microwave software®

SData+® User Manual

Scattering Parameter & Noise Data Enhancement Utility

S-Data User Manual
TABLE OF CONTENTS

SData+ User Manual Navigation Menu

INTRODUCTION

Program Overview

SData+is an interactive s-parameter and noise data enhancement & parameter calculation utility for the RF/Microwave engineer. It allows you to scan device data files, and then fit a 3rd degree natural spline function, a *french curve*, to either the s-parameter or noise data.

In addition to spline fitting the data, which is ***much*** closer than the *usual* linear interpolation type fit, you can perform a variety of calculations on the data itself, and then display the results on-screen.

Special Features

If you decide it is reasonable to make a unilateral assumption, because S12, the reverse transmission coefficient, is *small*, the **Calculations Module** will generate the impedance equivalents of S11 or S22, thus providing reasonably close approximations to the in/out impedance of the device.

If noise data is chosen for analysis, this module will calculate the conjugate noise impedance, i.e., the conjugate of the impedance equivalent of the optimum noise match reflection coefficient. This is what you would want to translate, via a match network, to 50 ohms, (or whatever), to create a proper noise match.

This module will ***also*** create .IMP impedance files for use by **SmithMatch**, our *on-screen* Smith Chart match network design program. In addition, it will create .CKT files for use by **Sceptre**, our frequency domain circuit analysis program. See Appendix C for information on these file formats.

Note: If you click on the link above to view Appendix C, use your browser ***BACK*** button to return here.

SYSTEM REQUIREMENTS

This program is for use on a Windows PC, where it will run *in a DOS window*. You will receive information on how to download a self-extracting file named SData+__INSTALL.exe via E-Mail, after making a purchase. The current size of the download file, as of the latest revision date, is *about* 222 KB. For our automated installer routine to work, you need to be using Windows 95, 98, 98SE, Me, NT, or XP operating in 32-bit mode.

To print to either a parallel port, or a USB printer in Windows, from within a program running in a small DOS window, you must first copy the data in the Windows Clipboard.

There are two ways to do this:

Method #1:
To print via Windows Paint, Wordpad, or Microsoft Word, from a small DOS window, use the two icons on the left, at the top. First, press 'Mark,' then left-click & drag the white cursor to highlight all the data you wish to transfer to the Clipboard. Now, with the data highlighted, press 'Copy' to paste in to the application of your choice, from which you may print.

Method #2:
To print via Windows Notepad, press the 4th icon from the left, at the top of the small DOS window, marked 'Full screen.' Now, press 'Alt + Print Scrn' to transfer the data to the clipboard. To return to the original small DOS window, press 'Alt + Enter.' Open Windows Notepad, press 'Edit' and then 'Paste.' Now you can print.

INSTALLATION

Introduction

This program is for use on Windows PC's, where the programs run *in a DOS window*.

When we receive confirmation of your order, from either Google, PayPal, or Kagi, we will send you an acknowledgement E-Mail. It will contain a software download link and your Registration Code. The current size of the "auto install" **SData+** file is *about* 222 KB. To "extract" the compressed files, you need to be using a 32-bit Windows PC.

Before the **SData+** program can be run, it must be *installed*. The installation does *NOT* write to your system registry, autoexec.bat, config.sys, or any *other* place on your PC. It is *totally* self-contained within a directory named c:\mwsoft\sdata.

Note: If you already *have* one or more of our software products, the directory c:\mwsoft will *already* exist. In that event, the installation routine will create sub-directories within c:mwsoft named c:\mwsoft\sdata and c:\mwsoft\sdata\mwdata6.

How to Install SData+ to a Windows PC

To begin the automatic installation process, double-click on the SData+__INSTALL.exe file, (its a black disk shaped icon), which you have downloaded to your desktop.

The install routine will first de-compress the files, create *three* directories (if needed), in your c:\ root, and then *transfer* all files to them. The main program file, named sdata.exe, is placed within c:\mwsoft\sdata, while all data and support files are placed in a sub-directory just *below* c:\mwsoft\sdata; its path is c:\mwsoft\sdata\mwdata6. The installer will *also* create an attractive desktop shortcut icon for your use.

The *first* time you run the program you'll be asked to enter your 17 character Registration Code. Be *careful* as it is a *mix* of upper/lower case letters and numbers, and is *case* sensitive. If you make an entry error, you'll be asked again to enter the code.

The **SData+** program is now *ready* to run!

How to Uninstall SData+

In Windows, using 'My Computer,' or 'Explorer,' navigate to c:\mwsoft. Now, if this is the ONLY program of ours that you have, delete the c:\mwsoft folder. If you also have *other* programs, and only want to delete this program, open c:\mwsoft and delete \sdata.

MAIN MENU

Introduction

The **Main Menu** is your *starting* point. From here you may select from two different types of calculations. One lets you sline fit your s-data, while the other lets you calculate a wide variety of useful data.

The **File Utility** allows you to edit your data, delete it, or create *new* data files.

The **Spline Fit** routine fits a 3rd degree natural spline function, a mathematical french curve, that *closely approximates* "the real world" at points ***in between*** vendor supplied scattering parameter data.

