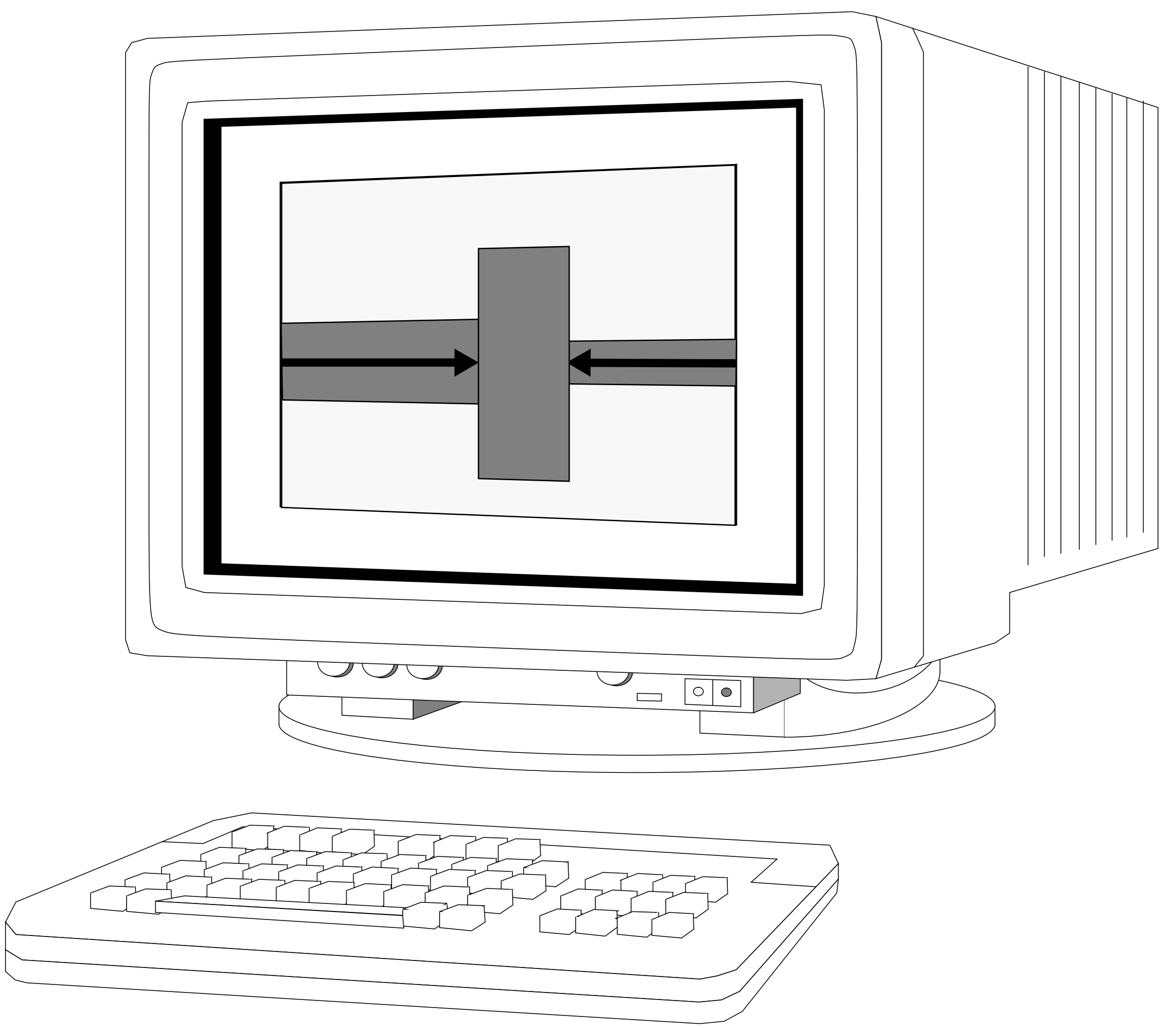
GDS Manual



GDS 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 GDSII Conversion

♦ Introduction

The ***gds*** command allows you to convert a GDSII file to a ".geo" compatible file. See your local circuit layout program on how to produce a GDSII file of your circuit. Once you have converted your GDSII file to a ".geo" format, you then need to tweak it using ***xgeom*** before you can analyze the circuit with ***em***. This document uses some advanced features of ***xgeom*** and assumes that the user is very familiar with release 2.2 or above.

♦ Changes in This Release

A few minor ***gds*** bugs have been fixed in this release. All other aspects of ***gds*** are identical to the previous release. A new section, called "Troubleshooting" , found on page 16, has been added to the manual.

♦ A Brief Overview

Gds needs two files as input: the GDSII file ("*.gds"), and a "layer file" ("*.lay"). The layer file tells ***gds*** how to map stream layers and data types to ***xgeom*** levels. ***Gds*** will then create an ***xgeom*** file ("*.geo") as output (see Figure 1).

