have only right angle edges (called a a Manhattan polygon). Manhattan polygons are automatically detected and the new subsectioning is automatically used. This means that, for example, a spiral inductor with 9 micron gaps and 10 micron wide lines can be analyzed easily and quickly. Just set XMIN and YMIN to 10 cells (assuming a 1 micron cell size) for all polygons of the spiral and the subsections become exactly 10 microns (one subsection wide) with exactly 9 micron gaps.

If the maximum subsection size happens to be greater than the subsections per wavelength (lambda) parameter, the subsections per wavelength (*Param ⇒ Box ⇒ Cell Size*) takes priority. Increasing the subsections per wavelength parameter gives more accurate results, but uses more subsections.

To work best, the large polygon should have a rectangular geometry. Diagonal geometries can negate the beneficial effect of the large subsection size.

These subsectioning limits: XMIN, YMIN, XMAX, and YMAX; can be set using *Objects ⇒ Polygons ⇒ Param ⇒ [Select Polygon] ⇒ Subsections*. See section "Menu: Objects - Polygons" on page 61 of the ***Xgeom*** **User's Manual** for more details.

♦ Parallel Subsection Removal Option

A transmission line connected to a port is subsectioned with both x and y directed subsections included. However, in most cases, most of the current flows in the longitudinal direction and there is very little current flowing in the transverse direction.

For example, in a horizontal, x directed transmission line, most of the current is flowing in the x direction. There is very little transverse, y directed current flowing. Without parallel subsection removal, all the y directed subsections are still included, even though their current is almost zero. This wastes subsections, memory and time.

To remove the transverse subsections, in ***xgeom***, select "Param ⇒ Parallel Subs...". Then select a side and specify a distance for which parallel subsections are to be excluded. The transverse subsections are parallel to the side of the box.

To gain confidence, first do a simple circuit without parallel subsection removal. Then do it with parallel subsection removal. You should get the same answer. If not, either reduce the distance from the sidewall that parallel subsections are being removed, or do not use this option.

This option should be used carefully. Here are a few rules:

- Never remove subsections for lengths greater than 1/4 to 1/2 wavelength.
- Remove subsections only from transmission lines with very small transverse currents such as feed lines. For example, do not remove subsections from discontinuities or lines that change in width.
- Since this option affects all levels, check all other levels for possible problems.

Chapter 6 Automatically De-embedding the Discontinuity

The ***em*** analysis is very physical and complete. All the effects that are seen in an actual measurement (with the exception of thermal noise) are also seen in the ***em*** analysis. This includes things like box resonances, dispersion, loss, higher order transmission line modes and port discontinuities.

How ***em*** handles port discontinuities is dealt with in this section. When a physical device is measured, it is often first placed in a microstrip-to-coaxial fixture (see Figure 5). In order to get accurate measurement of the device, two things must be removed from the measured data:

- The microstrip-to-coaxial transition (connector) discontinuity.
- The length of interconnecting transmission line (reference plane shift).

The process of doing this is called "de-embedding".

In order to get accurate data on our device (frequently a discontinuity) from ***em***, we use very much the same kind of de-embedding. Instead of a physical connector, ***em*** uses what is called a "source" in electromagnetics. Any source, in any electromagnetic analysis, introduces its own discontinuity. The source discontinuity is analogous to the connector discontinuity of a physical measurement. It is something which corrupts the desired data and is to be removed.

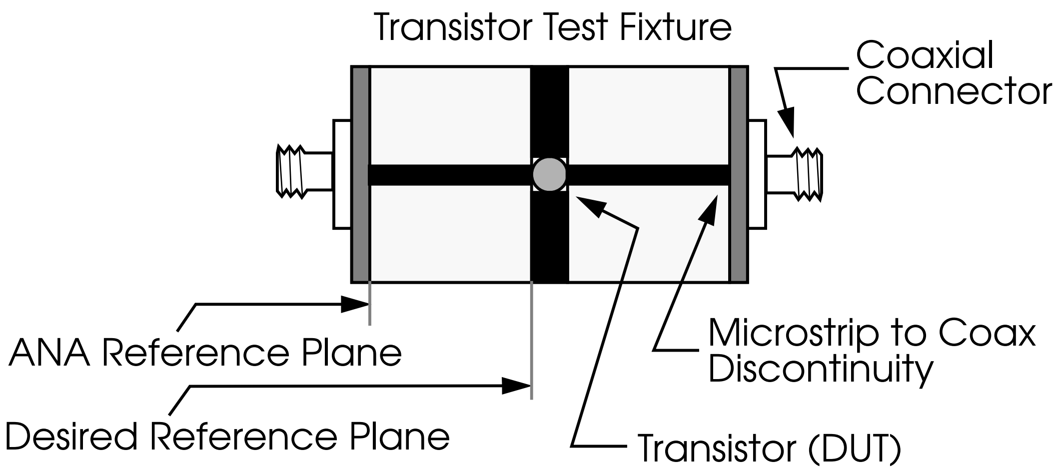


Figure 5 In order to get accurate data from a physical measurement, we must remove the port discontinuities and shift the reference planes to the face of the structure of interest.

To put the problem in microwave terms, we call the source discontinuity a "port discontinuity". In an actual physical measurement, the connector discontinuity must be removed. In the ***em*** analysis (and all other electromagnetic analyses) we must remove the port discontinuity. Once done, we may also, optionally, shift the reference plane right up to the face of the device (discontinuity).

Keep in mind that the reference plane shift is optional. If no reference plane is specified, a zero length reference plane is assumed. This means that the reference plane is placed at the port on the edge of the substrate.

The complete theory of the de-embedding algorithm used is presented in [8] (see the references at the end of the first chapter) and an abbreviated summary is presented in [7].

♦ A De-embedding Example

Em's entire de-embedding process is enabled by simply adding "-d" to the command line. The "steps_sy.geo" file has the reference planes already set to the face of the discontinuity, as indicated by the reference plane arrows (see Figure 4 on page 19). The reference planes are ignored until we invoke the automatic de-embedding.

To run ***em***, using both circuit symmetry and automatic de-embedding, type:

em -vd steps_sy

Note that the "-v -d" options have been combined to the equivalent "-vd". The resulting output to the computer screen is:

```
Circuit geometry file -- steps_sy.geo --
Analysis control file -- ctl.an --
De-embedded response file -- steps_sy.d --

               ELECTROMAGNETIC ANALYSIS OF 3-D PLANAR CIRCUITS
                                 Version 2.4

               (C) Copyright 1987 - 1993 Sonnet Software, Inc.
                            All rights reserved.

Starting subsectioning...done.
Circuit uses 41 subsections, 1 Mbyte.
Primary structure -- Analyzing 10.000000 GHz.
 Loading matrix, source level 0, field level 0.
 Reducing matrix.
De-embedding matrix, side 0.
First standard, 14 subs, 60.000000 mils -- Analyzing 10.000000 GHz.
 Loading matrix, source level 0, field level 0.
 Reducing matrix.
Second standard, 26 subsections -- Analyzing 10.000000 GHz.
 Loading matrix, source level 0, field level 0.
 Reducing matrix.
De-embedding matrix, side 1.
First standard, 5 subs, 60.000000 mils -- Analyzing 10.000000 GHz.
 Loading matrix, source level 0, field level 0.
 Reducing matrix.
Second standard, 9 subsections -- Analyzing 10.000000 GHz.
 Loading matrix, source level 0, field level 0.
 Reducing matrix.
 Analysis time per frequency (mm:ss) -- 0:00 user, 0:00 system, 0:00 real
10.0000000 0.513865 -141.5 0.857871 -53.90 0.857871 -53.90 0.513865 -146.3
! Eeff = ( 6.68), Z0 = ( 44.73).
! Eeff = ( 6.20), Z0 = ( 73.09).

em: finished.
```

The main difference between the output from this example and the previous output is the extra lines at the end. There are two ports in "steps_sy.geo", one on the left (side 0) and the other on the right (side 1). In order to de-embed the left side, ***em*** automatically creates two standards (analogous to ANA measurement standards) and analyzes each one. The same is done for the right side. The number of subsections is listed for each standard as well as the physical length for the first standard. The second standard is always double the length of the first standard (both standards are through lines).

♦ The De-embedded Result

When complete, both ports are de-embedded (port discontinuity removed and reference plane shifted) and the resulting discontinuity data is saved to disk. This time, the output file name is “steps_sy.d”, to indicate de-embedded data. An output file name ending in “.nd” indicates non-de-embedded data.

Em also calculates and saves to disk the effective dielectric constant (Eeff) and the TEM equivalent characteristic impedance of each port connecting transmission line, Zo (this is not the input impedance!). The first Eeff line is for port 1, the second for port 2, etc. This data is all listed in the last few lines of the circuit response file, steps_sy.d:

```
10.0000000 0.513865 -141.5 0.857871 -53.90 0.857871 -53.90 0.513865 -146.3
! Eeff = ( 6.68), Z0 = ( 44.73).
! Eeff = ( 6.20), Z0 = ( 73.09).
```

If loss is included, Eeff and Zo both become complex.

To see the degree to which the port discontinuity affects the S-Parameters, just compare the above de-embedded results with the non-de-embedded results obtained previously.

De-embedded data, such as the above, can be used directly in the popular microwave circuit simulators, just treat it as if it were measured data. If you need a different file format, see Appendix I, "The Em Command Line" describing the ***em*** command line.

It is also possible to get both the non-de-embedded data and de-embedded data in a single run. To do so, just specify both a “.d” and a “.nd” file in the command line. For example, “em -v steps_sy.geo steps_sy.d steps_sy.nd”.

De-embedding is where some microwave design background is useful in properly interpreting the results as described in the next chapter.

♦ Specifying Port Normalizing Impedances

Whenever results are to be incorporated in a circuit in a circuit theory based analysis program, the normalizing impedance for each port should be 50.0 ohms and the information in this section should be ignored. In rare cases, S-parameters normalized to some other impedance is desired.

For example, you may want to see what the reflection coefficient of a structure is when port 2 is connected to a 1.0 pF capacitor in parallel with a 10 ohm resistor (a power FET input model), while the input is being driven with a source which has a 35 ohm internal impedance. In this case, normalize port 1 to 35 ohms and port 2 to 10 ohms plus 1.0 pF.

In another example, pure electromagnetics is frequently carried out using S-parameters normalized to the characteristic impedance of each connecting transmission line. In the de-embedded steps, above, the researcher would want S-parameters normalized to 44.7 ohms on port 1 and 73.1 ohms on the output. These are often called “generalized” S-Parameters, in spite of the fact that the line impedance is usually not also specified, thus precluding precise conversion to 50 ohms for use in circuit theory software.

To understand the physical meaning of normalized S-parameters, recall that our standard 50 Ohm S-parameters are measured by terminating all ports in 50 Ohms and then measuring ratios of incident to reflected (or transmitted) wave amplitudes. If we were to use 60 Ohm terminations, instead of 50 Ohm terminations, the resulting measurements of traveling wave ratios would yield S-parameters normalized to 60 Ohms. If we were then to take the S-parameters which had been measured using 60 Ohm port terminations and use it in a circuit theory program (which expects you to use 50 Ohm terminations), we would get incorrect results.

However, 60 Ohm S-Parameters do have uses. Say you want to know what percentage of power is absorbed by a 60 Ohm load terminating port 2 of a two port circuit. That is simply one minus the magnitude squared of S_{11}, using 60 Ohm S-parameters. If you are interested in a different load, use a different normalizing impedance.

S-Parameters can be renormalized to any real impedance by using a circuit theory based program to cascade appropriate transformers on each port. This technique can not be used with a complex impedance, as is the case with a lossy line or complex load.

In ***em***, the default normalizing impedance is 50.0 ohms. If you would like a different normalization, refer to the section "Menu: Objects - Ports" on page 66 of the ***Xgeom* User’s Manual** on how to specify the normalizing impedance for each port.

The normalizing impedance is represented by four numbers. First is the real part in ohms. Next comes the imaginary part in ohms. Third is the inductive part in nanohenries (nH). The last number is the capacitive part in picofarads (pF). The inductive and capacitive part modify only the reactive portion of the load, they are included so you don't have to manually re-calculate the reactive part at each frequency.

The resistance, reactance, and inductance are all connected in series when the specific normalizing impedance is calculated at each frequency of analysis (see Figure 6). The capacitance is connected in parallel with the result and then the final normalizing impedance at the frequency of analysis is calculated.

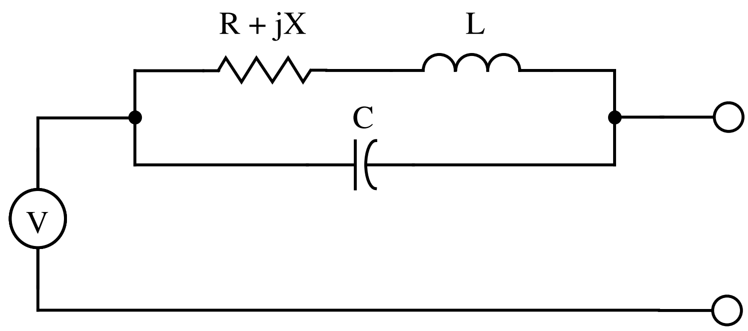


Figure 6 Equivalent Circuit of an ***em*** Port

☞ **NOTE**

The normalizing impedances are ignored if Z- or Y-parameters are specified for output.

This capability should be used only by the most advanced users. If the above discussion is not clear, seek assistance. Under no circumstances should this capability be used on data which is to be incorporated in a circuit in a standard circuit theory program. Such programs assume S-parameters normalized to exactly 50 Ohms.

For example, the 50 Ohm S-parameters of a microstrip step junction look like a very small shunt capacitance and a very small series inductance. In other words, the reflection coefficient is almost zero and the transmission coefficient is almost unity with a couple degrees of phase. If, instead of 50 Ohm S-parameters, we were to use S-parameters normalized to the characteristic impedances of the lines connecting to each of the two ports (this is common in electromagnetics), the S-parameters would look like a transformer between the two impedances. This causes the circuit theory programs, which are expecting 50 Ohm S-parameters, to give grossly incorrect results.

Chapter 7 Automatic De-embedding Limitations

Care should be taken when making use of de-embedded results. With some experience in microwave design, the similarities between potential difficulties in de-embedding and actual, physical microwave measurements are easily seen.

♦ Higher Order Transmission Line Modes

De-embedding removes the port discontinuity and the connecting length of transmission line. The de-embedding assumes that there is only one mode propagating on the connecting transmission line, usually the fundamental quasi-TEM mode. If higher order modes are propagating, the de-embedded results are not valid (the same is true for actual, physical measurements). If this is the case, we strongly recommend using a thinner substrate, unless, for some reason, multi-mode operation is desired.

Even when higher order microstrip modes are evanescent, there can still be problems. If the port is so close to the discontinuity of interest that their fringing (evanescent) fields interact, the de-embedding looses validity. Again, this is a problem which also arises in an actual physical measurement if the device to be de-embedded is close enough to the fixture connector. This problem is explored in a subsequent section.

Another related difficulty is that the resulting data, when it is used in a microwave circuit analysis program, should not be connected so close to another discontinuity that there is interaction (again, via evanescent fringing fields) between the discontinuities. Circuit analysis programs do not account for this interaction, invalidating the results. For a detailed description see [11] in the references at the end of the first chapter.

If there are two discontinuities close enough to interact, put both (all) of them in ***em*** where the interaction is completely included. In fact that is just what we did for the cascade of two steps.

♦ Box Resonances

A problem which affects actual measurements, as well as the ***em*** analysis, is box resonances. If present, resonances put glitches into the de-embedded data. To remove box resonances, just take the box lid off. How? A good approximation is to set the cover resistivity to 377 ohms per square, the impedance of free space. This solution, and its limitations, is demonstrated in a subsequent section.

Em also has the ability to detect and/or remove box resonances. Use the “-b” to detect resonances and “-B” to both detect and remove. All detected resonances are listed in the “.nd” file just before the S-parameters for the frequency. This is described in detail in section "Verifying the Box Resonance Problem" on page 44.

♦ Potential Reference Plane Difficulties

Em allows one reference plane per box side, no matter how many ports are on that side. All the ports on a side are de-embedded with a single pass through the de-embedding algorithm. Thus, no matter how many ports there are on a circuit, there are never more than four passes (for four sides) through the de-embedding algorithm. In the above example, there were ports on only two sides resulting in only two passes through the algorithm.

If there is more than one port per side, we have coupled lines and Zo and Eeff are not calculated. The de-embedded data is still accurate. The multiple ports may be as close to each other as desired, all interactions on any given side are taken into account.

Note that if the reference plane is close to zero electrical length (at very low frequency) or is close to a multiple of one half wavelength, the values of Zo and Eeff are not valid (and are usually not printed out). However the de-embedded data is still just fine. An example of this is provided later.

Keep in mind that the reference plane shift is optional. If no reference plane is specified, a zero length reference plane is assumed. This means that the reference plane is placed at the port on the edge of the substrate.

If the length of the reference plane is more than one wavelength, incorrect Eeff results might be seen. However, the S-parameters are completely valid. ***Em***'s calculation of Eeff is based on phase length. If the reference plane is, say, 365 degrees long, the Eeff calculation is based on a phase length of 5 degrees. But ***em*** has some "smarts" built in. If a non-physical result is seen, ***em*** increases the calculated phase length by 360 degrees at a time until physical (i.e., Eeff >= 1.0) results are seen. This usually corrects the problem.

Thus it takes a particularly long reference plane before the Eeff calculation fails. When it does fail, it suddenly jumps down to a value just above 1.0. Zo and the de-embedded data still have full validity. This failure mode is rarely seen.

♦ *Em*'s Intelligent De-embedding Algorithm

Em's de-embedding has even more "smarts" built in. It checks to see if two opposite sides of a circuit are identical (including reference plane length and the order of the port numbering). If so, it cancels the second (identical) pass through the de-embedding and simply uses the results from the first de-embedding. This can save a lot of time on structures like the gap or cross discontinuity.

♦ Why Box Sidewalls are Important

The box sidewalls can, on occasion, be frustrating, especially in regards to box resonances. This is true both with ***em*** and with actual circuits. However, the sidewalls also provide an invaluable service, they provide an unambiguous short circuit ground reference. In other words, the box sidewalls all form a perfect zero volts ground.

Microwave ports consist of two terminals. One terminal carries signal (e.g., the center conductor of a coaxial connector) and the other terminal is ground. When incorporating measured or calculated N-port S-parameter data into a circuit theory based analysis, all ports are assumed to have exactly the same ground reference.

Now let's say we incorporate loss in the sidewalls. When current flows in the sidewalls, that current generates a tangential electric field, a voltage. Thus the grounds at any two ports can have a voltage between them, violating the circuit theory assumption of all ground terminals being at the same potential. This effect is also known as a ground loop.

If we were to use an analysis that had no sidewalls at all, we would now have to integrate electric field through the substrate to calculate the circuit voltage. While such a voltage depends on the path taken to get to the ground plane, it can be done. In fact, if you use a via port in ***em***, this is just what is done. Now the problem that is introduced is a port discontinuity that includes a series inductance as well as a shunt capacitance. And in this case, as is pointed out in [7] and [8] (references are at the end of the first chapter), it is impossible to precisely de-embed internal S-parameters based on external current-voltage relationships.

Additionally, if ground loss is included in an analysis which has no sidewalls, a ground loop, as described above, can exist. This, again, destroys the assumption that all ports have the same ground potential. This means that such results can not be used accurately in a circuit theory analysis.

An additional benefit of sidewalls is that they form a perfect ground an infinitesimal distance from the signal line, there is no ambiguity as to the value of the port voltage.

Thus, the box sidewalls are frustrating on occasion, but they really are useful.

♦ Multiple Ports on the Same Side of The Box

When there is only one port on a box side, Zo and Eeff are calculated. When there are multiple ports on a box side, two dashes, "--" are output in the place of Zo and Eeff for each port on that side.

The reason for this is that there is not yet a standard in the microwave field for what impedances are to be used to define a multiple coupled line in general. We could use even and odd mode impedances. But these work only if the line is symmetric. Otherwise, electric and magnetic walls do not exist. We could use balanced and unbalanced impedances, but these are not common in microwave work.

When a standard emerges, and we can write a set of equations which take the set of N characteristic impedances and N effective dielectric constants and gives us the S-Parameters of an N-coupled line, then we can start generating Zo and Eeff for multiple coupled ports.

Chapter 8 De-embedding Situations

The following sections show what happens when unexpected results are encountered, how to recognize a problem, and what to do (if anything) when a problem is present.

Remember, ***em*** is a very physical analysis. If problems are encountered using ***em***, the same problems can be expected if the circuit is actually built.

♦ De-embedding Reference Plane Length -- How Close is Too Close?

