PXI SOFTWARE

BELDEN

OCTOBER 17, 2017

BELDEN
[Company address]

1. TABLES OF CONTENTS

2. LIST OF TABLES

3. LIST OF FIGURES

4. SETUP

4.1. COMMUNICATION

Everything is done on the VNA. This is the setup you need to do if you want to keep the firewall on. You can also just turn the firewall off and everything should work without having to open any ports.

Control Panel > System and Security > Windows Firewall > Advanced Settings

Shortcut: command prompt - **wf.msc**

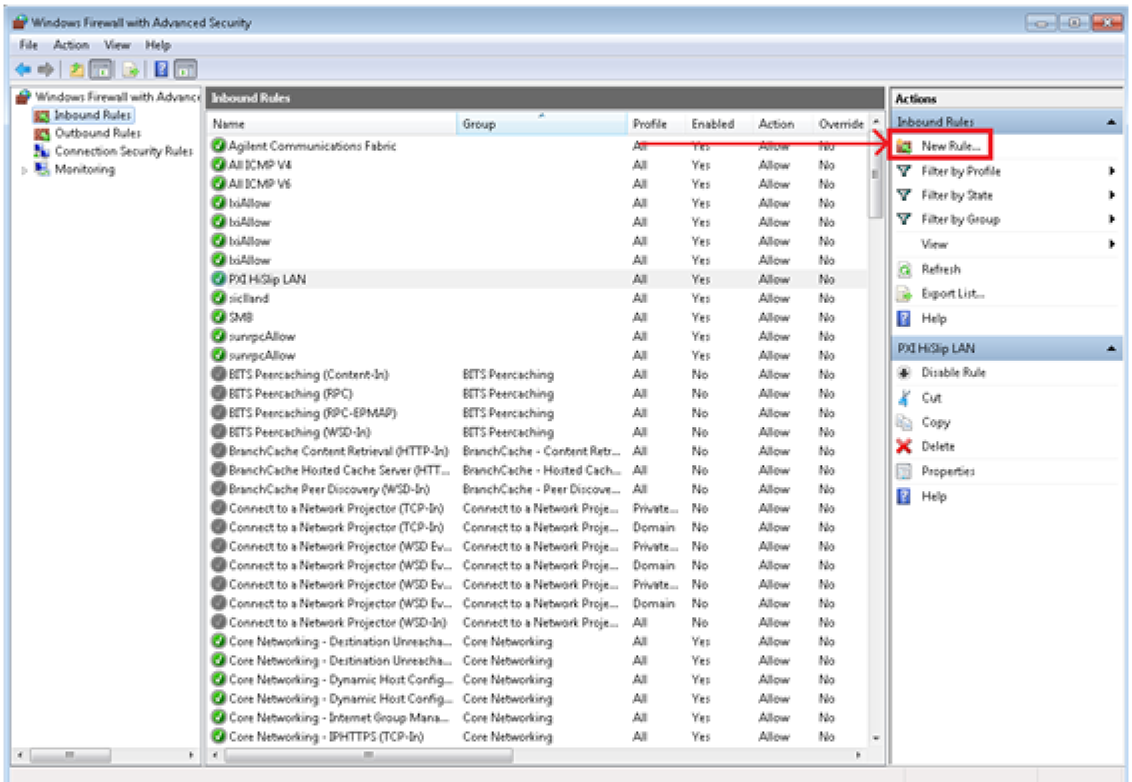


Figure 1: Screenshot of Windows' firewall advanced settings

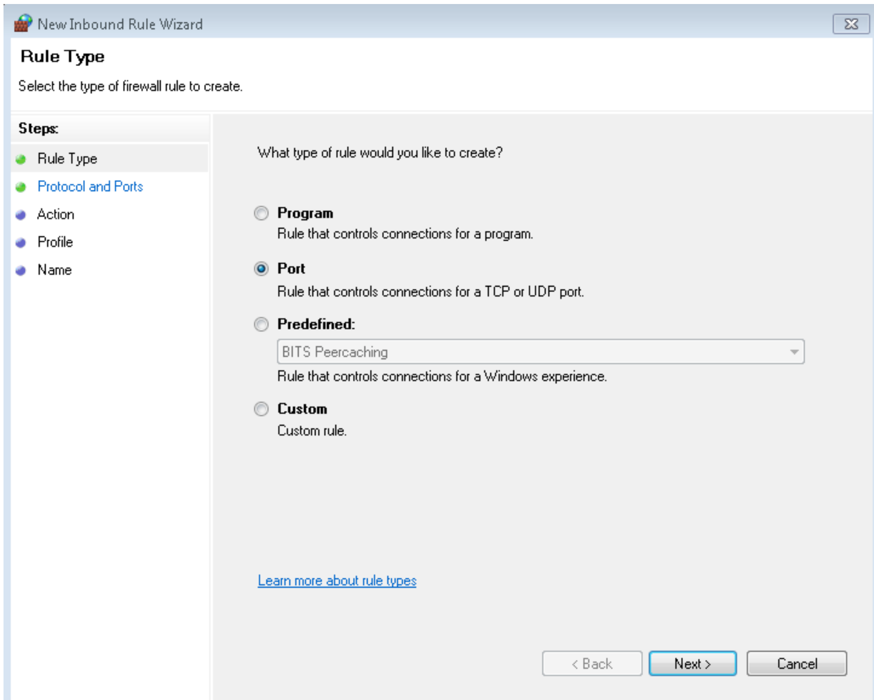


Figure 2: Choosing the port option as the rule type

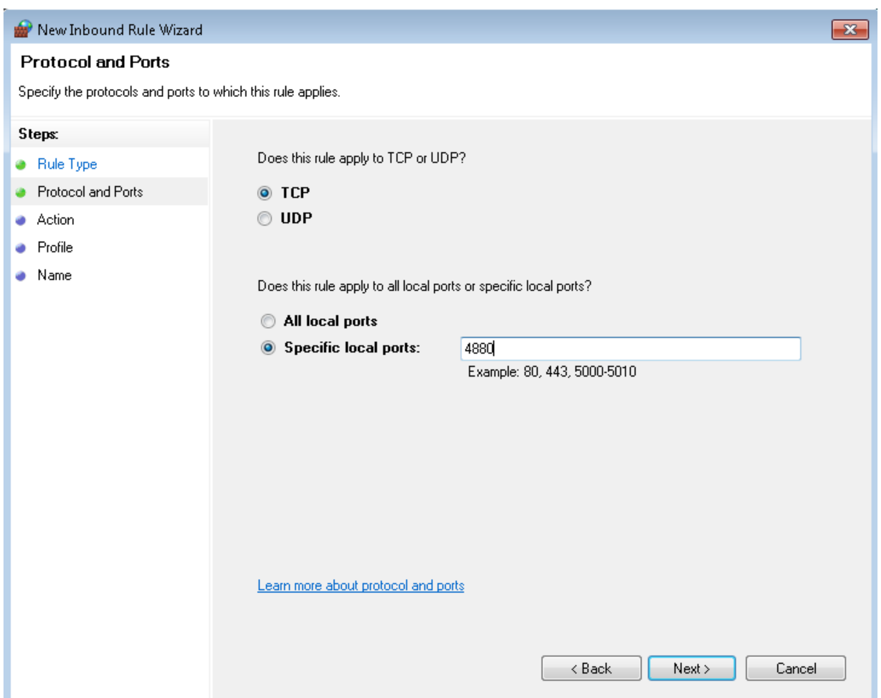


Figure 3: Choosing the port you want to open

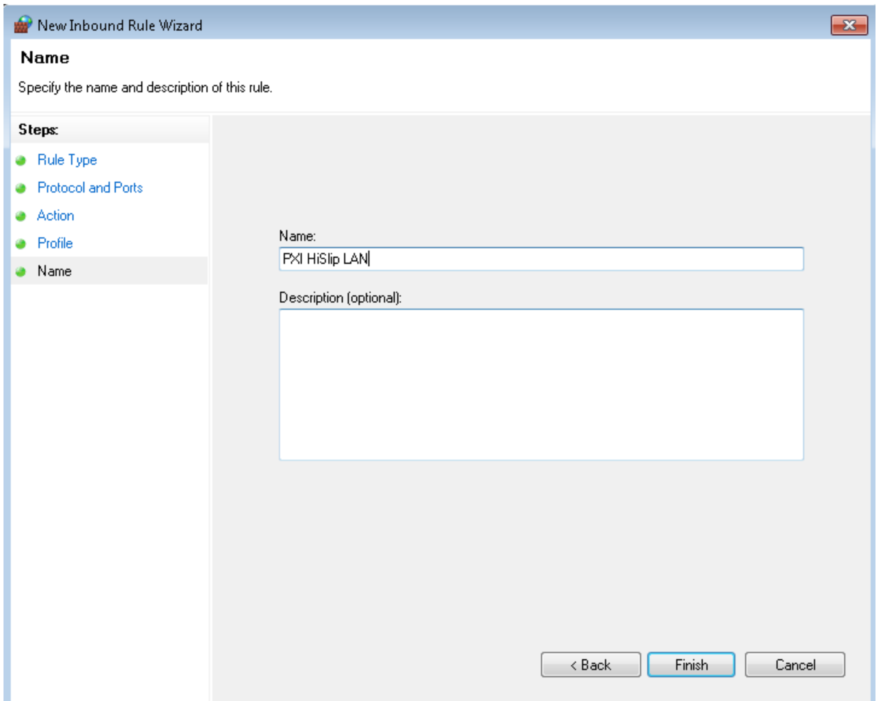


Figure 4: Name and description for the rule's list

4.2. NETWORK DRIVE

Using a network drive is very useful for managing the states and snp files. Creating one is pretty easy, but you might need to open a port on the firewall to make sure it is accessible for other computers on the network. For this, you can open the port 445 by using the same method. Once this is done, you can then use the default state folder to see what states you are currently using.

Important: Add a scope in the rule properties to make it more secure.

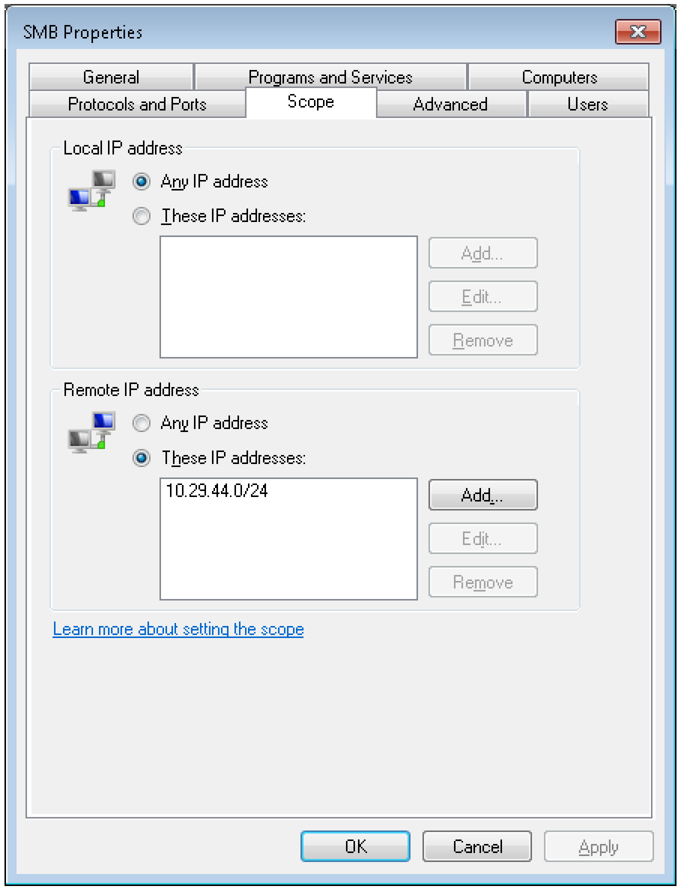


Figure 5: IP Address Range

IPADDRESS/24 means 24 is the mask : (255.255.255.0) meaning all IP address from 10.29.44.1 to 10.29.44.254 in our example.

4.3. REMOTE DESKTOP

This can be useful if you want to see how the VNA is acting when you are sending commands from a distance, or if you want to manually change some parameters. To connect to another computer, you need to have either a user account other than administrator, or set a password on the administrator account. Since the VNA default setup uses the administrator account by default, you want just want to setup a password on this account.

To setup an administrator password: Control Panel>User Accounts and Family Safety>Change your Windows password

You might all need to enable the remote desktop. To do so, go in Control Panel>System and Security>System. This enables every connection in the lan to remote access the pc by default.

Shortcut: command prompt – **control system**

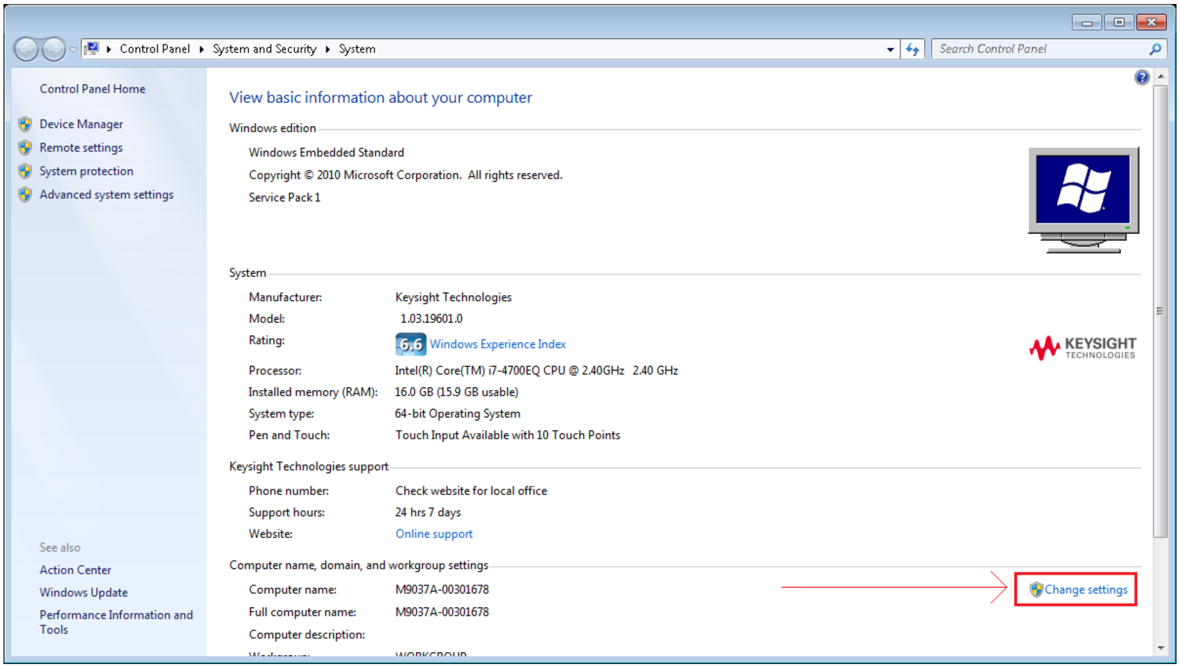


Figure 6: Allowing Remote Assistance

Then in remote and check “Allow Remote Assistance connections to this computer”.

If you want to restrict it to an IP address or IP address range, you need to do these steps: Control Panel > System and Security > Windows Firewall > Advanced Settings > Inbound Rules Find the Rule called “Remote Desktop (TCP-In)” and right-click properties. In scope, you can add the IP address allowed to remote control this computer in “Remote IP address”.

10.29.48.46

Administrator

Pass: space

5. SOFTWARE

5.1. USER INTERFACE

This is the area where the user can put the parameters needed. You have two choices: you can setup manual parameters on the user interface or use the connect VNA button to get the parameters from the VNA. You need to use the correct parameters, because the software uses these parameters to create arrays and manipulate data. This is why the file>Import from VNA button is disabled, since the VNA will generate a snp file from its values, writing different values would not be beneficial for the user.

5.1.1. Persistent settings

This is for changing the persistent settings for the user interface. When the program launches, it loads a text file called defaultset.txt to set the text box values. You can change those values by saving as default in the settings tab. This option overwrites the default.txt file with the values you currently have.

You can also save your custom files in the same folder and chose to load it later.

PS: Do not change the default settings text file unless you know what you are doing. It's recommended to change the values using the software rather than manually change them in the file.

Agilent doesn't have any options for persistent settings, unless you are using VEE's text box. However, as soon as you give it a design name to change its values, it loses its ability to keep these settings. You also don't have access to Visual Studio's ability so save its settings using

```
Properties.Settings.Default["SomeProperty"] = "Some Value";
Properties.Settings.Default.Save();
```

There's also an options to use serialization, which takes the data of an object to save it as a string that can be than reused. However, you can't really create custom objects with vee, unless you import a custom library in C, but you also need something to compile your code as a library.