The **Calculations** routine allows you to perform several useful types of calculations to aid and speed along your design.

The **System File** keeps track of program usage, and sets a ***pointer*** corresponding to whether or not there is a system clock.

You may exit the program by simply choosing "**5**" from the **Main Menu**.

Note that all menu choices are made via "*hot keys.*" When you are at "**Choose (1-5) ? _,**" and type a number from **1** thru **5**, you will immediately ***go*** to the page selected; it is ***not*** necessary to press **<Enter>**.

Main Menu

(1) File Utility
(2) Spline Fit
(3) Calculations
(4) System File

(5) Quit

Choose (1-5) ?
_

The colored ***function keys*** along the bottom edge of the page are also "*hot*" in that, when you press one on your keyboard, the response will be immediate.

FILE UTILITY

Introduction

The **File Utility** allows you to perform a wide variety of data file tasks. You may list, print* (see bottom of page), create, delete, and edit .S2P device s-parameter and/or noise data files. While this utility is quite adequate for its intended task, it's *not* a full featured word processor; it doesn't have features like "search and replace," etc.

File Utility Menu

When you choose "**(1) File Utility**" from the **Main Menu**, either by pressing "**1**," or by using the "**F1**" function key, you'll enter the **File Utility Menu**. The screen display will look *similar* to the following:

File Utility Menu

(1) List File *
(2) Print Logfile *
(3) Create File
(4) Edit File
(5) Delete File
(6) Directory
(7) Main Menu

Choose (1 - 7) ?

*** Windows Print: Use (1) List File, then 'Mark' & 'Copy' into Clipboard.**
*** Print Logfile (2) will direct output to Logfile.txt in the \mwdata6 sub-directory.**

If you choose the option to 'Print Logfile,' the data will be directed to 'Logfile.txt' within the \mwdata6 sub-directory, and *not* to the screen. Right-click on 'Logfile.txt' to print it, and then DELETE the file; it will re-create when next needed.

You may make a selection from the **File Utility Menu** by entering a number from **1** to **7**. You *don't* have to press <**Enter**>; these are *hot keys* and will immediately *do* what you want.

Listing, Printing, and Deleting Files

These three file operations are *identical* in that you are asked *one* simple question: **Filename ? _**

Listing

If you want to ***list*** the **NE02135A** device data (demo) file that exists in the \mwdata6 data sub-directory, simply choose "**(1) List File**," then type **NE02135A**, and press <**Enter**>**.**

Printing.

On Windows PC's, follow the directions given on-screen and at the *bottom* of this page.

Deleting.

Simply choose "**(5) Delete File**" and press <**Enter**>.

Directory Listings

By choosing "**(6) Directory**," and pressing <**Enter**>, you will see a list of *all* .S2P device data files within the \mwdata6 data sub-directory. Initially, the *only* files you will have will be one that we've included for demo purposes.

Creating a .S2P Device Data File

When you choose "(**3**) **Create File**," and press <**Enter**>, you'll see a black DOS screen *similar*, except in color, to the following:

Note 1: Max # lines = 255.
Note 2: Enter s-data in the following format. Note that "<>" = a *space*.

F<>S11M<>S11P<>S21M<>S21P<>S12M<>S12P<>S22M<>S22P

Note 3: Enter noise data in this format. Note that "<>" = a *space*.

F<>Fmin.<>Gamma opt. Mag<>Gamma opt. Phase<>Rn norm.

If you *wish*, you may type comment lines before entering data, *BUT*, the line must
begin with an exclamation "!" character. See below under "Headers."

Note 4: Press "Enter" to end.

Enter line # 1 : ? _

In accordance with industry wide convention, s-parameter files carry the three-letter DOS extension of S2P. An s-data file *must* have ***nine*** entries per line; the first number is frequency, the next eight is the s-data in standard format: S11, S21, S12, and S22. Noise data files requires ***five*** entries per line. Let's create a *short* s-data file and save it in order to *see* how to do it, O.K.?

Also, keep in mind that while ***our*** programs use frequency in MHz, many vendors provide data in GHz. There is a routine in the **Calculations Module** that detects GHz vs MHz, and *asks* if you wish to *switch* so you can use the data with our other programs.

After we *create* this file, we'll show you how to edit it

In answer to the request above to "**Enter line # 1**," type the following at the prompt. Be careful to put only ONE ***space*** between each number:

Enter line # 1 : ? _ 100 .8 22 3.1 129 .01 -23 .7 45

As in all our programs, press **<Enter>** to complete the line entry. Now type the line as shown below:

Enter line # 2 : ? _ 200 .9 30 2.9 134 .02 -27 .9 51

Press **<Enter>** to complete the line entry.

Enter line # 3 : ? _

At this point, we'll *stop* entering data and save the file we've just created. To tell **SData+** that you're ***done*** entering data, just type one ***extra*** **<Enter>**.

You'll now be asked:

Enter Filename ? _

Let's call this simple two-line file "**TRY**." Just type it after the prompt *above* and press **<Enter>**.

You'll see a brief message on-screen telling you that the file "**TRY**" is being *saved* and

then you'll be returned to the **File Utility Menu**.