As mentioned previously, if the port (which is, itself, a discontinuity) is too close to the discontinuity we are trying to de-embed, the resulting data is in error. This is because the fringing fields of the discontinuity interact with the fringing fields of the port. The de-embedding algorithm (which is virtually identical to algorithms used in de-embedding ANA measured data), being based on circuit theory, cannot handle fringing field interaction. See [11] in the references at the end of the first chapter for a detailed description of this problem.

As an experiment, we captured the circuit shown in Figure 7. The open circuited stub is 50 mils wide and 120 mils long. The box is 300 mils by 250 mils with a cell size of 10 mils square. The substrate is 25 mils thick with a dielectric constant of 9.8. You can capture this circuit using ***xgeom*** or copy the file “open_120.geo” using copyex.

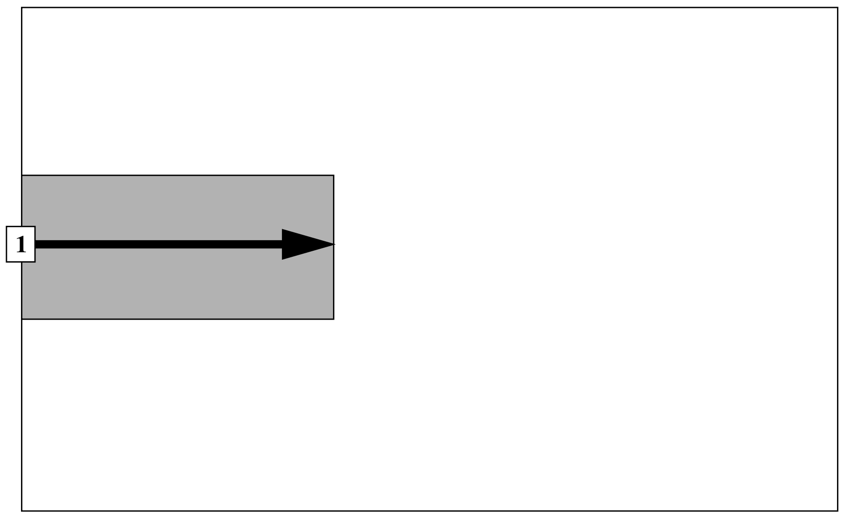


Figure 7 The open end discontinuity (“open_120.geo”) is used to check de-embedding interaction between the discontinuity and the port.

Both the line length and reference plane are 120 mils. We also shortened the line to 40 mils and to 20 mils and used *Objects ⇒ Ref. Planes* to move the reference plane. Those circuits were saved in files “open_40.geo” and “open_20.geo” respectively. In order to make the reference plane four cells long (the minimum length allowed), open_20.geo cell size was changed from 10 mils to 5 mils.

The result of each analysis (using the -d option) is:

```
120 mils between port and open end:
10.0000000 1.000000 -18.71
! Eeff = (  7.39), Z0 = (   33.63).

40 mils between port and open end:
10.0000000 1.000000 -19.96
! Eeff = (  7.26), Z0 = (   33.57).

20 mils between port and open end:
10.0000000 1.000000 -18.13
! Eeff = (  8.51), Z0 = (   31.40).
```

We see that, even with a reference plane on the order of the substrate thickness, there is only a moderate effect. At 40 mils, even that effect is almost gone with the Zo, Eeff and reflection phase very close to their values at 120 mils.

From this, we can suggest (this is not a rigorous conclusion!) that a reference plane equal to one substrate thickness or more is reasonable provided the transmission line is not over-moded. For reference planes shorter than one substrate thickness, the error introduced by fringing field interaction increases rapidly.

Zero length reference plane is a special case. Recall that de-embedding is a two step process. First, the port discontinuity is removed, then the reference plane shifted. A zero length reference plane is realized by simply skipping the second step and is entirely valid.

♦ Reference Plane Length at Multiples of a Half Wavelength

Eeff and Zo are not available when the length of the reference plane is an integral multiple of a half wavelength. For example, at an extremely low frequency the electrical length of the reference plane may be a fraction of a degree (i.e., zero half wavelengths). In this case, the analysis is unable to accurately evaluate the electrical length and, especially, the characteristic impedance.

If it gets bad enough, the analysis refuses to make the calculated values available. Occasionally, inaccurate values do get through and it is important to recognize the situation. It is also important to realize that only Eeff and Zo are affected. The de-embedded data is still available to full accuracy.

As an example, modify your “ctl.an” file to the following:

```
Ver 2.4
GHZ
FRE 16.0 19.0 0.5
```

Next run ***em*** with the command “em -vd open_120”, thus using the 120 mil reference plane set up in the previous example. This reference plane is precisely 180 degrees long at 17.5 GHz. The results are:

```
16.0000000 1.000000 -28.28
! Eeff = ( 7.90), Z0 = ( 38.25).
16.5000000 1.000000 -29.22
! Eeff = ( 7.94), Z0 = ( 40.93).
17.0000000 1.000000 -30.21
! Eeff = ( 7.98), Z0 = ( 51.40).
17.5000000 1.000000 -31.26
! Eeff = ( -- ), Z0 = ( -- ).
18.0000000 1.000000 -32.36
! Eeff = ( 7.98), Z0 = ( 26.93).
18.5000000 1.000000 -33.53
! Eeff = ( 8.01), Z0 = ( 30.85).
19.0000000 1.000000 -34.77
! Eeff = ( 8.03), Z0 = ( 32.56).
```

No results are printed out for Eeff and Zo at 17.5 GHz. Even so, we can see that the data (S11 phase) is completely reasonable at all frequencies, including 17.5 GHz. At frequencies around 17.5 GHz we can see that Eeff is affected a small amount, while Zo is obviously “walking” into and out of the (Zo) singularity.

The main point of this section is that if your results show a rapidly changing Zo, check the electrical length of the reference plane. If it is a multiple of a half wavelength, the de-embedded data is still correct. Disregard Zo.

♦ Box Resonances and De-embedding

Being based on circuit theory, ***em***’s de-embedding algorithm (like all de-embedding algorithms) is unable to de-embed a structure contained inside a resonant cavity (box). This section shows the effect of a box resonance and its resolution.

Modify the “ctl.an” file to the following:

```
VER 2.4
GHZ
FRE 24.0 24.6 0.1
```

Run ***em*** with the command “em -vd open_120.geo”, thus using the same “.geo” file as in the previous section. The results are:

```
24.0000000 1.000000 -54.87
! Eeff = ( 8.14), Z0 = ( 36.51).
24.1000000 1.000000 -56.83
! Eeff = ( 8.13), Z0 = ( 36.26).
24.2000000 1.000000 -61.56
! Eeff = ( 8.10), Z0 = ( 35.58).
24.3000000 1.000000 -172.2
! Eeff = ( 7.70), Z0 = ( 29.29).
24.4000000 1.000000 -43.23
! Eeff = ( 8.26), Z0 = ( 39.72).
24.5000000 1.000000 -49.19
! Eeff = ( 8.21), Z0 = ( 38.27).
24.6000000 1.000000 -51.42
! Eeff = ( 8.20), Z0 = ( 37.87).
```

Here we see the effects of a box resonance with the jump in phase (at 24.3 GHz) of the de-embedded open end. The phase of S11 suddenly sweeping through 360 degrees is a sure clue of a box resonance. Note that Eeff and Zo are also affected. This is because there is also a resonance in at least one of the standards that ***em*** creates for de-embedding.

In some cases, the standards do not have a box resonance and Zo and Eeff are unaffected. To see an example of this, reduce the top layer thickness from 1000 mils to 50 mils. The box resonance moves to 24.85 GHz while the Zo and Eeff calculations remain smooth.

A box resonance is not directly related to the length of the reference plane. Thus, it is easily distinguished from the problem described in the previous section where the reference plane is precisely a multiple of a half wavelength long. Notice that the box resonance is also a much more narrow band problem, at least for a lossless box. Compare the above results with those of the previous section.

♦ Verifying the Box Resonance Problem

When you see a “glitch” in the S-parameter data or in the characteristic impedance, you can verify whether or not it is a box mode. To do this, invoke the “-b” option to detect box resonances: “em -vbd open_120”. The output at 24.3 GHz is:

```
! Box resonances, primary structure.
! TE: 8(0,1,*) 1(1,1,*)
! Box resonances, first standard, side 0.
! TE: 1(1,1,*) 29(1,0,*) 1(0,3,*) 0(1,1,*)
! Box resonances, second standard, side 0.
! TE: 0(1,1,*) 19(1,0,*) 1(0,1,*) 0(0,1,*)
24.3000000 1.000000 -172.2
! Eeff = ( 7.70), Z0 = ( 29.29).
```

Notice that the resonance detection information is printed out before the frequency of analysis, not after. Also note that the S-parameters are the same, the resonance has not been removed. We see box resonances listed for the primary structure (the open end) and for both standards.

The box resonance information tells us that the resonance is Transverse Electric (TE), i.e., there is no Z directed electric field (perpendicular to the substrate surface). Several of the modes are (0,1,*) indicating that there is no variation (first digit) of field in the X direction (the direction of the open stub) and a half sine wave of variation (second digit) in the Y direction.

The “*” in the mode number indicates that the variation in the Z direction is unknown. Since we generally have several different dielectrics as we go along the Z axis, a mode number here is not always clear.

The digit in front of the parentheses is the mode significance. For example, the first mode in the primary structure has a significance of 8. This means that we are closer to exciting this mode than we are to the second mode, which has a significance of 1. The most important mode is the (1,0,*) mode in the first standard, with a significance of 29.

♦ Removing Box Resonances Artificially

To remove (as well as detect) box resonances, use the “-B” option. This option reduces the effect of the offending waveguide mode(s) in the calculation of coupling. There is a problem in doing this. We have now created a non-physical problem where all boundary conditions are no longer exactly met. The nature of this problem is such that if a mode is removed in one of the standards during analysis, the de-embedding becomes very confused and bad data can result. The bad data is usually obviously bad and, on top of that, ***em*** also prints out a warning message if both -B and -d are used on the same command line. You can see this by using -Bd on the “open_120.geo” file.

The -B option works reasonably well when not using de-embedding. Compare the output of “em -vb open_120” with the output from “em -vB open_120”. The data with the box mode is still several degrees off, however it is much improved from the case without box modes removed.

On some occasions, the box mode detection function “detects” a resonant mode which does not exist, this is called “false-ing”. In some cases, these are actually modes which could exist, but are simply not excited by the specific circuit you are analyzing.

♦ Removing Box Resonances Naturally

The box is a resonator because it is completely enclosed. One way to keep it from resonating is to just take off the top cover. We can use an approximation to this by setting the top cover resistivity to 377 ohms/square, the impedance of free space. This is an accurate approximation provided the cover is not so close that it interacts with the evanescent fringing fields surrounding the circuit.

If you wish to model a sheet of resistive material in the box, rather than removal of the top cover, setting the top cover to that resistivity is accurate no matter how close the cover is to the circuit. It is an approximation only when used to model the removal of the top cover.

Modify the file “open_120.geo” using ***xgeom*** to set the top cover resistivity to 377 ohms/square. This is the DC resistivity. Leaving the skin effect coefficient (which has a square root of frequency dependence) at 0.0, the DC resistivity is used at all frequencies. Use the ***xgeom*** option P*aram* ⇒ *Metallization* ⇒ *Param* to accomplish the modification and save the result in “openloss.geo”

Run *em* with the command line “em -vd openloss.geo”. The results are:

```
24.0000000 0.989660 -54.76
! Eeff = ( 8.14+j 0.00542), Z0 = ( 36.52+j 0.09512).
24.1000000 0.975578 -56.32
! Eeff = ( 8.13+j 0.01197), Z0 = ( 36.34+j 0.22884).
24.2000000 0.930526 -58.07
! Eeff = ( 8.13+j 0.03207), Z0 = ( 36.08+j 0.66651).
24.3000000 0.853834 -54.69
! Eeff = ( 8.16+j 0.06969), Z0 = ( 36.54+j 1.69237).
24.4000000 0.906228 -50.22
! Eeff = ( 8.20+j 0.04406), Z0 = ( 37.73+j 1.30906).
24.5000000 0.950918 -50.97
! Eeff = ( 8.20+j 0.02082), Z0 = ( 37.80+j 0.69223).
24.6000000 0.968893 -52.16
! Eeff = ( 8.19+j 0.01190), Z0 = ( 37.68+j 0.43988).
```

Most of the resonance has been removed. To remove the resonance more completely, the top cover can be moved closer to the substrate surface. In fact, if the cover is moved to within 50 mils, there is absolutely no indication of a resonance (now approaching 24.85 GHz) left. Another solution to the problem is to change the size of the box, larger or smaller, to move the resonant frequency.

With the top cover off, *em* is also including radiation loss in the de-embedded discontinuity (see the S11 magnitude). The loss has a second effect in making Eeff and Zo complex as well.

Taking the top cover off works, provided the sidewalls of the box are large enough to form a propagating waveguide up to the top cover, or you can place the top cover close enough to the substrate surface to catch the fields in the box mode. High order “box” modes tend to be confined primarily to the substrate and are difficult to remove in this manner. As you make the box bigger by increasing the substrate surface area, the modes “loosen up” so that they can propagate to the top cover and become absorbed.

To completely absorb any single waveguide mode, set the surface impedance of the top cover equal to the impedance of the waveguide mode. If you have multiple modes to absorb, setting the impedance to 377 ohms/square is a nice compromise, because some modes are above 377 ohms and others are below.

Chapter 9 Vias and 3-D Structures

Up to this point, most of the consideration has been with two-dimensional (2.5-D) structures, i. e., only X and Y current has been needed. However, ***em*** can also handle full 3-D current as well. The third (Z) dimension of current is handled by a special kind of subsection called a via.

♦ Adding Vias to the Circuit

Vias, as used in ***em***, are actually more general than the vias usually used in microwave design. In microwave design, vias connect metal on the substrate surface to the groundplane beneath the substrate, a ground via. In ***em*** vias can connect metallization between any substrate or dielectric layer, not just bottom layer to ground. We call these "level-to-level" vias, they can connect any two adjacent metallization levels.

Thus, ***em***'s vias can be used in air bridges, spiral inductors, wire bonds, and probes as well as the standard ground via. ***Em***'s vias use a uniform distribution of current along their length and thus are not intended to be used to model resonant length vertical structures. Keep the via lengths small with respect to a wavelength.

To create vias, use ***xgeom*** to specify one or more polygon edges as edge-vias (see Chapter 12, "Using Vias" in the ***Xgeom*** **User's Manual** for details). ***Em*** places subsectional vias (called "via-posts") along the entire length of all edge-vias. Vias always go up from the selected polygon edge with the length of the via equal to exactly the thickness of the dielectric layer. The via-posts are rectangular cylinders with a horizontal cross-sectional area equal to one cell. If you make the cell size smaller, the vias become smaller with more of them along the edge-via. Of course, the length of the edge-via is unchanged. Current in a subsectional via is uniform through out the body of the via and is Z directed. Via loss is determined by the polygon from which the via originates.

♦ Internal Ports and Via Ports

When using Sonnet, ports are usually placed on the edge of the substrate next to the sidewall. This allows the very precise de-embedding algorithm to take advantage of the precise ground reference formed by the short circuiting sidewall.

Internal ports, ports not on the edge of the substrate, may also be specified in Sonnet. Such a port is formed by attaching a via to ground and then specifying that via as a port. Keep in mind that such a port adds a small amount of series inductance and shunt capacitance due to the length of the via and the fringing fields surrounding it. In some cases, the small discontinuity is of no concern. In other cases, like a patch antenna, it is desirable because that is what you actually have.

For example, in an amplifier design, one may wish to have three or four ports on the interior of a circuit in which to insert the measured S-parameters of an FET. To do this, put ground edge-vias at the points where you wish to place the FET terminals. Then specify the bottom (ground) edge-via as a port. For the most accurate result, you may want to characterize the via port separately in order to de-embed it manually.

The example file "patch.geo" has an example of a via-port used in a patch antenna. The example file "intports.geo" shows conceptually how via ports are used in an amplifier. The designer would analyze the four port in intports.geo and then use the usual circuit theory programs to attach a transistor between the internal ports, ports 3 and 4 in intports.geo. Both example files can be obtained with the copyex command.

Another kind of internal port, which does not have any attachment to ground, can also be used. This kind of port is placed between two adjacent polygons. If there is only one port in the entire circuit, there is generally no problem. If there are two or more ports, the resulting S-parameters should not be used. This is because each port may have a different ground reference. Circuit theory based programs assume all the ports have the same ground reference. Use of such data in a circuit theory program is likely to give bad results.

On the other hand, if S-parameters are not the primary result, multiple internal ports with different ground references can be useful. For example, in calculating the radiation pattern of a phased array of dipoles, this approach works well. The example file "twoarray.geo" illustrates this use. A simple phased array of two dipoles, it is described in detail in the **Patvu User's Manual**.

♦ Restrictions on Vias

With the present version of ***em***, automatic de-embedding is not available for via ports so do not use the -d option if you have specified any vias to be ports. There is a small series inductance and shunt capacitance associated with a via port. The port discontinuity is small compared to a typical FET. However, if you are using the port via to include, say, a small measured capacitor (say, 0.1 pF), the port discontinuity may be of comparable magnitude, be careful!

The via ports can also be used to model coaxial feed for microstrip antennas. In this case the port discontinuity is desired, as it approximates what is actually seen.

Another factor to keep in mind is that the height of the via should be a small fraction of a wavelength. The via height is the same as the thickness of the substrate or dielectric layer it penetrates. So, for ground vias, this is usually no problem. If a microstrip substrate were a significant fraction of a wavelength thick, over-moding also becomes a major problem. If one is using vias to form, say a septum, or an interior wall, keep an eye on the potential problem. A via which happens to be a half wavelength long does not give reasonable results.

♦ Simple Via Example

A simple via is stored in the file "via.geo" and is shown in Figure 8. This file was generated by adding a via to ground at the end of the stub in "open_120.geo", converting an open circuited stub to a short circuited stub. The via subsections are indicated by small triangles pointing down, indicating metal going into the screen. Triangles pointing up indicate metal coming out of the screen. The file is also available by typing, "copyex via.geo".

The vias were captured by placing a polygon on the ground level indicating the periphery of the ground via. A square was used to model a via that is actually round. Such a model gives surprisingly accurate results. If an even more accurate model is needed, an octagon can be used.

After the polygon is in place at the base of the via, each side of the base polygon is converted into vias going up. See Chapter 12, "Using Vias" in the ***Xgeom*** **User's Manual** for details

Note that the top end of the via (see Figure 8) is a “hat” which is larger than the via itself. There are no restrictions on the polygons at the top of a via. ***Em***’s subsectioning algorithm handles the subsectioning accurately.

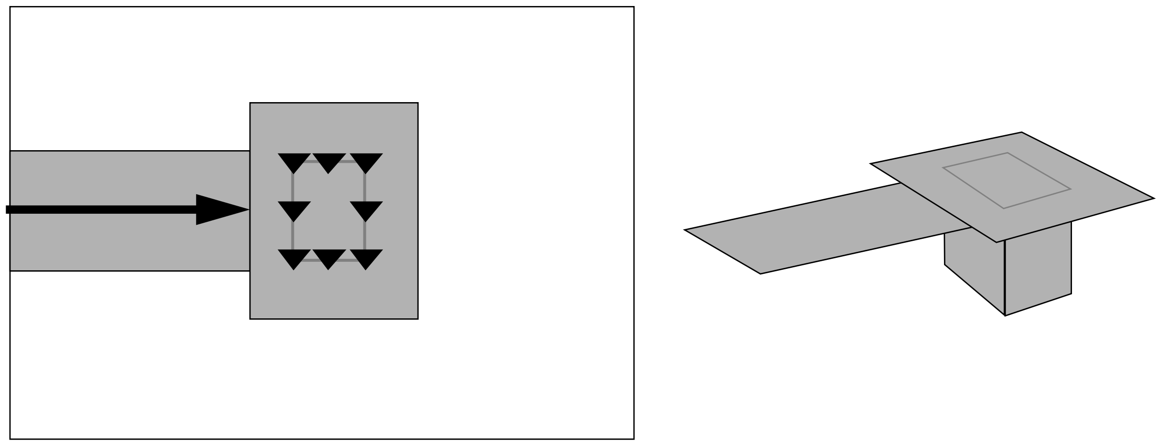


Figure 8 A simple via to ground. On the left, as it would appear in ***xgeom*** (top view). On the right, a view in perspective.

An analysis of the above via at 10 GHz with de-embedding gives a phase of 148.33 degrees, a perfect short circuit is 180 degrees. This means the via has an inductive phase of about 32 degrees.

♦ A Conical Via

One may simulate a conical ground via with a staircase approximation. Simply divide, say, a 100 µM GaAs substrate into four 25 µM substrates. Then put polygons (and specify edge-vias) at appropriate places to form the step approximation to the via sides. For an example, see the file “cvia.geo” in the examples directory. This circuit is a conical via to ground placed in the center of a through line, the purpose being to measure the via inductance.