5.1.2. DUT and baluns

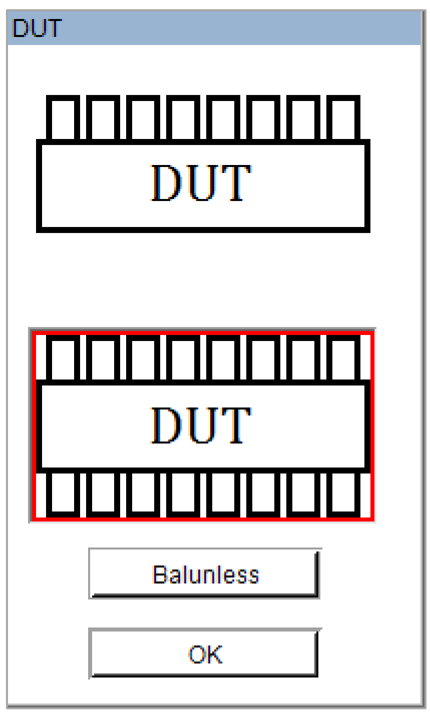


Figure 7: Changing the Device Under Test

In this menu, you can chose between two types of DUTs. The first one is a port>load DUT. It is useful to know what type of DUT the user is using, because this type of DUT can only measure the return loss and the NEXT loss.

The second DUT can measure everything else such as insertion loss and FEXT loss.

The third option is for what type of cable the measurement is done. Balunless is using single ended measured while balun is differential. Don't forget to change the number of ports accordingly.

5.2. IMPORTING DATA

5.2.1. From file

This is useful if you already have the touchstone file you want to use.

5.2.2. From VNA

First off, you need to make sure the software can communicate with the VNA. You can use the connect button to confirm this:

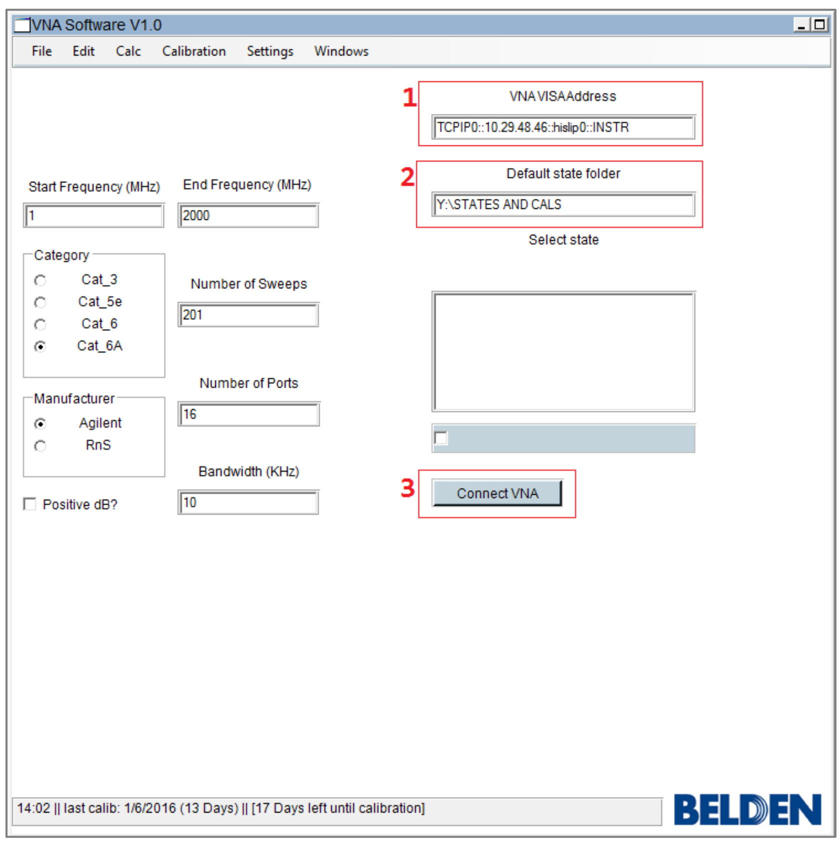


Figure 8: Connecting to the VNA

The VNA VISA Address is the visa address you have in Agilent Connection Expert. If you would change the IP on the VNA, you can also change it easily in the VISA Address.

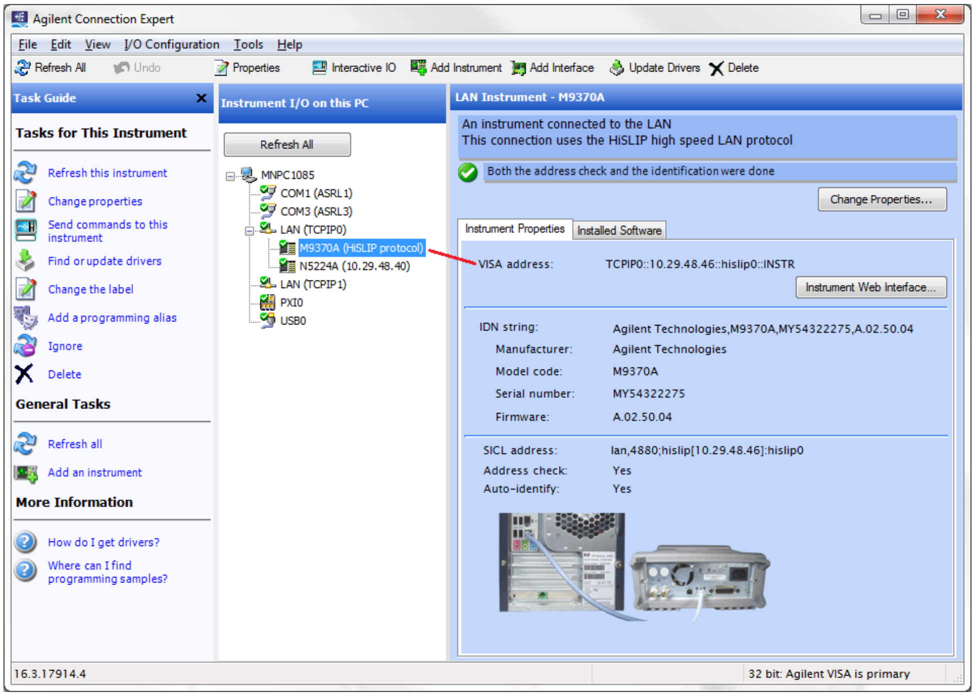


Figure 9: Finding the VISA address of the VNA

1. This is where you would store all the sta, cst, csa and cal files. For the .NET Framework to work, you need to have this folder on a network drive accessible to the client. In this example, we are using the Y:/ drive on the VNA.

   Once the previous steps are done, you can press this and the software will check if the communication is possible. This is what the software would look like if the connection was successful:

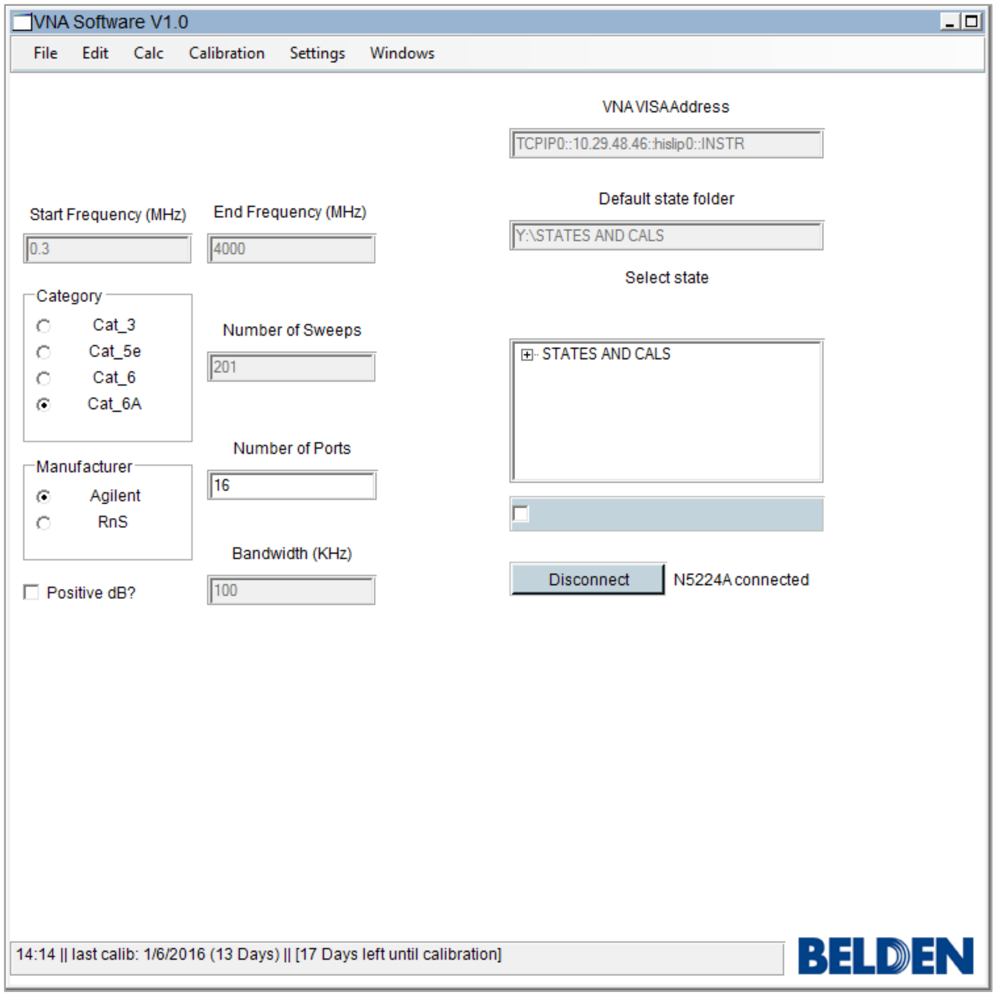


Figure 10: VNA Connected

Now, the "Import from VNA" option should be enabled.

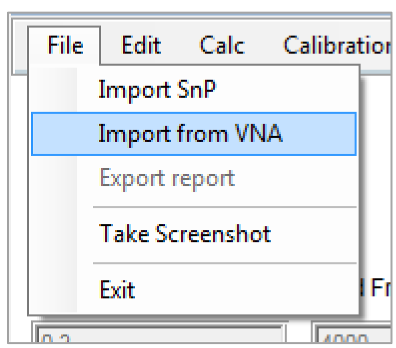


Figure 11: Importing from the VNA

This menu creates a snp file in the folder specified. It will then use this file for the processing.

5.3. FORMPOSTPROCESSING

This is the form that gets called after you import form an existing file or from the VNA.

5.3.1. Connecting Hardware (PLUG)

The limits represented in this function are the limits found in the document TP-10027: TNEXT PLUG Qualification.

5.3.2. Permanent Link

Permanent Link limits found in the belden formulas.

5.3.3. Channel

Channel limits found in the belden formulas.

5.4. PROCESSING

This is for the processing of the functions located in the first portion of the calc tab on the menu trip. Those include:

- Backbone Cable
- Channel
- Cord Cable
- Connecting Hardware
- Cords and Jumpers
- Horizontal Cable
- Permanent Link

These functions are different than the other ones since they are using the formulas.txt file to get their limits.

5.4.1. R_data record parameters

A record is a collection of field, similar to a struct in other languages. We use r_data for every parameter that we need, such as the number of sweeps, the number of ports, the DUT used and so on.

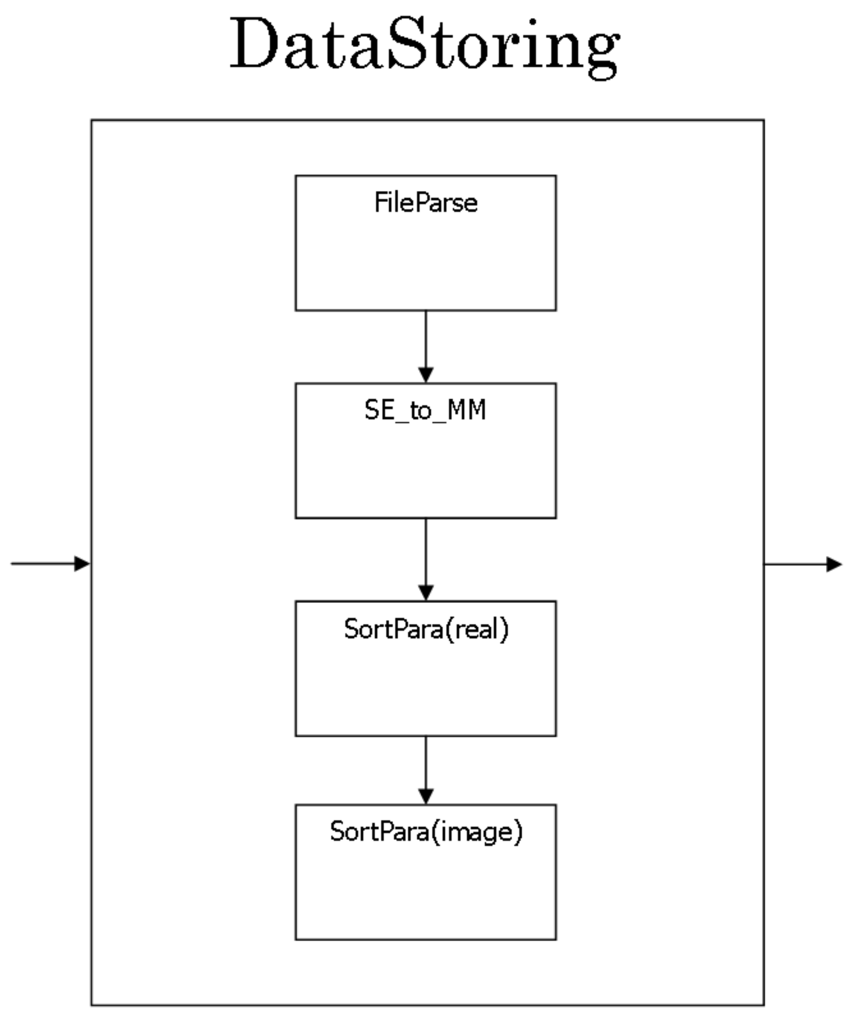
If you want to manually change the parameters in this record programmatically, you should use the functions to change them, rather than changing the fields directly. This makes sure no errors will happen by manually changing the fields. Also, some parameters, like the balun used, can affect others parameters in the record. All the functions start with “rDataSet”.

5.4.2. DataStoring

Scope: This function reads the date from the touchstone file generated from the VNA and generated arrays used for post processing such as the SDD values in real, imaginary dB and phase and finally the frequency points.

There are two methods for importing snp files. The first method is from an existing file. If you already have a snp file on your computer, you can import it if you already know the number of ports and the number of sweeps.

The second method is importing from the VNA. Depending on the state chosen, it will load the state and the calibration if it is still valid. It will then create a snp file on a network drive accessible on your computer. The software will then process it normally.



5.4.2.1. FILE PARSE

This block reads the touchstone file from the vna or from an already existing file. It first skip everything that's starts with "!" or "#" which are part of the header. It then reads every REAL64 values and stores it in a one dimensional array. If we know that the extension of the file is the number of physical ports, when can then calculate the number of sweeps like so:

$$n = number\ of\ physical\ ports \tag{1}$$

$$Number\ of\ sweeps = \frac{Totsize(array)}{2 * n^2 + 1} \tag{2}$$

The array is then processed in a 4D array for matrix operations.

RawData4d is simply for future processing. This step is needed so simplify greatly the following steps and also saving time. This function used the a_RawData array created before, to generate a four-dimension array. The generated array will be [Number Of Ports][Number of Ports][Number of Sweeps][2]. Using the same example as before, the array would be [16][16][2000][2]. To visualize a 4D array, you can simply think of having two cubes with a width and height of 16 blocks, and a 2000 block depth. This array only contains the S-parameters. Also, [x][x][x][0] would be the real part and [x][x][x][1] the imaginary part.

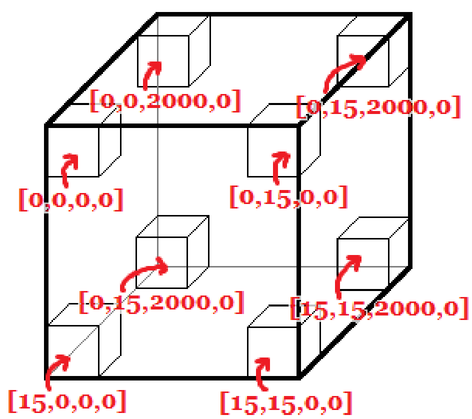


Figure 12: 4 Dimension arrays as a visual representation

5.4.2.2. SE_TO_MM

This functions takes the previously generated array to convert it to mixed mode. For any information on the conversion from singled ended to mixed mode, you can refer to.