As you can see, its very ***easy*** to create files! If you want to create a noise data file, the steps are the ***same***, only you type ***five*** entries per line instead of ***nine*** as in s-data files.

Please note that lines must be entered in ***ascending*** order, frequency-wise, i.e., from low to high, and that the ***maximum*** number of lines in a file is 255.

Headers

If you want to enter a "header" into a file, when you begin to create one, each text line, even a blank one, ***must*** have an exclamation point "**!**" character at the ***beginning*** of each line. You might *want* a header in order to record device bias information, the device part #, etc. The following shows a short header added to our "**TRY**" file, the short two-liner we created earlier.

```
     ! FMT 2060 Data
     !
     100 .8 22 3.1 129 .01 -
23 .7 45
     200 .9 30 2.9 134 .02 -
27 .9 51
```

For now, let's look into editing a file.

Editing Files

When you choose "**(4) Edit File**" from the **File Utility Menu**, either by pressing "**4**" or by using the "**F4**" function key, you'll see the following:

Enter Filename ? _

Let's edit the short two-line .S2P data file we created. Type "**TRY**" to edit the file. Except for some of the colors, here is what you should see on the screen:

File: TRY
 # Lines : 2

Edit Menu

1: List File
2: Delete Line
3: Insert Line
4: Replace Line
5: Save File

```
      6: Menu

      Edit Command (1-6)
[<Enter>=Menu] ? _
```

It's always a good idea to choose "**1: List File**" first. However, in this case, we *know* that there are just *two* lines in this file, so we'll ***skip*** this step.

Choose "**3: Insert Line**" by typing "**3**" above and then pressing **<Enter>** (or just press the "**F3**" key).

You'll be asked ***where*** in the file you want to insert the ***new*** line:

Insert line between (A,B): ? _

If you want to insert a line at the ***top*** of the file, as in this case, enter "**0,1**" to tell **SData+** that the line is to go just *before* the *first* line. By the *same* token, if you were adding a line at the ***bottom*** of this two line file, you would type "**2,3**" to add the line in 3rd position just *below* line 2. Make sense?

Type the following at the prompt above:

Insert line between (A,B): ? _0,1 and press **<Enter>**.

Enter the following at the next prompt:

Enter new line: ?_ ! FMT 2060 Data

Note that the ***first*** character we typed above was the "**!**" ***exclamation*** symbol. As always, press **<Enter>** to complete the line entry.

Repeat this "**Insert Line**" operation once again, but *this* time insert a *blank* line between what is now "**1,2**." Don't forget, a blank line ***must*** start with a "**!**" symbol.

Finally, choose the "**5: Save File**" option and *save* the modified file using the name "**TRY**." Don't *worry* about typing in the .S2P three-letter extension, **SData+** will do *that* for you.

O.K., now when you choose "**1: List File**" you should see the modified file with the *header* in place:

```
     File: TRY

      ! FMT 2060 Data
      !
      100 .8 22 3.1 129 .01 -
23 .7 45
```

200 .9 30 2.9 134 .02 -27 .9 51

When you "**Press any Key**" as the message at the *bottom* of the screen asks, you'll be returned to the **File Utility Menu**.

We've now covered *everything* you need to know to work with files in **SData+**. Practice until you feel comfortable.

To print to either a parallel port, or a USB printer in Windows, from within a program running in a small DOS window, you must first copy the data in the Windows Clipboard. There are two ways to do this:

Method #1:
To print via Windows Paint, Wordpad, or Microsoft Word, from a small DOS window, use the two icons on the left, at the top. First, press 'Mark,' then left-click & drag the white cursor to highlight all the data you wish to transfer to the Clipboard. Now, with the data highlighted, press 'Copy' to paste in to the application of your choice, from which you may print.

Method #2:
To print via Windows Notepad, press the 4th icon from the left, at the top of the small DOS window, marked 'Full screen.' Now, press 'Alt + Print Scrn' to transfer the data to the clipboard. To return to the original small DOS window, press 'Alt + Enter.' Open Windows Notepad, press 'Edit' and then 'Paste.' Now you can print.

SPLINE FIT MODULE

Spline Fit Module

When you choose "**(2) Spline Fit**" from the **Main Menu**, either by pressing "**2**," or by using the "**F2**" function key, you'll enter the **Spline Fit Module**. The screen display will look as follows:

Spline Fit Module

Enter Filename [<Enter>=Quit] ? _

Let's *look* at the summary of a data file that was placed in the mwsoft\sdata\mwdata6 sub-directory during the installation.
At the prompt above, type "**NE02135A**" for the bipolar transistor data file we have in the\mwdata6 sub-directory, and press **<Enter>**:

Data File Summary

```
File  :  NE02135A

!
Filename:                    NE02135A.S2P   VERSION:
1.0
! NEC PART
NUMBER:  NE02135             DATE:       1/83
! BIAS CONDITIONS:      VCE = 10V.   IC = 5 MA
# GHz S MA R 50

SData:   9 Freqs   FLo = 0.100   FHI = 4.000

! NOISE DATA        2/81

NData:   7 Freqs   FLo  = 0.500   FHI = 3.500
                       Fmin = 1.200  Fmax = 3.700

                            - Press any Key -
```

A variety of useful information appears in the NEC Data File Summary on their low cost NE02135A device. For instance, you are shown both the s-data and noise header text, if they exist. This data often contains at *least* bias information. You can also see the number of data frequencies in the file. In this case, there are nine s-data frequencies over 0.1 to 4.0 GHz, and 7 noise frequencies over 0.5 to 3.5 GHz. Also, the min/max noise figure is given as 1.2 and 3.7 dB, respectively. If you wanted to add any *other* data of your own, you could use the **SData+** File Utility to do so.