To look at the “cvia.geo” file type:

copyex cvia.geo

xgeom cvia

The “cvia.geo” file is a very detailed model of a conical via. If you are modeling a large circuit (say, an inter-stage matching network) with multiple vias, you may want to use a simpler model for faster analysis.

Another, more complex circuit (see Figure 9), which uses a simple via, shows a 20 µM

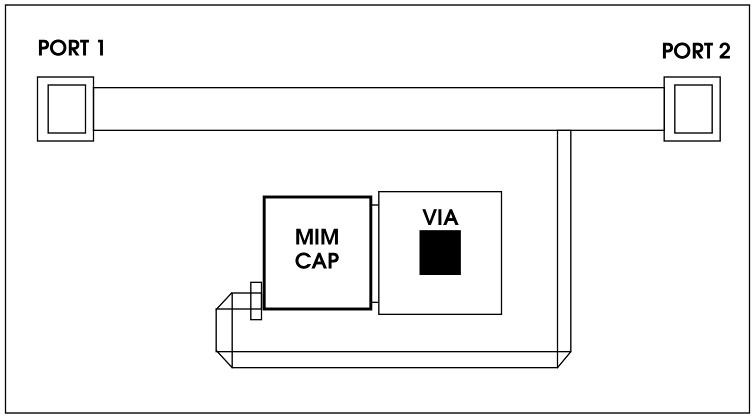


Figure 9 The coupling between the 20 µM wide stub and the via/capacitor combination requires an electromagnetic analysis.

wide stub shunted across a 50 ohm line on 100 µM GaAs (courtesy Raytheon Research Division). The stub is so close to the via/capacitor combination that there is significant coupling, causing the error seen in the circuit theory calculation (see Figure 10).

The ***em*** analysis includes the coupling and would have eliminated a re-design and re-fabrication cycle. The circuit theory analysis requires only a fraction of a second per frequency while ***em*** requires 13 Seconds per frequency (HP-710). But, in this case, the circuit theory analysis, because it gives the wrong answer, is a very expensive way to do the analysis!

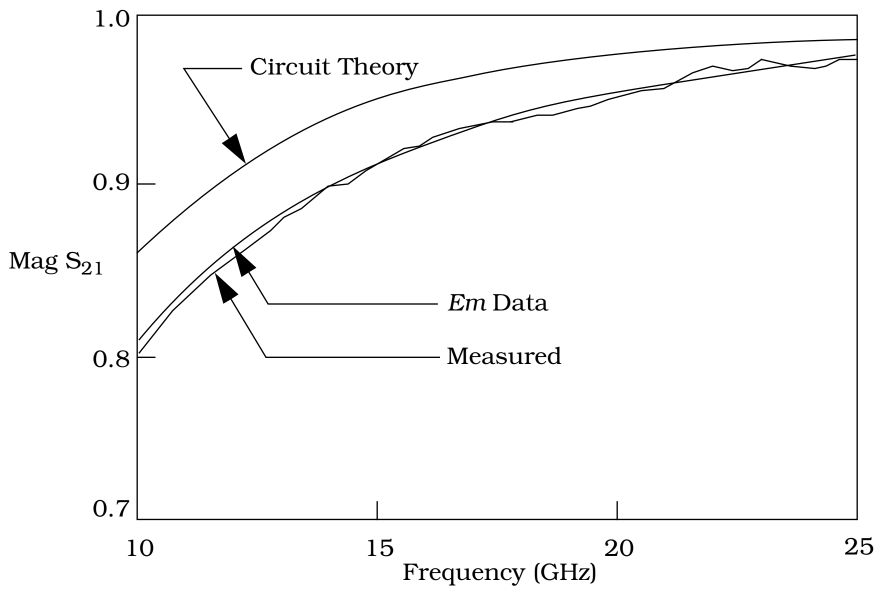


Figure 10 The ***em*** calculated data, which required a minute per frequency to analyze, accurately predicts the measured data, which required fabrication of a GaAs MMIC wafer.

If the circuit had been spread out (the stub straightened out and kept at least 100 μM from any other components), circuit theory would probably have provided an accurate analysis, and electromagnetic analysis would be unneeded. This illustrates the point that we should use a combination of circuit theory and electromagnetic analysis, either one alone compromises the quality of resulting designs.

This circuit can be obtained by typing “copyex raystub.geo”. This particular circuit, because of the short vias used for connecting to the capacitor top plate, requires quad precision (“-q” option) below 2 GHz, otherwise a residual error of several percent starts to accumulate.

Chapter 10 Metallization Loss and Super Conductor Surface Reactance

Metallization loss is specified in ***xgeom*** by using the *Param ⇒ Metallization* option, see section "Menu: Param - Metallization" on page 85 in the ***Xgeom*** **User's Manual** for details. Losses may be assigned to circuit metal, top cover, and ground plane.

A common misconception is that only one type of metallization is allowed on any given level. In fact, different metallizations (i.e., different losses) can be mixed together on any and all levels. It is possible to have, say, a thin film resistor right next to a superconductor all on the same level.

♦ Surface Resistance

The metallization loss is based on surface resistance. Two numbers are required. The R_{DC} determines loss at low frequency (where the conductor is much thinner than the skin depth). Surprisingly, electromagnetic analyses often predict zero loss at low frequency because they assume R_{DC} is zero. R_{DC} is useful for thin film resistors as well as low frequency metal loss.

The second required number is the skin effect coefficient, R_{RF}. ***Em*** multiplies this number by the square root of the frequency (in Hertz) to yield the ohms/square at high frequency.

Em also properly models the transition between electrically thin (low frequency) and electrically thick (high frequency) conductors. See the section "Menu: Param - Metallization" on page 85 in the ***Xgeom*** **User's Manual** for examples. The equations for R_{DC} and R_{RF} are repeated here (σ is the bulk conductivity in mhos/meter, t is the metallization thickness, and $\mu = 4\pi \times 10^{-7}$):

$$R_{DC} = \frac{1}{\sigma t}$$

$$R_{RF} = \text{skin effect coefficient} \approx \sqrt{\pi \frac{\mu}{\sigma}}$$

Typical values for R_{DC} and R_{RF} are 0.004 and 3e-7. If you start getting very strange results, check R_{RF} especially to make sure it didn't slip an exponent!

The transition frequency, from R_{DC} to R_{RF} is the square root of R_{DC}/R_{RF}. At this frequency, and a relatively narrow band around it, both coefficients are important.

If an extremely precise evaluation of loss is needed, we recommend building (or finding) and measuring a simple structure of the desired metallization. Then adjust the values of R_{DC} (at low frequency) and R_{RF} (at high frequency) until the calculated loss matches the measured loss. Then you can use these values for any circuit which uses the same metallization. You are now effectively using measured values for R_{DC} and R_{RF}. The typical values given above work well as a first approximation.

The alternative to measurement of R_{RF} is to develop a physical model. Such a model would have parameters such as bulk conductivity, bottom side surface roughness, top side surface roughness, metal porosity, and edge roughness as at least a few of its parameters. Developing such a model accurately is expected to be very difficult and is beyond the scope of this analysis software. Even if developed, such a model only converts the problem of measuring R_{RF} to the problem of measuring the physical parameters needed for the model. Measuring the required physical parameters (like metal porosity) is likely to be even more difficult than just measuring R_{RF} in the first place.

Em's modeling of loss is very accurate, provided accurate values of R_{DC} and R_{RF} are used. Expect the final result to be just as accurate as the values of R_{DC} and R_{RF} specified.

In another aspect of loss, it is well known in electromagnetics that the surface impedance of a good conductor has an imaginary part which is equal to the real part. This reactive surface impedance is physically due to the increased surface inductance caused by the current being confined closer and closer to the surface of the conductor. This surface reactance is included in R_{RF}. The effect is small but potentially significant in certain cases.

Some electromagnetic analyses use a "perturbational" approach for loss. This means that they assume the current flowing everywhere is the same as the lossless case. This approximation works for low loss (good conductor). However for thin film resistors (high loss), the lossless (short circuit) current is not the same as the lossy current and a perturbational approach fails.

Em's loss analysis is not perturbational. It works just as well for 100 ohms/square resistor material as it does for 0.004 ohms/square good conductor.

♦ Surface Reactance

Until recently, the only surface resistivities of practical interest were pure real, i. e., pure loss. With the growing application of super-conductors in high frequency work, a reactive (inductive) surface impedance may see actual application. A super conductive effect known as "kinetic inductance" slows the velocity of the electrons with no loss of energy. This can be modeled as a surface inductance.

The effect of the surface inductance is to make ε_{eff} larger, i. e., the velocity of propagation slower. For normal conductors, ε_{eff} can never be larger than ε_{rel}. In a super conductor, this is no longer true. This unusual effect becomes significant for very thin substrates. As an example (data provided by Miter Corporation), take Niobium at 4^o Kelvin, line width of 80 µM, substrate thickness 0.7 µM (!), $\varepsilon_{rel} = 8.0$ has a surface impedance of 2.65e-5 + j 0.00678 ohms/square at 10 GHz. Z_0 (as determined by ***em*** using the -d option on a through line) is 1.27 ohms and $\varepsilon_{eff} = 9.65$ with very low loss. What can a creative microwave designer do with such a transmission line? Only time will tell.

Surface reactance is included in ***em*** by using the *Param* ⇒ *Metallization* option and specifying the surface reactance in ohms/square. ***Em*** uses the same reactance at all frequencies. As more is understood about kinetic inductance, a more sophisticated model will be used.

As an added side light, the surface resistance can be used to include lumped resistors (just make a square patch and set the ohms/square to the desired resistance). In addition, the surface reactance can be used to make a lumped reactance. Since the reactance of an inductor and capacitor both change with frequency, one must manually change the reactance as the frequency changes, but at least lumped reactive components can be used.

The surface reactance effect can also be seen in normal conductors.

♦ Viewing Tangential Electric Fields

One reason ***em*** is so fast is that all of the electric and magnetic fields were solved for analytically, with pencil, paper, and many equations. The computer need only do an FFT and solve for the current distribution.

However, on some occasions, we want to view the fields. The charge distribution can be viewed in ***emvu***; just select the appropriate display option. The charge distribution is proportional to the normal electric field discontinuity at the dielectric interface.

There is also a way to view the tangential electric fields as well. We do this with what is called a “sense layer”. The sense layer is a rectangular patch of conductor placed where we want to see the tangential electric field.

Actually, describing the sense layer as a conductor is a little misleading. This is because we set the surface resistance of the conductor to some large value, say 1,000,000 Ohms per square, or even more. We set the resistance to such a large value so that the sense layer has little influence on the original fields. An intuitive analogy is to view a sense layer like inserting a sheet of paper (very high resistance) into the fields. The sheet of paper does not change the fields. When capturing the sense layer, it is best to set YMIN and YMAX to 1 for the best image, see the section "Adjusting the Subsectioning" on page 27.

Because the resistance of the sense layer is so high, the currents are very small. But even though they are small, what are the currents? The current density is proportional to the tangential electric fields over the area of the sense layer. This is just a two dimensional version of Ohms Law: Current is proportional to Voltage.

An example is shown in Figure 11 as viewed by ***emvu***. Type “copyex tane.geo” to get

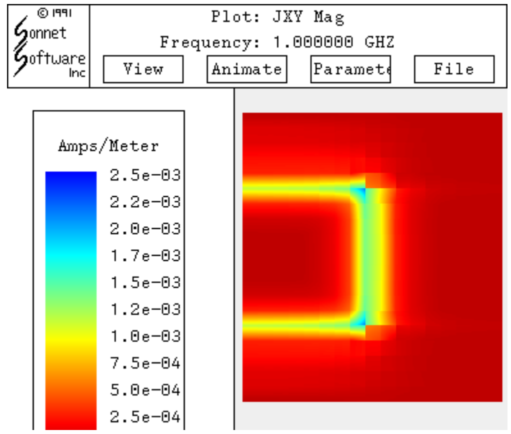


Figure 11 The tangential electric fields just above a gap discontinuity. The input voltage comes from the left. Strong fields are present across the gap, especially at the corners. This analysis was at 1 GHz.

the geometry file. The tane.geo file is based on the gap20.geo file.

Chapter 11 Corner Fill and Triangular Subsections

Corner fill places a special kind of subsection in the vicinity of all small gap discontinuities. These subsections accurately model the corner singularity in the current distribution and allow more accurate results to be generated. Using corner fill increases the number of subsections used. If there are no gap discontinuities, corner fill has no effect.

Circuits with diagonal edges may benefit from including triangular subsections. The edges of such structures are frequently much better approximated with the diagonal edges allowed by triangular subsections. Triangle fill must be selected manually. Include it only on polygons with diagonal edges that also carry significant current. See [9] and [21] in the references at the end of the first chapter for a detailed description of triangle subsections.

Both of these options involve additional computation time. However, depending on the circuit, using these options may allow a given level of precision to be achieved more quickly. The alternative to using corner fill and triangular subsections is to increase the resolution (smaller cell size).

An example of each option is given next.

♦ The Effect of Corner Fill on a Gap

There are two example circuits to show the effect of corner fill, “gap20.geo” and “gap20_nc.geo”. You can get copies by using copyex. The first circuit uses corner fill. The second circuit is identical to the first except it does not use corner fill.

Set the “ctl.an” file to 1.2 GHz and analyze the first circuit by typing:

em -vd gap20

The resulting S_{21} magnitude is 0.063 or -24.0 dB. When the above command line is used with the file “gap20_nc.geo”, the result is 0.051 or -25.9 dB. for a total of 1.9 dB difference. The use of corner fill increases the number of subsections but also increases the accuracy of the analysis.

Beginning with release 2.2 of ***em***, corner subsections are included only when there is a gap of two cells or less. For larger gaps, the normal subsectioning is sufficient to model the corner singularity.

♦ The Triangle Fill Option Applied to a Coupled Open-Miter

To demonstrate the triangle subsections, run ***xgeom***, again, on “open_120.geo”. This time, we shall analyze a right angle mitered bend which is closely interacting with an open end. Attempts to model this discontinuity using a circuit theory based program do not include the fringing field interaction between the two discontinuities.

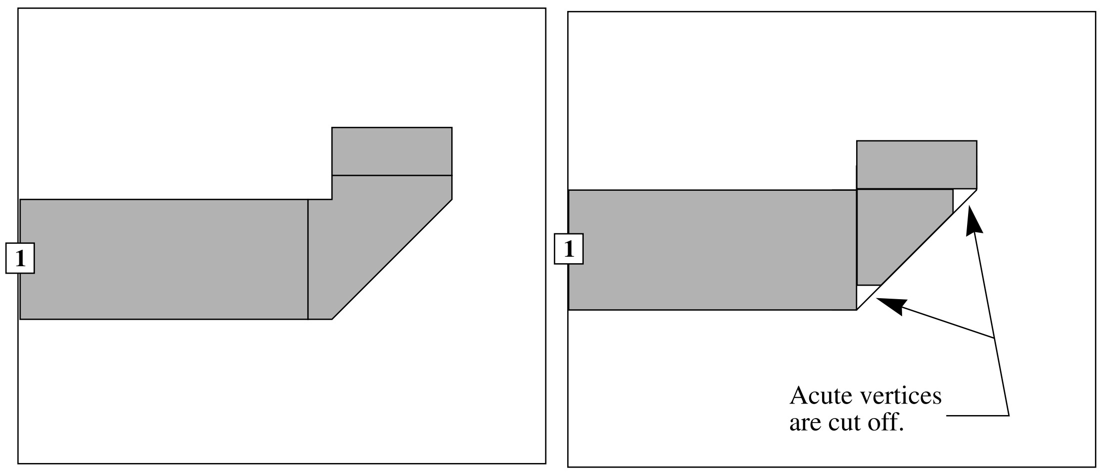


Figure 12 On the left, closely interacting mitered bend and open end discontinuity. On the right, if the mitered region is captured as a simple triangulate (no tabs), the vertices with acute angles are cut off.

The circuit (see Figure 12) is a multi-polygon circuit. The transmission line connected to port 1 is the original stub from “open_120.geo”. It is unchanged and forms the first polygon. The mitered region is the second polygon with the new open end forming the third polygon. Polygons with edges in common are electrically connected together along those edges.

To capture the circuit, use *Objects ⇒ Polygons ⇒ Add.* After the circuit is drawn, select *Objects ⇒ Polygons ⇒ Param*, click the left mouse button inside the miter polygon, and change the miter polygon fill mode to “Diagonal”. If the circuit is captured as a single polygon, change that polygon to Diagonal fill.

Notice that there are “tabs” where the miter polygon attaches to the transmission line. Without these tabs, the miter polygon becomes a simple triangle. However, due to the nature of triangle subsections, acute angles (less than 90 degrees) cannot be modeled.

Em automatically cuts acute angles off. This can be seen by capturing a triangle polygon with ***xgeom*** and setting it to triangle fill. In the case of our example stub, small notches are formed where the triangle meets the lines, as illustrated on the right in Figure 12. Thus, the tabs on the miter polygon.

Be sure to click on *Param* ⇒ *Box* ⇒ *Symm*. to turn off symmetry. With the miter included, the circuit no longer has symmetry. Save the file under “openmite.geo”. Alternatively, use copyex to get a copy of the file “openmite.geo”.

With “ctl.an” set to 10 GHz, use the command line “em -vd openmite”. The result is:

```
10.0000000 1.000000 -150.6
! Eeff = ( 7.39), Z0 = ( 33.64).
```

This analysis required more time. The analysis time is sensitive to whether or not triangle subsections are included. Once included, however, the ***em*** analysis time is relatively insensitive to the number of triangle subsections, provided the total number of triangle subsections is small compared to the overall total of all subsections.

Chapter 12 A Multi-Port Discontinuity for Microstrip Coupled Line Filters

Em can handle circuits (with complete de-embedding) of any number of ports, without limit. In this section we look at the junction between two sections in the classic microstrip bandpass filter.

In a typical microwave design, this junction (see Figure 13) is usually accounted for by adding a microstrip open end model at the end of each resonator. This model often takes the form of a simple length extension. In reality, the open end discontinuity is modified by the close proximity of the adjacent resonator. In addition, the fringing fields also generate coupling between the open end and the adjacent resonator. There is even coupling between the open end of the port 1 resonator and the open end of the port 4 (non-adjacent) resonator!

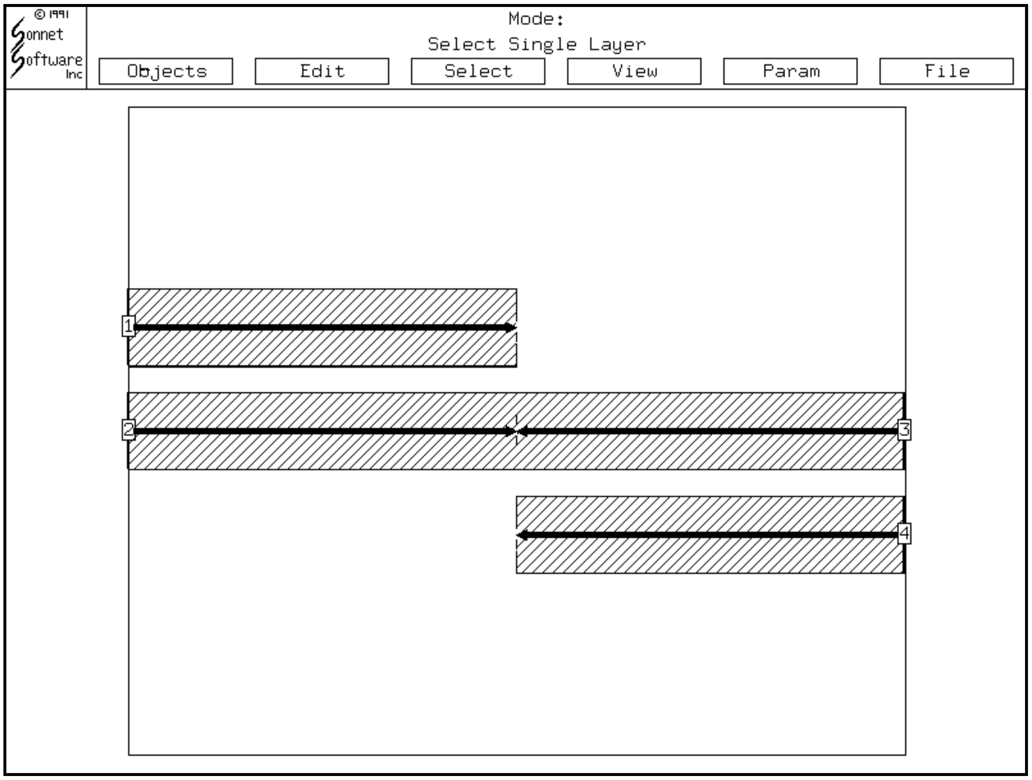


Figure 13 A four-port discontinuity is formed at the point where two sections of a coupled line filter join.