This function is separated in multiple steps for easier understanding. The first step executed in the function is generating the conversion matrix. A_ConvMatrix is [Number of Ports][Number of Ports] or [16][16] and always has the same values. A 4 by 4 conversion matrix looks like this:

$$\begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} & 0 & 0 \\ 1/\sqrt{2} & 1/\sqrt{2} & 0 & 0 \\ 0 & 0 & 1/\sqrt{2} & -1/\sqrt{2} \\ 0 & 0 & 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \tag{3}$$

The matrix is the same for more ports; just repeat the same thing for a bigger matrix. We then create the sort matrix. This matrix is needed because of the formula we are using. Using the sort matrix, we can easily get the data we needed, Sdd, and at the same time reduce the time of execution.

For the actual conversion, we simply do a matrix multiplication with the conversion matrix.

$$a_{ConvMatrix} * a_{RawData4D} * a^{-1}_{ConvMatrix} \tag{4}$$

Finally we sort the matrix, store the real and imaginary part to a new array and store the dB and Phase. The two final arrays are a_Differential_RI and a_Differential_dB and their size are [Number of Sdd Ports][Number of Sdd Ports][Number of Sweeps][2] or [8][8][2000][2].

5.4.2.3. SORTPARA

Using the port form, we can rearrange ports to make it correspond to how we actually made the connections on the VNA. Here's an example that could happen:

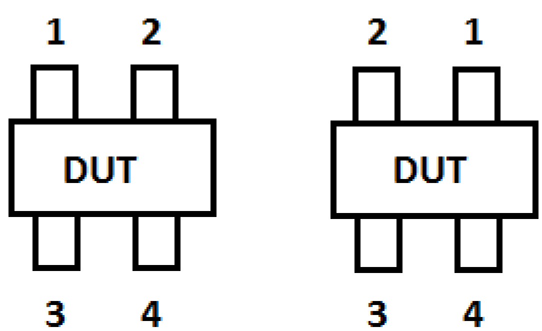


Figure 13: What we want (left) and what we have (right)

The sort matrix is then generated by subtracting the logical ports from the configuration we have to the configuration we want. The array would give: +1, -1, 0, 0. because the first logical port is now 2, and the second logical port is now 1.

Then:

$$MatrixLeft = \begin{bmatrix} S11 & S12 & S13 & S14 \\ S21 & S22 & S23 & S24 \\ S31 & S32 & S33 & S34 \\ S41 & S42 & S43 & S44 \end{bmatrix}$$

$$MatrixRight = \begin{bmatrix} S22 & S21 & S23 & S24 \\ S12 & S11 & S13 & S14 \\ S32 & S31 & S33 & S34 \\ S42 & S41 & S43 & S44 \end{bmatrix}$$

To sort the right matrix correctly, we need this matrix:

$$a_{Swap} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

This matrix is generated with the array mentioned earlier: {+1, -1, 0, 0}. These are used by generating an identity matrix, and changing its configuration. For each column, it changes the position of the number, for example, in the first column the first one is one row lower (+1) and the second column is one row higher (-1).

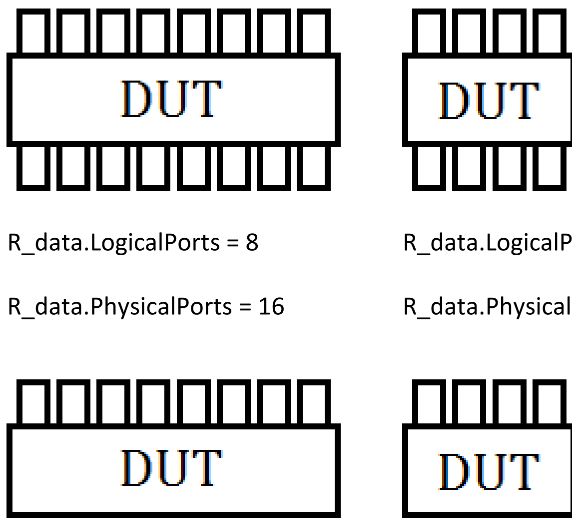
Final step:

$$a_{Swap} * MatrixRight * a_{Swap}^{-1} = MatrixLeft \tag{6}$$

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} S22 & S21 & S23 & S24 \\ S12 & S11 & S13 & S14 \\ S32 & S31 & S33 & S34 \\ S42 & S41 & S43 & S44 \end{bmatrix} = \begin{bmatrix} S12 & S11 & S13 & S14 \\ S22 & S21 & S23 & S24 \\ S32 & S31 & S33 & S34 \\ S42 & S41 & S43 & S44 \end{bmatrix} \rightarrow \begin{bmatrix} S12 & S11 & S13 & S14 \\ S22 & S21 & S23 & S24 \\ S32 & S31 & S33 & S34 \\ S42 & S41 & S43 & S44 \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} S11 & S12 & S13 & S14 \\ S21 & S22 & S23 & S24 \\ S31 & S32 & S33 & S34 \\ S41 & S42 & S43 & S44 \end{bmatrix} \tag{7}$$

Also, in order to differentiate the different kind of port arrangement, we use the port form and the DUT form. We then use the record r_data to differentiate the number of logical ports for the different setups. Since a 16 port balunless DUT will have a matrix as large as a 8 port balun DUT, the software needs to recognize the difference between the two.

Table 1: How the number of ports will affect our formulas



	BALUNLESS	BALUN
THRU	R_data.LogicalPorts = 8 R_data.PhysicalPorts = 16	R_data.LogicalPorts = 8 R_data.PhysicalPorts = 8
	R_data.LogicalPorts = 4 R_data.PhysicalPorts = 8	R_data.LogicalPorts = 4 R_data.PhysicalPorts = 4

5.4.3. InitArraysTHRU

This block is where all the titles for the excel report is generated. Normally, we would just give some static list of strings for the titles, however it wouldn't be an answer with all the possible file type (s8p, s16p, etc.). Therefore, I created some algorithms that will generate them automatically. To explain my thinking process, I'll show an exemple.

In this exemple, I'm going do the titles of the END1 Next losses and use a s16p file with no baluns, or in others words, a 8 port sdd matrix that looks like this:

Table 2: 8 port sdd thru matrix

Sdd11	Sdd12	Sdd13	Sdd14	Sdd15	Sdd16	Sdd17	Sdd18
Sdd21	**Sdd22**	Sdd23	Sdd24	Sdd25	Sdd26	Sdd27	Sdd28
Sdd31	Sdd32	**Sdd33**	Sdd34	Sdd35	Sdd36	Sdd37	Sdd38

Sdd41	Sdd42	Sdd43	**Sdd44**	Sdd45	Sdd46	Sdd47	Sdd48
Sdd51	Sdd52	Sdd53	Sdd54	**Sdd55**	Sdd56	Sdd57	Sdd58
Sdd61	Sdd62	Sdd63	Sdd64	Sdd65	**Sdd66**	Sdd67	Sdd68
Sdd71	Sdd72	Sdd73	Sdd74	Sdd75	Sdd76	**Sdd77**	Sdd78
Sdd81	Sdd82	Sdd83	Sdd84	Sdd85	Sdd86	Sdd87	**Sdd88**

Assuming this matrix is correctly sorted like explained earlier, the loss parameters are easily identified with a pattern.

- The return loss is always the oblique of the matrix, starting at the top left
- Insertion loss is always the oblique parallel of the return loss, and start at the half of the width and height of the matrix plus one.

We can also determine that all the values under the return loss oblique, in bold, are homologues values of the top values. Therefore, depending on the usage, we might not want to use them, because they will have the same values as their homologues.

Now, if we want to create an algorithm to have the column titles names of END1 Next Loss, we first going to reduce the size of the matrix to make it easier to understand.

Sdd11	Sdd12	Sdd13	Sdd14
Sdd21	**Sdd22**	Sdd23	Sdd24
Sdd31	Sdd32	**Sdd33**	Sdd34
Sdd41	Sdd42	Sdd43	**Sdd44**

This represents the END1 next loss values. The output string array we want is : {Sdd12, Sdd13, Sdd14, Sdd23, Sdd24, Sdd34}. Also, the size of this array can be determined by this formula:

$$sizeArray = \frac{x * (x - 2)}{8} \mid x = number\ of\ logical\ ports \quad (8)$$

So how can we generate those strings depending on the matrix size? This issue is easier to understand once we represent this matrix with indexes.

[0,0]	[0,1]	[0,2]	[0,3]
[1,0]	**[1,1]**	[1,2]	[1,3]
[2,0]	[2,1]	**[2,2]**	[2,3]
[3,0]	[3,1]	[3,2]	**[3,3]**

Now, what can we notice between the values above the oblique? Let's say that the row parameters can be represented by "i", the column index can be represented by "j" and "n" the number of logical ports(8 in our example). We notice that the values we need are when i<j.

We can now represent it as a simple code:

```
for(int i=0; i<n/2; i++)
  for(int j=0; j<n/2; j++)
    if(i<j)
      list.Add("S"+(i+1)+(j+1));
```

Therefore, if n = 8:

i	j	ouput
0	1	S12
0	2	S13
0	3	S14
1	2	S23
1	3	S24
2	3	S34

This process is the same for every other NEXT or FEXT possibilities. The only thing that changes is the condition and the ouput format. If the values we need is in the bottom left corner of a square matrix, the condition will be "i>j". Also, if the values we need is not in the first quadrant, we might have to add some constant to our value. For example, if we want to calculate the values of the END2 NEXT loss, those values are in the bottom right quadrant of the total matrix. Therefore, doing "`("S"+(i+1)+(j+1)`" will not give us the right name. We need to increment those values by half of "n". The formulas would then look like this:

```
for(int i=0; i<n/2; i++)
  for(int j=0; j<n/2; j++)
    if(i<j)
      list.Add("S"+(i+1+n/2)+(j+1+n/2));
```

i	j	ouput
0	1	S56
0	2	S57

0	3	S58
1	2	S67
1	3	S68
2	3	S78

5.5. FORM POST PROCESSING

This function gets called after importing a touchstone file. It can be used for the qualification of the plug, according to TP10026. When using the connecting hardware, four graphs are shown to the user. Each of these graphs have their own limits. Usually, the left is for the NEXT in decibels, while the right graphs are for the phase in degrees. Some of these data might also have 2 limits by graphs.

We use the standard limits for the graphs and for calculating the marge, however you can also use Belden's limits by changing the formulas that calls the function "Formulas" and "BuildLimitArray".

5.6. POST PROCESSING 8-8 THROUGH

5.6.1. Loss Calcs

This function gets the data needed and stores it in arrays for display or post processing.

If every port is lined properly on the VNA or using the port setup all the NEXT, FEXT and insertion loss should be like the examples shown.

These function are made to work with every size of matrix possible. The algorithms used are simple and always depends on the number of ports. Here's an example with the NEXT:

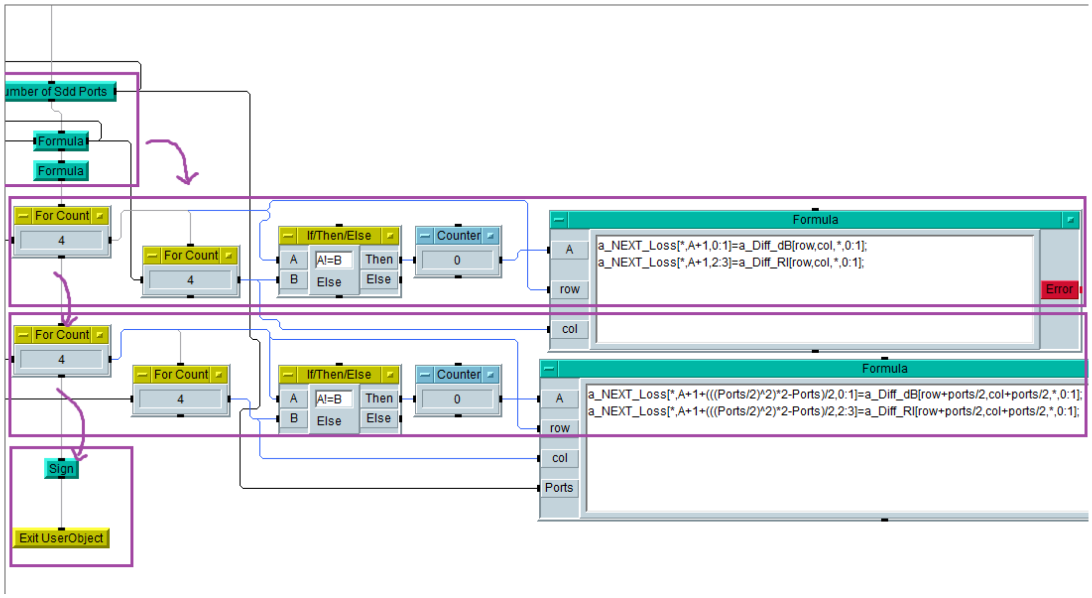
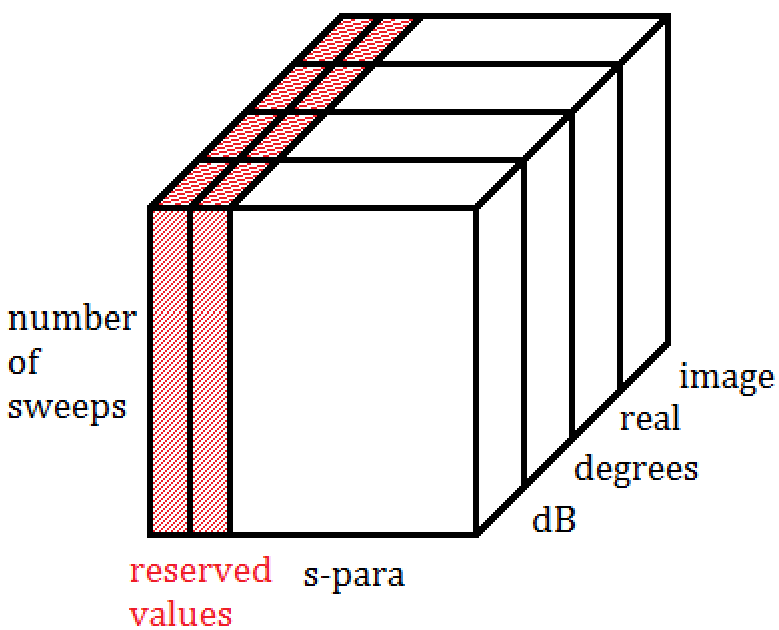


Figure 14: Modified screenshot of the VNA software on the IDE Agilent VEE pro v9.32

The blocks are supposed to be read from top to bottom. Inside the blocks, you read it from left to right. A_Next_Loss array is not quite like a_diff_dB and a_diff_RI. This is because the excel library accepts arrays in a certain configuration, so we prepare it in advance for future use. The next array looks like this:



Font: Cambria 14px

The reserved values are the first two columns of the array. The former is for the frequency points and the latter for the limits.

This code can be read as a nested for loop inside another for loop going to the number of logical ports/2. Let's go through a few iterations.

```
for(row=0;row<4;row++)
  for(col=0;col<4;col++)
    if(row!=col)
      [FORMULABLOCK]
```

Table 3: Example of the iterations possible

row	col	A	Formula	a_Diff_dB para
0	0	0	NaN NaN	NaN
0	1	1	a_NEXT_LOSS[*,2,0:1] = a_Diff_dB[0,1,*,0:1] a_NEXT_LOSS[*,2,2:3] = a_Diff_dB[0,1,*,2:3]	S12
0	2	2	a_NEXT_LOSS[*,3,0:1] = a_Diff_dB[0,2,*,0:1] a_NEXT_LOSS[*,3,2:3] = a_Diff_dB[0,2,*,2:3]	S13
0	3	3	a_NEXT_LOSS[*,4,0:1] = a_Diff_dB[0,3,*,0:1] a_NEXT_LOSS[*,4,2:3] = a_Diff_dB[0,3,*,2:3]	S14
1	0	4	a_NEXT_LOSS[*,5,0:1] = a_Diff_dB[1,0,*,0:1] a_NEXT_LOSS[*,5,2:3] = a_Diff_dB[1,0,*,2:3]	S21
1	1	4	NaN NaN	NaN
1	2	5	a_NEXT_LOSS[*,6,0:1] = a_Diff_dB[1,2,*,0:1] a_NEXT_LOSS[*,6,2:3] = a_Diff_dB[1,2,*,2:3]	S23
1	3	6	a_NEXT_LOSS[*,7,0:1] = a_Diff_dB[1,3,*,0:1] a_NEXT_LOSS[*,7,2:3] = a_Diff_dB[1,3,*,2:3]	S24

This is for the first formula block, the second formula block is the same thing, but the row and col in incremented by the number of logical ports divided by two.