When you "**- Press any Key -**" as instructed above, you'll be given the option to *view* the input data file. You'll be asked:

Display input file data (Y/N) [<Enter>=Quit] ? _

If you answer "**Y**" to the question above, here, except for the color, is what you'll see:

```
! FILENAME:       NE02135A.S2P  VERSION:
1.0
! NEC PART NUMBER:
NE02135      DATE:    1/83
! BIAS CONDITIONS: VCE=10V, IC=5MA
# GHZ S MA R 50
0.100    .84   -
36    13.82  156    .02   73    0.94   -18
0.500    .68  -
126     7.18  106    .08   35    0.51   -53
1.000    .66  -
```

```
163    4.02   81    .09   27    0.34   -66
1.500   .65   178    2.75   64    .10   27    0.31   -
74
2.000   .65   163    2.10   52    .12   30    0.31   -
83
2.500   .66   151    1.68   39    .13   26    0.31   -
95
3.000   .66   141    1.46   27    .14   26    0.33   -
106
3.500   .67   129    1.24   17    .16   26    0.36   -
116
4.000   .68   121    1.14    5    .17   23    0.38   -
127
! NOISE DATA     2/81
0.500   1.2    .36   69    0.14
1.000   1.5    .31   124   0.12
1.500   2.0    .50   165   0.05
2.000   2.4    .44  -175   0.06
2.500   2.6    .52  -161   0.10
3.000   3.6    .68  -141   0.14
3.500   3.7    .71  -139   0.21
```

If you just press <**Enter**>, you'll return to the **Spline Fit Module** *opening* prompt.

On the other hand, if you choose "**N**" up above and press <**Enter**>, you'll be able to *spline fit* data in the file, and you'll see the *following* on-screen message.

Spline fit S-data or Noise data (S/N) [<Enter>=Quit] ? _

Note: You're *seeing* the above message that refers to ***both*** s-data *and* noise data because the file NE02135A.S2P includes ***both*** types of data. If the file *ONLY* contained s-data, you would see:

Spline fit S-data (Y/N) [<Enter>=Quit] ? _

Or, if the file *ONLY* contained noise data, you would see:

Spline fit Noise data (Y/N) [<Enter>=Quit] ? _

O.K., we "diverted" a bit to pass along some information. Let's get back to where we were. We looked at the NE02135A input file, and then you were asked the question:

Spline fit S-data or Noise data (S/N) [<Enter>=Quit] ? _

Type "**S**" (to spline fit s-data) in response to the above question and press <**Enter**>.

Enter S-Data Output Freq Start, Stop, & Step ? _

If you *remember* from up above, the **Data File Summary** for the NE02135A device we listed above showed that it contained s-data over the frequency range of 0.100 to 4.000 GHz. Suppose we wanted to do a design for an amplifier working in L-Band over the range of 1 to 2 GHz? As it is, we *only* have s-data at ***three*** points, 1, 1.5, and 2 GHz. Our *first cut* design would likely be *closer* to optimum if we had ***better*** data. Let's spline fit data over 1 to 2 GHz in 0.1 GHz steps!

In response to the question above, about "**Start, Stop, & Step,**" enter the following and press <**Enter**>.

1,2,.1

Here is what you'll see:

Freq, S-Data Mag/Phase

1.000	**0.660**	**-163**	**4.02**	**81**	**0.090**	**27**	**0.340**	**-66**
1.100	0.661	-167	3.72	78	0.090	27	0.330	-67
1.200	0.659	-171	3.43	74	0.092	27	0.323	-69
1.300	0.656	-174	3.18	71	0.094	27	0.317	-71
1.400	0.653	-178	2.95	67	0.097	27	0.313	-72
1.500	**0.650**	**178**	**2.75**	**64**	**0.100**	**27**	**0.310**	**-74**
1.600	0.648	175	2.58	61	0.104	28	0.309	-76
1.700	0.647	171	2.44	59	0.108	28	0.309	-77
1.800	0.647	168	2.31	57	0.112	29	0.309	-79
1.900	0.648	166	2.20	54	0.116	30	0.310	-81
2.000	**0.650**	**163**	**2.10**	**52**	**0.120**	**30**	**0.310**	**-83**

- Press any Key -

Please note that is the *above* on-screen display, there are ***three*** lines of s-data that are ***highlighted*** in bold print. These lines are highlighted because they represent ***real*** data; and they ***exist*** in the vendor s-data file; all the *other* lines are *spline fit*.