All of these couplings are precisely evaluated by ***em***. To demonstrate, capture the circuit shown in Figure 13, or use copyex to get "coup_end.geo". This circuit uses the same box and dielectrics as "open_120.geo", just load the file and use *File* ⇒ *Clear* to remove the existing circuit. Don't forget to also click on *Param* ⇒ *Box* ⇒ *Symm.* to turn off symmetry. This circuit is not symmetric.

Save the file as "coup_end.geo". Run the analysis with "em -vd coup_end.geo" for the results (listed in matrix order using magnitude, angle format):

```
10.0000000 0.998678 -16.18 0.036174 90.731 0.036346 71.343 0.003616 -126.3
           0.036174 90.731 0.010972 -97.47 0.998624 -0.967 0.036346 71.343
           0.036346 71.343 0.998624 -0.967 0.010972 -97.47 0.036174 90.731
           0.003616 -126.3 0.036346 71.343 0.036174 90.731 0.998678 -16.18
```

As described in the section "Multiple Ports on the Same Side of The Box" on page 38, Eeff and Zo are not calculated for coupled port lines.

♦ Using *Em*'s Results in a Filter Design

As an example of the use of ***em*** in a filter design, one would do the first cut of the microstrip filter with conventional circuit theory based programs. With the resulting approximate final dimensions, calculate the above discontinuity. You may want to include radiation loss (Chapter 16, "Antennas and Radiation") and metallization loss (Chapter 10, "Metallization Loss and Super Conductor Surface Reactance"). Dielectric loss can even be included in the discontinuity.

Once calculated, incorporate the discontinuity (as though it were measured data) between ideal circuit theory coupled lines in circuit theory based software. Now, the lengths of the coupled lines can be adjusted to account for the effects of the junction discontinuity. The resonator couplings (gap widths) can even be modified slightly since the junction discontinuity is insensitive to the gap width.

In contrast with an entire filter structure, the coupled line discontinuity S-parameters vary slowly with frequency. Thus, an analysis of the discontinuity at three or four frequencies is all that is needed. A circuit theory program may then be use to interpolate and assemble the entire, frequency sensitive, filter.

If desired, even the coupled lines can be done using ***em*** (and the automatic de-embedding). A small section of line can be calculated with no loss of precision relative to a large section. Just analyze a section of line longer than one or two substrate thicknesses and set the reference planes so that, say, the S-Parameters for a one cell length of line result. This short length of line may be cascaded as needed to build any desired length of line in most any circuit theory based program.

The obvious approach to the problem is to analyze the entire filter with ***em***. Given sufficient computing power (a very small cell size is needed to accurately position the edges of a typical filter), this is a viable alternative. You may want to select several frequency points to validate the design, say the 3 dB points and one or two other points, and restrict the analysis to just those points. This approach takes into account coupling between all resonators and discontinuities in the entire filter.

Yet another approach is to take a symmetric filter and split it in half (midway between the input and output). Analyze the half filter (typically a 3-port) and then use a circuit theory program to cascade it with itself to form the complete filter.

Chapter 13 Co-Planar Waveguide Discontinuities and Balanced Ports

Em also handles coplanar waveguide well because ground planes and dielectric layer thicknesses can be made any value while still maintaining full accuracy and without compromise in speed. This section describes a very simple, yet important, coplanar waveguide discontinuity, the short circuit. A second example is the coplanar cross junction. The cross junction also illustrates the use of level-to-level vias to form air bridges. Air bridges are needed to suppress the slot line mode as well as provide a ground current return path in some cases.

♦ The Coplanar Short

The coplanar short (see Figure 14) finds wide use as a measurement calibration standard.

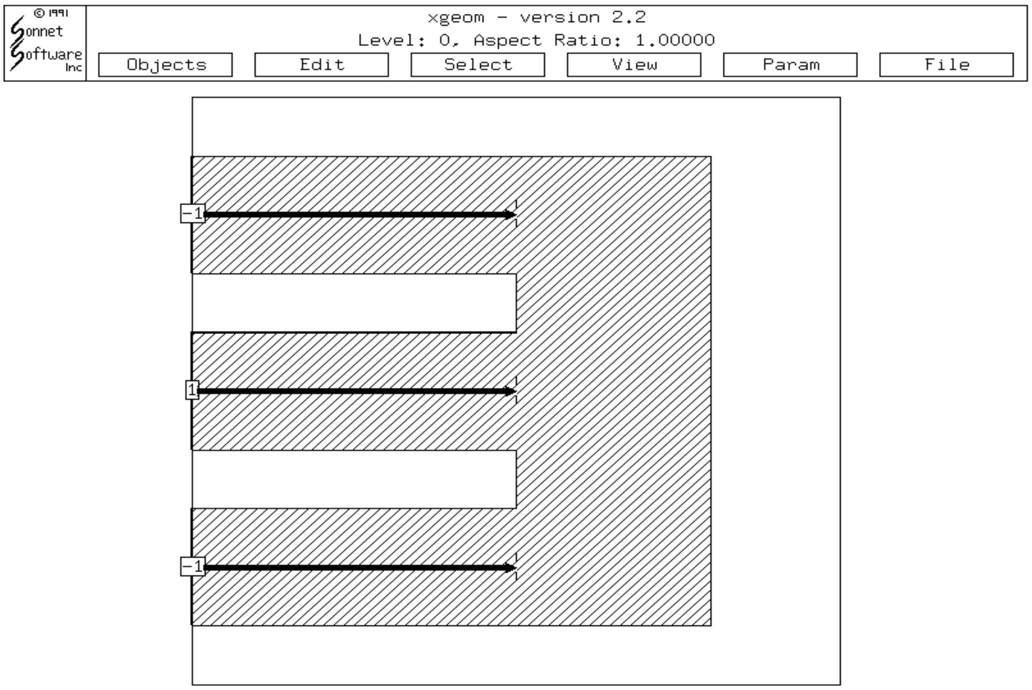


Figure 14 A co-planar waveguide short circuit discontinuity is easily handled by ***em***.

Notice that both ground ports have been labeled "-1", while the signal port is labeled "1". This represents a balanced port. Any number of ports can have the same positive or negative labels (this is done in ***xgeom*** using the Objects ⇒Ports ⇒Renumber option). ***Em*** sums the total current going into all the positive ports with the same port number and sets that equal to the total current going out of all the ports with that same negative port number. Thus the name "balanced", or "push-pull" port.

This circuit is symmetric, however, there is a special consideration in invoking symmetry. This issue is addressed at the end of this section.

Slot line can be represented by removing one of the "-1" ports, above. Be careful, however, the sign of the slot line port can be reversed by swapping the remaining "-1" port with the "1" port. It is possible, in fact, to get an S21 phase of 180 degrees for a zero length line. This happens if the ports on either end of a slot line have opposite sign.

Note that we end the width of the ground lines just before they would have touched the sidewall. If the side of the ground line touches the side wall, it shorts out to the sidewall, thus allowing ground current to return via the sidewall instead of via the ground line. This defeats the purpose of the balanced port. Be sure that your ground lines touch the sidewall only at the location of negative port numbers when using balanced ports.

For large coplanar (or slot line) structures, there may be significant current on the outside edges of the ground strip. This can be verified with ***emvu***. If this current is undesired, it can be eliminated by connecting the outside edge of the ground strips to the sidewalls of the box about a quarter wavelength from each port. This forms a quarter wavelength shorted (slot line) stub. At the port, the stub presents an open circuit to the current which, otherwise, would have started flowing along the outside edge of the ground strip.

To analyze the coplanar short, type “em -vd cosht”. You can get a copy of the file using copyex, if desired. The result is

```
10.0000000 1.000000 169.31
! Eeff = ( 5.25), Z0 = ( 58.19).
```

A perfect short is 180 degrees, thus we have a short that is about 11 degrees inductive.

It is possible to get the same result, only faster. Notice that the circuit is physically symmetric. To add in symmetry, use ***xgeom***. If this were the only change we were to make before analyzing the circuit, it would not work.

The reason for this is that when a circuit is symmetric, ***em*** does not subsection any part of the circuit below the center line. After subsectioning the circuit, ***em*** checks to make sure all ports have at least one subsection. But notice that the “-1” port below the center line is still there. This port does not have any subsections and ***em*** prints an error message to that effect and terminates. So also remove the lower of the two “-1” ports and save the circuit under “cosht_sy.geo”. An analysis of this circuit provides almost identical data in much less time.

Whenever you see an error message stating that there are no subsections for a given port, check for ports below the axis of symmetry, if symmetry is turned on.

♦ The Coplanar Cross Junction

The coplanar cross junction (see Figure 15) illustrates the analysis of coplanar waveguide structures including the effect of air bridges as an integral part of the coplanar discontinuity. The air bridges are needed in order to suppress the slot line mode which would otherwise propagate in the coplanar transmission line. In the case of the cross junction, the air bridges are needed for a second purpose, to provide a return path for ground current. The ground conductor on both sides must be continuous between all ports, otherwise the ground current on one side is interrupted. If so, very strange results are seen, both in the ***em*** analysis and in actual measurements.

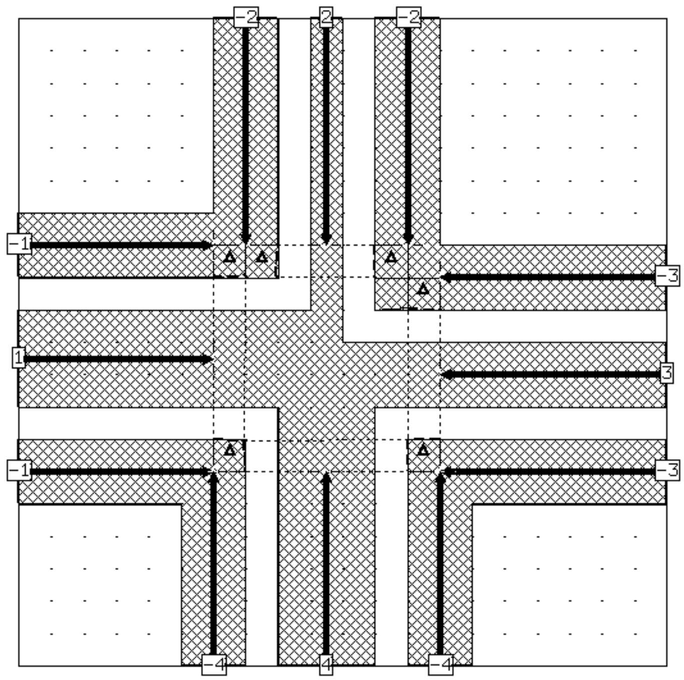


Figure 15 A coplanar cross junction with airbridges (dotted lines) to short out slot line modes and to provide a return path for current.

The air bridges are indicated by dashed lines. Both ends of each air bridge are supported by a via (indicated by the up triangle) which also provides electrical connection to the ground conductors. The air bridges are 1.5 µM above the GaAs.

This circuit is stored in the file “cocross.geo” in the examples directory and is accessible with the copyex command.

Chapter 14 Thick Metal with Arbitrary Cross-Section

The previous chapter discussed balanced or "“push-pull” ports. In this section, we use “push-push” ports to create thick metal lines where the vertical cross-section has an arbitrary geometry. To demonstrate this capability, we use a simple trapezoidal geometry, the cross section shown in Figure 16.

To analyze the thick metal, set up the dielectrics so that there is one layer of dielectric with the same thickness as the metal. Then, place a polygon representing the top side of the thick metal on the top side of that dielectric layer. Also place a polygon representing the bottom side of the thick metal on the bottom side of that dielectric layer.

Next, place ports on both top and bottom sides. To connect the top side port to the bottom side port, give both ports the same number. This can be done with the Objects ⇒Ports ⇒Renumber option in ***xgeom***. That’s all there is to it!

As many physical ports as desired may be given the same numeric label, and all ports with the same label are automatically connected together. Such ports can be called “push-push”, in contrast with the “push-pull” ports of the previous chapter.

A circuit implementing the above transmission line is stored in “thkthru.geo”. A copy can be obtained by using copyex. An analysis, using “em -vd thkthru.geo” yields:

```
! 10.0000000 0.308708 -125.3 0.951157 144.70 0.951157 144.70 0.308708 -125.3
! Eeff = ( 7.62), Z0 = ( 26.43).
! Eeff = ( 7.62), Z0 = ( 26.43).
```

For extremely thick lines, where the assumption that there is no current wrapping around the edge might not be valid, vias must be included along the edges of the line so that current flowing there is included. Note that vias, representing Z-directed current, are not needed to represent current flowing along the edge of the line. They are only needed for current which flows over and around the edge of the line, a rare situation in most planar microwave circuit designs.

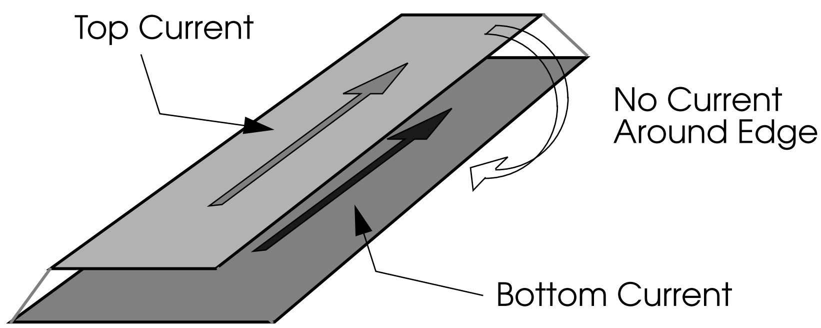


Figure 16 A trapezoidal cross-section transmission line viewed in perspective. If the line has no current going around the edge, it can be modeled, as shown, as two infinitely thin sheets of current, one at the top and the other at the bottom of the actual metal.

If a more detailed cross-section is desired, one can use more than two levels of infinitely thin metal to model the cross-section. Don’t place metal on the interior (i.e., metal which is not on the top, bottom or edge) of the thick line, there is no current inside a good conductor and there is no need to waste subsections there.

When a discontinuity is encountered in thick metal, vias probably should be included between the top and bottom metal as current may need to flow up or down at that point. Also if a line is longer than about an eighth wavelength, it would be a good idea to “tack” the top and bottom together with vias periodically to prevent unexpected resonances.

Do not place vias at the edge of the substrate, next to a sidewall on a port. In this situation, the vias short out to the sidewall. This shorts out the port. Such a situation can be identified by the port refection coefficient magnitude close to unity and the phase close to 180 degrees, a short circuit.

The example directory contains an example of a thick step junction in a file called "thkstep.geo". An analysis of that file with de-embedding gives:

```
10.0000000 0.246579 -106.4 0.969120 -19.15 0.969125 -19.15 0.246579 -111.9
! Eeff = ( 7.70), Z0 = ( 26.67).
! Eeff = ( 8.24), Z0 = ( 16.51).
```

Chapter 15 Determination of Package Resonances

The microwave circuit is designed, fabricated and meets all specifications. So put it in the package, button it up and ship it. Right?

Of course not. The circuit has to work in the package. Once the lid goes on, it must be tested. What could possibly go wrong? Package resonances and chip-to-chip coupling could go wrong. And when you are just about to ship is the wrong time for something to go wrong.

Em can help with the package design. Recall that ***em*** analyzes a circuit in a conducting 6-sided metal box. The sidewalls of the box are always perfect conductors. Thus, you can find out, on the computer, early in the design cycle, if the package is going to have resonances that are of any concern. As an example, we analyzed a model of an amplifier in a box (see Figure 17). The circuit is stored in the file “package.geo” in the examples directory.

We have left the transistor location open in this circuit for two reasons. First, ***em*** can not do active devices (if you want to use a circuit theory program to include a transistor, place via ports here). Second, if there is a resonance, it shows up as a value of S_{21} approaching unity. This indicates a strong, package resonance induced, coupling between input and output. And this means problems!

We have also included the bias lines (ports 3 and 4) to make sure that a package resonance doesn't kill the bias isolation. Feedback through bias leads is a significant cause of low frequency oscillations and smoked devices!

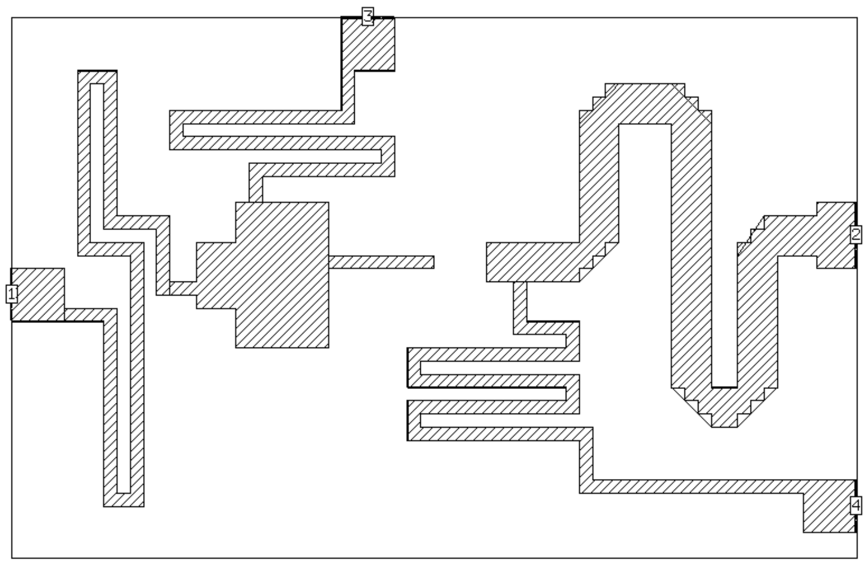


Figure 17 A model of an amplifier to check for package resonances. The entire box is not shown.

The amplifier model is only approximate, we don't need exact line widths and precise layout. The purpose of this analysis is only to check for resonances. The detail used in this case is probably more than is needed. A simpler circuit should be sufficient for a package resonance search.

The lowest resonance found was at 31.7625 GHz (see Figure 18). Given the approximate nature of the way we captured the circuit, we can conclude that there is a package resonance somewhere around 31 -32 GHz. There may be other resonances, the search was not exhaustive. Coupling between bias ports (not shown) also becomes large at resonance, making instability a virtual certainty.

In addition to searching for resonances by looking at the S-parameters, as we have done here, we can also search for resonances by invoking the "-b" option. The "-b" option is described in section "Verifying the Box Resonance Problem" on page 44.

This package has big problems at resonance. Fortunately, with the ***em*** analysis in hand, modifications can be made and tested before costly fabrication.

A second analysis was performed from 20 GHz to 60 GHz with the top cover removed (i.e., resistivity set to 377 ohms/square). This analysis showed several low Q (wide bandwidth) resonances, none of which exceeded 35 dB down (see Figure 19).

A related application of this analysis is to use the circuit package as the resonant, frequency determining element in an oscillator design.

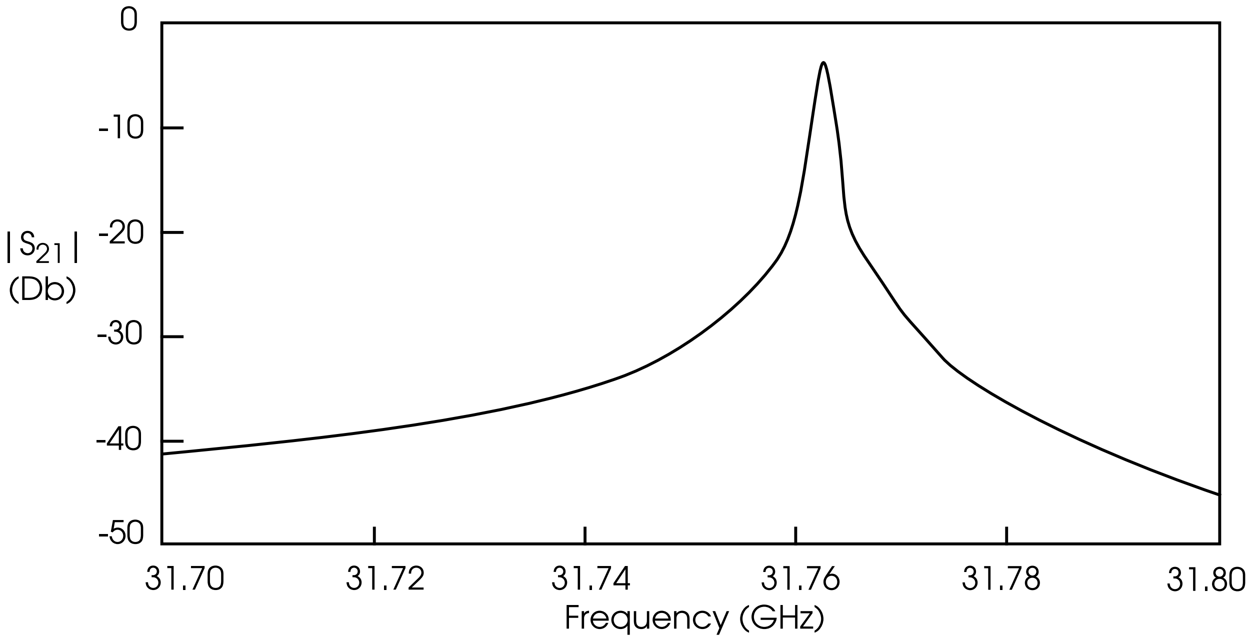


Figure 18 Results of a search for package resonances shows strong coupling between input and output at 31.7625 GHz.