5.6.1.1. RETURN LOSS

SDD11 to SDDnn where n = number of logical ports

Table 4: Return Loss

Sdd11	Sdd12	Sdd13	Sdd14	Sdd15	Sdd16	Sdd17	Sdd18
Sdd21	**Sdd22**	Sdd23	Sdd24	Sdd25	Sdd26	Sdd27	Sdd28
Sdd31	Sdd32	**Sdd33**	Sdd34	Sdd35	Sdd36	Sdd37	Sdd38
Sdd41	Sdd42	Sdd43	**Sdd44**	Sdd45	Sdd46	Sdd47	Sdd48
Sdd51	Sdd52	Sdd53	Sdd54	**Sdd55**	Sdd56	Sdd57	Sdd58
Sdd61	Sdd62	Sdd63	Sdd64	Sdd65	**Sdd66**	Sdd67	Sdd68
Sdd71	Sdd72	Sdd73	Sdd74	Sdd75	Sdd76	**Sdd77**	Sdd78

Sdd81	Sdd82	Sdd83	Sdd84	Sdd85	Sdd86	Sdd87	**Sdd88**

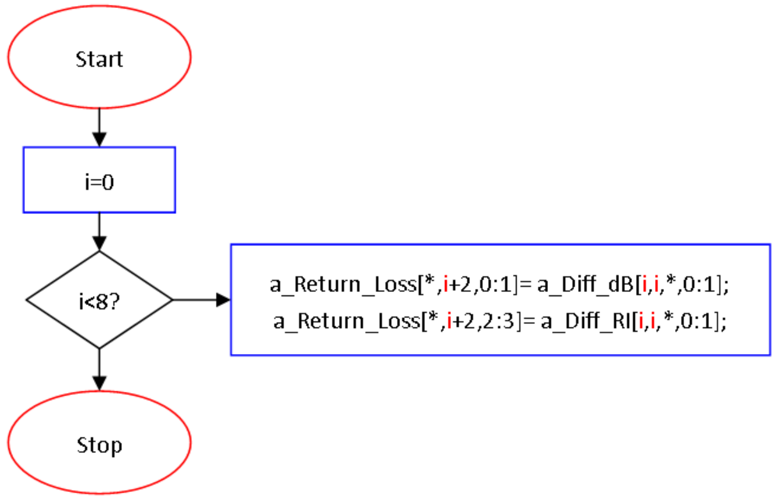


Table 5: Example of iterations for a 8 logical port DUT

i	Formula
0	A_Return_Loss[*,2,0:1] = a_Diff_dB[0,0,*,0:1]; A_Return_Loss[*,2,2:3] = a_Diff_RI[0,0,*,0:1];
1	A_Return_Loss[*,3,0:1] = a_Diff_dB[1,1,*,0:1]; A_Return_Loss[*,3,2:3] = a_Diff_RI[1,1,*,0:1];
2	A_Return_Loss[*,4,0:1] = a_Diff_dB[2,2,*,0:1]; A_Return_Loss[*,4,2:3] = a_Diff_RI[2,2,*,0:1];]
3	A_Return_Loss[*,5,0:1] = a_Diff_dB[3,3,*,0:1]; A_Return_Loss[*,5,2:3] = a_Diff_RI[3,3,*,0:1];
4	A_Return_Loss[*,6,0:1] = a_Diff_dB[4,4,*,0:1]; A_Return_Loss[*,6,2:3] = a_Diff_RI[4,4,*,0:1];
5	A_Return_Loss[*,7,0:1] = a_Diff_dB[5,5,*,0:1]; A_Return_Loss[*,7,2:3] = a_Diff_RI[5,5,*,0:1];
6	A_Return_Loss[*,8,0:1] = a_Diff_dB[6,6,*,0:1]; A_Return_Loss[*,8,2:3] = a_Diff_RI[6,6,*,0:1];

7	A_Return_Loss[*,9,0:1] = a_Diff_dB[7,7,*,0:1]; A_Return_Loss[*,9,2:3] = a_Diff_RI[7,7,*,0:1];

5.6.1.2. INSERTION LOSS

Also known as the **attenuation**

Table 6: Insertion Loss

Sdd11	Sdd12	Sdd13	Sdd14	**Sdd15**	Sdd16	Sdd17	Sdd18
Sdd21	Sdd22	Sdd23	Sdd24	Sdd25	**Sdd26**	Sdd27	Sdd28
Sdd31	Sdd32	Sdd33	Sdd34	Sdd35	Sdd36	**Sdd37**	Sdd38
Sdd41	Sdd42	Sdd43	Sdd44	Sdd45	Sdd46	Sdd47	**Sdd48**
Sdd51	Sdd52	Sdd53	Sdd54	Sdd55	Sdd56	Sdd57	Sdd58
Sdd61	**Sdd62**	Sdd63	Sdd64	Sdd65	Sdd66	Sdd67	Sdd68
Sdd71	Sdd72	**Sdd73**	Sdd74	Sdd75	Sdd76	Sdd77	Sdd78
Sdd81	Sdd82	Sdd83	**Sdd84**	Sdd85	Sdd86	Sdd87	Sdd88

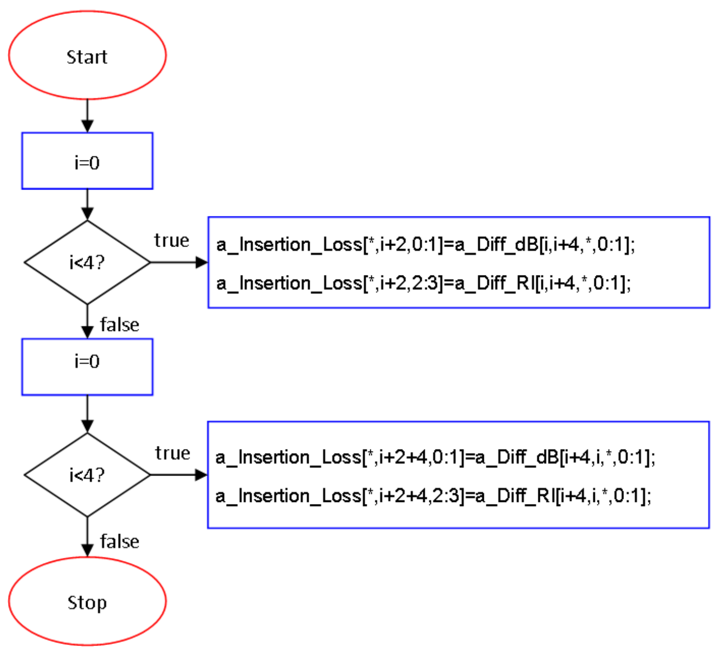


Table 7: Example of iterations for an 8 logical port DUT

i	Formula
0	a_Insertion_Loss[*,2,0:1]=a_Diff_dB[0,4,*,0:1];

```
      a_Insertion_Loss[*,2,2:3]=a_Diff_RI[0,4,*,0:1];

1     a_Insertion_Loss[*,3,0:1]=a_Diff_dB[1,5,*,0:1];

      a_Insertion_Loss[*,3,2:3]=a_Diff_RI[1,5,*,0:1];

2     a_Insertion_Loss[*,4,0:1]=a_Diff_dB[2,6,*,0:1];

      a_Insertion_Loss[*,4,2:3]=a_Diff_RI[2,6,*,0:1];

3     a_Insertion_Loss[*,5,0:1]=a_Diff_dB[3,7,*,0:1];

      a_Insertion_Loss[*,5,2:3]=a_Diff_RI[3,7,*,0:1];

0     a_Insertion_Loss[*,6,0:1]=a_Diff_dB[4,0,*,0:1];

      a_Insertion_Loss[*,6,2:3]=a_Diff_RI[4,0,*,0:1];

1     a_Insertion_Loss[*,7,0:1]=a_Diff_dB[5,1,*,0:1];

      a_Insertion_Loss[*,7,2:3]=a_Diff_RI[5,1,*,0:1];

2     a_Insertion_Loss[*,8,0:1]=a_Diff_dB[6,2,*,0:1];

      a_Insertion_Loss[*,8,2:3]=a_Diff_RI[6,2,*,0:1];

3     a_Insertion_Loss[*,9,0:1]=a_Diff_dB[7,3,*,0:1];

      a_Insertion_Loss[*,9,2:3]=a_Diff_RI[7,3,*,0:1];
```

5.6.1.3. NEXT LOSS

Also known as the near-end crosstalk

Table 8: Near-end Crosstalk

Sdd11	**Sdd12**	**Sdd13**	**Sdd14**	Sdd15	Sdd16	Sdd17	Sdd18
Sdd21	Sdd22	**Sdd23**	**Sdd24**	Sdd25	Sdd26	Sdd27	Sdd28
Sdd31	**Sdd32**	Sdd33	**Sdd34**	Sdd35	Sdd36	Sdd37	Sdd38
Sdd41	**Sdd42**	**Sdd43**	Sdd44	Sdd45	Sdd46	Sdd47	Sdd48
Sdd51	Sdd52	Sdd53	Sdd54	Sdd55	**Sdd56**	**Sdd57**	**Sdd58**

Sdd61	Sdd62	Sdd63	Sdd64	**Sdd65**	Sdd66	**Sdd67**	**Sdd68**
Sdd71	Sdd72	Sdd73	Sdd74	**Sdd75**	**Sdd76**	Sdd77	**Sdd78**
Sdd81	Sdd82	Sdd83	Sdd84	**Sdd85**	**Sdd86**	**Sdd87**	Sdd88

5.6.1.4. FEXT LOSS

Also known as the far-end crosstalk

Table 9: Far-end crosstalk

Sdd11	Sdd12	Sdd13	Sdd14	Sdd15	**Sdd16**	**Sdd17**	**Sdd18**
Sdd21	Sdd22	Sdd23	Sdd24	**Sdd25**	Sdd26	**Sdd27**	**Sdd28**
Sdd31	Sdd32	Sdd33	Sdd34	**Sdd35**	**Sdd36**	Sdd37	**Sdd38**
Sdd41	Sdd42	Sdd43	Sdd44	**Sdd45**	**Sdd46**	**Sdd47**	Sdd48
Sdd51	**Sdd52**	**Sdd53**	**Sdd54**	Sdd55	Sdd56	Sdd57	Sdd58
Sdd61	Sdd62	**Sdd63**	**Sdd64**	Sdd65	Sdd66	Sdd67	Sdd68
Sdd71	**Sdd72**	Sdd73	**Sdd74**	Sdd75	Sdd76	Sdd77	Sdd78
Sdd81	**Sdd82**	**Sdd83**	Sdd84	Sdd85	Sdd86	Sdd87	Sdd88

5.6.1.5. ACR

Also known as attenuation to crosstalk ratio, ACR is calculated by subtracting the NEXT by the insertion of the same victim in **dB**. Examples of ACR values:

Table 10: Attenuation to crosstalk ratio

Sdd11	**Sdd12**	**Sdd13**	**Sdd14**	**Sdd15**	Sdd16	Sdd17	Sdd18
Sdd21	Sdd22	**Sdd23**	**Sdd24**	Sdd25	**Sdd26**	Sdd27	Sdd28
Sdd31	**Sdd32**	Sdd33	**Sdd34**	Sdd35	Sdd36	**Sdd37**	Sdd38
Sdd41	**Sdd42**	**Sdd43**	Sdd44	Sdd45	Sdd46	Sdd47	**Sdd48**
Sdd51	Sdd52	Sdd53	Sdd54	Sdd55	**Sdd56**	**Sdd57**	**Sdd58**
Sdd61	**Sdd62**	Sdd63	Sdd64	**Sdd65**	Sdd66	**Sdd67**	**Sdd68**
Sdd71	Sdd72	**Sdd73**	Sdd74	**Sdd75**	**Sdd76**	Sdd77	**Sdd78**
Sdd81	Sdd82	Sdd83	**Sdd84**	**Sdd85**	**Sdd86**	**Sdd87**	Sdd88

$$Sdd12 - \boldsymbol{Sdd15}, \quad Sdd13 - \boldsymbol{Sdd15}, \quad Sdd14 - \boldsymbol{Sdd15}, \quad Sdd21 - \boldsymbol{Sdd26}, \quad Sdd23 - \boldsymbol{Sdd26} \ldots \tag{9}$$

5.6.1.6. ACRF

Also known as attenuation to cross talk ratio (far end), ACRF is calculated by subtracting the FEXT by the insertion loss of the same victim in **dB**.

Table 11: Attenuation to crosstalk ratio (far-end)

Sdd11	Sdd12	Sdd13	Sdd14	**Sdd15**	**Sdd16**	**Sdd17**	**Sdd18**
Sdd21	Sdd22	Sdd23	Sdd24	**Sdd25**	**Sdd26**	**Sdd27**	**Sdd28**
Sdd31	Sdd32	Sdd33	Sdd34	**Sdd35**	**Sdd36**	**Sdd37**	**Sdd38**
Sdd41	Sdd42	Sdd43	Sdd44	**Sdd45**	**Sdd46**	**Sdd47**	**Sdd48**
Sdd51	**Sdd52**	**Sdd53**	**Sdd54**	Sdd55	Sdd56	Sdd57	Sdd58
Sdd61	**Sdd62**	**Sdd63**	**Sdd64**	Sdd65	Sdd66	Sdd67	Sdd68
Sdd71	**Sdd72**	**Sdd73**	**Sdd74**	Sdd75	Sdd76	Sdd77	Sdd78
Sdd81	**Sdd82**	**Sdd83**	**Sdd84**	Sdd85	Sdd86	Sdd87	Sdd88

$$Sdd16 - \boldsymbol{Sdd15}, \quad Sdd17 - \boldsymbol{Sdd15}, \quad Sdd18 - \boldsymbol{Sdd15}, \quad Sdd25 - \boldsymbol{Sdd26}, \quad Sdd27 - \boldsymbol{Sdd26} \ldots \tag{10}$$

5.6.1.7. PSNEXT

Also known as power-sum near-end crosstalk, PSNEXT is calculated by calculating the sum of 3 NEXT on a same victim.

Table 12: Power-sum near-end crosstalk

Sdd11	**Sdd12**	**Sdd13**	**Sdd14**	Sdd15	Sdd16	Sdd17	Sdd18
Sdd21	Sdd22	**Sdd23**	**Sdd24**	Sdd25	Sdd26	Sdd27	Sdd28
Sdd31	**Sdd32**	Sdd33	**Sdd34**	Sdd35	Sdd36	Sdd37	Sdd38
Sdd41	**Sdd42**	**Sdd43**	Sdd44	Sdd45	Sdd46	Sdd47	Sdd48
Sdd51	Sdd52	Sdd53	Sdd54	Sdd55	**Sdd56**	**Sdd57**	**Sdd58**
Sdd61	Sdd62	Sdd63	Sdd64	**Sdd65**	Sdd66	**Sdd67**	**Sdd68**
Sdd71	Sdd72	Sdd73	Sdd74	**Sdd75**	**Sdd76**	Sdd77	**Sdd78**
Sdd81	Sdd82	Sdd83	Sdd84	**Sdd85**	**Sdd86**	**Sdd87**	Sdd88

$$PSNEXT\ Vic1 = Sdd12 + Sdd13 + Sdd14 \tag{11}$$

$$PSNEXT\ Vic2 = Sdd21 + Sdd23 + Sdd24 \tag{12}$$

…

PS: When subtracting or adding s-parameters, real and imaginary values must be used or the values won't be correct. Linear values also works, but you must not add the phase if you do so.

5.6.1.8. PSFEXT

Also known as power sum far-end crosstalk, PSFEXT is calculated by calculating the sum of 3 FEXT on a same victim.

Table 13: Power sum far-end crosstalk

Sdd11	Sdd12	Sdd13	Sdd14	Sdd15	**Sdd16**	**Sdd17**	**Sdd18**
Sdd21	Sdd22	Sdd23	Sdd24	**Sdd25**	Sdd26	**Sdd27**	**Sdd28**
Sdd31	Sdd32	Sdd33	Sdd34	**Sdd35**	**Sdd36**	Sdd37	**Sdd38**
Sdd41	Sdd42	Sdd43	Sdd44	**Sdd45**	**Sdd46**	**Sdd47**	Sdd48
Sdd51	**Sdd52**	**Sdd53**	**Sdd54**	Sdd55	Sdd56	Sdd57	Sdd58
Sdd61	Sdd62	**Sdd63**	**Sdd64**	Sdd65	Sdd66	Sdd67	Sdd68
Sdd71	**Sdd72**	Sdd73	**Sdd74**	Sdd75	Sdd76	Sdd77	Sdd78
Sdd81	**Sdd82**	**Sdd83**	Sdd84	Sdd85	Sdd86	Sdd87	Sdd88

$$PSFEXT\ Vic1 \;=\; Sdd16 + Sdd17 + Sdd18 \tag{13}$$

$$PSFEXT\ Vic2 = Sdd25 + Sdd27 + Sdd28 \tag{14}$$

…

PS: When subtracting or adding s-parameters, real and imaginary values must be used or the values won't be correct. Linear values also works, but you must not add the phase if you do so.

5.6.1.9. PSACR

Also known as power sum attenuation to crosstalk ratio, PSACR is calculated by subtracting the PSNEXT of a victim by the insertion loss of the same victim in dB.