Note: At this point, you'll be asked "**Print Logfile (Y/N) [<Enter>=No] ? _**" Please press <**Enter**>.

If you choose the option to 'Print Logfile,' the data will be directed to 'Logfile.txt' within the \mwdata6 sub-directory, and *not* to the screen. Right-click on 'Logfile.txt' to print it, and then DELETE the file; it will re-create when next needed.

You'll next see:

Write data to file (Y/N) [<Enter>=Quit] ? _

If you type "**Y**" and press <**Enter**>, you'll be asked to enter a *filename* for the data. Pick a name whose ***length*** is less than or equal ***eight*** characters, and press <**Enter**> once again. Only type a file *NAME* - do *NOT* add a three-letter extension! This data file will be *saved* in the c:\mwsoft\sdata\mwdata6 sub-directory. You'll be able to see it by choosing "**Directory**" from the "**File Utility Menu**."

If you respond to the "Write data" question with "**N,**" or just <**Enter**>**,** you'll be returned to the **Spline Fit Module** opening prompt.

CALCULATIONS MODULE

Calculations Module

When you choose "**(3) Calculations**" from the **Main Menu**, either by pressing "**3**," or by using the "**F3**" function key, you'll enter the **Spline Fit Module**. The screen display will look as follows:

Calculations Module

Enter Filename [<Enter>=Quit] ? _

Let's *look* at the file content summary of a bipolar transistor data file that was placed in the mwsoft\sdata\mwdata6 sub-directory during the installation.

At the prompt above, type "**NE02135A**" and press <**Enter**>. Here is, except for the color, what you will see:

File: NE02135A

This file contains 9 lines of s-data and 7 lines of n-data.

Perform 'S' or 'N' Calculations (S/N) [<Enter>=Quit] ? _

Be ***careful*** above. You must make a *conscious* choice and choose either "**S**" or "**N**." If you *accidentally* press **<Enter>** you'll be back at the **Calculations Module** opening prompt.

Gain/Stability Data

In this instance, type "**S**" and then press **<Enter>**. Here is what you'll see:

Gain/Stability Data in a 50 Ohm System

Note: GT Max (dB) only defined for K>=1

File: NE02135A

Freq	K-Factor	B1	GT Max (dB)	MSG (dB)
0.100	0.023	0.220		28.395
0.500	0.368	1.057		19.530
1.000	0.664	1.288		16.500
1.500	0.890	1.318		14.393
2.000	0.960	1.324		12.430
2.500	1.075	1.338	9.441	
3.000	1.125	1.322	8.030	
3.500	1.099	1.305	6.974	
4.000	1.069	1.297	6.656	

- Press any Key -

Unilateral Assumption

After you "**- Press any Key -**" as requested above, you'll see the following information about the data just presented, and be offered a suggestion about when you may make a unilateral assumption.

This device is unconditionally stable at 4 frequencies. It has useful gain at 4 frequencies.

NOTE: Given that K>=1 and B1 is positive at these points, you MAY, (if S12 is
small), make a unilateral assumption and compute the impedance equiv. of S11

and S22. These impedances may then be used to create SmithMatch .IMP load
files. Take care to insure stability outside this band. Our Sceptre program
will aid you in the calculation of stability circles using the `S2' option.

- Press any Key -

Here, we are talking about the parameter S12, i.e., the reverse transmission coefficient, as possibly being ***small***, and say that if it ***is*** small, S11 and S22 then become fair approximations to the devices in/out impedances. But, how *small* is *small* anyway? In general, we've found that is the magnitude of S12 is ***less than*** ~ 0.1, the assumption is reasonable.

Sceptre .CKT Files

This time, if you "**- Press any Key -**" above, you'll see:

Would you like to create a circuit file for use with our Sceptre program
(Y/N) [<Enter>=Quit] ? _

Be ***careful*** once again in your response to the above question. If you choose "**Y**" and press **<Enter>,** (or just use the "**F9**" function key), you see the following:

NOTE: All Sceptre files will be directed to the \mwsoft\sceptre\mwdata2 sub-directory with the .CKT extension.

Enter Filename [<Enter>=Quit] ? _

Frequency Units

After typing in the filename of your choice, ***8 characters max.*** please, you'll be told:

The data in this file appears to be in terms of GHz. If so, would you like to convert
the data to MHz for use with Sceptre (Y/N) ? _

Since our **Sceptre** program requires that frequency data be in MHz, you should type "**Y**"

and press **<Enter>** (or use the "**F9**" function key). You'll See:

Data file saved.

If you use **SData+** to create files for use with *other* analysis programs, you might want to leave it in terms of GHz. That is your choice, and now you've seen ***how*** to choose.

Unilateral Impedance Data

Then, *Yes* there's ***more***, <g> you'll be asked:

Would you like to see unilateral impedance data
(Y/N) [<Enter>=Quit] ? _

If you choose "**Y**" and press **<Enter>** you'll see:

Choose S11 or S22 impedance data (11 or 22)[<Enter>=Quit] ? _

Type "**11**" and press **<Enter>**. After a question about *printing*, which Users should ***ignore*** by pressing **<Enter>**, you'll see the following tabulation. This data represents the impedance equivalent of S11 *after* having made a unilateral assumption for the NE02135A device.