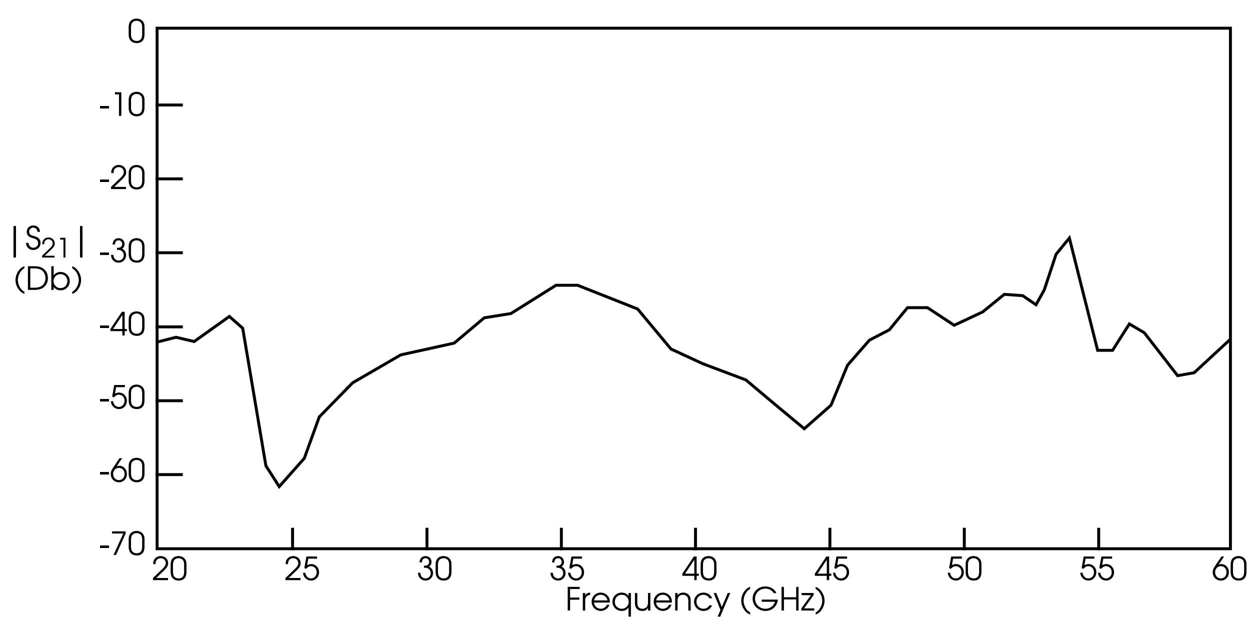


Figure 19 The package resonances disappear when the top cover is removed.

Chapter 16 Antennas and Radiation

While analysis of radiating structures is not the primary thrust of the ***em*** analysis, there is a large class of radiating problems for which ***em*** can provide useful results. This chapter, based on Sonnet Application Note 93-01 (see reference [20] at the end of the first chapter), describes how to use ***em*** to analyze 3-D planar radiating structures, such as microstrip patch arrays and microstrip discontinuities using the "Open Waveguide Simulator" technique. The underlying assumptions of this technique are described in detail. Common modeling mistakes are also pointed out. Examples are provided to illustrate the correct use of the modeling technique.

♦ Background

Since ***em*** is an analysis of 3-D planar circuits in a completely enclosing, shielding, rectangular box, the analysis of radiating structures is not something that one would ordinarily consider.

However, ***em*** can be used to simulate infinite arrays using a waveguide simulator. In this technique, as shown in Figure 20, a portion of the array is placed within a waveguide. The waveguide tube is vertical, connecting the radiating patches to the termination. The images formed by the waveguide walls properly model the entire infinite array scanned to a specific angle. The waveguide must be terminated in a matched load.

The waveguide simulator inspired what we now call the Open Waveguide Simulator Technique described in the next section.

♦ Modeling Infinite Arrays

The sidewalls of the shielding box in the ***em*** analysis easily represent the sidewalls of the waveguide in the infinite array waveguide simulator. A side view is shown in Figure 20. Providing a termination for the end of the waveguide requires a little more thought. After working with the equations, we found that any waveguide mode can be perfectly terminated by making the top cover resistivity in ***em*** equal to the waveguide mode impedance:

$$Z_{TE} = \frac{\eta}{\sqrt{1 - \frac{f_c}{f}}} \qquad f > f_c$$

$$Z_{TM} = \eta\sqrt{1 - \frac{f_c}{f}} \qquad f > f_c$$

$$f_c = \frac{v_c}{2\pi}\sqrt{\left(\frac{m\pi}{A}\right)^2 + \left(\frac{n\pi}{B}\right)^2}$$

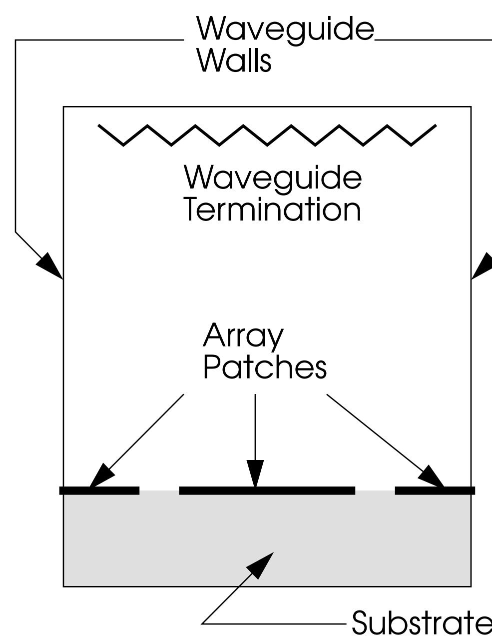


Figure 20 The waveguide simulator for infinite arrays inspired the technique described here. In this side view, the waveguide walls form images of the array of microstrip patches, simulating an infinite array.

In a phased array, with the array scanned to a specific direction, a single waveguide mode is generated. The ***em*** software can model the waveguide simulator of that infinite array just by setting the top cover impedance to the impedance of the excited waveguide mode. This modeling technique has been used by our customers since 1988.

♦ Modeling Finite Arrays

If we can use a closed (i.e., terminated) waveguide to model an infinite array, is there any way that we can model radiation from a finite array? Of course, the answer is yes, under certain conditions. It is important to keep in mind that, unless the analysis is carefully prepared, these conditions are easily violated, yielding incorrect results. When the conditions are met, useful results can be obtained, as we shall demonstrate.

First Condition: *Make both of the lateral substrate dimensions greater than one or two wavelengths.*

When using the Open Waveguide Simulator, we view the sidewalls of the shielding box as forming a waveguide whose tube extends in the vertical direction, propagating energy from the antenna toward the “Termination” in Figure 20. Radiation is then approximated as a sum of many waveguide modes. If the tube is too small, there are few, if any, propagating modes, violating the First Condition.

There is an easily made mistake when modeling radiation from small discontinuities. Discontinuities are usually small with respect to wavelength. For a discontinuity analysis, the sidewalls are usually placed one or two substrate thicknesses from the discontinuity. In this case, the substrate dimensions are unlikely to meet the First Condition. If the sidewalls form a cut-off waveguide, there is no radiation.

Second Condition: *Make sure the sidewalls are far enough from the radiating structure that the sidewalls have no affect.*

Another way to look at this condition is to consider the image of the structure (discontinuity or antenna) created by the sidewall. Position the sidewall so that the image it forms has insignificant coupling with the desired structure.

Usually several substrate thicknesses separation from the sidewall is sufficient for discontinuities. For single patch antennas, at least one half wavelength is needed. Requirements for specific structures can easily be greater than these guidelines. If the First Condition requires a larger substrate dimension than the Second Condition, the larger dimension is to be used. This is very important.

Third Condition: *Place the top cover outside the fringing fields (i.e., near field) of the radiating structure.*

If this condition is violated, the resistive top cover becomes involved in the reactive fringing fields which form the near field of the radiator. This changes what would have been reactive input impedance into resistive input impedance, incorrectly overestimating the radiation loss.

Don’t place the top cover thousands of wavelengths away from the radiator. Extreme aspect ratios of the box should be avoided. For a patch antenna, place the top cover one or two box widths away from the radiator. This usually works out to be about one or two wavelengths.

Fourth Condition: *Set the top cover resistivity to 377 Ohms/square.*

This value is a compromise. As shown by the equations on the previous page, all TE modes have a characteristic impedance larger than 377 Ohms (η), while all TM modes are lower. As the waveguide dimensions (i.e., lateral substrate dimensions), A and B, become large (see the First Condition), f_c decreases, causing all modal impedances to approach 377 Ohms. Thus, while a 377 Ohms/square top cover does not perfectly terminate any mode, it forms an excellent compromise termination for many modes. This approximates removing the top cover of the box. If the box is large, it, in turn, approximates radiation, as we shall demonstrate.

Fifth Condition: *The radiating structure can not generate a significant surface wave.*

If there is a significant (compared to required accuracy) surface wave, it is reflected by the sidewalls of the box. Unless this is the actual situation, such antennas are inappropriate for this technique. Actually, the Fifth Condition is a special case of the Second Condition. If there is significant surface wave, the Second Condition cannot be satisfied. We state the Fifth Condition explicitly because of its importance.

In general, any surface wave is both reflected and refracted when it encounters the edge of the substrate. This boundary condition is different from either the conducting wall of Sonnet or the infinite substrate provided by a true open space analysis.

A dual patch antenna is illustrated conceptually in Figure 21. The feed point is created

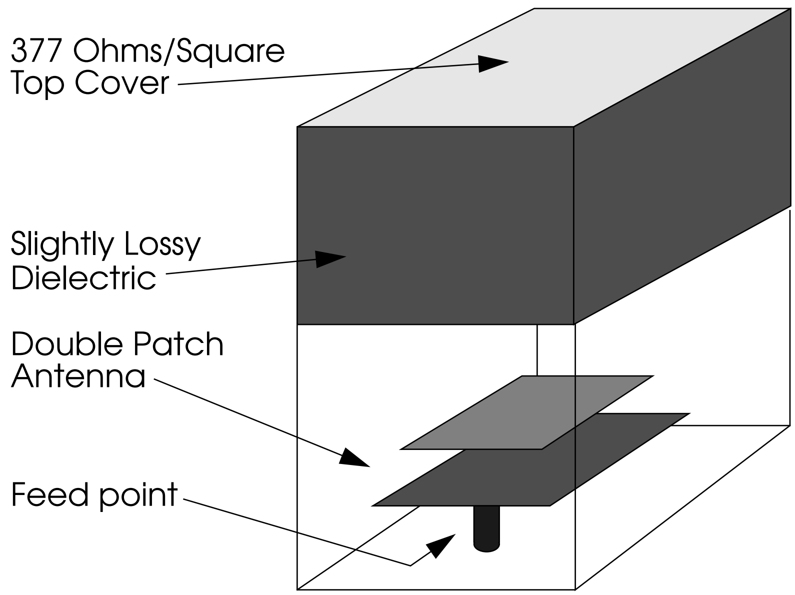


Figure 21 Radiation can be simulated by including a lossy top cover, a lossy dielectric layer (optional) and by placing the sidewalls far from the radiator (drawing not to scale!). Place the top cover one or two wavelengths from the radiator.

in ***xgeom*** by creating a via to ground at the feed point. Then the ground end of that via is specified as a port, just as one would specify a more typical port on the edge of the substrate at a box sidewall. A file showing an antenna similar to this one is named “patch.geo” and is available using the copyex command.

♦ Validation Example

For validation, we present work performed by E. Ongareau, Matra Defense, Antennas & Stealthness Dept., France, presented at the 1993 EEsof User's Group meeting at HYPER in Paris, reprinted with permission. The antenna is a triple patch structure, with a top view shown in Figure 22. The antenna is a test realization intended only for validation. It is not designed for optimum VSWR.

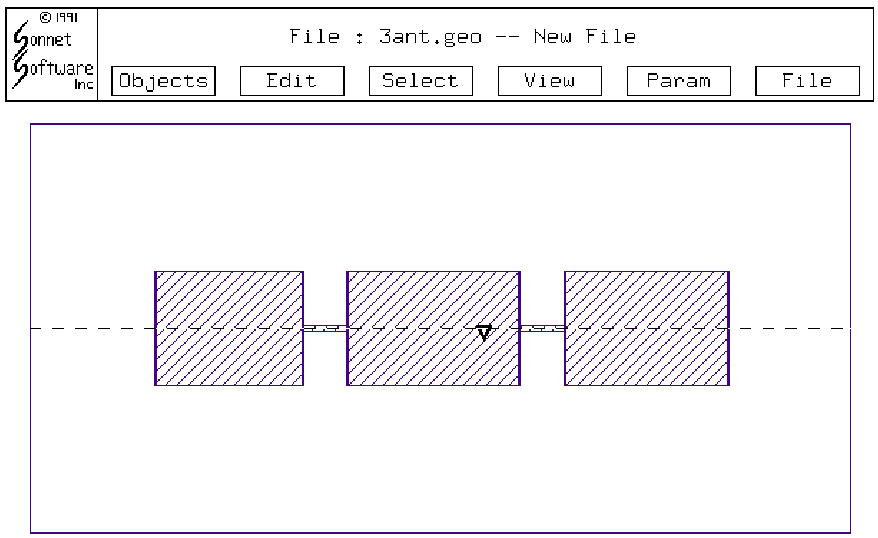


Figure 22 Top view of a triple patch antenna courtesy of Matra Defense. The central patch is fed with a coaxial probe (indicated by a down pointing triangle). Each patch is resonant at a different frequency to increase the overall antenna bandwidth.

Good results are also regularly obtained on single microstrip patch antennas. We present this example as one of the more sophisticated antennas analyzed using the Open Waveguide Simulator technique.

In this antenna, each patch has a slightly different resonant frequency. This results in an increased bandwidth. The antenna is fed from below with a coax probe attached to the central patch. The feed point is indicated with a triangle.

The substrate is 3.04 mm thick with a dielectric constant of 2.94. The drawing is to scale with substrate dimensions of 200 mm x 100 mm. The top cover is 200 mm above the substrate surface. Cell size is 0.78125 mm square. A loss tangent of 0.001 is used in both air and substrate. The small air loss helps terminate the propagating modes. The antenna geometry file is available using the copyex command to get the file "tri-pat.geo".

This validation was performed using a “double blind”. In using a double blind, the measured data and the calculated data are obtained completely separately, by different researchers. There is no opportunity to “tweak” the model to get it to match the measurement.

Figure 23 shows the result. We see that the resonant frequencies of each patch (i.e., the low VSWR points) have differences between measured and calculated of about 1%. This is typical of most analyses of patch antennas using this technique. The differences in resonant frequency (i.e., the reflection zeros) then determine the differences in the rest of the VSWR plot. The degree to which these differences are due to analysis error, fabrication error, and measurement error, cannot be determined from this data.

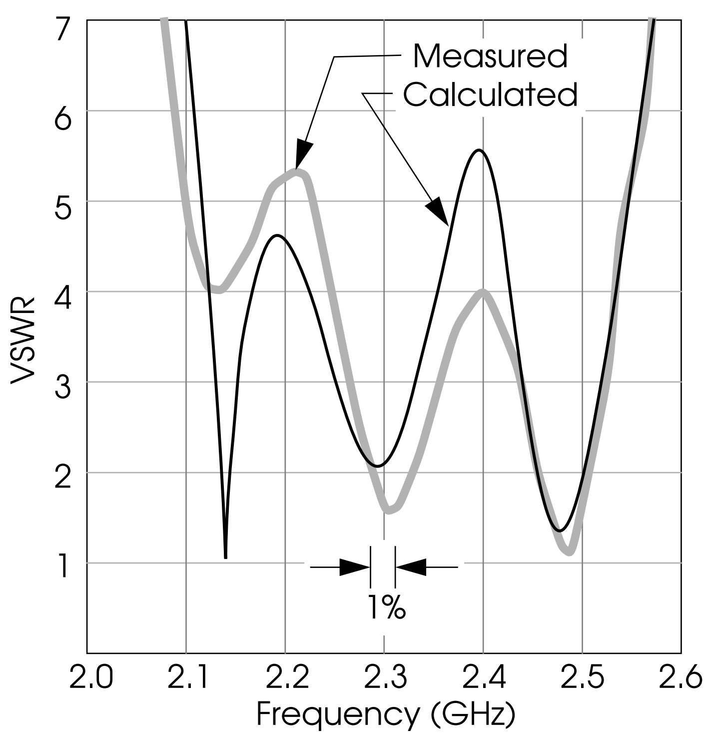


Figure 23 The measured and calculated data for the triple patch antenna were obtained completely separately, so there was no chance to “tweak” the model for agreement.

If the typical differences between measured and calculated data shown in Figure 23 are acceptable, given the specific requirements for a particular project, then the Open Waveguide Simulator technique can provide useful results.

Chapter 17 SPICE Lumped Model Synthesis

This chapter describes how to use ***em*** to automatically synthesize SPICE files. This capability is useful for circuits which are small with respect to the wavelength of the highest frequency of interest. This includes structures such as microwave discontinuities like step, tee, and cross junctions. Other applications include modeling cross-talk and propagation delay in digital interconnect circuits, and multiple spectrum circuits that combine digital, analog, and RF functions.

The option automatically takes the results of the electromagnetic analysis of a circuit and synthesizes a lumped model of inductors, capacitors, resistors, and mutual inductors. This information is then formatted into an ASCII SPICE file segment and saved on disc.

A second SPICE related capability of ***em*** is the generation of L, C, R, and G matrices for N-coupled transmission lines. These matrices represent the distributed parameters of the transmission lines, for example, inductance per unit length.

This chapter has been adapted from Sonnet Application Note 91-01. The theory of the synthesis technique is described in [3] (references are at the end of Chapter 1).

♦ Class of Problems

The SPICE generation capability is intended for any circuit which is small with respect to the wavelength of the highest frequency of excitation. Typically, 1/10th wavelength is an appropriate limit. If a circuit is too large, split it into two or more circuits and analyze each circuit separately. This limitation is due to the circuit theory limitations of modeling a circuit with lumped elements. The Sonnet electromagnetic analysis is not intrinsically limited in this fashion.

The model generated by the analysis includes any lumped elements (including mutual inductors) between any ports of the circuit layout. Lumped elements from any port to ground are also included. The synthesis capability does not allow internal nodes with the single exception of the internal node required to specify a resistor in series with an inductor. Internal nodes are nodes which are not connected to a port in the layout.

Any circuit which requires internal nodes for an accurate model should be split into several parts so that the required points become nodes. Internal ports without ground reference give incorrect results. Any internal ports should be carefully specified and checked for reasonable results. See the section "Internal Ports and Via Ports" on page 48.

The SPICE file generation capability is usually not appropriate for large microwave circuits as such circuits are usually larger than a small fraction of a wavelength. However, there are exceptions, for example, a de-embedded step discontinuity has zero physical size and can be used with this capability.

The SPICE model synthesis capability is fast enough that it can be used on circuits with hundreds of ports.

♦ Using The Option

First, select two frequencies for analysis and specify them in the ctl.an file. The SPICE synthesis needs electromagnetic results at two frequencies to accomplish its work. If four frequencies are specified, two lumped models are generated. If an odd number of frequencies is specified, ***em*** terminates with an appropriate error message.

Select the two frequencies so that they are separated by at about 10%. Also, make sure they do not exceed (or even come close to) the highest useful frequency, where the circuit is a sizeable fraction of a wavelength.

Make sure the frequency is not too low. When the subsection size is less than 0.00001 wavelength, numerical precision can be a problem. For example, if the subsection size (or vertical via length) is 1 mm, it would be unwise to analyze below 1 MHz.

If the frequency specified is low enough, ***em*** automatically starts using quadruple precision. The quadruple precision is used only for the matrix fill. The matrix is still stored and solved in the usual double (default) or single precision. The quadruple precision option can be enabled manually with the “-q” option.

When quadruple precision is enabled, a message is printed to the screen (if the -v option is selected) and a comment is placed in all output data files. Quadruple precision eliminates the low frequency precision problem, however execution time can be large.

After completing the analysis, always do a “reality check” for reasonable values. If you have bad data, either a too high or a too low frequency is likely to be at fault. To be absolutely sure your results are good, select a second pair of frequencies, different from the first pair by, say, a factor of two, and re-analyze the circuit. You should obtain almost the same answer.