Table 14: Power sum attenuation to crosstalk ratio

Sdd11	**Sdd12**	**Sdd13**	**Sdd14**	**Sdd15**	Sdd16	Sdd17	Sdd18
Sdd21	Sdd22	**Sdd23**	**Sdd24**	Sdd25	**Sdd26**	Sdd27	Sdd28
Sdd31	**Sdd32**	Sdd33	**Sdd34**	Sdd35	Sdd36	**Sdd37**	Sdd38
Sdd41	**Sdd42**	**Sdd43**	Sdd44	Sdd45	Sdd46	Sdd47	**Sdd48**
Sdd51	Sdd52	Sdd53	Sdd54	Sdd55	**Sdd56**	**Sdd57**	**Sdd58**
Sdd61	**Sdd62**	Sdd63	Sdd64	**Sdd65**	Sdd66	**Sdd67**	**Sdd68**
Sdd71	Sdd72	**Sdd73**	Sdd74	**Sdd75**	**Sdd76**	Sdd77	**Sdd78**
Sdd81	Sdd82	Sdd83	**Sdd84**	**Sdd85**	**Sdd86**	**Sdd87**	Sdd88

$$PSACR\ Vic1 = (Sdd12 + Sdd13 + Sdd14)dB - \boldsymbol{Sdd15dB} \tag{15}$$

$$PSACR\ Vic2 = (Sdd21 + Sdd23 + Sdd24)dB - \boldsymbol{Sdd26dB} \tag{16}$$

...

PS: When subtracting or adding s-parameters, real and imaginary values must be used or the values won't be correct. Linear values also works, but you must not add the phase if you do so.

5.6.1.10. PSACRF

Also known as power sum attenuation to crosstalk ratio (far-end), PSACRF is calculated by subtracting the PSFEXT of a victim by the insertion loss of the same victim.

Table 15: Power sum attenuation to crosstalk ratio (far-end)

Sdd11	Sdd12	Sdd13	Sdd14	**Sdd15**	**Sdd16**	**Sdd17**	**Sdd18**
Sdd21	Sdd22	Sdd23	Sdd24	**Sdd25**	**Sdd26**	**Sdd27**	**Sdd28**
Sdd31	Sdd32	Sdd33	Sdd34	**Sdd35**	**Sdd36**	**Sdd37**	**Sdd38**
Sdd41	Sdd42	Sdd43	Sdd44	**Sdd45**	**Sdd46**	**Sdd47**	**Sdd48**
Sdd51	**Sdd52**	**Sdd53**	**Sdd54**	Sdd55	Sdd56	Sdd57	Sdd58
Sdd61	**Sdd62**	**Sdd63**	**Sdd64**	Sdd65	Sdd66	Sdd67	Sdd68
Sdd71	**Sdd72**	**Sdd73**	**Sdd74**	Sdd75	Sdd76	Sdd77	Sdd78
Sdd81	**Sdd82**	**Sdd83**	**Sdd84**	Sdd85	Sdd86	Sdd87	Sdd88

$$PSFEXT\ Vic1 = (Sdd16 + Sdd17 + Sdd18)\text{dB} - \boldsymbol{Sdd15}\mathbf{dB} \quad (17)$$

$$PSFEXT\ Vic2 = (Sdd25 + Sdd27 + Sdd28)\text{dB} - \boldsymbol{Sdd26}\mathbf{dB} \quad (18)$$

…

PS: When subtracting or adding s-parameters, real and imaginary values must be used or the values won't be correct. Linear values also works, but you must not add the phase if you do so.

5.6.2. Limit calc

Most of the formulas are from the TIA-568-C.2 standard[1]. Everything else can be found in TP 10026, TP 10027 and TP 10028.

This function reads the formulas from a text file, called Formulas.txt, and sets the limits in the arrays used previously for the loss calculations.

5.7. GET FORMULAS

This function reads from the formula text file. Depending on which connection type the user chose, it puts the formula and the limits corresponding in a two dimensional array. In example, the return loss limits of a channel, category 6A cable, the output array would be:

Table 16: Output array of get Formulas

1<=f<10	19
10<=f<40	24-5*(log10(f))
40<=f<398.1	32-10*(log10(f))
398.1<=f<=500	6

5.8. PROCESS LIMITS

Since the If/Then/Else block cannot have a control input for its formula, we have to do a workaround to make it work. The FormulaPara separates the different variables of the frequency limits. If we would have an input of 398.1<=f<=500, the two outputs would give us:

[1] Reference\TIA 568-C.2 Final.pdf

$$MinMaxArray = \{398.1\ , 500\} \tag{19}$$

$$SymbolArray = \{\leq , \leq\} \tag{20}$$

The symbol array is the issue for the If Block. We can have value inputs for the min max array, but we can't change the symbols at will. To remedy the situation, we had to make a function for each possibility of symbol combination. The three combination are {<, <=}, {<=, <} and {<=, <=}. Those three functions then compare the min and max values with the frequency array, and set the minimum and maximum index for the main function. A for loop is then used to calculate the limits formulas, from the getFormula outputs.

5.9. ALIEN CROSSTALK

5.9.1. Form

5.9.2. Data

Table 17: Alien power sum End1 and End2

Sdd11	Sdd12	Sdd13	Sdd14	**Sdd15**	**Sdd16**	**Sdd17**	**Sdd18**
Sdd21	Sdd22	Sdd23	Sdd24	**Sdd25**	**Sdd26**	**Sdd27**	**Sdd28**
Sdd31	Sdd32	Sdd33	Sdd34	**Sdd35**	**Sdd36**	**Sdd37**	**Sdd38**
Sdd41	Sdd42	Sdd43	Sdd44	**Sdd45**	**Sdd46**	**Sdd47**	**Sdd48**
Sdd51	**Sdd52**	**Sdd53**	**Sdd54**	Sdd55	Sdd56	Sdd57	Sdd58
Sdd61	**Sdd62**	**Sdd63**	**Sdd64**	Sdd65	Sdd66	Sdd67	Sdd68
Sdd71	**Sdd72**	**Sdd73**	**Sdd74**	Sdd75	Sdd76	Sdd77	Sdd78
Sdd81	**Sdd82**	**Sdd83**	**Sdd84**	Sdd85	Sdd86	Sdd87	Sdd88

This is how the data is taken. Blue being END1 and red being END2. These are normally the FEXT and insertion loss if you would use one jack. However, for our test, this is the measurement of the effect of a pair on a victim. Depending on how the setup is done, you can measure the alien NEXT or the alien FEXT. When the measurements are done, we then calculate the power sum of each victim:

END 1:

$$PS1(Pair1) = Sdd15 + Sdd16 + Sdd17 + Sdd18 \tag{21}$$

$$PS2(Pair2) = Sdd25 + Sdd26 + Sdd27 + Sdd28 \tag{22}$$

END 2:

$$PS1(Pair1) = Sdd51 + Sdd52 + Sdd53 + Sdd54 \tag{23}$$

$$PS2(Pair2) = Sdd61 + Sdd62 + Sdd63 + Sdd64 \tag{24}$$

In the form, you can then decide to only show certain disturbers. Since the disturbers are actually pairs from the same connector, doing so will affect the previous formula. In example, if we would only want to show the Disturber 1 and 4, the formula would then be:

END 1:

$$PS1(Pair1) = Sdd15 + Sdd18 \tag{25}$$

$$PS2(Pair2) = Sdd25 + Sdd28 \tag{26}$$

END 2:

$$PS1(Pair1) = Sdd51 + Sdd54 \tag{27}$$

$$PS2(Pair2) = Sdd61 + Sdd68 \tag{28}$$

5.9.3. Algorithm

A_Diff_RI_ANEXT/a_Diff_RI_AFEXT: This is the raw data stored when measuring using the two buttons on the form.

Size: [8,8,Number of Sweeps,2]
The first two Digits are the 64 S-Parameters

- [0,0,*,0] is S11 in real
- [7,7,*,1] is S88 in image

The last digit is for the Real/Image

- [*,*,*,0] is real
- [*,*,*,1] is image

Pair1 to Pair4: These are the arrays of the data converted in dB

Size: [Number of Sweeps, 1]

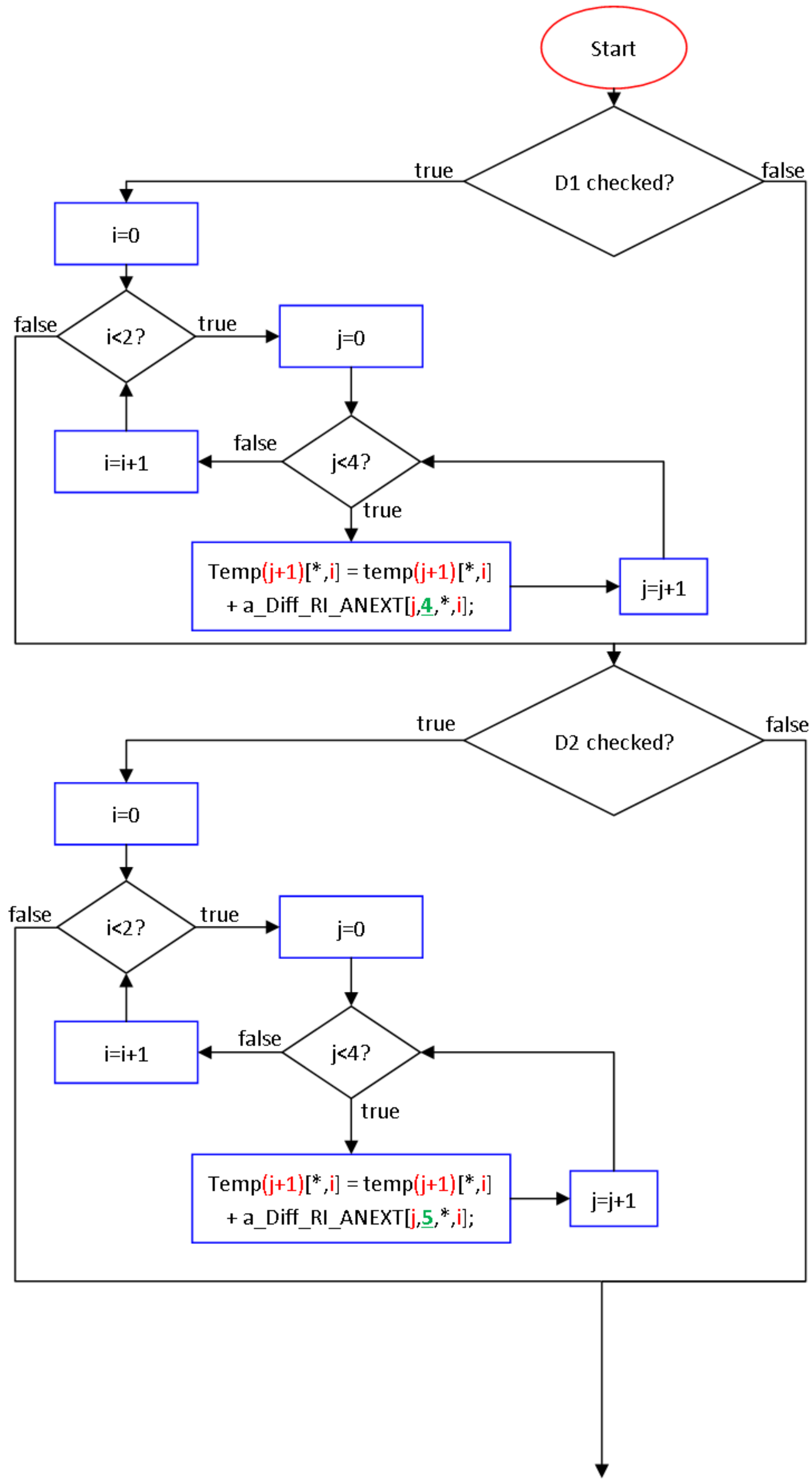
Since either FEXT or NEXT and END1 or END2 is shown, you only need a 1D array to show the data on the graph. How, the reason we use a 2D arrays with the second dimension being size 1, is for other functions. Vee differentiates 2D arrays and 1D arrays even if they are the same size, so 1D arrays may not work for some functions like combining two arrays.

Temp1 to Temp4: These are the arrays using temporary for holding the data before converting it to dB.

Size: [Number of Sweeps,2]
The last digit is for the Real/Image

- [*,0] is real
- [*,1] is image

Here's an example for the END1 NEXT.


Start
D1 checked?
true
false
i=0
i<2?
true
false
j=0
j<4?
false
true
i=i+1
Temp(j+1)[*,i] = temp(j+1)[*,i]
+ a_Diff_RI_ANEXT[j,4,*,i];
j=j+1
D2 checked?
true
false
i=0
i<2?
true
false
j=0
j<4?
false
true
i=i+1
Temp(j+1)[*,i] = temp(j+1)[*,i]
+ a_Diff_RI_ANEXT[j,5,*,i];
j=j+1

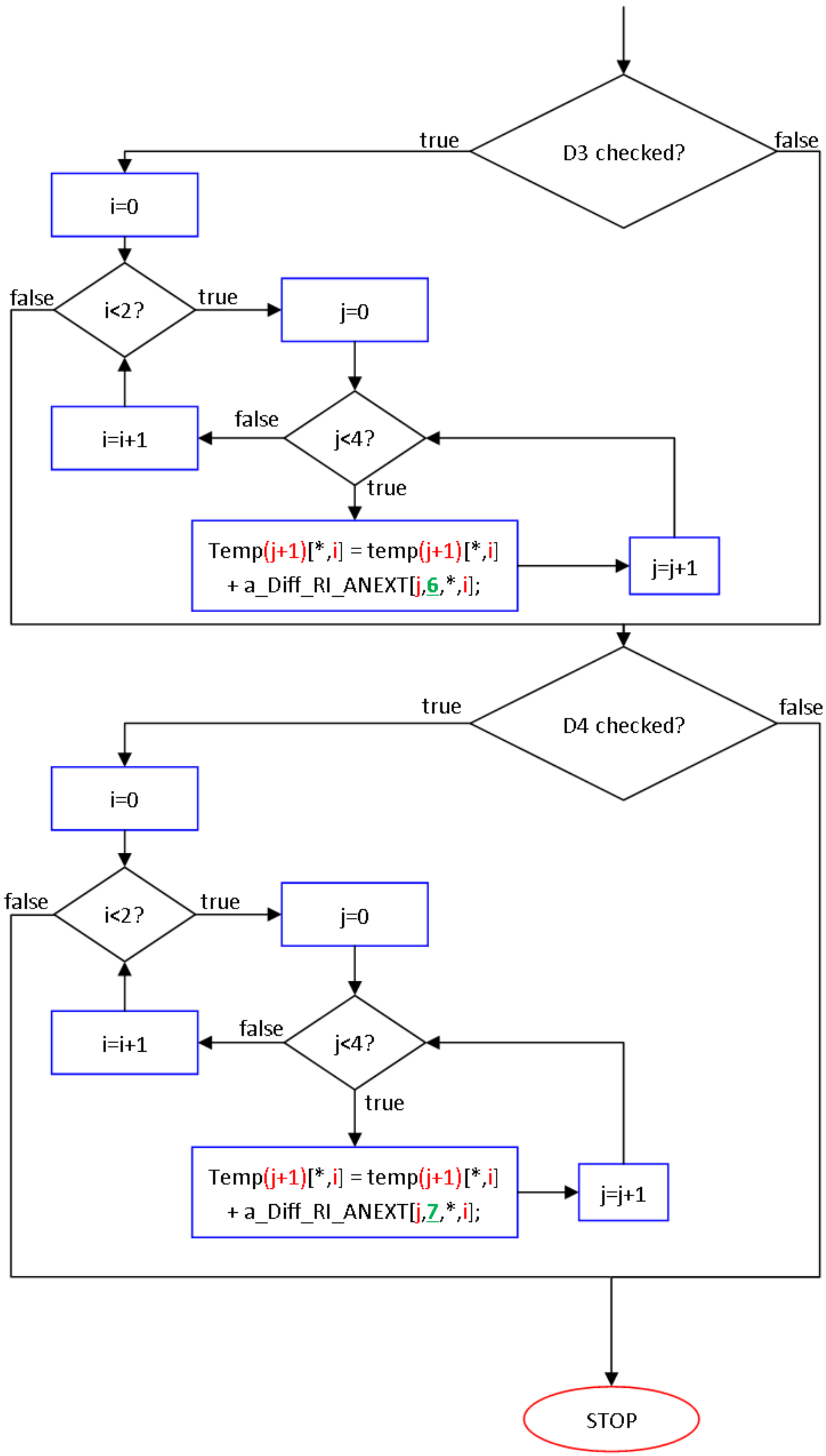

D3 checked?
true
false
i=0
i<2?
false
true
j=0
j<4?
false
true
i=i+1
Temp(j+1)[*,i] = temp(j+1)[*,i] + a_Diff_RI_ANEXT[j,6,*,i];
j=j+1
D4 checked?
true
false
i=0
i<2?
false
true
j=0
j<4?
false
true
i=i+1
Temp(j+1)[*,i] = temp(j+1)[*,i] + a_Diff_RI_ANEXT[j,7,*,i];
j=j+1
STOP