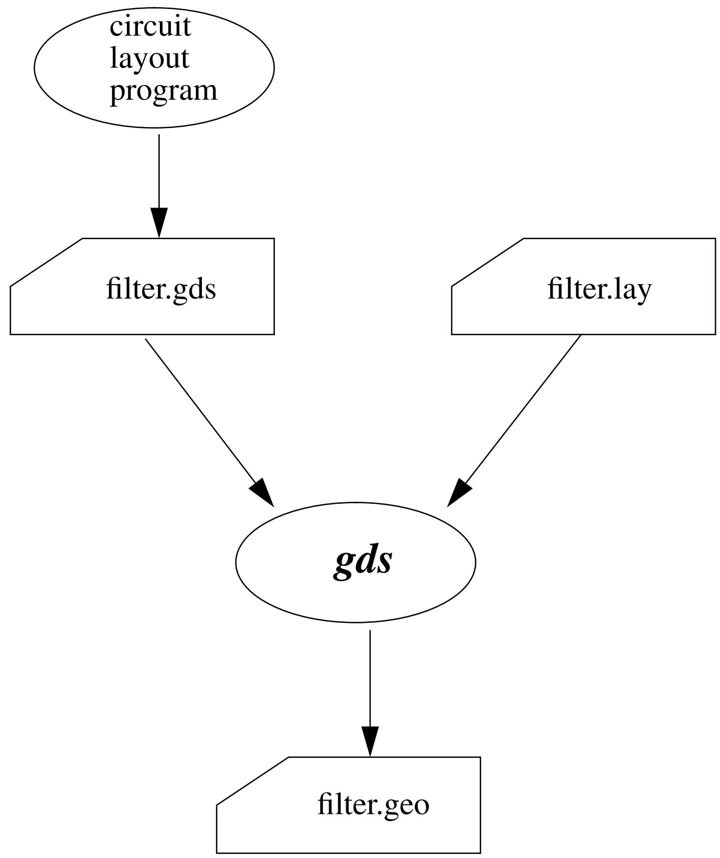


Figure 1 The gds program reads two files as input and creates an ***xgeom*** ".geo" file as output.

The ***gds*** program supports GDSII formats for release 3.0, 4.0, 5.0, 6.0.1 and 6.0.2.

♦ The Layer File

Before you convert the file, you must create a GDSII file from your circuit layout program. The file should have a suffix of ".gds", although it is not required. For example, say you have a filter that you want to analyze, you could call your file "filter.gds". Next you must create a file, known as the "layer file", to tell the GDSII conversion program the mapping of stream layers and data types to ***xgeom*** levels. To help you with this process, ***gds*** may optionally create a template file that can be used as a starting point. Type:

gds -L filter .gds

This command creates a layer file called “filter.lay”. Figure 2 shows an example of a Layer file created by the gds program.

```
#    Name        Purpose     Stream      Data      Geo      Metal
#                            Layer       Type      Level    Name
#                            Number      Number    Number
#
Not_defined   boundary   5          0         5        Unknown_metal
Not_defined   boundary   8          0         8        Unknown_metal
Not_defined   boundary   8          1         8        Unknown_metal
Not_defined   boundary   10         0         10       Unknown_metal
Not_defined   boundary   11         0         11       Unknown_metal
```

Figure 2 Sample Layer File

The above file maps Stream Layer Numbers and Data Type Numbers to Geo Level Numbers and Metal Names. You will probably want to taylor this file to your specific needs.

✧ Layer File Format

The following are the definitions for each field:

Field Name	**Purpose**
Name	A name you can give to the layer/data type pair. This field is not used by the software and is there to help you remember what the purpose of the field is. **Change this field.**
Purpose	Purpose of the layer. Should always be set to “boundary”. May have different values in the future. **Do not change this field.**

Stream Layer Number	The layer number as defined in your circuit layout program. Note that the layer number has no relationship to the xgeom's dielectric layer number or metallization level number. **Do not change this field**
Data Type Number	The data type number as defined in your circuit layout program. Each layer, defined by your circuit layout program, may have multiple data types. **Do not change this field.**
Geo Level Number	Xgeom's metallization level number. Top level is number 0. To put metal on the ground plane, for via placement, set the Geo level number to "gnd" or "GND". **Change this field.**
Metal name	Metal name being used, like copper, gold, etc. This is the same name being used in xgeom. You define the characteristics of the metal in xgeom. Do not call a metal "lossless", instead give this a different name and simply do not add loss to this metal. This gives you the option to add loss to the metal at a later time. **Change this field.**

Each field is separated by one or more tabs (or spaces). If you are not using the layer/-data type pair, comment out the line by putting a "#" at the beginning of the line. Any line beginning with a # is a comment. These layers are ignored in the conversion. Do not use spaces in the field. Use instead "_".

✧ Using Xgeom to Determine Level Mapping

If you do not already know the mapping of Stream Layer Number to Geo Level Number, then you may want to create an ***xgeom*** file using the template layer file that was created by using the -L option. To do this, type:

gds filter .gds

This command uses the "filter.lay" file to create a file called "filter.geo" that can be viewed by ***xgeom***. You can view all of the separate stream layers on separate ***xgeom*** levels. Use ***xgeom*** to view "filter.geo" and determine the mapping between layers to ***xgeom*** levels. It is quite possible that may stream layers are not needed by ***em*** and may be commented out.

✧ Modifying the Layer File

Now you should modify the layer file to meet your specific needs. Use any standard text editor, like ***vi***, to make the changes. See the section entitled "Layer File Format" on page 3 for the format of the layer file.

NOTE

The top level of your circuit is on level 0 in ***xgeom***. Next level below is level 1. (See Figure 3)