Unilateral device input impedance in a 50 Ohm System

File: NE02135A

Freq	Real	Imag
100.000	42.488	-142.513
500.000	11.884	-24.323
1000.000	10.460	-7.152
1500.000	10.609	0.833
2000.000	10.832	7.129
2500.000	10.895	12.354
3000.000	11.465	16.874
3500.000	12.021	22.716
4000.000	12.428	26.949

- Press any Key -

SmithMatch .IMP Files

Now, if you "**- Press any Key -**" above, you'll see:

```
      Would you like to create a load impedance file for
use with our
      SmithMatch program  (Y/N) [<Enter>=Quit] ? _
```

Note: Please keep in mind when you create a .IMP file for use with **SmithMatch**, that it ***cannot*** have more than 10 lines of data in it!

Be ***careful*** again in your response to the above question. If you choose "**Y**" and press <**Enter**> you'll see:

```
 NOTE: All SmithMatch files will be directed to the
\mwsoft\smithmat\mwdata1 sub-directory  and given the .IMP extension.

      Enter Filename [<Enter>=Quit] ? _
```

After typing a name, (8 characters or less), you'll see "**Data file saved**" and be returned to the **Calculations Module** opening prompt.

We've gone thru a highly *detailed* scenario working with s-data. We could, (and you can), work thru the *same* type thing with the noise data in the NE02135A data file.

To print to either a parallel port, or a USB printer in Windows, from within a program running in a small DOS window, you must first copy the data in the Windows Clipboard. There are two ways to do this:

Method #1:
To print via Windows Paint, Wordpad, or Microsoft Word, from a small DOS window, use the two icons on the left, at the top. First, press 'Mark,' then left-click & drag the white cursor to highlight all the data you wish to transfer to the Clipboard. Now, with the data highlighted, press 'Copy' to paste in to the application of your choice, from which you may print.

Method #2:
To print via Windows Notepad, press the 4th icon from the left, at the top of the small DOS window, marked 'Full screen.' Now, press 'Alt + Print Scrn' to transfer the data to the clipboard. To return to the original small DOS window, press 'Alt + Enter.' Open Windows Notepad, press 'Edit' and then 'Paste.' Now you can print.

PRODUCT SUPPORT

If you think of something we could *add* to this program to make it *better*, please let us know. If we *use* your suggestion, you'll get credit for it, and our sincere thanks. We'll also send you a free up-grade!

Product Upgrades

From time to time we *up-grade* our software. We do this so we can offer a *better* product, ***and*** because *we* use it ourselves, on a day to day basis, in our RF/Microwave engineering consulting practice.

As a *registered* User of our software, when an up-date becomes available, its FREE, and we'll post information about it on our web site.

Technical Support

We will help you with problems involving the installation and use of our software, via e-mail, and free of charge.

If you wish *design assistance* with a task you're working on, tell us about it, and we will give you a fixed price quote for a specific task.

Contact Technical Support

On-Site Support

We can provide on-site support ***and*** training, at ***your*** facility, for all of our software products. When we say support, we ***mean*** support. We'll do a lot *more* than just show you how to install the software, we'll walk you thru trial designs. When we leave, you'll have a good idea as to *how* to design an amplifier, compute line dimensions for a directional coupler, or *how* to synthesize a match network for an antenna, etc. Contact us for availability and details.

Contact Sales

Software "Bugs"

Bugs happen, even to the best of us. Usually, they come about as a result of a software change that wasn't thoroughly "beta tested,' (our fault), ***or***, because ***you*** did something that none of us ever *thought* you might do, (our fault again). If, by chance, you should uncover a bug, please let us know immediately. We'll correct it, up-date your software, and offer our apologies.

APPENDIX A

S-Parameter Data Files

SData+ can read *most* vendor supplied device *two-port* data files having the standard .S2P three-letter DOS extension. These are formatted as standard ascii text files. They can, if you're *careful*, be created or edited with a program like Windows Notepad. The proviso is that the edit utility can ***not*** insert any control characters, etc. in to the file. Notepad ***meets*** this requirement.

These two-port .S2P vendor files may contain device s-data, noise data, or both. An *example* of a file that contains ***both*** is the NE02135A sample data file that was included with you installation.

Certain formats have become *common* in the industry, and **SData+** can *read* them, or at least the ones we show below. Knowing this, any files ***you*** create should *follow* accepted format.

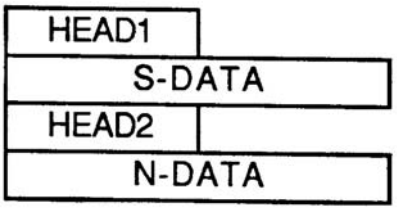
In addition to actual measured device data, vendor files frequently contain ***header*** text. This type text is added to identify, or in some way clarify, *how* a device should be biased or connected. It may be as simple as just the type number of the device, or it may include details about bond wire length, etc. Given that a vendor file could contain ***both*** s-data and noise data, it may have ***two*** blocks of header text.