Table 18: Alien Next END1 Disturber 1

i	j	Formula	S-Parameter equivalent
0	0	Temp1[*,0] = temp1[*,0] + a_Diff_RI_ANEXT[0,4,*,0]	S15 (real)
0	1	Temp2[*,0] = temp2[*,0] + a_Diff_RI_ANEXT[1,4,*,0]	S25 (real)
0	2	Temp3[*,0] = temp3[*,0] + a_Diff_RI_ANEXT[2,4,*,0]	S35 (real)
0	3	Temp4[*,0] = temp4[*,0] + a_Diff_RI_ANEXT[3,4,*,0]	S45 (real)
1	0	Temp1[*,1] = temp1[*,1] + a_Diff_RI_ANEXT[0,4,*,1]	S15 (image)
1	1	Temp2[*,1] = temp2[*,1] + a_Diff_RI_ANEXT[1,4,*,1]	S25 (image)
1	2	Temp3[*,1] = temp3[*,1] + a_Diff_RI_ANEXT[2,4,*,1]	S35 (image)
1	3	Temp4[*,1] = temp4[*,1] + a_Diff_RI_ANEXT[3,4,*,1]	S45 (image)

Table 19: Alien Next END1 Disturber 2

i	j	Formula	S-Parameter equivalent
0	0	Temp1[*,0] = temp1[*,0] + a_Diff_RI_ANEXT[0,5,*,0]	S16 (real)
0	1	Temp2[*,0] = temp2[*,0] + a_Diff_RI_ANEXT[1,5,*,0]	S26 (real)
0	2	Temp3[*,0] = temp3[*,0] + a_Diff_RI_ANEXT[2,5,*,0]	S36 (real)
0	3	Temp4[*,0] = temp4[*,0] + a_Diff_RI_ANEXT[3,5,*,0]	S46 (real)
1	0	Temp1[*,1] = temp1[*,1] + a_Diff_RI_ANEXT[0,5,*,1]	S16 (image)
1	1	Temp2[*,1] = temp2[*,1] + a_Diff_RI_ANEXT[1,5,*,1]	S26 (image)

1	2	Temp3[*,1] = temp3[*,1] + a_Diff_RI_ANEXT[2,5,*,1]	S36 (image)
1	3	Temp4[*,1] = temp4[*,1] + a_Diff_RI_ANEXT[3,5,*,1]	S46 (image)

Table 20: Alien Next END1 Disturber 3

i	j	Formula	S-Parameter equivalent
0	0	Temp1[*,0] = temp1[*,0] + a_Diff_RI_ANEXT[0,6,*,0]	S17 (real)
0	1	Temp2[*,0] = temp2[*,0] + a_Diff_RI_ANEXT[1,6,*,0]	S27 (real)
0	2	Temp3[*,0] = temp3[*,0] + a_Diff_RI_ANEXT[2,6,*,0]	S37 (real)
0	3	Temp4[*,0] = temp4[*,0] + a_Diff_RI_ANEXT[3,6,*,0]	S47 (real)
1	0	Temp1[*,1] = temp1[*,1] + a_Diff_RI_ANEXT[0,6,*,1]	S17 (image)
1	1	Temp2[*,1] = temp2[*,1] + a_Diff_RI_ANEXT[1,6,*,1]	S27 (image)
1	2	Temp3[*,1] = temp3[*,1] + a_Diff_RI_ANEXT[2,6,*,1]	S37 (image)
1	3	Temp4[*,1] = temp4[*,1] + a_Diff_RI_ANEXT[3,6,*,1]	S47 (image)

Table 21: Alien Next END1 Disturber 4

i	j	Formula	S-Parameter equivalent
0	0	Temp1[*,0] = temp1[*,0] + a_Diff_RI_ANEXT[0,7,*,0]	S18 (real)
0	1	Temp2[*,0] = temp2[*,0] + a_Diff_RI_ANEXT[1,7,*,0]	S28 (real)
0	2	Temp3[*,0] = temp3[*,0] + a_Diff_RI_ANEXT[2,7,*,0]	S38 (real)
0	3	Temp4[*,0] = temp4[*,0] + a_Diff_RI_ANEXT[3,7,*,0]	S48 (real)

1	0	Temp1[*,1] = temp1[*,1] + a_Diff_RI_ANEXT[0,7,*,1]	S18 (image)
1	1	Temp2[*,1] = temp2[*,1] + a_Diff_RI_ANEXT[1,7,*,1]	S28 (image)
1	2	Temp3[*,1] = temp3[*,1] + a_Diff_RI_ANEXT[2,7,*,1]	S38 (image)
1	3	Temp4[*,1] = temp4[*,1] + a_Diff_RI_ANEXT[3,7,*,1]	S48 (image)

Therefore if all four disturbers are checked:

$$temp1[*,0] = a\_Diff\_RI\_ANEXT[0,4,*,0] + a\_Diff\_RI\_ANEXT[0,5,*,0] + a\_Diff\_RI\_ANEXT[0,6,*,0] + a\_Diff\_RI\_ANEXT[0,7,*,0] \quad (29)$$

$$temp2[*,0] = a\_Diff\_RI\_ANEXT[1,4,*,0] + a\_Diff\_RI\_ANEXT[1,5,*,0] + a\_Diff\_RI\_ANEXT[1,6,*,0] + a\_Diff\_RI\_ANEXT[1,7,*,0] \quad (30)$$

$$temp3[*,0] = a\_Diff\_RI\_ANEXT[2,4,*,0] + a\_Diff\_RI\_ANEXT[2,5,*,0] + a\_Diff\_RI\_ANEXT[2,6,*,0] + a\_Diff\_RI\_ANEXT[2,7,*,0] \quad (31)$$

$$temp4[*,0] = a\_Diff\_RI\_ANEXT[3,4,*,0] + a\_Diff\_RI\_ANEXT[3,5,*,0] + a\_Diff\_RI\_ANEXT[3,6,*,0] + a\_Diff\_RI\_ANEXT[3,7,*,0] \quad (32)$$

The s-parameter equivalent would be:

$$temp1[*,0] = S15 + S16 + S17 + S18 \quad (33)$$

$$temp1[*,0] = S25 + S26 + S27 + S28 \quad (34)$$

$$temp1[*,0] = S35 + S36 + S37 + S38 \quad (35)$$

$$temp1[*,0] = S45 + S46 + S47 + S48 \quad (36)$$

And vice-versa for the image part.

Unchecking the disturber 3 would give you:

$$temp1[*,0] = S15 + S16 + S18 \quad (37)$$

$$temp1[*,0] = S25 + S26 + S28 \quad (38)$$

$$temp1[*,0] = S35 + S36 + S38 \quad (39)$$

$$temp1[*,0] = S45 + S46 + S48 \quad (40)$$

5.9.4. Modification

5.10. ALIEN BUNDLE CROSSTALK

This is for testing the crosstalk of multiples connector around a victim. In this case, the disturbers are different jack on the same panel. For that reason, checking the disturbers will affect the total power sum of the previous formula.

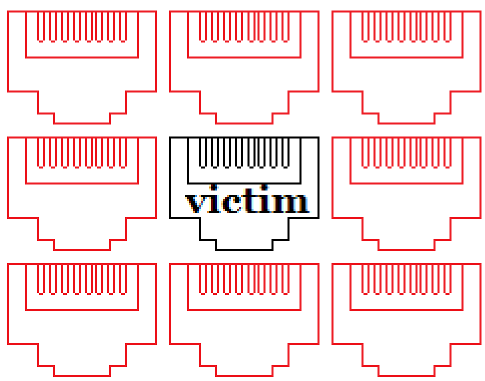


Figure 15: Representation of alien bundle disturbers affecting a victim

5.10.1. Form

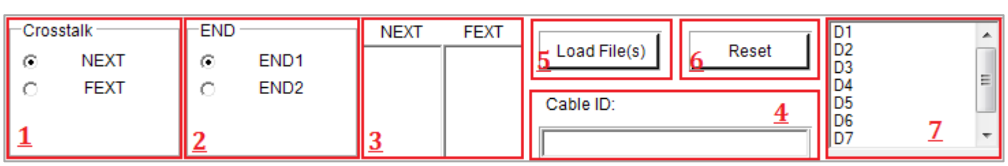


Figure 16: Alien bundle crosstalk's software form

1. Crosstalk
   selects between the NEXT and FEXT measurements.

2. End
   selects between the END1 and END2 measurements.

3. NEXT/FEXT
   these are the list of the current measured disturbers for the NEXT and FEXT.

4. Cable ID
   this will determine the name of the folder and the touchstone files.

5. Load Files
   this is for loading your own touchstone files without measuring them.

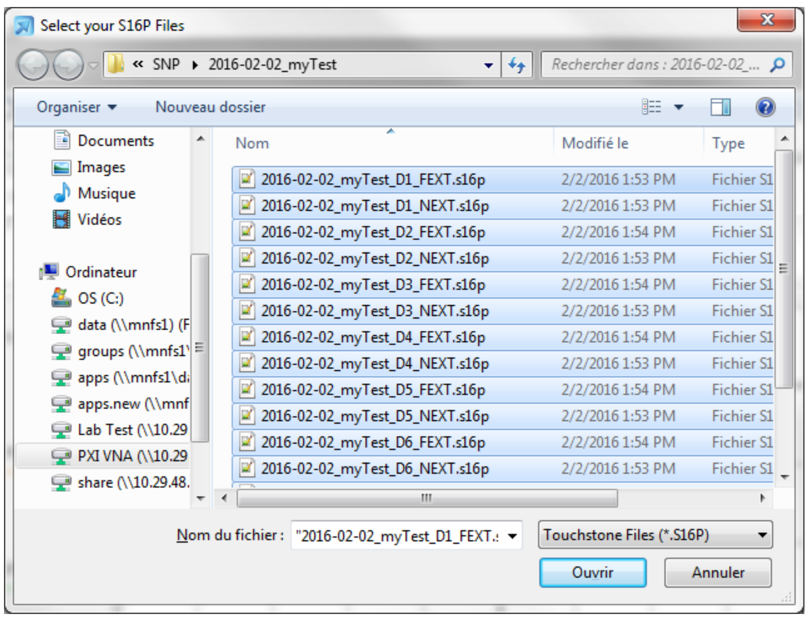


Figure 17: Loading s16p files via the load button

6. Reset
   Clears the NEXT and FEXT measurements.

7. Disturbers
   Measures and disturbers individually. To do so, you must double click on one of the disturbers and it will automatically add it to the measurements list (#3).

5.10.2. Data

This is how the data is calculated in the program. It calculates the power sum of each victim, then does the sum of every disturbers for each power sum previously calculated.

END 1:

$$PS1_{D1} = Sdd15_{D1} + Sdd16_{D1} + Sdd17_{D1} + Sdd18_{D1} \tag{41}$$

$$PS2_{D1} = Sdd25_{D1} + Sdd26_{D1} + Sdd27_{D1} + Sdd28_{D1} \tag{42}$$

$$PS3_{D1} = Sdd35_{D1} + Sdd36_{D1} + Sdd37_{D1} + Sdd38_{D1} \tag{43}$$

$$PS4_{D1} = Sdd45_{D1} + Sdd46_{D1} + Sdd47_{D1} + Sdd48_{D1} \tag{44}$$

…

$$PS1_{D2} = Sdd15_{D2} + Sdd16_{D2} + Sdd17_{D2} + Sdd18_{D2} \tag{45}$$

$$PS2_{D2} = Sdd25_{D2} + Sdd26_{D2} + Sdd27_{D2} + Sdd28_{D2} \tag{46}$$

$$PS3_{D2} = Sdd35_{D2} + Sdd36_{D2} + Sdd37_{D2} + Sdd38_{D2} \tag{47}$$

$$PS4_{D2} = Sdd45_{D2} + Sdd46_{D2} + Sdd47_{D2} + Sdd48_{D2} \tag{48}$$

END 2:

$$PS1_{D1} = Sdd51_{D1} + Sdd52_{D1} + Sdd53_{D1} + Sdd54_{D1} \tag{49}$$

$$PS2_{D1} = Sdd61_{D1} + Sdd62_{D1} + Sdd63_{D1} + Sdd64_{D1} \tag{50}$$

$$PS3_{D1} = Sdd71_{D1} + Sdd72_{D1} + Sdd73_{D1} + Sdd74_{D1} \tag{51}$$

$$PS4_{D1} = Sdd81_{D1} + Sdd82_{D1} + Sdd83_{D1} + Sdd84_{D1} \tag{52}$$

…

$$PS1_{D2} = Sdd51_{D2} + Sdd52_{D2} + Sdd53_{D2} + Sdd53_{D2} \tag{53}$$

$$PS2_{D2} = Sdd61_{D2} + Sdd62_{D2} + Sdd63_{D2} + Sdd64_{D2} \tag{54}$$

$$PS3_{D2} = Sdd71_{D2} + Sdd72_{D2} + Sdd73_{D2} + Sdd74_{D2} \tag{55}$$

$$PS4_{D2} = Sdd81_{D2} + Sdd82_{D2} + Sdd83_{D2} + Sdd84_{D2} \tag{56}$$

…

$$PS_{Pair1} = PS1_{D1} + PS1_{D2} + PS1_{D3} + PS1_{D4} + PS1_{D5} + PS1_{D6} + PS1_{D7} + PS1_{D8} \tag{57}$$

$$PS_{Pair2} = PS2_{D1} + PS2_{D2} + PS2_{D3} + PS2_{D4} + PS2_{D5} + PS2_{D6} + PS2_{D7} + PS2_{D8} \tag{58}$$

…

Finally, if we would want to only show D1 and D7 for END 1 :

$$PS1_{D1} = Sdd15_{D1} + Sdd16_{D1} + Sdd17_{D1} + Sdd18_{D1} \tag{59}$$

$$PS2_{D1} = Sdd25_{D1} + Sdd26_{D1} + Sdd27_{D1} + Sdd28_{D1} \tag{60}$$

$$PS3_{D1} = Sdd35_{D1} + Sdd36_{D1} + Sdd37_{D1} + Sdd38_{D1} \tag{61}$$

$$PS4_{D1} = Sdd45_{D1} + Sdd46_{D1} + Sdd47_{D1} + Sdd48_{D1} \tag{62}$$

…

$$PS1_{D2} = Sdd15_{D7} + Sdd16_{D7} + Sdd17_{D7} + Sdd18_{D7} \tag{63}$$

$$PS2_{D2} = Sdd25_{D7} + Sdd26_{D7} + Sdd27_{D7} + Sdd28_{D7} \tag{64}$$

$$PS3_{D2} = Sdd35_{D7} + Sdd36_{D7} + Sdd37_{D7} + Sdd38_{D7} \tag{65}$$

$$PS4_{D2} = Sdd45_{D7} + Sdd46_{D7} + Sdd47_{D7} + Sdd48_{D7} \tag{66}$$

…

$$PS_{Pair1} = PS1_{D1} + PS1_{D7} \tag{67}$$

$$PS_{Pair2} = PS2_{D1} + PS2_{D7} \tag{68}$$

…

5.10.3.Algorithm

To simplify the formula, we use an algorithm to calculate every power sum. Here's the algorithm for the software's graph.

First off, here are the arrays we are going to use:

A_AlienBundle_D1 to A_AlienBundle_D2: This only contains de parameters for the **NEXT** from END 1 to END 2.

Size: 5 Dimension arrays: [4,4,Number Of Sweeps,2,2]
The first two digits are for the data. The last two digits is for the Real/Image and END1/END2.

- [*,*,*,0,0] is real END1
- [*,*,*,1,0] is image END1
- [*,*,*,0,1] is real END2
- [*,*,*,1,1] is image END2

If we would show [*,*,0,0,0]:

Table 22: Real END1 S Parameters at the first frequency point

S15	S16	S17	S18
S25	S26	S27	S28
S35	S36	S37	S38
S45	S46	S47	S48

A_AlienBundleF_D1 to A_AlienBundleF_D2: This only contains de parameters for the **FEXT** from END 1 to END 2.

Size: 5 Dimension arrays: [4,4,Number Of Sweeps,2,2]
The first two digits are for the data. The last two digits is for the Real/Image and END1/END2.

- [*,*,*,0,0] is real END1
- [*,*,*,1,0] is image END1
- [*,*,*,0,1] is real END2
- [*,*,*,1,1] is image END2

If we would show [*,*,0,0,0]:

Table 23: Real END1 S Parameters at the first frequency point

S51	S52	S53	S54
S61	S62	S63	S64
S71	S72	S73	S74
S81	S82	S83	S84

Temp1 to Temp4: These are the arrays using temporary for holding the data before converting it to dB.

Size: [Number of Sweeps, 2]
The last digit is for the Real/Image

- [*,0] is real
- [*,1] is image

Pair1 to Pair4: These are the arrays of the data converted in dB

Size: [Number of Sweeps, 1]

Since either FEXT or NEXT and END1 or END2 is shown, you only need a 1D array to show the data on the graph. How, the reason we use a 2D arrays with the second dimension being size 1, is for other functions. Vee differentiates 2D arrays and 1D arrays even if they are the same size, so 1D arrays may not work for some functions like combining two arrays.