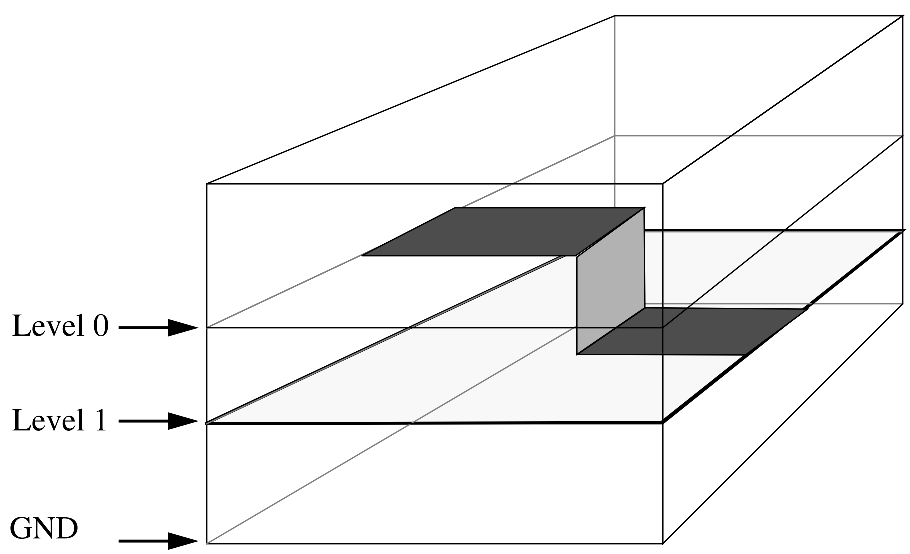


Figure 3 ***Xgeom*** Layer Numbering

Figure 4 shows an example of how the layer file could have been changed.

```
#      Name         Purpose     Stream      Data       Geo        Metal
#                               Layer       Type       Level      Name
#                               Number      Number     Number
#
top_plate       boundary    5           0          1          thick_metal
gate_metal      boundary    8           0          1          copper
#               The next layer includes only text
#Text           boundary    8           1          8          Unknown_metal
bottom_plate    boundary    10          0          1          thick_metal
air_bridge      boundary    11          0          0          gold
```

Figure 4 Sample layer file after modifications.

♦ Converting Your GDSII file

Now that you have a valid layer file, you are ready to convert the GDSII file. To do the conversion, type:

gds filter .gds

This is the same command that was typed before, but this time ***gds*** uses the updated layer file. The ***gds*** program reads both the “filter.gds” and “filter.lay” file to create a converted file called, “filter.geo”.

♦ Special Cases

✧ Reconversion

At times you may want to do another conversion of the GDSII file. For example, after you have done an analysis of your circuit with ***em*** you may have made extensive changes. You then update the circuit using your layout program and want to do another analysis using ***em***. To do this you must create a new copy of the GDSII file of your circuit then reconvert this. To reconvert the GDSII file to a “.geo” file, type:

gds -g filter .gds

The “-g” options tell the gds program to use data in the “filter.geo” file as input into the conversion process.

The GDSII file is converted again, but this time the program ***gds*** reads the already existing “.geo” file to determine the dielectric parameters, metallization parameters, box size, and cell size including the subs/lambda value.

✧ Using Your Favorite Layout Program to Replace *Xgeom*

You may use any layout program that writes GDSII files as a replacement for ***xgeom***. When you do this, you will probably want to convert the GDSII file without any translation in the coordinates. Normally, when ***gds*** converts a file, it moves the circuit to a convenient place for use in ***xgeom***. To convert a file without this translation, use the “-t none” option. This will ensure that the location of your circuit does not change when converted. For example, type:

gds -t none filter .gds

♦ Tweaking the Circuit

Now you are ready to tweak the circuit using ***xgeom***. To do this, type:

xgeom filter .geo

Some of the items still to be done are:

1. Define metallization and dielectric layer parameters.
2. Remove parts of the circuit not being analyzed.
3. Decide on a proper size substrate and cell size.
4. Change polygons to have the proper fill.
5. Align the circuit to grid points.
6. Move points around as needed.
7. Add vias.
8. Add ports and reference planes.

Depending on your circuit, all of these steps may not be needed. However, it is recommended that you try to follow the steps in the order given. The next section uses an example to describe each of these step in detail.

♦ A Simple Example

The example is found in the examples directory provided with your software and is called “dcblock.gds”.

To copy the example file into your account, type:

cp $SONNET_DIR/examples/dcblock.gds .

This file is a GDSII file that contains two circuits, a coupled line structure to be used as a DC block, and a bent transmission line (see Figure 5). The objective of this example is to convert the GDSII file into an ***xgeom*** file and to tweak the file so that the DC block may be analyzed in em.

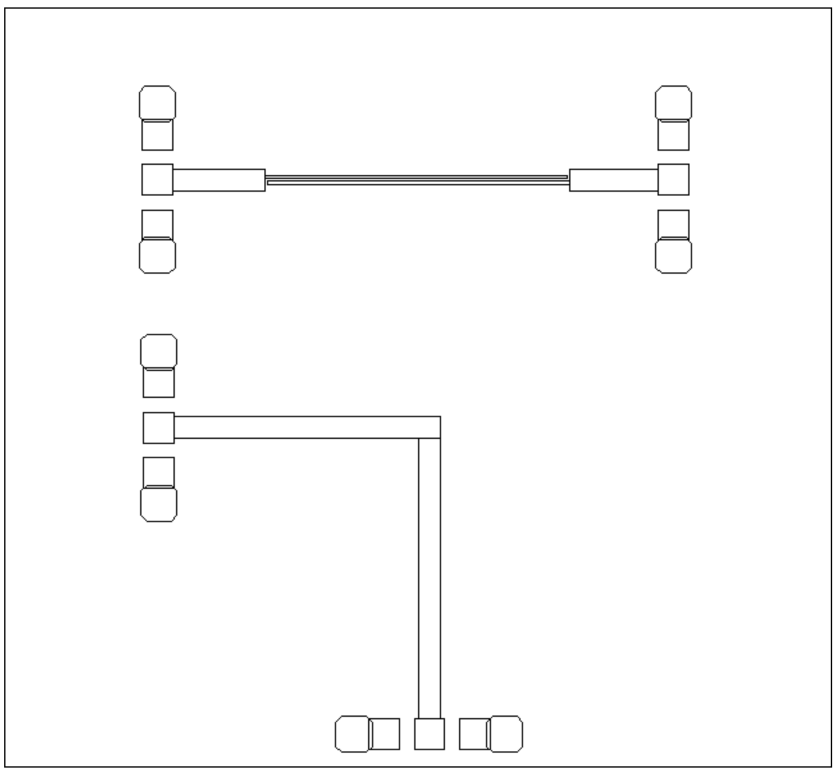


Figure 5 The GDSII example file contains two circuits. Our example uses the top circuit.