To use the SPICE generation option, include the option “-x” in the command line. The “x” may be followed by a single digit specifying the displayed precision for the lumped element values. Default is “-x1”, resulting in output like, “C1 1 3 12.1pf”. When working with very small structures, specifying, for example, “-x4” may be desired. This results in output like, “C1 1 3 0.0312pf”. Values from 0 to 7 may be used. The digits 8 or 9 generate the same output as 7. Any capacitors which would be displayed as zero are not included in the model. Any inductors or resistors which are essentially open circuits are also excluded.

To reduce the number of lumped elements in the model, open circuit limits can be specified in the ctl.an file. The following table specifies allowed ctl.an lines.

Table 1: Allowed ctl.an Lines for SPICE Open Circuit Limits

Default Command	Description
CMIN 0.1	Minimum allowed capacitance (pF).
LMAX 100.0	Maximum allowed inductance (nH).
RMAX 1000.0	Maximum allowed resistance (Ohms).
KMIN 0.01	Minimum allowed mutual inductance (dimensionless ratio).
RZERO 0.0	Resistor to go in series with all lossless inductors (Ohms). Needed for some versions of SPICE.

All calculated component values which fall outside the allowed range specified by the user in the ctl.an file are excluded from the resulting lumped model. The RZERO line is provided for those versions of SPICE which need inductors to have some small loss, to avoid numerical difficulties. The default value of 0.0 disables this capability.

Output is sent to a file ending in ".lc".

In most cases, the de-embedding option, "-d" should also be used. Otherwise the small shunt capacitance from the port discontinuity is also included in the lumped model. See Chapter 6, "Automatically De-embedding the Discontinuity"; Chapter 7, "Automatic De-embedding Limitations"; and Chapter 8, "De-embedding Situations".

♦ N-Coupled Line Option

If the structure being modeled is an N-coupled line, the "-x" option described above can be applied to a short length of line to generate a single section of the L-C-L-C... lumped model of a transmission line. This approach has two disadvantages. First, the lumped LC model of a transmission line is approximate. Second, the LC model can be very time consuming to analyze. There is a better alternative.

That alternative is to use distributed LC parameters of a transmission line, specifically, inductance per unit length and capacitance per unit length. Analyses which use such data are much faster than those which use simple lumped models. In addition, accuracy is maintained at all frequencies for which TEM mode propagation is an adequate approximation.

For a single line, the L and C distributed parameters are each a single number. For N-coupled lines, L and C become N by N matrices. When metal loss is included, we now also have an R matrix. The resistance is in series with the inductance. When there is dielectric loss, a G matrix is also calculated. The conductance is in parallel with the capacitance. The synthesis determines whether a G or R matrix are needed only from the calculated S-parameters. The circuit geometry is not referenced at any time.

To generate LCRG matrices, use the option "-dX" in the command line. The "d" enables de-embedding, often required as described in the previous section. The ctl.an file needs only one frequency specified. If two frequencies are specified, two LCRG matrix sets are generated.

Output is sent to a file ending in ".lct".

Your .geo file must be an N-coupled line with ports 1 through N as input and ports N+1 through 2N as output. The input of line M should be port M and its output should be port M+N. ***Em*** does not check for this condition. This restriction does not apply to the "-x" option, only to the "-X" option. There is no limit on N.

The results are per unit length, where a unit length is the de-embedded length of the N-coupled line. The length must be short compared to the wavelength at the frequency of analysis. Don't forget that the non-de-embedded length of line must be long enough that the problems described in the section "De-embedding Reference Plane Length -- How Close is Too Close?" on page 39, do not occur.

♦ A Simple Microwave Example

Let's say we want a lumped model for the steps discontinuity used in the beginning of this manual. A copy of the file can be obtained by typing "copyex steps.geo".

In order to synthesize a lumped model, two frequencies need to be analyzed. Set your ctl.an file for 1.0 and 1.1 GHz. Analyze steps.geo with the command "em -vdx steps". The lumped model is stored in the file "steps.lc":

```
* Limits: C>0.1 pF, L<100 nH, R<1000 Ohms, K>0.01.
* Analysis frequencies: 1000.000000, 1100.000000 MHz
.subckt 1 2
C1 1 0 0.3pf
C2 2 0 0.2pf
L1 1 2 0.3nh
.ends
```

The first line indicates the open circuit limits. For example all capacitors less than 0.1 pF are excluded from the model. These limits can be modified as in Table 1 at the beginning of this chapter.

The next line documents the analysis frequencies. This is followed by the SPICE lumped model. In this case we have two capacitors to ground (node 0) and one inductor connected between port 1 and port 2. Ground is node 0 and all the remaining node numbers correspond to ports of the same number. If two decimal places of precision are desired, use the option "-x2" instead of "-x".

♦ Topology Used for SPICE Output

The topology of the lumped element model generated by ***em*** depends on the circuit being analyzed. In general, the model contains an inductor (in series with a resistor if using loss), a capacitor, and a resistor (when using loss) connected in parallel from each port to ground. A similar parallel RLC network is also connected between each port. Therefore, a four-port circuit can contain more elements than a two-port circuit. Each inductor may also have a mutual inductance to any other inductor in the network.

Figure 24 shows the most complex equivalent circuit possible for a two-port (mutual inductances not shown). Any values that are outside of the open circuit limits are not included.

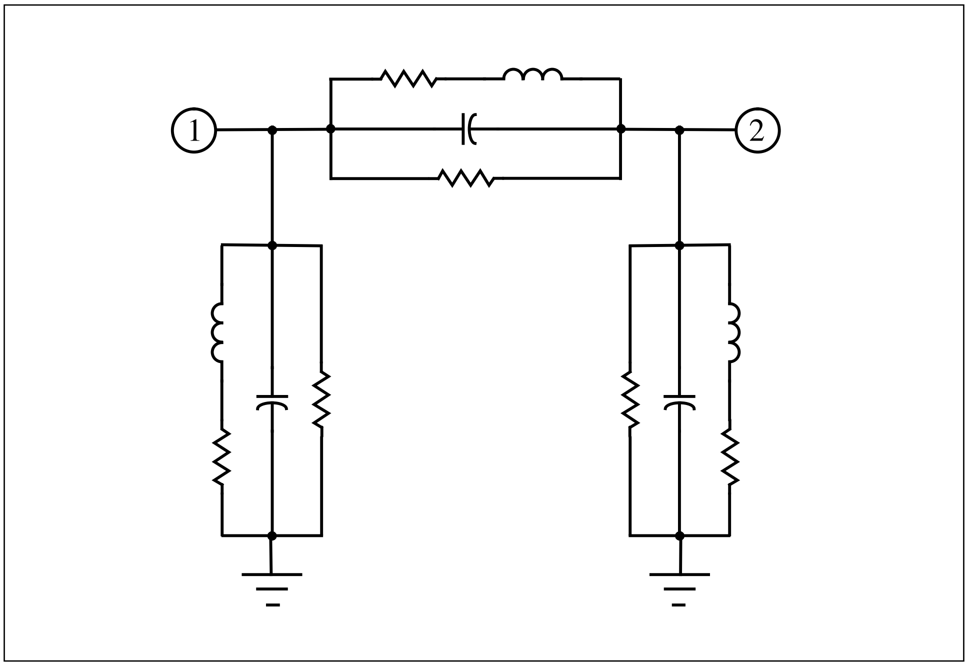


Figure 24 Equivalent circuit of a two-port structure generated by em's SPICE option. Mutual inductances also exist between all inductors but are not shown. Any component whose value is outside of the open circuit limits are not printed in the SPICE output file.

♦ A High Speed Digital Example

Figure 25 shows the two levels of an example circuit. There are 32 input ports and 32 output ports for a total of 64 ports. The first eight bits (ports 1-8) go into the circuit, down to the second level, underneath all the other lines, and come up on the right hand side as the last eight bits. Thus, this is a byte-reversal network.

Top Level

Bottom Level

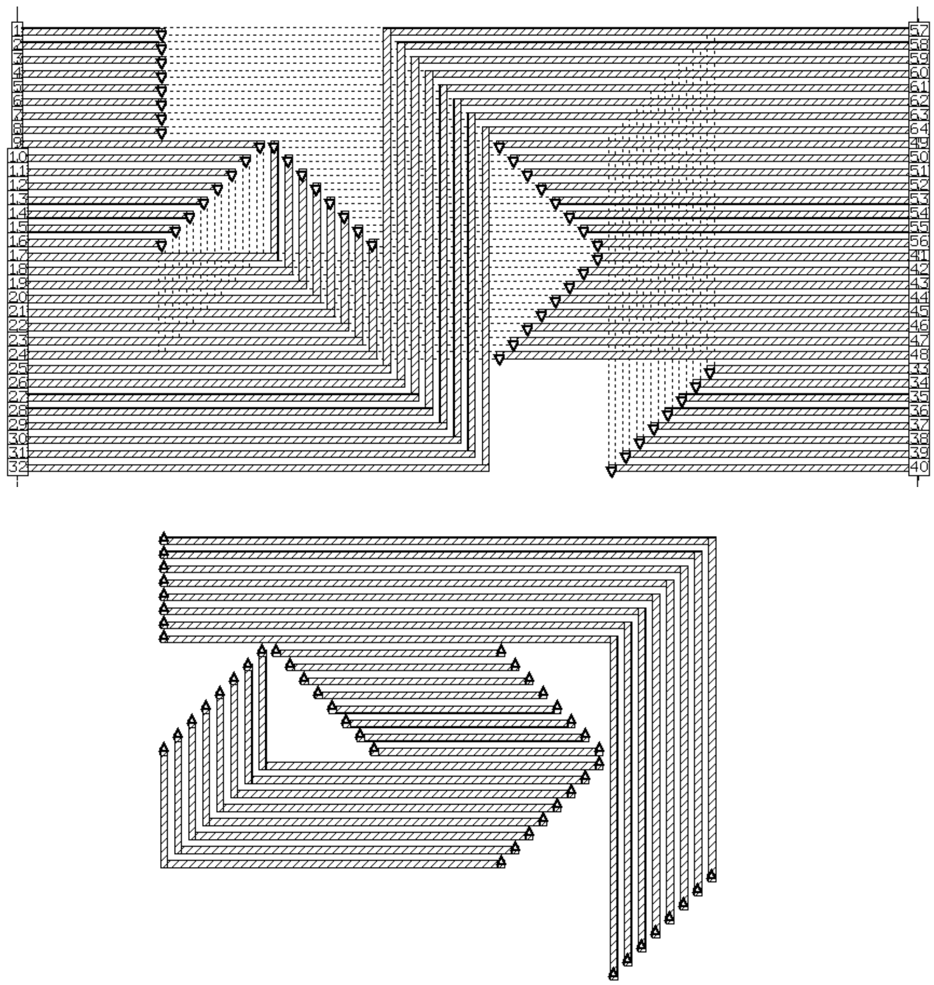


Figure 25 The top and bottom level of the byte-reversal network. The byte order on input (left side) is reversed on output (right side). This is a top view. The circuit is on a 1 mm snap grid. The triangles indicate connections between levels (up and down)

The circuit layout can be imported from EEsof, GDSII, Cadence Analog Artist, or captured directly by mouse. A copy of the above file can be obtained by typing, “copyex br32.geo”.

The circuit was analyzed at 10 and 15 MHz, where each line is about 1 degree long. The analysis required just under 1 minute per frequency on an HP-710, using the “-e” option to reduce the number of subsections at cross-over points. The following is a portion of the resulting SPICE model (... indicates information left out):

```
.subckt bytervrs 1 2 3 4 . . . 63 64
C1 1 0 14.6pf
C2 1 2 1.6pf
C3 1 3 0.1pf
C4 1 9 0.1pf
.
.
.
C1057 64 0 9.2pf
L1 1 33 77.2nh
L2 2 34 77.3nh
.
.
.
L32 32 64 79.6nh
K1 L1 L2 0.2
K2 L1 L3 0.1
.
.
.
K70 L31 L32 0.2
.ends
```

Nodes 1 - 64 correspond to the ports of the same number in the circuit layout, Figure 25. Node 0 is ground. For example, C1 represents the capacitance from port 1 to ground. L1 represents the inductance from port 1 to port 33 (i.e., port 1 is connected to port 33). There is also a capacitance (not listed above) from port 33 to ground.

C2 is a stray capacitance coupling ports 1 and 2, generating cross-talk. The capacitive coupling causes cross-talk when ever there is a time varying voltage difference between ports 1 and 2.

Mutual inductance K1 inductively couples the port 1 to port 33 line (L1) to the port 2 to port 34 line (L2), also generating cross-talk. The mutual inductance causes cross-talk whenever either the port 1 or port 2 line carries a time varying current. A quick inspection of this file reveals the worst cases for cross-talk (i.e., largest mutual inductors and capacitors).

The sub-circuit was placed in a complete SPICE file with an example analysis shown in Figure 26.

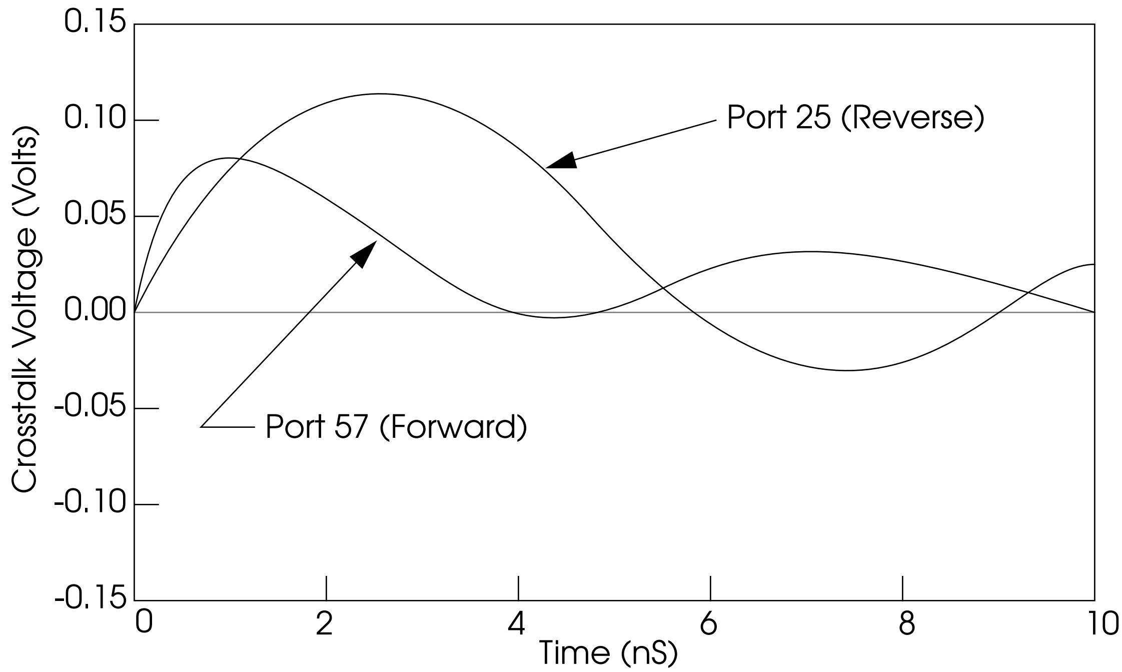


Figure 26 A SPICE analysis was performed using the ***em*** synthesized model of the byte-reversal network. Shown here is the cross-talk to the port 25 - port 57 line caused by a 1 Volt signal with a 10 pS rise time on port 1. All ports are terminated in 50 Ohms. Analysis courtesy of CONTEC Microelectronics USA.

♦ Other Techniques

Classical techniques use, for example, just an electrostatic or just a magnetostatic analysis to derive a model. This is adequate for uniform transmission lines embedded in homogenous dielectric (no different layers). In an arbitrary predominantly planar circuit, as we have here, a single static analysis provides only half a circuit model, just the capacitors or just the inductors.

Since ***em*** is a full dynamic analysis, both the inductive and capacitive portion of the model are obtained with one analysis (at two frequencies).

In addition, the techniques usually used for the static analyses are of a volume gridding variety (e.g., finite elements, finite difference). Even under the simplifications allowed by static analysis, the circuit shown in Figure 25 is well beyond the capability of such software tools.

However, because ***em*** is a surface meshing analysis, it can do the circuit of Figure 25 in under 1 minute on an HP-710 using about 1 MByte of memory. The SPICE file is generated after analyzing two frequencies.

The model which results has 1057 capacitors, 32 inductors, and 70 mutual inductors. If loss is included, the model would also include resistors. Many of the capacitors in the lumped model are 0.1 and 0.2 pF, just over the default minimum capacitance of 0.1 pF. A much simpler model (with a little less accuracy) is possible by including the command, “CMIN 0.3” in the ctl.an file. Nearly all of the 1057 capacitors are then excluded from the model because they are too small.

Note that the experimental approach to modeling this circuit involves building the circuit, measuring a 64 port structure (requiring 2080 separate complex measurements at each frequency), developing and entering an appropriate 1000+ element model in a circuit simulator, and optimizing each of the 1000+ variables for a best fit. Such a task is well beyond the state-of-the-art.

In contrast, by using the ***em*** analysis, the total, end-to-end time is a few minutes. This includes the time required for manual circuit layout capture and inspection of the final results.

Chapter 18 Accuracy Benchmarking

Electromagnetic analyses are often described as providing what is called, “Good Agreement Between Measured And Calculated” (GABMAC). However, in the past, there has been little effort to decide just what “good” means. Also, everyone seems to enjoy finding “good agreement”, rather than looking for what we really need to know in order to do our job, which is the “Difference Between Measured And Calculated” (DMAC).

In this chapter, we describe a precise benchmark which allows the evaluation of DMAC for any 2.5-D or 3-D electromagnetic analysis down to the 1 x 10^{-8} level of accuracy.

♦ An Exact Benchmark

What we need to calculate DMAC is an exact benchmark. One source of an exact benchmark is stripline. The characteristic impedance of a stripline has an exact theoretical expression:

$$Z_0\sqrt{\varepsilon_r} = 30\pi\frac{K(k')}{K(k)}$$

$$k = \tanh\left(\frac{\pi}{2}\frac{w}{b}\right) \qquad k' = \sqrt{1-k^2}$$

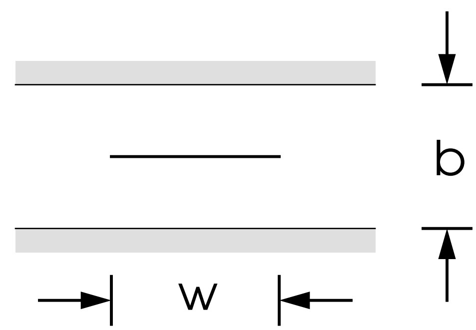


Figure 27 Characteristic impedance of stripline has an exact theoretical expression. K(k) is the complete elliptical integral of the first kind. For evaluation on a computer, a polynomial for K(k) is available in Abramowitz and Stegun, **Handbook of Mathematical Functions**, pg. 590 - 592. Be sure to note the errata, $m_1 = 1\text{-}m^2$, not 1-m.

The expression for K(k) cited in the caption of Figure 27 provides an accuracy of about 1 x 10^{-8}. When programmed on a computer, the following values are obtained for three different transmission line impedances (unity dielectric constant):

Table 2: Stripline Benchmark Dimensions

Z_0 (Ohms)	*w/b*
25.0	3.3286400
50.0	1.44369356
100.0	0.50460578

For a length of stripline, there are two parameters of interest: characteristic impedance and propagation velocity. With the w/b given above, we know the exact answer (to within 1 x 10^{-8}) for Z_0. With a dielectric constant of 1.0, we also know the exact answer for the propagation velocity. It is speed of light, known to about 1 x 10^{-9}. Any difference from these values is error, or, DMAC.

Each of the above three benchmarks are available via the copyex command. To get the 50 Ohm line, type, “copyex s50.geo”. The other benchmark circuits are in s25.geo and s100.geo.

The “b” dimension is exactly 1.0 mm, the “w” dimension is given by the above table, and the length of each line is 4.99654097 mm with a dielectric constant of 1.0. Each of these lines is precisely 0.25 wavelengths long at 15.0 GHz. The .geo files have the subsectioning set so the lines are 16 cells wide and 128 cells long. Analysis time is a few seconds on an HP-710.

To evaluate DMAC, do an analysis of the line at 15 GHz, with de-embedding enabled. For the error in characteristic impedance take the percent difference between the calculated value and the exact value, above. For the error in propagation velocity, take the percent difference between the calculated S_{21} phase and -90 degrees. Total error, in percent, is the sum of the two errors.

Some types of analyses do not calculate characteristic impedance. A detailed error analysis shows that, to first order, for a 1/4 wavelength long 50 Ohm line, the value of $|S_{11}|$ is equal to the error in characteristic impedance. For example, an $|S_{11}| = 0.02$ means that there is about 2% error in characteristic impedance. To use this approximation for, say, a 25 Ohm line, the S-parameters must be converted to 25 Ohm S-parameters. This may be done by adding transformers in a circuit theory program.