In an s-data file, each line has *nine* entries per line, while noise data has *five* entries per line. The first entry on any line is ***always*** the measurement frequency. The frequency *may* be either in terms of MHz ***or*** GHz.

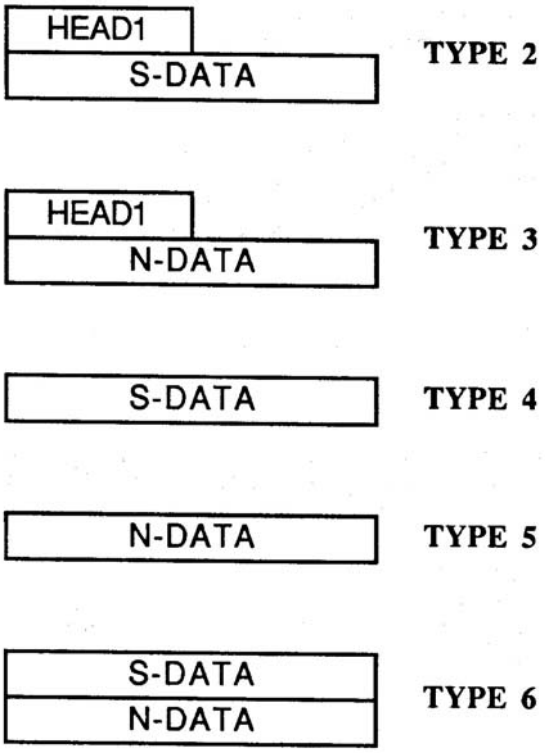
This program does *not* work with four-port files which are those having a .S4P three-letter DOS extension.

Standard Format

There are ***six*** basic file structures that we're currently *aware* of, and that **SData+** recognizes. Each of the six types is shown in a *sketch* below:



TYPE 1



Each line, even a ***blank*** line, in a header text block ***must*** have a "**!**" exclamation character in *first* position on the far left.

From the descriptions of the various formats shown above, you'll recognize that the NE02135A device we've been *working* with is a TYPE 1.

The TYPE 1 file structure we've been using is the ***maximum*** sized file allowed. This max type file has ***four*** blocks of *mixed* text and data. A proviso here is that data ***cannot*** be *interleaved*, i.e., you cannot ***mix*** lines of s-data with noise data. Each type data must be in a block by itself, and the frequencies in each ***must*** be in ***ascending*** (low to high) order.

If you try to use **SData+** to read a file that is *different* from one of the *six* types shown above, you'll either get a *non-standard file* message on-screen, ***or*** the file may be read incorrectly.

If you ***do*** come across a *different* type file structure than the six we currently support, please let us know in detail. We'll add it to our list and give you a **f r e e** up-grade!

APPENDIX B

Directories & Files

Our installation software creates ***three*** directories for use by the ***first*** program of ours that you install. The *top* directory, just off the root, is C:\MWSOFT, the second is the program folder \SDATA, and the last, (see about 14 lines below), is \MWDATA6. The tree structure is like what's shown below: However, once you have *one* of our programs, each additional install only creates a *program folder*, and a *datasub-directory* for that software program.

So, if you have all *six* programs, you will have one main directory, six program folders, and *six* sub-directories.

Here are the *names*, below \MWSOFT, of our various program folders and their corresponding data sub-directories:

\SMITHMAT	\MWDATA1	SmithMatch
\SCEPTRE	\MWDATA2	Sceptre
\MSTRIP	\MWDATA3	MStrip+
\OPTIMATC	\MWDATA4	OptiMatch
\UTILITIE	\MWDATA5	Utilities+
\SDATA	**\MWDATA6**	**SData+**

In the above sketch, your hard drive "Root" is C: Below, (or aside it) are your system, program, and User created directories. We use the \MWSOFT directory for *all* of our main program .exe files. Its "path" on your hard drive is c:\mwsoft.

In the sketch, only *one* sub-directory named MWDATA is shown for simplicity. In reality, we use a separate data directory for each of our programs. If you have all *six* programs, you will have *six* sub-directories.

A significant advantage to this directory scheme is that, to access a file, you do not need to type a path, just a file name. Each of our programs "knows" that the data files are one level below the program folder.

Support Files

For **SData+**, support files are transferred into the \MWDATA6 sub-directory during the installation. The use of such files varies from program to program. **OptiMatch** uses three, while **MStrip+** uses only one.

These files perform a number of functions. At a minimum, they record program usage and whether or not you have an on-board clock. In the case of **SmithMatch**, this file records **AR**, your screen aspect ratio. In **OptiMatch**, the files store your preferred unit set, and changes made to algorithm variables.

.IMP Files

These are impedance files that are used by **SmithMatch**, **Sceptre**, **OptiMatch**, and the **SData+** programs. They are in simple ascii text format and contain three entries per line. These are frequency, Re (Z), and Im(z). A maximum of *ten* lines of data are allowed.