✧ Determine Level Mapping

First you must determine the mapping of GDSII layers to ***xgeom*** levels. To do this, type:

gds -L dcblock.gds

This will create an initial layer file, called “dcblock.lay”. You may now convert the GDSII file to an ***xgeom*** file by typing:

gds dcblock.gds

This command converts the file “dcblock.gds” to an ***xgeom*** file, “dcblock.geo” using “dcblock.lay” for layer information. Now you must decide which GDSII layer(s) are important. To do this, type:

xgeom dcblock.geo

and look at each of the ***xgeom*** levels. Since the cell size has not yet been defined, it may be difficult to see your circuit. Toggle the metal on and off at your discretion (using ^M) during the rest of this example.

Notice that there are many duplicate levels and many empty levels. Level 8 has all of the information that we need to do an ***em*** analysis. You should now edit the file “dcblock.lay” so that the conversion uses only this layer. To do this, use any text editor (such as vi) and comment out all of the lines except the line containing stream layer 8. Also, since our example is a single layer circuit, set the Geo Level Number to “0” (the first level in ***xgeom***). Your layer file should look similar to Figure 6.

```
#    Name      Purpose    Stream     Data     Geo      Metal
#                         Layer      Type     Level    Name
#                         Number     Number   Number
#
#Not_defined  boundary   7          0        7        Unknown_metal
circuit_layer boundary   8          0        0        Copper
#Not_defined  boundary   9          0        9        Unknown_metal
#Not_defined  boundary   10         0        10       Unknown_metal
#Not_defined  boundary   11         0        11       Unknown_metal
#Not_defined  boundary   12         0        12       Unknown_metal
#Not_defined  boundary   20         0        20       Unknown_metal
```

Figure 6 Layer file used for example. Changes from original file are shown in bold type.

Any line that starts with a “#” is a comment and is ignored by ***gds***.

To use the new layer file, type:

gds dcblock.gds

This command will create a new “dcblock.geo” file using the new layer file “dcblock.lay”. You are now ready to tweak your circuit. Type:

xgeom dcblock.geo

✧ Define Dielectric Layer and Metallization Parameters

Now define the metallization and dielectric layer parameters in ***xgeom***. The procedure is the same as making these changes for any normal “.geo” file.

In this example, set the top dielectric layer to 1000 microns of air (Erel=1.0) and the bottom dielectric to 100 microns of GaAs (Erel=12.9). Also, set the DC resistivity to 0.006 ohms/square and the skin effect coefficient to 2.6E-07.

✧ Remove parts of the circuit not being analyzed

Before you do the deletes and clips, you may want to save your file.

You may have converted a much larger section of the circuit than you want to analyze with ***em***. Thus you may want to do massive deletes on multiple levels. You may even need to delete entire levels using *Param* ⇒ *Dielectrics* ⇒ *Delete.* In our example, we used the layer file “dcblock.lay” to delete extra layers that we did not need. Either way is acceptable.

Use *Edit* ⇒ *Clip* to window out a part of the circuit you wish to analyze. To window out a precise area, you may need to turn the snap off before selecting *Edit* ⇒ *Clip*.

You can also delete several polygons at once by using *Select* ⇒ *Single Layer* or *Select* ⇒ *Mult. Layer* to select several polygons and points. Press the “Delete” key to delete the selected objects. Remember, any place where you can remove metal that does not effect your answer speeds up ***em*** execution time.

In our example, save your file and then use *Edit* ⇒ *Clip* as shown in Figure 7. The RF probe structure is calibrated out in the measurement so it should not be included in the ***em*** analysis.

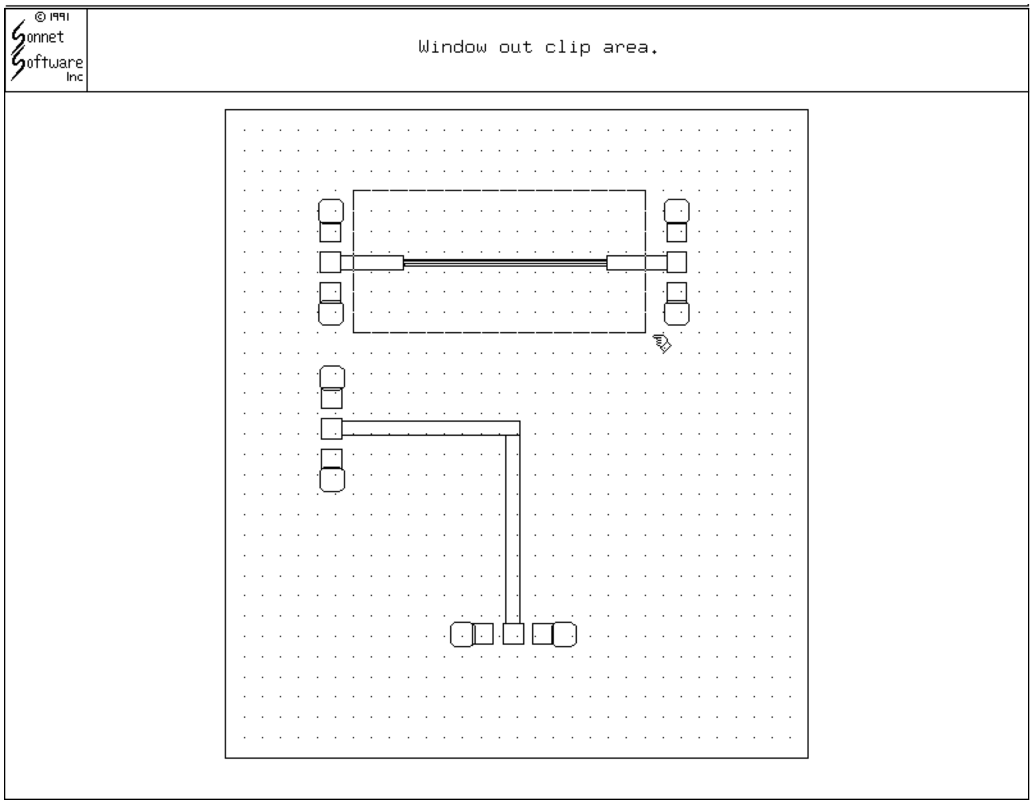


Figure 7 Clipping out the circuit.