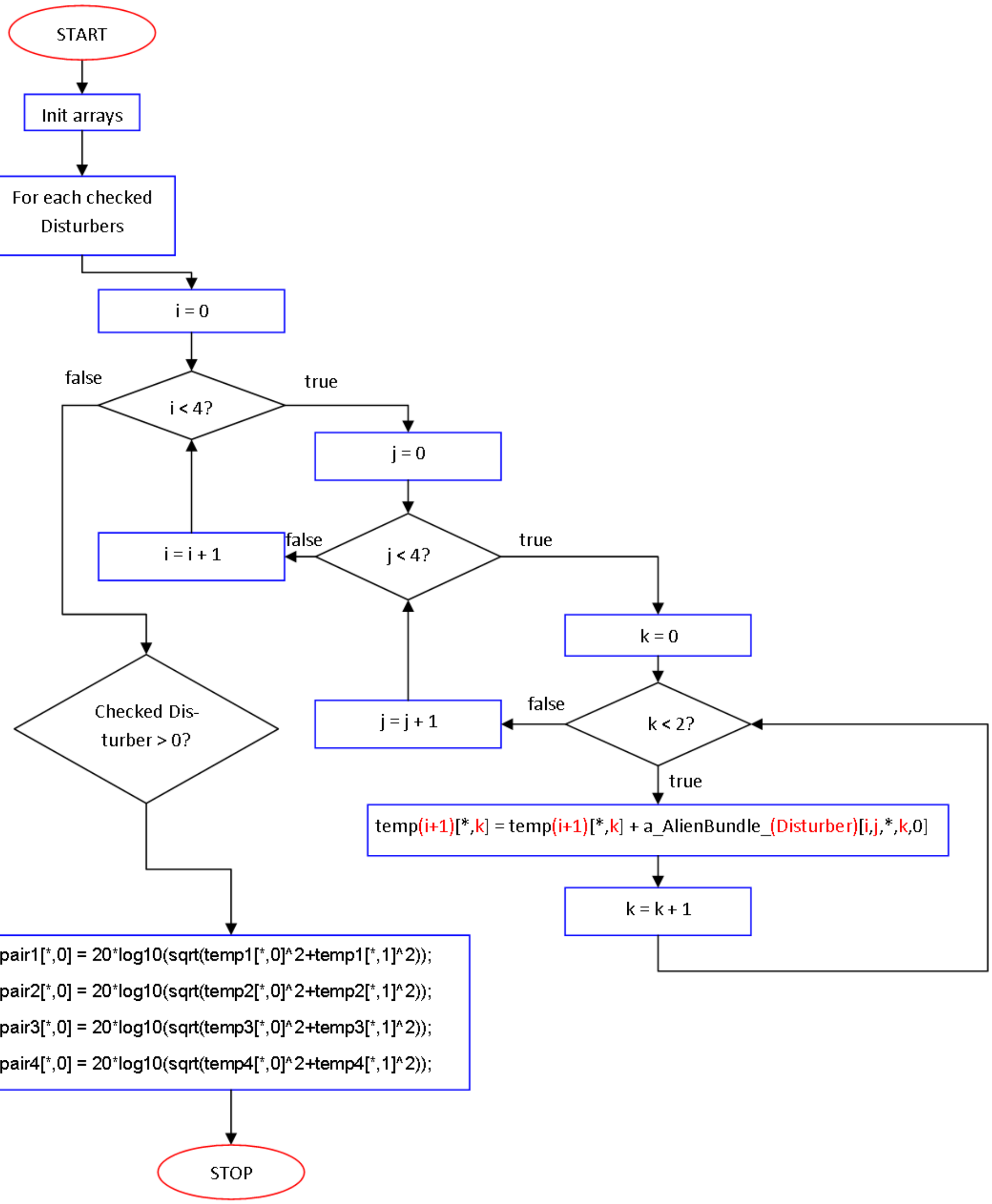


Figure 18: Alien Bundle crosstalk calculation workflow

*PS: There are 4 possible formulas for this algorithm:

$$temp(i+1)[*,k] = temp(i+1)[*,k] + a\_AlienBundle\_(Disturber)[i,j,*,k,0] \quad (69)$$

$$temp(i+1)[*,k] = temp(i+1)[*,k] + a\_AlienBundle\_(Disturber)[i,j,*,k,1] \quad (70)$$

$$temp(i+1)[*,k] = temp(i+1)[*,k] + a\_AlienBundleF\_(Disturber)[i,j,*,k,0] \quad (71)$$

$$temp(i+1)[*,k] = temp(i+1)[*,k] + a\_AlienBundleF\_(Disturber)[i,j,*,k,1] \quad (72)$$

Equation (1) and equation (2) calculates the END1 and END2 NEXT respectively while Equation (51) and Equation (52) calculates the END1 and END2 FEXT. (EQUATIONS ARE FIELDS SO THEY CAN CHANGE)

Here are a few iterations to show what the algorithm actually does:

Table 24: Disturber1 iterations

i	j	k	Formula
0	0	0	Temp1[*,0] = Temp1[*,0] + a_AlienBundle_D1[0,0,*,0,0]
0	0	1	Temp1[*,1] = Temp1[*,1] + a_AlienBundle_D1[0,0,*,1,0]
0	1	0	Temp1[*,0] = Temp1[*,0] + a_AlienBundle_D1[0,1,*,0,0]
0	1	1	Temp1[*,1] = Temp1[*,1] + a_AlienBundle_D1[0,1,*,1,0]
0	2	0	Temp1[*,0] = Temp1[*,0] + a_AlienBundle_D1[0,2,*,0,0]
0	2	1	Temp1[*,1] = Temp1[*,1] + a_AlienBundle_D1[0,2,*,1,0]
0	3	0	Temp1[*,0] = Temp1[*,0] + a_AlienBundle_D1[0,3,*,0,0]
0	3	1	Temp1[*,1] = Temp1[*,1] + a_AlienBundle_D1[0,3,*,1,0]
1	0	0	Temp2[*,0] = Temp2[*,0] + a_AlienBundle_D1[1,0,*,0,0]
1	0	1	Temp2[*,1] = Temp2[*,1] + a_AlienBundle_D1[1,0,*,1,0]

Table 25: Disturber2 iterations

i	j	k	Formula
0	0	0	Temp1[*,0] = Temp1[*,0] + a_AlienBundle_D2[0,0,*,0,0]

0	0	1	Temp1[*,1] = Temp1[*,1] + a_AlienBundle_D2[0,0,*,1,0]
0	1	0	Temp1[*,0] = Temp1[*,0] + a_AlienBundle_D2[0,1,*,0,0]
0	1	1	Temp1[*,1] = Temp1[*,1] + a_AlienBundle_D2[0,1,*,1,0]
0	2	0	Temp1[*,0] = Temp1[*,0] + a_AlienBundle_D2[0,2,*,0,0]
0	2	1	Temp1[*,1] = Temp1[*,1] + a_AlienBundle_D2[0,2,*,1,0]
0	3	0	Temp1[*,0] = Temp1[*,0] + a_AlienBundle_D2[0,3,*,0,0]
0	3	1	Temp1[*,1] = Temp1[*,1] + a_AlienBundle_D2[0,3,*,1,0]
1	0	0	Temp2[*,0] = Temp2[*,0] + a_AlienBundle_D2[1,0,*,0,0]
1	0	1	Temp2[*,1] = Temp2[*,1] + a_AlienBundle_D2[1,0,*,1,0]

$$\begin{aligned}
&Temp1[*,0](Real) = \\
&a_AlienBundle_D1[0,0,*,0,0] + a_AlienBundle_D1[0,1,*,0,0] + \\
&a_AlienBundle_D1[0,2,*,0,0] + a_AlienBundle_D1[0,3,*,0,0] + \\
&a_AlienBundle_D2[0,0,*,0,0] + a_AlienBundle_D2[0,1,*,0,0] + \\
&a_AlienBundle_D2[0,2,*,0,0] + a_AlienBundle_D2[0,3,*,0,0]
\end{aligned} \tag{73}$$

$$\begin{aligned}
&Temp1[*,1](Image) = \\
&a_AlienBundle_D1[0,0,*,1,0] + a_AlienBundle_D1[0,1,*,1,0] + \\
&a_AlienBundle_D1[0,2,*,1,0] + a_AlienBundle_D1[0,3,*,1,0] + \\
&a_AlienBundle_D2[0,0,*,1,0] + a_AlienBundle_D2[0,1,*,1,0] + \\
&a_AlienBundle_D2[0,2,*,1,0] + a_AlienBundle_D2[0,3,*,1,0]
\end{aligned} \tag{74}$$

This example only shows the first pair of D1 and D2, but it will loops for every checked item in the check list.

5.10.4. Detail

This function calculates the margin between the limit and the values for each pair of the cable. Here's the formula for every pair:

$$array_{marge} = |array_{limit}| - |array_{loss}| \quad (75)$$

As a result, if one of the values is negative in the resulting array, the result is fail. The lowest value in the array also means it's the minimum marge value that we want to show in the detail Form.

Meanwhile, even though the limits go farther than what TIA specifies, the marges are only calculated within the range the specifications indicate.

5.10.5. Modification

Clicking on the alien bundle menu strip calls the event handle(or function because the IDE sometimes likes to create a function) called "*CalcAlienBundleMenu_Click*". This function simply initializes the a_AlienBundle arrays and call the process function called "*FormBundleAlienXT*".

This function does every processing needed for the bundle alien, and also initializes the .net framework objects if they weren't initialized before. Every button or checklist simply recalls this function to recalculate the new values and update the graph.

The data is fetched from the checklist event handler called: "*listboxBundleAlien_DoubleClick*". This function is called everytime you double click on one of the disturbers in the listbox. This function also adds the check boxes in the checked list box called "*ChkListFEXT*" and "*ChkListNEXT*".

If you want to change any formulas, they are located in the "*Process Data*" userblock.

If you want to change the limits, you can change the text arrays in the "Process Limits" userblock. They are processed using the function "BuildLimitArray".

5.10.6. Export Report

This section is about the generation of the excel files. The user can decide to generates excel charts with every loss calculations and charts associated with them. They can then use a preconfigured macro to reposition and resize them.

5.11. GENERATE EXCEL FILE

Agilent VEE Pro has an Excel library that can be used to create Excel worksheets. When the user decides to create an Excel report, the software waits for an user input for the cable ID. The cable ID is going to be used for the file name, created in the report folder located in the root folder. The exact name of the file will be [YYYY]-[MM]-[DAY]_[CableID].xlsx. A sheet will be created for each loss calculations and a sheet will be created for all the charts.

The data used in Excel are the following to make 6 charts:

Table 26: Excel generated charts for a Through DUT

16 Port (8 Port Diff)	
Return Loss	Insertion Loss
Sdd11, Sdd22, Sdd33, Sdd44, Sdd55, Sdd66, Sdd77, Sdd88	Sdd15, Sdd26, Sdd37, Sdd48, Sdd51, Sdd62, Sdd73, Sdd84
END1 NEXT	END1 FEXT
Sdd12, Sdd13, Sdd14, Sdd23, Sdd24, Sdd34	Sdd16, Sdd17, Sdd18, Sdd25, Sdd27, Sdd28, Sdd35, Sdd36, Sdd38, Sdd45, Sdd46, Sdd47
END2 NEXT	END2 FEXT
Sdd56, Sdd57, Sdd58, Sdd67, Sdd68, Sdd78	Sdd52, Sdd53, Sdd54, Sdd61, Sdd63, Sdd64, Sdd71, Sdd72, Sdd74, Sdd81, Sdd82, Sdd83

Table 27: Excel generated charts for a Load DUT

16 Port (8 Port Diff)	
Return Loss	NEXT
Sdd11, Sdd22, Sdd33, Sdd44	Sdd12, Sdd13, Sdd14, Sdd23, Sdd24, Sdd34

5.11.1.Graph macro

When using the excel library on the Agilent VEE Pro 9.32, there are a few problems that don't seem to be easy to fix. If you decide to put every chart in the same sheet and change their size and location, the library reuses the same graph every time and can't notice the difference between the first chart created and the others. Therefore, if you generate a report using this program, you'll notice how every chart is stacked on top of each other. This is because the original location when a chart is generated by Agilent is that location. However, if you zoom out you can also notice the return loss chart being oversized, since it's the first graph created by Excel's library.

To fix this problem, I integrated a macro in the macro folder. This macro resizes and repositions all the charts in the chart sheet. You can either install it directly using the install.bat or copy the code in your own personal.xlsb. This file is located in:

C:\Users\<username>\AppData\Roaming\Microsoft\Excel\XLSTART.

Personal.xlsb is a special Excel binary file executed when opening a standard excel file. Every macro presents in that file will be available to your excel files on the same computer.

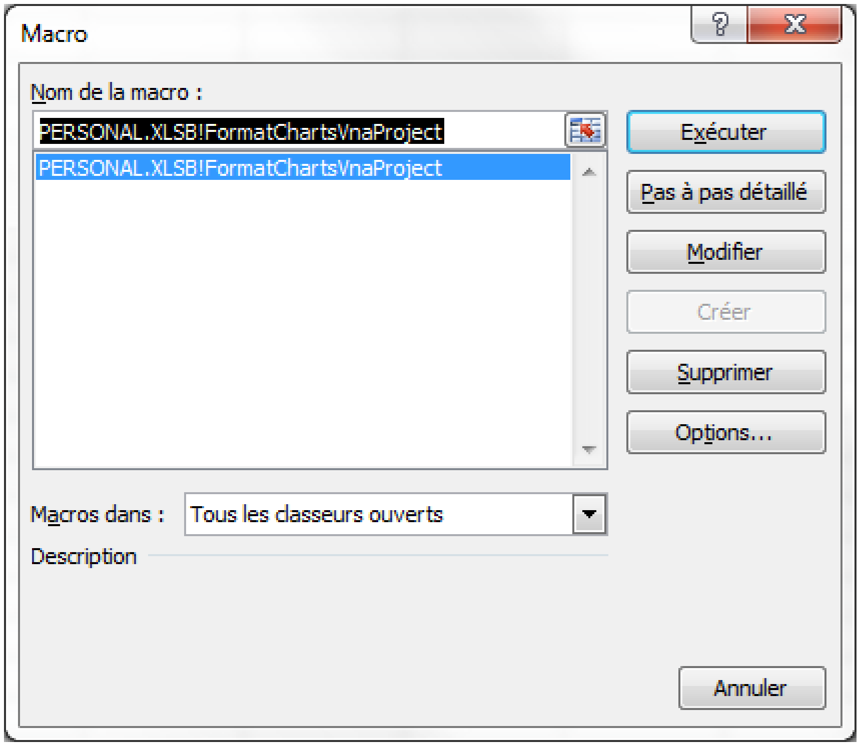


Figure 19: Macro used to rearrange the charts

Excel's macro window, it can be access in developer tab, or by pressing ALT+F8

PS: If there's already a personal.xlsb file in your folder, the program will ask you if you want to make a backup of it and replace it. The backup is going to be in your macro folder. However, it is recommended that you simply add the macro in it, since you probably already have personal macros.

5.11.2. Show data

This section is for visualizing the data before saving it. Nothing will be kept on the hard drive.

5.12. GENERATE TEMPLATE

Using the template made for this software, you can export the data you need by using this function. It exports the data in RawData. The same data is also processed by the software to calculate the marges, worst and averages values.

The data is also separated in two different interfaces: the telecom side and the work area. These are determined by the ports used to take the measurements. Ports 1 through 8 are telecom room and ports 9 through 16 are work area.

There's two main function for this module, Raw data and Reports.

RawData: Contains the functions for the raw data and the resume. The raw data are used for the graphs in the resume. This function does more than just putting the data in the cells. It also need to reorganize the matrices, because they are not stored the same way as the table in the Excel template. To fix this issue, we use a sort matrix.

Resume is the marging, average and worst values for every loss. The marge is the marge minus the value. Worst value is the highest marging values in a specific end, and finally the average is the mean value of a specific end.

Reports: The reports sheet are the values for the important freq points next to the charts:

$$\{1, 4, 8, 10, 6, 20, 25, 31, 62, 100, 160, 200, 250, 300, 350, 400, 450, 500, 550, 600, 625\}$$

This function only works with the specified template used this software. It will export every data needed, from the software and the snp files, and paste it on the template.

This function requires a few snp files to work:

Plug

- Case 1: Mated, 16 port,
- Case 2: Mated, 8 port forward, with the delay correction enabled,
- Case 3: Mated, 8 port reverse, with the delay correction enabled,

Jack

- Case 1: Mated, 8 port forward (reembed)
- Case 1: Mated, 8 port reverse (reembed)

You can either do the next by doing the measure while importing the snps in the function, or by having them saved on your hardware.