♦ Residual Error Evaluation

We have performed a detailed analysis of the relationship between subsectioning and residual error (DMAC). The simplest way to subsection a line is to use subsections the same width as the line. In Sonnet, and in many other analyses, this results in a uniform current distribution across the width of the line. In reality, the current distribution is singular at the edges of the line.

Since the current distribution is symmetrical about the center line, using either one or two subsections across the width of the line gives the same amount of error.

We find that a one or two subsection wide line gives 5% to 6% error. If there is not much stray coupling, circuit theory can often give a better result. When the line is 16 cells wide, we see about 1% error, much more reasonable. We have found (and you can verify) that convergence is very strong: Double the number of cells per line width and the error is cut in half.

When we vary the number of cells per wavelength, along the length of the line, we see an inverse square relationship. Double the number of cells per wavelength along the length of the line and the percent error decreases by a factor of four.

An equation which expresses the error as a function of subsectioning is:

$$E_T \cong \frac{16}{N_W} + 2\left(\frac{16}{N_L}\right)^2 \qquad N_W \geq 4 \qquad N_L \geq 16$$

where

N_W = Number of cells per line width,

N_L = Number of cells per wavelength along line length,

E_T = Total Error (DMAC) (%).

This equation estimates subsectioning error only. For example, any de-embedding errors are added to the above error. This error estimate should be valid for any electromagnetic analysis which uses roof-top subsectioning.

Notice that the quantities used for the error estimate are in terms of cells, not subsections. Cells are the smallest possible subsections size. In Sonnet, subsections in the corners of polygons are one cell on a side. Subsections along the edge of polygons are one cell wide and can be many cells long. Interior subsections can be many cells in both dimensions.

We have found that, for most cases, the cell size is the important parameter in determining error. Or, in other words, the smallest subsection size is important. For example, the stripline benchmark .geo files, mentioned before, are set to make the lines 16 cells wide, even though those 16 cells may be merged into only 4 or 5 subsections. It is the 16 cells which determine the level of error, not the 4 or 5 subsections.

In performing this error evaluation, we also found that the error in characteristic impedance due to N_W is always high, never low. Also, there is very little variation in the error for different impedance lines. The above equation can be very accurate in evaluating error. And, finally, for N_L above about 40 cells per wavelength, all the error is in the characteristic impedance. The error in velocity of propagation is essentially zero. The above equation can be very accurate in evaluating error. With this precise knowledge of the error, we can now do something about it!

♦ Using the Error Estimates

The above error estimate can be used to estimate the error for an overall circuit. Let's say that a cell size is used that makes some high impedance transmission lines only 1 cell wide. Other, low impedance transmission lines, are, say, 30 cells wide. The 1 cell wide lines give us about 5% error. The 30 cell wide lines give about 0.5% error. In non-resonant situations, you can expect the total error to be somewhere between 5% and 0.5%. If most of the circuit is the low impedance line, the error is closer to 0.5%, etc.

However, let's say that our circuit has resonant structures. Let's say it is a low pass filter. It is easy to verify by means of circuit theory that the low pass filter is very sensitive to the high impedance lines. This means we can expect about 5% error, even though the high impedance lines only make up half the filter.

Given this information, there are several courses of action. First, if 5% error is acceptable, no further effort is needed.

More likely, we wish to analyze the filter with less error. Since we now know the error in the characteristic impedance is 5%, can physically widen the line so that the characteristic impedance is 5% lower to compensate for the known increase in characteristic impedance due to subsectioning the line only one cell wide. Very precise analyses are possible using this compensation technique.

♦ A Microstrip Benchmark

Although we do not present details here, we have used the confidence gained with the stripline benchmark to establish a microstrip benchmark. In microstrip, both the characteristic impedance and velocity of propagation vary with frequency. We present the calculated values here, determined by a series convergence approach. We believe the values to be accurate to +/- 0.07%. The microstrip line is 0.635 mm (25 mils) wide on

Table 3: Microstrip line Benchmark Parameters

F(GHz)	Z_0(Ohms)	ε_{eff}
2.0	48.68	6.578
4.0	48.61	6.621
6.0	48.54	6.691
8.0	48.51	6.786
10.0	48.62	6.901
12.0	48.97	7.027
14.0	49.71	7.138
16.0	50.51	7.268
18.0	52.37	7.426

0.635 mm thick substrate with dielectric constant of 9.8. A copy of the .geo file can be obtained by typing. “copyex m48.geo”.

Chapter 19 Range of Analysis Validity

Em is a complete electromagnetic analysis, all electromagnetic effects, such as dispersion, loss, stray coupling, etc., are included. Other than the finite numerical precision inherent in digital computers, there are only two approximations used by ***em***. First, ***em*** subdivides the metallization into small subsections. Second, the coupling between subsections is an infinite summation which ***em*** truncates. The user has full control over both approximations.

The number of terms taken in the infinite summation has little impact on execution time (try it!). In addition, nearly full accuracy is obtained, even at minimum setting. ***Em***'s -c option determines the number of terms taken. For example, -c2 takes four times the number of terms over -c1, while -c3 includes nine times the number of terms. Default is -c2. The -c option is of little concern.

In contrast, the cell size is important. By using a smaller cell size, metal edges can be more accurately defined and the microstrip current distribution is better represented. Both effects improve the accuracy of the result. The trade-off is increased execution time. A quantitative description of accuracy versus cell size is given in Chapter 18, "Accuracy Benchmarking".

The trade-off between execution time and accuracy increases the degrees of freedom available to the microwave design engineer. With previous software, if additional accuracy is needed (and when is it not needed?), the designer is stymied. With such software, the accuracy you get is what you have to take. Period.

♦ Subsection Size

There are actually two limits on subsection size, a minimum and a maximum. When we refer to a cell, we have in mind the minimum limit. For example, a circuit may cover 10,000 cells. ***Em*** can use variable size cells, which we refer to as subsections. That same 10,000 cell circuit could be subsectioned into only 200 variable sized subsections. The maximum limit determines the size of the largest subsection (which may cover many cells).

With a maximum frequency in mind, the cell size should be set so that a single cell is no more than a fraction of a wavelength at the maximum frequency of analysis.

The maximum subsection size is usually specified in terms of subsections per wavelength. For most circuits 10 or 20 subsections per wavelength provide good results. A large value, say 40,000 per wavelength assures that the cell size and subsection size are everywhere the same. If the cell size is larger than the specified number of subsections per wavelength, the subsections per wavelength specification is ignored.

In the limit, as subsection size (both maximum and minimum) is made smaller, the ***em*** analysis is asymptotically exact. Given sufficient computer resources, an arbitrarily accurate answer may be achieved.

♦ Metallization and Dielectric Thickness

The analysis assumes zero metallization thickness. This could result in some discrepancy for thick lines which are tightly coupled. An approximation for metallization thickness is to have two zero thickness metallization levels, one at the top and the other at the bottom of the actual metallization. Layer to layer vias may be used to form the sides of the thick lines, see Chapter 14, "Thick Metal with Arbitrary Cross-Section".

Dielectric layer thickness and dielectric constant have no impact on the accuracy as long as the layers are greater than 0.05 microns thick. For layers thinner than 0.05 microns, numerical precision does not allow accurate answers. The substrate area has no impact on accuracy, however, it does have a secondary impact on analysis time.

All coupling between all subsections is calculated and included in the analysis, even coupling between patches of metallization which are not connected to anything.

♦ Numerical Precision

Em experiences a numerical precision problem (difference between two large numbers) if the subsection size gets too small. For example, if the subsection size is several microns and the frequency is a few kiloHertz, there are precision problems. The results quickly become obviously bad. A higher frequency of analysis, or use of quadruple precision, -q, solves the problem. At a higher frequency, the subsection becomes larger, in terms of wavelengths.

The precision problem also arises with vias. This is especially true if, in addition to the small cell size, the via is in a thin layer, say a micron or less. Unreasonable results are seen as high as several hundred MHz, or even 1 GHz. The net effect of the loss of precision makes the via appear to have a very long phase length. These problems become worse if the -m (single precision) is used. Again, use of the quadruple precision option, -q, can solve the problem in exchange for increased analysis time.

Chapter 20 Time Required for Analysis

Em is a memory and computation intensive program. Small circuits are analyzed quickly while large circuits can require a considerable amount of time. In some cases, it may be desirable to run the program in background (i.e., under UNIX, end the command line with an ampersand, “&”).

There is no simple rule for calculating the time required for a particular analysis so some guidelines are presented.

The amount of time required is closely related to the number of subsections, which is printed out when ***em*** is run with the -v option. After a few trials, you will have a good idea of whether an analysis is a few seconds, a few hours, or just totally hopeless by looking at the number of subsections.

The most important factor in the execution time required is the amount of memory required by ***em*** compared with the amount of memory you have. This is also printed out by ***em*** if you use the -v option. If ***em*** says you need 36 Mbytes and you have 16 Mbytes installed, kill the run quickly! You can reduce the memory required by using the -m, memory saver, option, which makes the analysis single precision. You can also try to reduce the area of metallization in the circuit. Try to eliminate any metal that is not carrying current and make connecting lines as short as possible (but not too short, see the chapters on de-embedding).

Another approach, if your memory requirement is right on the edge, is to free up some of your computer's memory. Make sure no one else is also running a big number cruncher at the same time. Also, if you have a window system running, take it down before starting your job.

The estimate of required memory printed out by ***em*** is just an estimate. It is usually within 1 Mbyte or so, but could be off by much more. To get both the memory estimate and the number of subsections without going on to actually analyze the circuit, use the -N option.

For most circuits, the following equation can be used to estimate the amount of memory that will be used by ***em***:

$$B = K*N^2$$

where B is the number of bytes, and N is the number of subsections. K is equal to 8 if running with double precision and loss. This can be circuit metal loss, top or bottom cover loss, or dielectric loss. K is equal to 4 if you are running with loss but using single precision (-m), or running lossless and using double precision. K is equal to 2 if you are running a lossless circuit and using single precision. This equation should be used only as an estimate as it only includes the memory used by the final matrix in ***em***. Circuits with large boxes (in terms of number of cells) or many layers require more memory. You should use this equation to calculate an upper limit on the number of subsections for your computer.

For UNIX systems, you can check the memory actually used by typing the "ps" (process status) command. Consult your system administrator or UNIX manuals for details on the "ps" command.

To check how much of your system's memory is actually available for your use, use the "checkmem" command, provided with your software. See Chapter 1, "The Checkmem Utility" of **Sonnet Utilities** for information.

♦ The “Wall”

When using circuit theory analysis, an increase in circuit complexity gradually produces an increase in analysis time. With an electromagnetic analysis, the increase happens suddenly. A mere doubling of circuit complexity (say, by using a smaller subsection size) can result in one, or even two, orders of magnitude longer analysis. We call this the “Wall” (see Figure 28).

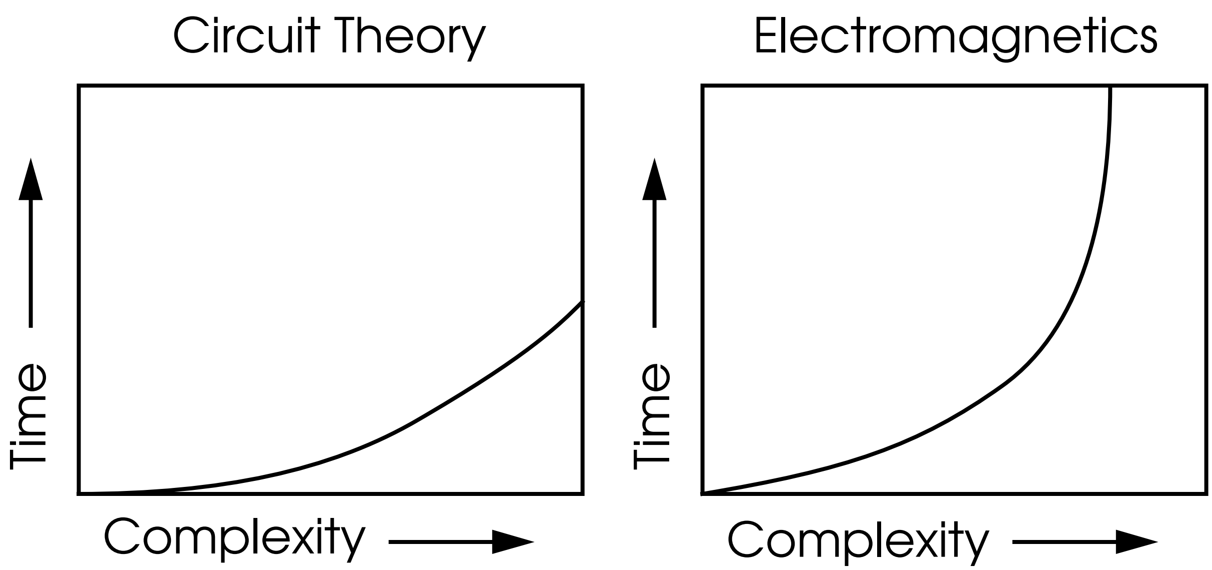


Figure 28 Circuit theory analysis time increases slowly with respect to circuit complexity. Electromagnetic analyses run into the “Wall” at some point.

The Wall is frequently encountered when ***em*** runs out of real memory, as described above, and is forced to start swapping out to disk. Execution time can quickly go from a few minutes to a few hours. Either get more memory or modify the circuit so that there are fewer subsections.

To avoid the wall, *start with no loss (metal or dielectric) and use a large subsection size*. Perform the first analysis at a single frequency to evaluate how long an analysis takes. Then, provided you get fast results, try adding loss or making subsection size smaller. Keep going until the analysis is as long as you can tolerate and then let it run over a full range of frequencies, perhaps overnight.

The main factor in analysis time is the number of subsections. ***Em*** prints out the number of subsections if the “-v” option is used. As you gain experience with ***em***, you will get a good feel for what can be tolerated. For example, on a Sun SPARCstation 1 with

16 Mbytes of memory, up to 1700 subsections (lossless) or 1200 subsections (with loss) can be calculated in an hour or so. At this point the computer runs out of memory and starts swapping to disk, resulting in huge increases in time.

To avoid the frustrations of getting on the slow side of the wall, *start lossless with big subsections*. You may find that big subsections provide all the accuracy you need!

♦ Detailed Parameter Dependencies

How do changes in the various input parameters affect the analysis time?

First, keep in mind that ***em*** has two stages in the analysis. In the first stage, ***em*** fills a large matrix. The matrix has one row/column for each subsection. This is where ***em*** is calculating the coupling between every possible pair of subsections. In the second stage, ***em*** is solving the matrix. Here, ***em*** is performing matrix inversion-like functions and is calculating the currents which allow all boundary conditions to be met. The different parameters affect each stage of the analysis differently.

During analysis of the first frequency in a run, ***em*** prints out the amount of time spent on various portions of the analysis. If the time spent in any particular section of the analysis is less than one second, it is not printed out. Sections which are timed include the waveguide mode calculation (prior to matrix fill), the matrix fill, and the matrix inversion.

Parameters which have no effect on analysis time include the substrate thickness, cover height and number of ports. Each of these parameters is unlimited and have no impact on speed while still maintaining complete accuracy.

Including metallization loss increases the matrix solve time only. And this is only if the rest of the structure is lossless. If there is any dielectric loss or ground plane (or top cover) loss, there is virtually no additional impact from also including metallization loss (the whole calculation is already fully complex). The matrix solution time is increased by about a factor of four (if it becomes complex). Metallization loss has no impact on any other segment of the analysis.

Including dielectric loss, or ground plane or top cover loss, makes the entire calculation complex. The matrix solve time is increased by about a factor of four, while the matrix fill time is increased by about a factor of two.

In calculating the values for the matrix elements during the matrix fill, several two dimensional DFT's (Discrete Fourier Transform) must be calculated. The size of the DFT's are the same size as the substrate in terms of cells. If a substrate is 128 x 64 cell, each DFT is 128 x 64 elements. Memory storage is required for only one DFT at a time and this is usually much smaller than the matrix being filled.

For large substrate dimensions (as measured in terms of cells), it is best to use a power of two. With a substrate dimension of 32 cells, there is little difference when a power of two is used. However if the substrate dimension is 512 cells, noticeably slower execution results if one were to use 500 cells. With a power of two, the FFT algorithm is automatically used to evaluate the DFT

Normally there are three DFT's performed. One for X current to X current coupling. A second for Y current to Y current coupling. The third is for X to Y current coupling. If there is more than one layer, the coupling between every possible pair of layers is also calculated, with another three DFT's for each coupling. If corner fill is used, there are now four kinds of subsections between which coupling must be calculated (10 DFT's). If triangle fill is used, there are another two kinds of subsections and vias require one more type of subsection.

For all the above detail, keep in mind that if the substrate is small (less than 256x256 cells), the FFT time is of little consequence.

If the quadruple precision option, -q, is used, all matrix fill operations are performed in quadruple precision. Depending on the computer and specific problem being solved, this can add substantial time to the analysis. The precision used in the matrix solve is not affected by the -q option.

Matrix fill time is proportional to the number of subsections squared for large circuits.

The number of subsections, for a given cell size, can be reduced by minimizing the number of vertices and the number of diagonal lines in the polygonal description of the circuit. If the circuit is symmetric with no more than two ports, with both ports on the axis of symmetry, invoke the symmetry option for a significant memory and time savings.

The matrix solve time is proportional to the number of subsections cubed and is the main limitation on the analysis at this time.

Appendix I The *Em* Command Line

NAME: ***em*** - Electromagnetic analysis of planar circuits.

SYNOPSIS: ***em*** [-options] [files]

DESCRIPTION: ***Em*** performs an electromagnetic analysis of 3-D planar circuits. The circuit geometry is subdivided into small rectangles. The analysis calculates the N-port circuit parameters (e.g., S-parameters) and is described in the "***Em*** **User's Manual**". The circuit geometry is stored in a file which can be created by a geometry capture program such as ***xgeom*** (see the ***Xgeom*** **User's Manual**).

If the last item on the command line is not a recognized file name (i.e., does not end with ".d", ".nd", ".rsp", ".jxy", or ".an") and is not recognized as an option (i.e., does not begin with "-"), then it is treated as the root file name. With the exception of the analysis control file (ctl.an), the root file name will have the appropriate extensions added to satisfy all specified options. Previously specified file names take priority. See the examples section at the end of this appendix.

♦ Options

-c Followed by an integer, 1-5, to indicate the number of cycles to be used in the modal summation. Values outside the valid range are limited to within the range. Larger numbers give slightly more precise answers. For example: "em -c1". Default is 2.

-m	Memory saver option. The system matrix is filled in single precision which reduces memory requirements for storing the matrix to one half of that of double precision. On Sun SPARCstations, matrix solution time is twice as fast. On Cray computers, this option has no effect. If this option is not used, the matrix is stored in double precision. This option affects only the matrix storage and the matrix solution. The precision of the matrix fill calculations is not affected (see “-q”).
-q	Quadruple precision option.If cell size is less than about 10^{-5} wavelength, numerical precision can cause numerical error. This option switches the matrix fill calculations to quadruple precision so that very low frequency analysis is accurate. For example, for a cell size of 1 micron and a frequency of 1 GHz, errors are often seen, and the -q option should be used. The option is enabled automatically if the frequency is low enough. This option affects only the matrix fill calculations. The matrix is still stored and solved in double (or single, if “-m”) precision. This option and “-m” are completely independent.
-s **-y** **-z**	N-port circuit parameters are stored in S, Y or Z parameter form. The default is -s. The Y and Z parameters may be normalized to 50 Ohms depending on the microwave circuit analysis software vendor (see -CT).
-G **-M**	Select GHz or MHz for the circuit response file listing. If neither option is used, the frequency units default to those last used in the analysis file. If no frequency units were explicitly specified in the analysis control file, MHz is used. This option has no effect on information specified in the analysis control file.
-C **-T**	Store the circuit response information in Super-Compact (-C) or Touchstone (-T) format. Default for this option is Touchstone.
-n	Specify a name for the output SPICE sub-circuit (see -x) or for the S-parameter output data (see -C). For example, “em -x -nModel_name”.
-r	Specify the file for circuit response (usually S-parameter) data. Default file is “cir.nd”. Specifying a command line file name which ends with a period followed by “nd” is equivalent to specifying the option. This option is usually used when the desired response file does not end with “.nd” or “.d”, for example, “-rAnswer.s2p”.
-g	Specify the file for circuit geometry. Default file is “cir.geo”. Specifying a command line file name which ends with a period followed by “geo” is equivalent to specifying the option. For example, “-gFile.xx”.
-a	Specify the file for the analysis parameters (usually frequencies for analysis). Default file is “ctl.an”. Specifying a command line file name which ends with a period followed by “an” is equivalent to specifying

the option. Similar to -r except the file “stdin” may be specified. If the option “-astdin” is specified or if the analysis control file does not exist, the user is prompted for a start, stop, and step frequency in MHz.