✧ Decide on a proper size substrate and cell size.

Based on your geometry, you should decide on your cell size and substrate size. Remember to first set the area size to a multiple of the cell size. Since you have deleted a large portion of the circuit, you may now need to resize your substrate.

WARNING

Make sure that you are not zoomed into a small area before you resize your box. If after resizing your box, your circuit is outside the entire box, doing a *Zoom* ⇒ *Reset* may put your circuit off screen. The best way to ensure that this does not happen is to do a *Zoom* ⇒ *Reset* before changing your box size.

For our example, use *Param* ⇒ *Box* ⇒ *Area* ⇒ *Mouse* to window out a piece of the substrate (see Figure 8). Now use *Param* ⇒ *Area* ⇒ *Keyboard* to precisely set the x dimension of the box to 1800 microns and the y dimension of the box to 800 microns.

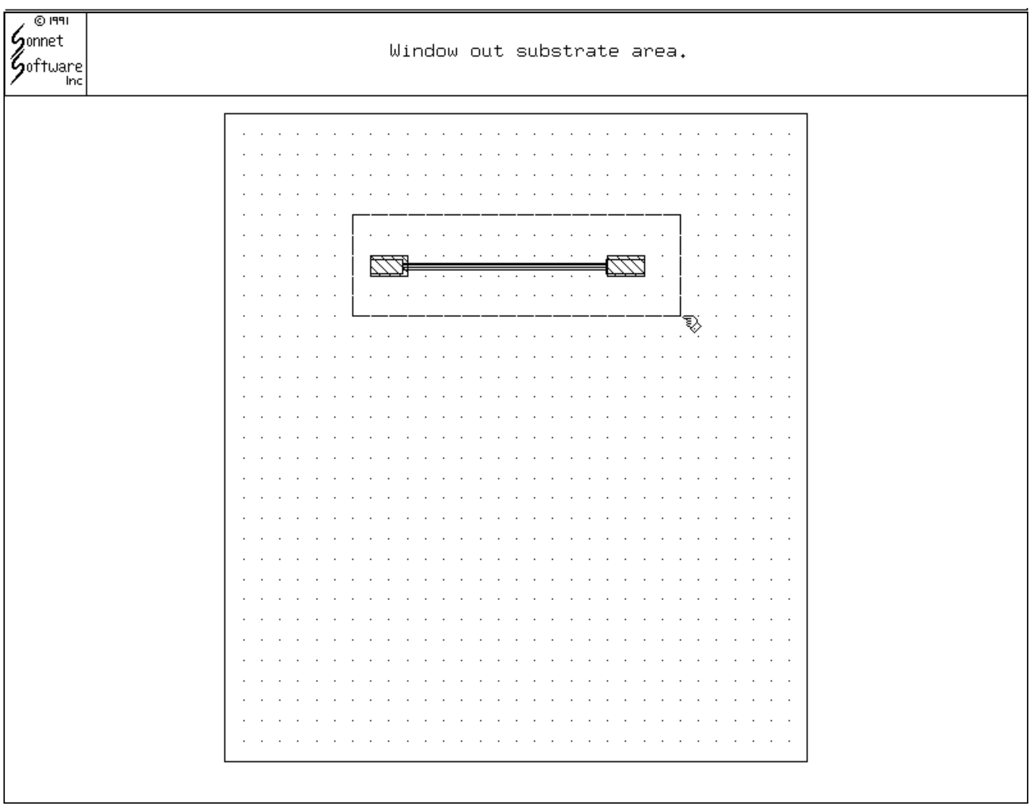


Figure 8 Using the mouse to set the substrate area to an approximate size.

Now set the cell size to 10 microns by 10 microns. Notice that we selected a substrate size that was a multiple of the cell size.

✧ Change polygons to have the proper fill.

Change the polygon fill (subsection type) if needed. For example, you may need to change several polygons from Staircase fill to Diagonal fill. To change several polygons at one time, you can select multiple polygons then use the *Objects ⇒ Polygons ⇒ Param*.

Our example does not need to have the polygon fill changed.

✧ Align the circuit to grid points.

Now check to see how ***em*** will subsection your circuit. To do this, make sure the metal is turned "On". The metal fill shows you how ***em*** will subsection your circuit. If you did not choose a small enough cell size or if your circuit is not oriented exactly on grid points, the metal may "short" two polygons together. This happens in our example (see Figure 9). The problem shown in the figure can be fixed by moving the entire circuit by a fraction of a cell.