.CKT Files

This file type is *only* used by **Sceptre**, our frequency domain analysis program. Like a .IMP file, it is in simple ascii format. However, unlike a .IMP file, it can have any number of lines, and each line carries more data. Parameters like frequency, an s-data matrix, stability factor elements K & B1, gain, etc.

.S2P Files

This file type is *only* used by **SData+**. It's a *special* type widely used in the industry and may contain s-data, noise data, or both. It was discussed in length in Appendix A.

APPENDIX C

Special .IMP/.CKT Files

The three-letter filename extensions **.IMP** and **.CKT** stand for, respectively, **impedance** and **circuit**. These files can be created by the **Calculations Module**.

The .IMP extension is *automatically* appended to files created by **SmithMatch**, our Smith Chart program, and also by **OptiMatch**, our match network optimization program. **Sceptre**, our frequency domain circuit analysis program can write a .IMP file for use by **SmithMatch**. It is in this way that our programs form a *suite*; they can ***talk*** to each other.

One proviso on files being created for use by either **SmithMatch** or **OptiMatch** is that each can have no more than ***ten*** lines maximum in it.

.IMP File Format

A .IMP file contains ***three*** entries per line. The three entries are, from left to right, frequency, Re(Z), and Im(Z). We used the NE02135A.S2P bipolar transistor data file in the **Calculations Module** to compute the unilateral input impedance of the device. We then created a .IMP file. The file created by **SData+** looks like this:

```
      100.000    42.488    -
142.513
      500.000   11.884     -
24.323
     1000.000   10.460      -
7.152
     1500.000   10.609       0.833
     2000.000   10.832       7.129
     2500.000   10.896      12.354
     3000.000   11.465      16.874
     3500.000   12.021      22.716
     4000.000   12.428      26.949
```

If the file is to be used by either **SmithMatch** or **OptiMatch**, it is limited in length to ***ten*** lines, and the frequency term ***must*** be in units of MHz.

When we *created* this file in **SData+** and went to save it as a .IMP file, **SData+** caught the fact that the frequency units were in terms of GHz, and asked if we wanted to *convert* them to MHz, to which we replied "**Y**."

.CKT File Format

A .CKT file contains ***nine*** entries per line. The nine entries are, from left to right, frequency, S11 Mag/Phase, S21 Mag/Phase, S12 Mag/Phase, and S22 Mag/Phase.

As we did with a .IMP file above, we used the NE02135A data file and opted to *create* a .CKT file to use with **Sceptre**. The file created by **SData+** looks as follows:

```
    1 TWO  AA  S1  50
    2 PRI   AA  S1  50
    3 END
    4 DATA  100  500  1000  1500  2000  2500  3000  3500  4000  -
1
    5 DATA  0.840  -36  13.82  156  0.020  73  0.940  -18
    6 DATA  0.680  -126  7.18  106  0.080  35  0.510  -53
    7 DATA  0.660  -163  4.02  81  0.090  27  0.340  -66
    8 DATA  0.650  178  2.75  64  0.100  27  0.310  -74
    9 DATA  0.650  163  2.10  52  0.120  30  0.310  -83
    10 DATA  0.660  151  1.68  39  0.130  26  0.310  -95
    11 DATA  0.660  141  1.46  27  0.140  26  0.330  -106
    12 DATA  0.670  129  1.24  17  0.160  26  0.360  -116
    13 DATA  0.680  121  1.14  5  0.170  23  0.380  -127
```

The .CKT file created for **Sceptre** includes ***more*** than just the s-data. It is a ***complete*** file, and includes the code which allows **Sceptre** to do a normal type "**S1**" analysis. In this case, **Sceptre** Users will *recognize* that the file contains two-port s-data over the range of 100-4000 MHz, and that **Sceptre** is being told to perform an s-parameter gain and stability analysis.

The file structure is such that you can easily edit it within **Sceptre**. You can add impedance match networks on either side of the "**TWO**" and create an amplifier. Naturally, using other programs within our **MicroWorks Design Suite**, i.e., **SmithMatch** and **OptiMatch**, you could create and optimize those match networks.

APPENDIX D

DOS Window Error Messages

Since our programs run in a DOS Window, you may at times get an error message. The most typical one is "File not found," or some such. Not to worry, our DOS Error Handler routine will deal with it.

In DOS, if you make a typing mistake, and enter the name of a file that doesn't exist, or leave a drive door open, etc., your PC will ***beep*** and you'll get an error message on-screen. It can look like the one shown *below*. To cause this error, we hit typed, by mistake, "NEC," when asked to enter a filename for analysis.

Sorry! Error in the Spline Module

DOS Error Code # 53
Reference Line # 4050

See your DOS Manual for an explanation of the error. A short list of *typical* DOS errors is given below:

Error	Translation
5	Illegal function call
7	Out of memory
11	Division by zero
27	Out of paper
53	File not found
61	Disk full
64	Bad file name
71	Disk not ready
76	Path not found

In the example we show above, DOS Error Code 53 means "**File not found**" The reference line number refers to our *internal code*, and is only listed in the unlikely event that you come across an error you cannot resolve.

To *recover* from a DOS Window error, follow the on-screen instruction to "**Press any Key**," and you will be returned to the module where the error occurred.

END.