-v Verbose mode, display messages during program execution describing the current state of the analysis.

-j Outputs current density information for the entire circuit. Default file name is cir.jxy. Specifying a command line file name which ends with “.jxy” is equivalent to specifying the option. If the “.jxy” file already exists, it is renamed “.jxb” and output is sent to the new, empty, “.jxy” file. In this case, any previous “.jxb” file is lost.

-J Outputs current density file containing subsectioning information only. ***Emvu*** will show circuit as completely red, but can be useful to view subsectioning. Default file name is cir.jxy. If the “.jxy” file already exists, it is renamed “.jxb” and output is sent to the new, empty, “.jxy” file. In this case, any previous “.jxb” file is lost.

-x SPICE lumped model synthesis. Requires at least two frequencies of analysis. Output is a lumped model suitable for incorporating as a “.subckt” directly in a SPICE deck. If followed by a single number, 0-7, that number is taken as the number of digits to the left of the decimal place to be used for formatting lumped element values. Default is 1. Output file name is “spice.lc”, unless there is a file ending in “.lc” on the command line.

-X SPICE distributed N-coupled line R, L, C, and G matrix synthesis. The geometry must be an N-coupled line. Only a single frequency need be specified. Output are the L and C NxN matrices for the N-coupled line. If there is metal loss, R is also generated. If there is dielectric loss, G is also generated. Output file name is “spice.lct”, unless there is a file ending in “.lct” on the command line.

-N Calculates the number of subsections followed by an estimate of the number of Mbytes of memory required. The analysis is not performed.

-Q Suppress printing of the license. Use of this option indicates that you agree to the terms of the license.

-d The circuit is automatically de-embedded to the specified reference planes (see the geometry file description). This is the only case in which the reference plane information is used and is called automatic de-embedding. Specifying a command line file name which ends with a period followed by “d” is equivalent to specifying the option. For example, “em -vd thru.geo” places the automatically de-embedded S-parameters in “cir.d” while “em -v thru.geo thru.d” places the same results in “thru.d”.

-b Detects potential box resonances, prints out warning message just before S-parameter data in the output file.

-B Same as -b except, in addition, it tries to remove the offending waveguide mode. Bad data may result if used in conjunction with the -d option.

-e Results in a less accurate subsectioning, but faster analysis, in some cases. This option disables the detection of polygon edges on other than the present level for subsectioning purposes. If thin dielectric layers are used (for example, capacitor dielectrics), this option is not recommended.

-Ver Prints out the current version of ***em*** and license id and exits. Must be the first option on the command line. All other options are ignored.

The following files may be specified on the command line:

Analysis Control File:

Default name “ctl.an”, the file name should end with “.an”, otherwise, the file name must be specified with the -a option. This file can set the frequencies for analysis, among other things. If the file does not exist, the user is prompted for a start, stop and step frequency in MHz.

Circuit Geometry File:

Default name “cir.geo”, the file name should end with “.geo”, otherwise, the file name must be specified with the -g option. The circuit geometry file can be created with a geometry capture program such as ***xgeom***, see the ***Xgeom*** **User’s Manual**. If the circuit geometry file does not exist, ***em*** terminates.

Circuit Response File:

Default name “cir.nd”, the file name should end with “.nd”. If the file ends with “.d”, automatic de-embedding is enabled. Otherwise, the file name must be specified with the -r option. This file contains the N-port circuit parameters (e.g., S-parameters) of the circuit being analyzed. This file can be used for input directly to any of a number of microwave circuit analysis programs.

Current Density File:

Default name “cir.jxy”, the file name should end with “.jxy”. This file stores current density information on the file for later viewing. Analysis time is increased by this option and the file can take up a large amount of disc space if the circuit is large with many ports.

EXAMPLES:

em -v sscross.geo

Analyze the circuit in sscross.geo using the analysis control parameters in ctl.an and place the results in the file cir.nd. Print out the status of the analysis as it executes.

em -vmd sscross

Analyze the circuit in sscross.geo using the analysis control parameters in ctl.an and place the results in the file sscross.d. Print out the status of the analysis as it executes. Use a single precision matrix.

em -j sscross.d sscross.nd sscross

Analyze the circuit in sscross.geo using the analysis control parameters in ctl.an and place the de-embedded results in the file sscross.d and non-de-embedded results in sscross.nd. Also calculate current distribution and place in sscross.jxy. Do no print out the status of the analysis as it executes.

em -rcap.mim -yd

Analyze the circuit in cir.geo using the analysis control parameters in ctl.an with the results going to cap.mim. The output is to be in y parameters and automatic de-embedding is invoked.

em -j array

Analyze the circuit in array.geo using the analysis control information in ctl.an with the results going to array.nd and current distribution going to array.jxy.

em array.nd -d array

Analyze the circuit in array.geo using the analysis control information in ctl.an with the non-de-embedded results going to array.nd and de-embedded results going to array.d.

Appendix II The Analysis Control File Format

The analysis control file controls the frequencies used for analysis as well as other analysis parameters. The file may have any name ending with ".an". If the analysis control file name does not end with ".an", the file name must be specified with the -a option, for example: ***em*** -aanalysis.file. If no file name is specified in the command line, ***em*** looks for "ctl.an" in the current directory. If that file is not found or if the file is specified as "stdin" (with the ***em*** -a option), ***Em*** interactively asks the user for the frequencies of analysis.

Any line with a first non-space character of "!" is ignored. Any blank line is also ignored. Comments following any complete line of data are allowed.

In the keywords that follow, only the specified number of letters (3 or 4) are significant. Upper and lower case letters are allowed. Additional letters may be used but they do not alter the program's execution. For example, "VER", "VERSION", and "VERTRFGH" all have the same effect. There may be no more than 255 characters per line.

VER Any line with the first three characters "VER" is taken to specify the ***em*** version number. Up to five characters following the end of the first space characters following "VER" are read into ***em***. This information is required for compatibility with future versions. The VER line should be the first non-comment line in the file.

ANN When the response file is created, the analysis parameter file is searched for any lines beginning with ANN. The remainder of any lines found are listed in the heading of the circuit response file. This is useful for the automatic documentation of the ***em*** analysis output.

CYC The number of cycles for the modal summation may be specified on a line starting with "CYC". Default is 2. The number of cycles (1-5) must follow on the same line. If the number of cycles is outside of the proper range, it is limited to within that range. The analysis is slightly more precise with more cycles. The number of cycles can also be specified on the command line and if it is, it will take precedence over the specification in the analysis file.

GHZ
MHZ Frequency units can be specified by a line with "MHZ" or "GHZ". The frequency units can be changed as often as desired within the same analysis file. All frequencies specified on a FRE line must be in the units most recently specified. If no units have been specified, MHz is assumed. If no frequency units were specified in the command line, the frequency units in effect at the end of the analysis file is used for the circuit response file. Frequency units specified in the command line have no effect on how the analysis file is read.

FRE A line starting with FRE specifies a list of analysis frequencies. FRE may be followed by one, two or three numbers. One or two numbers specify one or two frequencies. Three numbers specify a start, stop and step. The start, stop and step are all checked for error conditions. When the file is complete, the frequency list is sorted. As many FRE lines may be used as is needed. Duplicate frequencies are not removed. There is no limit on the number of frequencies.

FMAX The subsectioning parameter "subsections/wavelength" normally uses the highest analysis frequency to determine the wavelength. If this line is present and followed by a frequency in the units most recently specified for frequency, that frequency is used for the wavelength determination. Thus, the same subsectioning can be used for several analyses which differ in the highest frequency being analyzed.

CMIN Followed by a number, specifies the smallest capacitor allowed for inclusion in a SPICE lumped model, in pF.

LMAX Followed by a number, specifies the largest inductor allowed for inclusion in a SPICE lumped model, in nH.

RMAX Followed by a number, specifies the largest resistor allowed for inclusion in a SPICE lumped model, in Ohms.

KMIN Followed by a number, specifies the smallest mutual inductance allowed for inclusion in a SPICE lumped model, dimensionless ratio

RZERO Followed by a number, specifies the resistor to go in series with all lossless inductors, in ohms. Needed for some versions of SPICE.

Appendix III The Circuit Geometry File Format

The circuit geometry file specifies the circuit geometry to be analyzed by ***em***. The geometry of the metallization is represented in terms of polygons. Any part of a polygon outside the box is ignored by ***em***. The coordinates of the polygon vertices are called out in terms of actual dimensions using floating point values. The polygons are automatically subsectioned by ***em*** for analysis. Polygons with dimensions smaller than the selected resolution can provide unreliable results.

The file can be created by means of a geometry capture program such as ***xgeom*** (see the ***Xgeom*** **User's Manual**). Other information, including the box dimensions and substrate parameters, are also specified. The experienced user may want to make minor modifications to a specific geometry file without going thru the geometry capture program. With this in mind, the file format has been set up to be as forgiving as possible, but keep in mind that it is still possible to corrupt the file in a manner which will not be detected by the analysis software. For this reason, only experienced users should attempt any modification.

The file may have any name ending with ".geo". Otherwise, the file name must be specified with the -g option, for example: "em -g circuit.file". If no file name is specified in the command line, ***em*** looks for "cir.geo" in the current directory. If that file is not found, ***em*** terminates with an error message directed to standard error (which is usually your computer screen).

Any line with a first non-space character of “!” is ignored. Any blank line is also ignored. Comments following any complete line of data are allowed. Comments are not allowed in the polygon vertex lists.

In the keywords that follow, only the specified letters (3 or 4) are significant. Additional letters may be used but they do not alter the program’s execution. For example, “VER”, “VERSION”, and “VERTRFGH” all have the same effect. There may be no more than 255 characters per line. The following description is for version 2.2.

VER Any line with the first three characters “VER” will be taken to specify the ***em*** version number. Up to five characters following the end of the first space characters following “VER” are read into ***Em***. This information is required for compatibility with future versions. The VER line should be the first non-comment line in the file.

LIC Followed by a string representing the license id number. This string is output to the heading of the circuit response file. If the value is unknown, then ***xgeom*** uses the string “UNKNOWN”.

DAT Followed by a string which represents the last date the file was changed. This string is output to the heading of the circuit response file.

ANN When the circuit response file is created, the circuit geometry file is searched for any lines beginning with ANN. The remainder of any lines found are listed in the heading of the circuit response file. This is useful for the automatic documentation of the ***em*** analysis output.

LEN The length units used in the file (see BOX below) are specified by a line beginning with LEN followed by two items. The first is up to six letters representing the unit. The last item is a conversion constant which, when multiplied by a length in the specified units yields meters. For example -- LEN mm 0.001. Only one LEN line is allowed per file and, if present, it must be before the BOX line. If not present, LEN mils 0.0000254 is assumed.

BOX The dimensions of the box and the parameters of the enclosed substrates are described starting with a line whose first non-space characters are “BOX”. BOX is followed by at least five numbers with two more optional. First is the number of metallization levels (= 1, if using standard microstrip). Next come the A and B dimensions of the substrate surface given in the units specified by the LEN statement. These are followed by the integer size of the substrate along the A and B dimensions. These integers must be even. The smallest possible subsection dimension (i.e., the resolution) is equal to two units in each dimension. The first optional number is an integer which sets the minimum density of subsections per wavelength. Default is 20 subsections per wavelength, with anything below 6 set to 6. The wavelength used is a simple average of the dielectric constant (and magnetic permeability) over all dielectric layers (i.e., constants are summed and divided by the number of layers, air included). The second optional number (following the subsections/wavelength parameter) is the Eeff for calculating the wavelength for satisfying the subsections/wavelength parameter. If not specified, or if it is less than 1.0, the parameter is ignored and the simple estimate of Eeff is used. Please note that you must specify the first optional number in order to use the second number.

The lines following BOX provide information on each of the dielectric layers in the box. Each line has the thickness of the layer (in the units previously specified) followed by the relative dielectric constant followed by the relative permeability of the layer. The dielectric loss tangent and the magnetic loss tangent follow. All items except the substrate thickness are optional. Default is lossless free space. Only one dielectric layer per line is allowed.

SYM If the circuit AND excitation are symmetric about the center line parallel to the X axis, a significant reduction in computation time can be realized. To indicate this, include a line with "SYM" as the first non-space characters. In this case, a mirror image of the geometry of the half substrate with Y coordinate less than the center line is used on both halves of the substrate. Even though the data for the large Y coordinate half of the substrate is ignored, it is recommended that the data be consistent with the small Y coordinate half of the substrate so that the automatic subsectioning may proceed correctly.

REF Each side of the Box may have a reference plane offset associated with it. REF, followed by up to four numbers, indicates the reference plane offset for each side. All ports on a given side have the same offset. The first number is the offset for the x=0 side followed by the x=A, y=0 and y=B side. The offset information is used ONLY if ***em***'s "-d" option (automatic de-embedding) is used. Reference planes not specified default to zero. The reference planes are floating point numbers, in the units specified by LEN. The actual reference plane is moved to the nearest even integer coordinate (see BOX). Four additional numbers are allowed. These numbers indicate the distance for removal of transverse subsections from each of the four sides. The numbers have the same format and order as the first four reference plane numbers.

RES Needed if metallization loss is to be included. Followed by up to three real numbers, the first is the resistivity (Ohms/square) at zero frequency (equals one over the bulk conductivity times metallization thickness). The second number (if present) represents the skin effect coefficient which is multiplied by the square root of frequency in Hertz to obtain the resistivity. For example, for ideal copper, 3 mils thick, use: "RES

0.0002 2.6e-7”. For equations to calculate these numbers, see section "Menu: Param - Metallization" on page 85 in the ***Xgeom*** **User’s Manual**. The third, if present, is the surface reactance in Ohms/square. There is one RES line per metallization type. An index is assigned to each metallization type starting with zero and increasing according to their order of occurrence in the file.

TOP
BOT

Same as a RES line (above) except that the top or bottom plate resistivity is specified. Up to one TOP and one BOT line is allowed per file.

REN
TON
BON

Specifies the name of a RES, TOP or BOT resistivity specification. The first item is an integer representing the pattern to be used in drawing any polygon assigned to the metallization. The entire remainder of the line is used as the name of the most recent RES, TOP or BOT line. For example: “RES 0 Cap bottom plate”.

POR

Followed by three required integers. The first is the polygon number within a specific level (see NUM), starting with zero. Count only polygons on the same level as the port. The port is located along the entire length of one line segment of that polygon, the second integer indicates the number of the first vertex of that line segment in the polygon (again, starting with zero). The third integer is the metallization level on which the port is located, starting with zero. A port is usually placed at the edge of the substrate so that a connection to ground is readily available (the sidewall). Otherwise a port is placed at the junction of two polygons (each polygon has a line segment colinear with the other). In this case, the two terminals of the port are formed by the two polygons. If there is no second polygon, ***em*** terminates stating that the port has no subsections.

The ports are numbered according to their order of appearance in the file unless an optional fourth integer follows which lists the number by which the port will be known. This number cannot be zero, but may be any other value. It need not be consecutive. If the index is the same as the index of any other port, the ports will be connected together. If the index is negative, it will be connected together with any other negative indices of the same value and be used as the ground terminal for that port. Four optional floating point numbers follow representing the impedance to which the port S-parameters will be normalized. The first is resistance (ohms), the second is reactance (siemens), the third is inductance (henries) and the fourth is capacitance (farads).

VIA

Now obsolete. Still recognized by all programs, but no longer generated. Use “EVIA”, which is identical to “VIA” except, in some cases, the location that the via is placed.

EVIA Followed by three required integers. A VIA line points to a polygon edge to which ***em*** will attach vias starting on the indicated metallization level and going up exactly one layer to the next metallization level. Polygons on the bottom cover (ground plane) are used only for vias attached to ground. A edge-via on the top metallization level attaches to the top cover. The three integers are identical to those for the POR line. The same polygon edge may be both a edge-via and a port. If a via is a port, it should be attached to bottom (or top) cover.

NUM Followed on the same line by a single required number, the total number of polygons in the file. The next line (blank lines and comments not allowed) describes the first polygon. The first number is the metallization level number (starting with zero). The second is the number of x,y point pairs. The third item is the metallization index, default is lossless (= -1). The fourth item is optional and indicates the edge fill mode. Default (or "N" or "n") results in a staircase fit to the edge of the given polygon. Specifying "T" or "t" allows diagonal edges (i.e., triangle subsections). "C" or "c" includes subsections which improve the representation of current at corners, important for gaps. A value of "B" or "b" includes both "T" and "C" above. Other values result in the default. Next, an integer follows, it is used for GDSII bug tracing. Four more numbers terminate the line, xmin, ymin, xmax, ymax. These integers indicate the smallest and largest subsection size for the polygon in terms of cells. Default is min = 1, max = 100.

The proper number of x,y point pairs follow, one pair per line and separated by blanks. The data are floating point numbers in the previously specified units (see LEN). The last point must be identical to the first point. Immediately following the last line is an END statement. Subsequent polygons are presented in an identical manner, no blank lines allowed.

Once a NUM line is encountered, the other keywords, described above, are not recognized.

Polygons which have edges in common are electrically connected. Polygons with an edge in common with the edge of the substrate are automatically shorted to ground on that edge, unless there is a port at that point. The polygons may overlap, however it is recommended that they do not. Likewise, a polygon may be complex (it crosses over itself), however this is of little utility.

Appendix IV Summary of Unix Commands

The following is a summary of some basic Unix commands. This chart is intended for the user who has no experience with Unix. We have included a list of MSDOS equivalent commands.

UNIX	MSDOS	MEANING
ls	dir	Print the names of all the files in the directory
ls -l	dir	Print the names of all the files in the directory and gives the last time the file was modified
cat *file*	type *file*	Print a file on the screen
tail *file*	-	Print the last 10 lines of a file
rm *file*	del f*ile*	Delete a file
mkdir *directory*	mkdir *directory*	Make a new directory
rmdir *directory*	rmdir *directory*	Delete a new directory
cd *directory*	cd *directory*	Change the default directory.
pwd	cd	Print working directory.

Notes:

1. Unix file names use “/” not “\”, which is used in MSDOS.
2. Command and file names are case sensitive, so the file “filter.geo” and “FILTER.GEO” are different.

Appendix V A Very Brief Introduction to Vi

The vi (visual) editor is a screen based editor available on all Unix systems. While there are certainly better editors (vi is still very good), once one knows vi, it may be used on any Unix system whatever.

In this appendix, we provide a minimal tutorial on the vi editor so that one may create a “ctl.an” file for ***em***. Vi is much more flexible than suggested here. If you find yourself doing significant work on Unix systems, we recommend learning vi in more detail.

To start, type:

vi ctl.an

If the file exists, it will be displayed at the top of your screen. In any case, lines on your screen which are past the end of the file are displayed with a “~” in the first character position.

Vi has an input mode and a command mode. Vi starts in command mode, any letters typed at the keyboard are treated as commands, not text. In command mode, input from the keyboard is not displayed on the screen.

Some of the commands (which are usually just single letters) place vi into input mode. In input mode, text typed at the keyboard goes into the file and is displayed on screen. When finished with input mode, pressing "ESC" returns vi to command mode. Pressing "ESC" in command mode makes a beep. If you forget which mode you are in, press "ESC" and you can be certain vi is in command mode.

With vi running, type "i". This command places vi in input mode. Now type the following text (in upper or lower case):

```
VER 2.4
GHZ
FRE 10.0
```

To correct errors while still on the same line, try "backspace" or control-H. If you have already pressed return, just keep going for now.

When finished typing, press "ESC". You are now back in command mode. If you used the caps lock key, be sure to take it out of caps lock. The commands are very case sensitive. A command typed in upper case has a different function from lower case. The problem is especially confusing because commands are not typically displayed on the screen.

To make corrections, in command mode, move the cursor to the point you want to correct. Try any arrow keys which may be on your keyboard. If you have no arrow keys or they don't work as expected, you can use the commands "h" (left), "j" (down), "k" (up) and "l" (right). If you make a mistake in typing a command, press "ESC" and start over.

To remove an unwanted character, type "x". To replace a single character, type "r" followed by the new character. To insert text before the cursor, type "i" (sound familiar?) and press "ESC" when finished. To delete an entire line, type "dd".

There are a lot more commands, but these should be enough for very simple editing. Before we end the description, one final command. To save your file and exit vi, type ":wq".

SUMMARY OF VI COMMANDS

i	Insert (enter input mode)
ESC	Leave input mode
h	Move cursor left one character
l	Move cursor right one character
j	Move cursor up one line
k	Move cursor down one line
x	Delete a character
dd	Delete a line
r	Replace a single character
:wq	Write out the file, then quit the editor
:q!	Quit the editor without making any changes to the file

INDEX

L

M

N

O

P

V

W

X

Y

Z