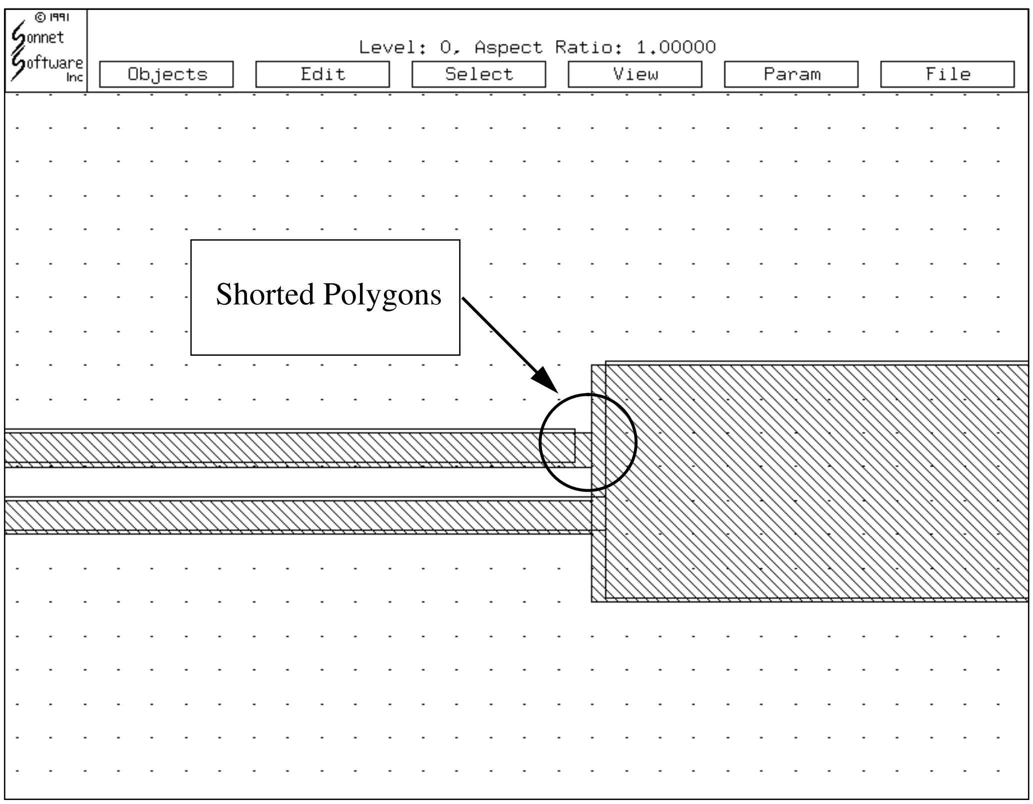


Figure 9 The example circuit has two polygons unintentionally shorted together.

For our example, turn the cell snap off, using *Param* ⇒ *Snap* ⇒ *No Snap*. Now, using *Select* ⇒ *All*, move your circuit so that the circuit is aligned to the grid points. Don't worry about getting the circuit exactly aligned. Notice that the polygons are no longer "shorted" together (see Figure 10).

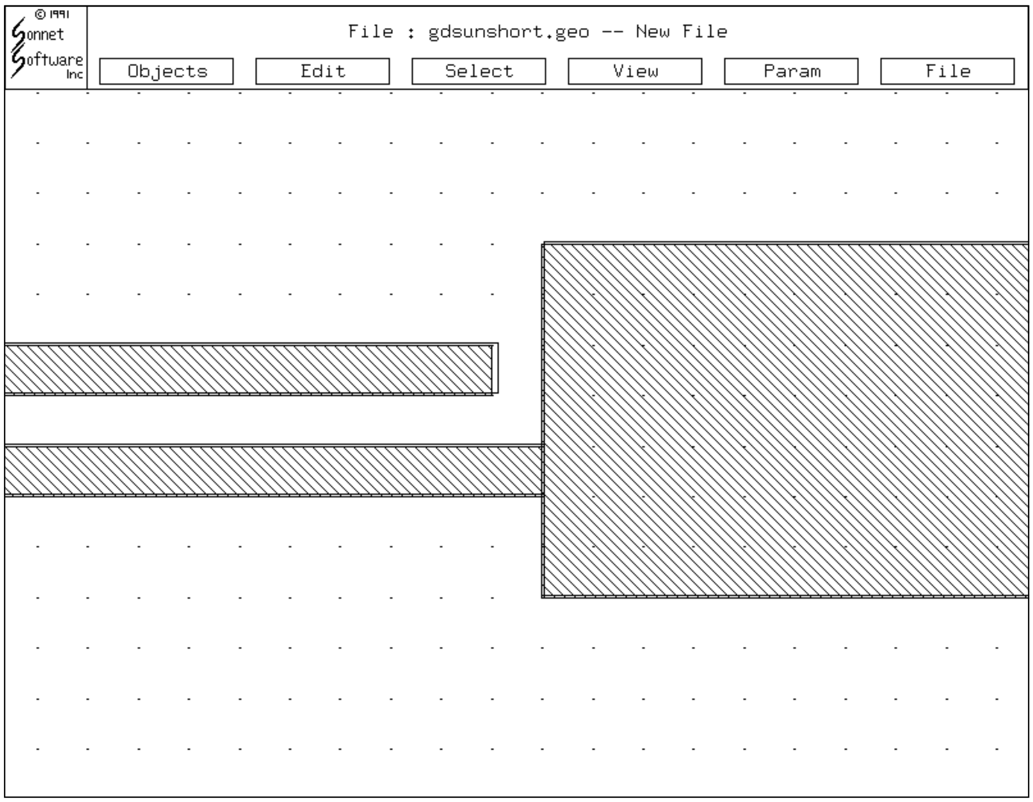


Figure 10 The entire circuit was moved closer to grid points. The two polygons are no longer shorted together.

At this point you may want to snap all of the polygons to grid points. Before you do the snap, you may want to save your file. For our example, first turn the snap on by using *Param* ⇒ *Snap* ⇒ *Cell Snap* then select all of the polygons. Now select *Edit* ⇒ *Snap*. If desired you can select only parts of the circuit to snap to the grid points.

✧ Move points around as needed.

Some points may now need to be moved. Pay particular attention to mitered corners and narrow lines and gaps.

Our example needs to be centered. Make sure the snap is still set to cell snap (*Param* ⇒ *Snap* ⇒ *Cell Snap*) and choose *Select* ⇒ *All.* Now move your circuit to the approximate center of your substrate.

✧ Add vias.

Next add vias as needed. The metallization being put down in a standard GDSII file may only represent the mask dimensions and layers and not the actual physical dimensions or location of the via. You may need to create each via by hand.

Our example does not need any vias.

✧ Add ports and reference planes.

To finish the example, add ports and reference planes. Typically this entails adding transmission lines to bring the circuit out to the edge. Figure 11 shows the finished circuit.

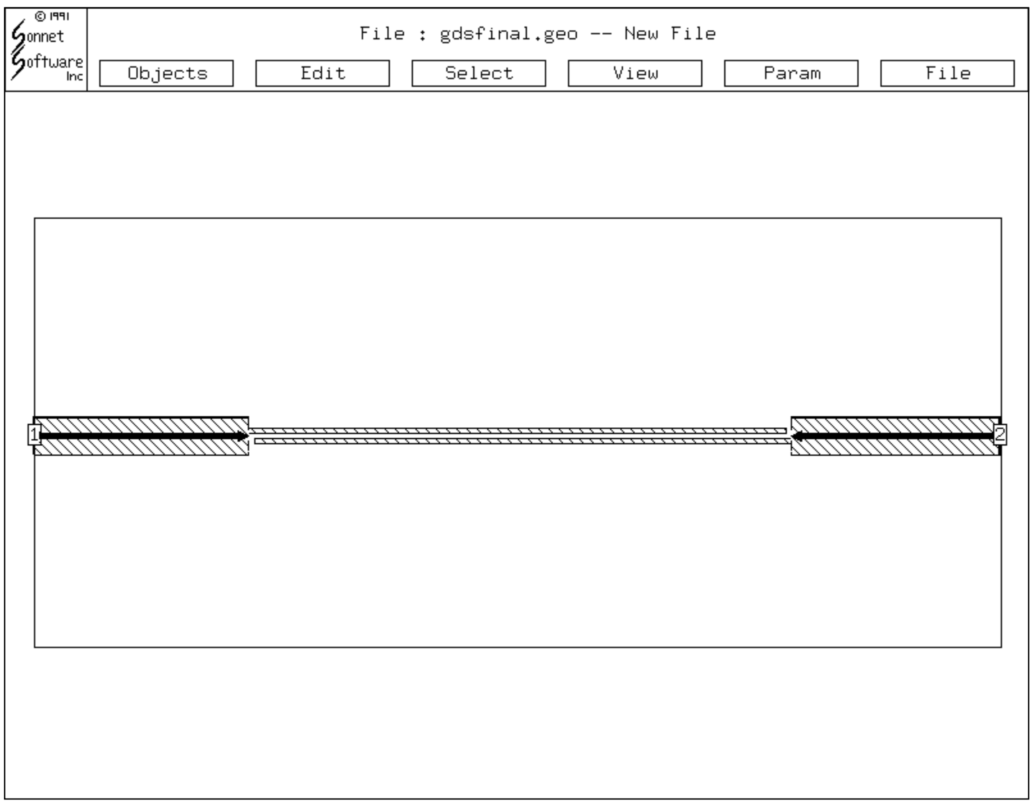


Figure 11 The final circuit.

You may now analyze this circuit using ***em***. It should have approximately 200 subsections if analyzed at 10 GHz.

♦ Troubleshooting

This section gives some procedures to trouble shoot through some simple conversion problems.

Error in Processing the GDSII File

You may receive an error message from ***gds*** stating that the format of the GDSII file is incorrect. The error message comes in many different forms. An example is:

```
ERROR: Encountered a record with the wrong data type
This record has a data type of NO_DATA
The data type should be TWO_BYTE_INT
Record type is HEADER
Error encountered in record 3
```

The most common reason for this message is that the GDSII file was changed when transferring between computers. This typically occurs when transferring the file from a PC to a UNIX workstation. The GDSII file is a binary file and not an ASCII file. Many who transfer programs between computers assume that you are transferring ASCII files and have to make changes to the files between the two computers. For example, MS-DOS and UNIX use different characters to indicate the end of a line in ASCII files. This is the reason that most UNIX computers have the commands “unix2dos” and “dos2unix”. These programs are used to convert ASCII files when transferring them between a PC and a UNIX computer. If a “dos2unix” conversion is done on a GDSII file, then it will be corrupted.

To determine if the GDSII file has been corrupted, determine how many bytes the GDSII file has on the original computer and also on the computer where the ***gds*** conversion is done. Both files should have the same number of bytes. When transferring files between computers, make sure that the transfer is using binary transfer mode. For example, FTP has two transfer mode: ASCII and binary. Also, do not perform a “dos2unix” conversion on the GDSII file if transferring the file from a PC to UNIX.

After conversion, *xgeom* file contains only part of circuit

You may have done a conversion but yet the ***xgeom*** file has no circuit or only part of the circuit. First verify that the layer file is correct. You may wish to recreate this file.

Another reason that only part of your circuit is shown in ***xgeom***, may be that only part of the GDSII file was converted. Every GDSII file contains structures. A structure may be referenced in another structure. The GDSII file may even have multiple structures that have no relation to each other. If a GDSII file contains multiple structures, then the ***gds*** conversion program must decide which structure to convert. Typically most layout programs create the GDSII file with the first structure as the parent structure. By default the ***gds*** program converts the first structure in the GDSII file. However, the layout program that you have may have created a GDSII file where the structure to convert is not the first structure. In this case you must tell the ***gds*** program which structure to convert.

For example for a file named “sprl.gds”, to get a list of structures type:

gds -d sprl.gds

An example of output may be:

```
Library: SPIRAL1
VIA
SPIRAL
PAD
```

In this case, from the layout program we may know that the SPIRAL structure references the VIA and PAD structures. The **gds** program would only convert the VIA structure. To convert the SPIRAL structure, type:

gds -p SPIRAL sprl.gds

Gds will now convert the entire SPIRAL structure instead of only the VIA structure.

♦ Options

Gds uses the following command line syntax:

gds *options* gdsfile

where *options* should be replaced with one or more of the options listed below, and gdsfile should be replaced with a valid GDSII stream file.

The following is a list of options available with ***gds***.

-t <type> Determines location of the circuit (See Figure 12). This option translates the gds coordinates, where <type> is:

none	The coordinates are used as is with no translation. This option is recommended when using another graphics program as an ***xgeom*** replacement.
min	The coordinates are translated so that the minimum X coordinate and the minimum Y coordinate are set to 0.
normal	The minimum X and Y coordinates are translated so that both are positive (**Default**).

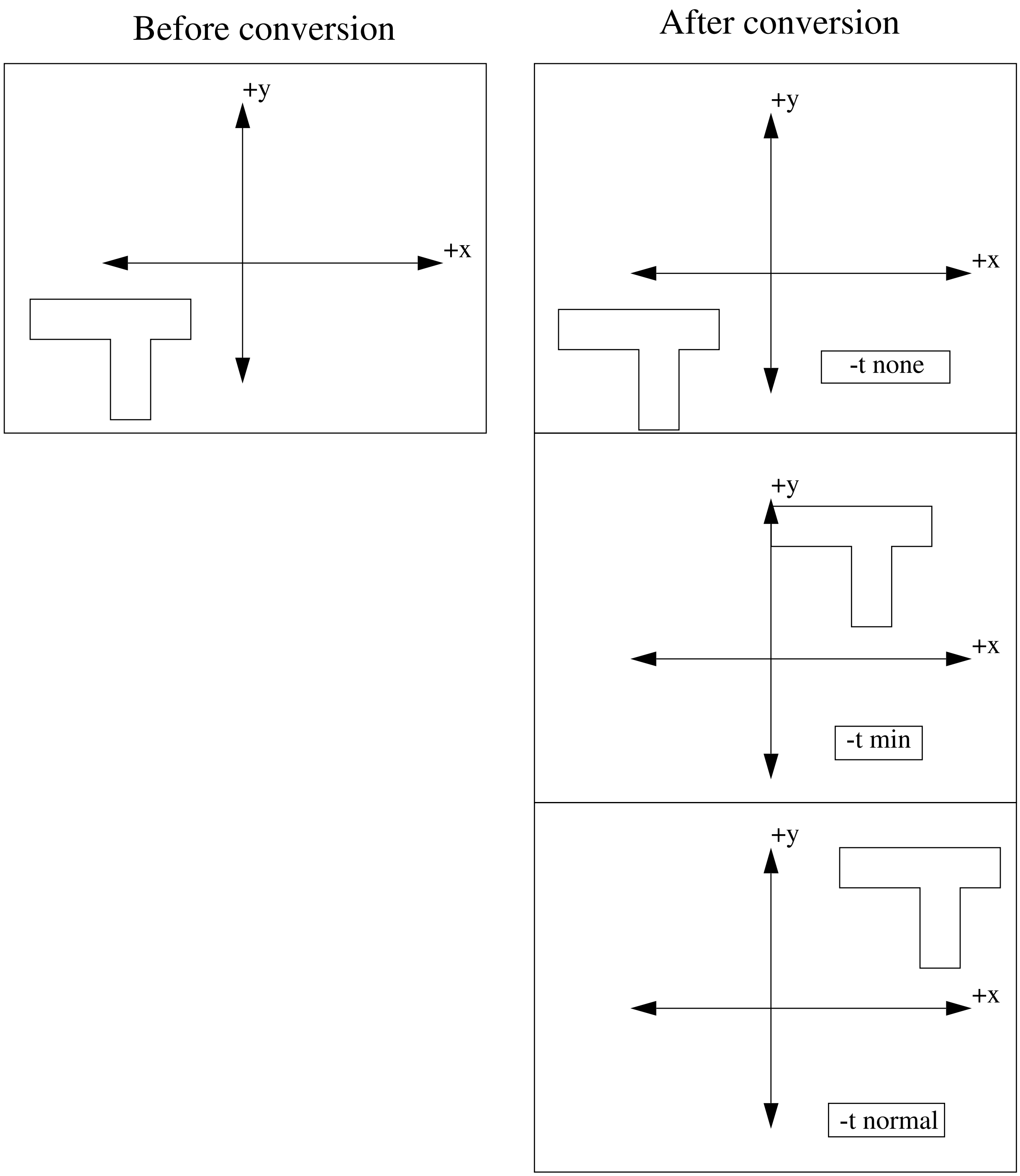


Figure 12 The "-t" option gives the user control over how ***gds*** transposes the GDSII coordinates.

-s <type> Determines the size of the substrate. The size of the substrate is independent of the location of the circuit. You could specify a substrate size so the circuit is completely outside of the substrate.

- max — Uses the maximum X and maximum Y coordinate of the circuit polygons as the size of the substrate.
- min — Makes the substrate minimum size. Specifically it makes the length of the substrate equal to the maximum X coordinated minus the minimum X coordinate. The height of the substrate is the maximum X coordinate minus the minimum X coordinate. May position circuit outside of box if used with the "-t normal" option.
- normal — Same as "min" except the substrate is made a little larger so that the circuit, if positioned properly, would not touch the edge of the substrate. **(Default)**

-d Lists out all defined structures in the GDSII file. Conversion is not done.

-L Creates a layer file. Conversion is not done

-p <struct> Converts <struct> structure instead of the default structure. See the -d option to list out your choices.

-n Prints out layers and data types that are used in the GDSII file but are not defined in the layer file.

-l <file> Use <file> as the layer file.

-f <file> Use <file> as the ".geo" file.

-o <file> Outputs the resultant conversion into <file>.

-V Prints the version number of the ***gds*** program.

-v Prints the version number of the GDSII file.

-g Used for reconverting a file. When this option is used, the box size, cell size, units, dielectric and metallization parameters are taken from the ".geo" file unless another option overrides this one.

-y Inverts Y coordinates before conversion.