For the function to work, you need to import the snp files in the program. You do this by importing them via the normal functions, such as “Import from snp” or “Import from VNA” for the Plug Case 1-2-3. Since these does not require reembedding, you don’t have anything else to do. Once you have imported them in the software, simply use the Export to Template function in the file menu and click the case you need to save in the program. For case 2 and 3, since you need deembeding, you need to make you sure to apply the delay correction accordingly. Also, you might need to set the DUT to load since you are probably using a 8 port balunless with a load.

The jack’s cases are a little bit different. Since you need to do a reembeding on the port, you need to use another function. You need to use the deembeded function normally used. There, you can measure the mated and direct fixture used to calculate the reembed cases. Once this step is done, you can then import it to the template function. You can repeat the same steps for the case 2 of the jack.

The case buttons simply take the previously imported snp files and save it in a temporary variable for future use. Once you have imported all the cases you need, you can export it to the template.

To change the limits, you need to go in the belden folder and change the formulas manually in the formula.vee file.

5.13. SHOW GRAPH

Graphs on Agilent VEE can be displayed using the X vs Y Plot block in the display menu. One of the problem of using the excel sheet is not being able to set the x axis to logarithmic. However, with this graph we can do that and change other useful things such as the color of the traces, their name etc.

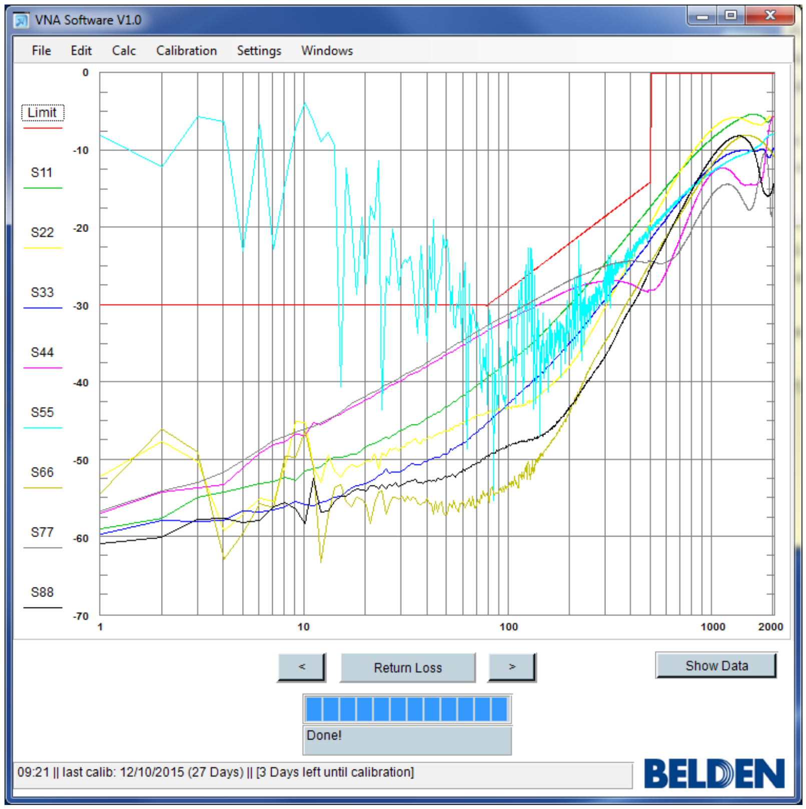


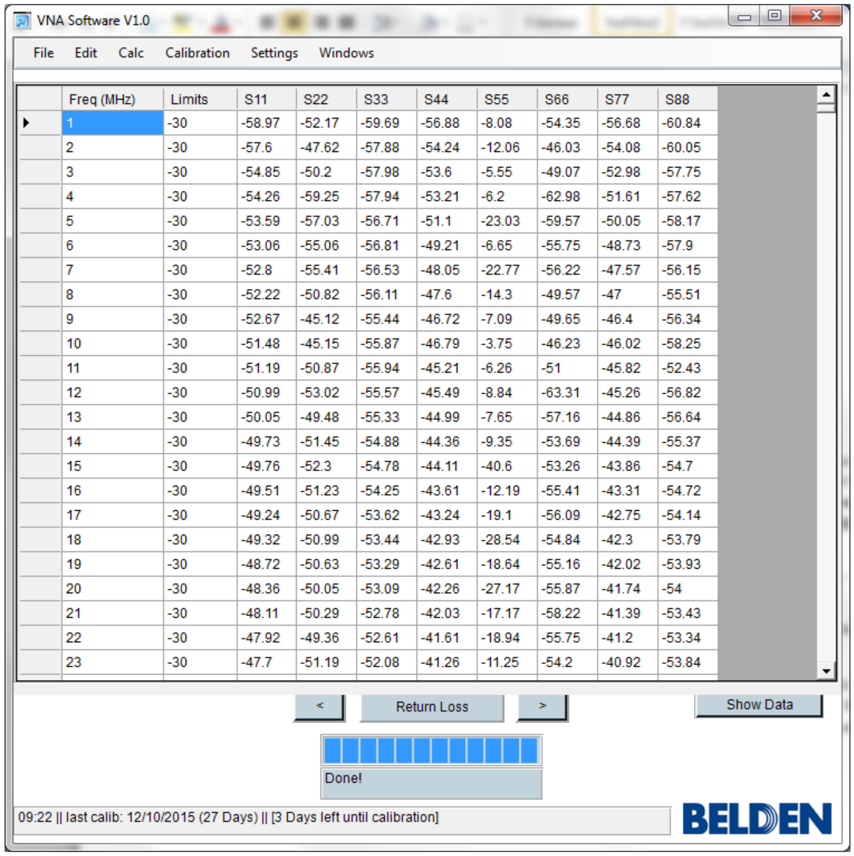
Figure 20: X vs Y Plot showing the Return Loss

PS: If you change the traces names, and not every trace name fits vertically, the chart will auto scale and cause it to extend outside the borders of the form. I fixed this issue by resizing the graph to its normal size after doing the manipulation. Also, we can change the colors of the traces; however the number of colors seems very limited, going from integer 0 to 9. Anything more than ten traces will have the same color.

5.14. SHOW NUMERIC DATA

This is simply a method of showing the data processed earlier. The program generates a data table and puts the frequency, limits and values in the table.

One of the problems of using the notepad to display data is its launch time. A 2000 sweeps and 16 ports touchstone file is around 10Mo. using notepad, it takes a long time to actually open the file. A workaround is using a better file editor, such as notepad++, however not every computer has it and/or have admin privilege to install it.



Freq (MHz)	Limits	S11	S22	S33	S44	S55	S66	S77	S88
1	-30	-58.97	-52.17	-59.69	-56.88	-8.08	-54.35	-56.68	-60.84
2	-30	-57.6	-47.62	-57.88	-54.24	-12.06	-46.03	-54.08	-60.05
3	-30	-54.85	-50.2	-57.98	-53.6	-5.55	-49.07	-52.98	-57.75
4	-30	-54.26	-59.25	-57.94	-53.21	-6.2	-62.98	-51.61	-57.62
5	-30	-53.59	-57.03	-56.71	-51.1	-23.03	-59.57	-50.05	-58.17
6	-30	-53.06	-55.06	-56.81	-49.21	-6.65	-55.75	-48.73	-57.9
7	-30	-52.8	-55.41	-56.53	-48.05	-22.77	-56.22	-47.57	-56.15
8	-30	-52.22	-50.82	-56.11	-47.6	-14.3	-49.57	-47	-55.51
9	-30	-52.67	-45.12	-55.44	-46.72	-7.09	-49.65	-46.4	-56.34
10	-30	-51.48	-45.15	-55.87	-46.79	-3.75	-46.23	-46.02	-58.25
11	-30	-51.19	-50.87	-55.94	-45.21	-6.26	-51	-45.82	-52.43
12	-30	-50.99	-53.02	-55.57	-45.49	-8.84	-63.31	-45.26	-56.82
13	-30	-50.05	-49.48	-55.33	-44.99	-7.65	-57.16	-44.86	-56.64
14	-30	-49.73	-51.45	-54.88	-44.36	-9.35	-53.69	-44.39	-55.37
15	-30	-49.76	-52.3	-54.78	-44.11	-40.6	-53.26	-43.86	-54.7
16	-30	-49.51	-51.23	-54.25	-43.61	-12.19	-55.41	-43.31	-54.72
17	-30	-49.24	-50.67	-53.62	-43.24	-19.1	-56.09	-42.75	-54.14
18	-30	-49.32	-50.99	-53.44	-42.93	-28.54	-54.84	-42.3	-53.79
19	-30	-48.72	-50.63	-53.29	-42.61	-18.64	-55.16	-42.02	-53.93
20	-30	-48.36	-50.05	-53.09	-42.26	-27.17	-55.87	-41.74	-54
21	-30	-48.11	-50.29	-52.78	-42.03	-17.17	-58.22	-41.39	-53.43
22	-30	-47.92	-49.36	-52.61	-41.61	-18.94	-55.75	-41.2	-53.34
23	-30	-47.7	-51.19	-52.08	-41.26	-11.25	-54.2	-40.92	-53.84

Figure 21: Using the function to display the Return Loss

5.15. CALIBRATION

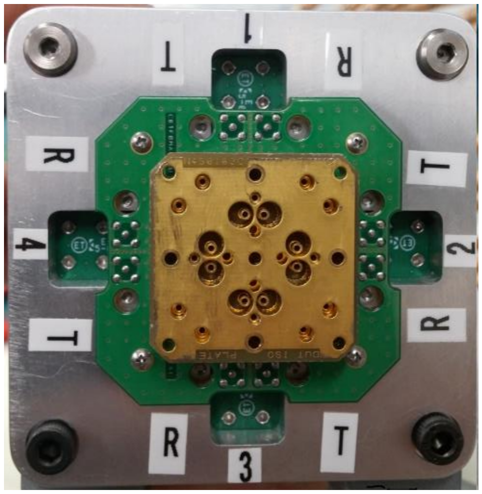
This calibration is done on the VNA's firmware. This software will determine the next calibration date and remind the user if a calibration is needed.

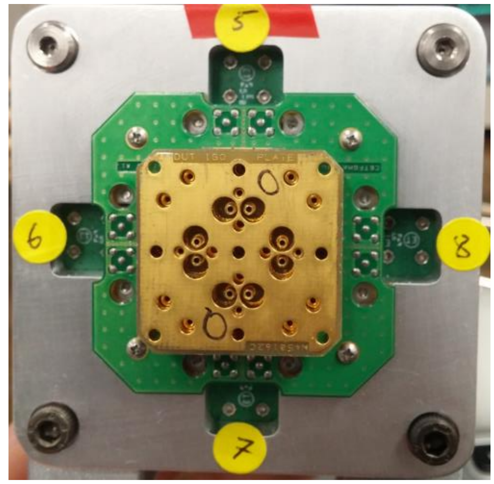
This function prompts the user with the calibration instruction. Once the user is ready, they can accept to save the state, and replace the old one.

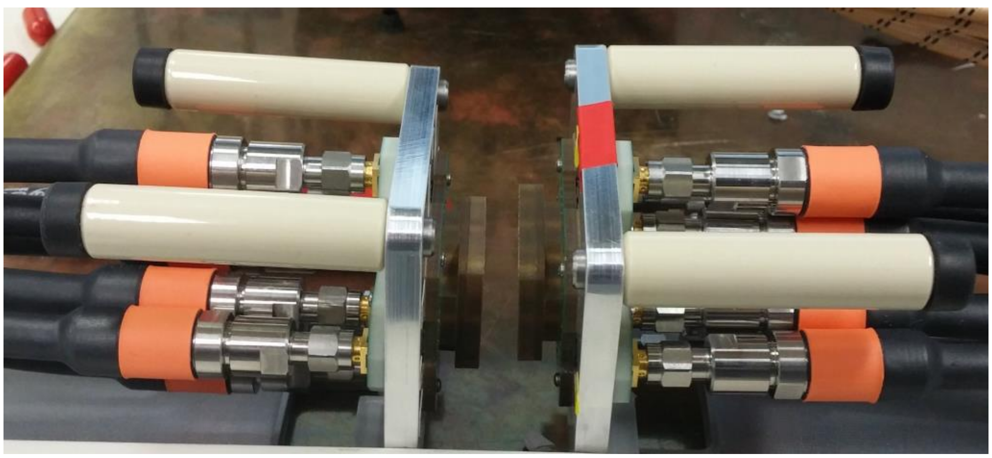
When the program imports the s-parameters from the VNA, it firsts verify that the calibration state is present. If the calibration isn't present, it will ask the user if they want to do a calibration or continue anyway. If there is a calibration but it's out of date (depending on the calibration interval) it will warn the user before doing its process.

5.15.1. Using the software

You can also use the software we made on Agilent Vee for the calibration. It makes the calibration much faster. There's also a few versions you can use. We did the calibration of the vna by putting the ports 1 to 8 as END 1 and ports 9 to 16 END 2. We then put the odd ports on tip and the even ports on ring.







- Calib8port.vee

  This is for the calibration of a 8 port configuration. It is configured to to every ports through except for 2-5 to 2-16.

- Calib16port.vee

  Basicly the same thing as the 8 ports but for a 16 ports calibration.

- Calib16portv2.vee

  The second version of the 16 port program. This removes the throughs ports that you can't do with the actual artifacts without turning the fixtures. Basicly, you only have a few types of artifacts and they can't cover every port configuration, unless you turn the fixture 90 degrees. You can use this excel sheet for ports configuration.

VNA calibration
template 2.xlsx

5.15.2. Set calibration interval

This button will prompt the user with a dialog box asking for an integer. This is the interval in days between the last calibration and today.

5.15.3. Set last calibration

This button manually sets the last calibration date. It is useful if the last calibration wasn't done using this software.

5.16. DELAY CORRECTION

5.16.1. Delay

We measure the delay of Sdd11 to Sdd44 from 100Mhz to 500Mhz using this formula:

$$Delay = \frac{-1}{360} * \frac{\Delta\varphi}{\Delta\theta} \tag{76}$$

$$\varphi = Phase\ in\ degrees \tag{77}$$

$$\theta = Frequency\ in\ Hertz \tag{78}$$

We then do the average of these measures for each Sdd.

$$Sddxx\ Delay = \frac{\left(Avg\left(Sddxx_{Open\ Delay}\right) + Avg\left(Sddxx_{Short\ Delay}\right)\right)}{4} \tag{79}$$

We can then calculate the delay from one pair to another by adding them. Once we have those values, we can correct the phases.

$$Phase_{corrected} = Phase_{old} + 360 * frequency_{Hertz} * Delay \tag{80}$$

The deembeding can be enabled or disabled using the edit category in the menu strip. When you decide to enable it, you can chose a delay file from the Delay Correction folder.

5.17. DEEMBEDDING

This module mainly explains the function *Deembed.*

5.17.1. Button: Short

This button measures a plug short sdd parameters and stores it in an array.

5.17.2. Button: Open

This button measures a plug open sdd parameters and stores it in an array.

5.17.3. Button: Calc Delay

Calculate de delay with the short and open measures. Reference: TIA-568-C.2, p.133

5.17.4. Button: Mated

Measure the first 8 ports (balunless) s-paramerers and stores the data in the arrays **a_diff_Db_Mated**, **a_Diff_Ri_Mated**.

NB: The delay correction must be enabled to mesure the mated values.

5.17.5. Button: Direct Fixture

Measure the first 8 ports (balunless) s-paramerers and stores the data in the array **a_Diff_Ri_DF**.

NB: The delay correction must be enabled to mesure the direct fixture values.

5.17.6. Button: Calc Re-Embed

This is where the main algorithms calculates the reembed values. To be able to calculate those values, we first need to import the mated and direct fixture data we need (you can also measure it directly with the PXI). Those values are already delay corrected, since we had the delay correction enabled in the previous steps.

With those values, we can calculate the jack vector.

$$a\_Diff\_RI\_JackVector = a\_Diff\_RI\_Mated - a\_Diff\_RI\_DF; \qquad (81)$$

And with this with can also get the db values using the standard formula.

Now, we need to get the case values. There is 14 possible case. Here is an example for the case one:

$$Case1_{mag} = -\left(38.1 - 20 * log10\left(\frac{f}{100}\right)\right) \tag{82}$$

$$Case1_{phase} = DF_{phase} * \frac{pi}{180} \tag{83}$$

$$Reembed_{real} = JackVector_{real} + Case1_{real} \tag{84}$$

The value we use 38.1 can be different from case to case, here are the values:

Case #	Value	Pair Combination
Case 1	38.1	3,6-4,5
Case 2	38.6	3,6-4,5
Case 3	39	3,6-4,5
Case 4	39.5	3,6-4,5
Case 5	46.5	1,2-3,6
Case 6	49.5	1,2-3,6
Case 7	46.5	3,6-7,8
Case 8	49.5	3,6-7,8
Case 9	57	1,2-4,5
Case 10	70	1,2-4,5
Case 11	57	4,5-7,8
Case 12	70	4,5-7,8
Case 13	66	1,2-7,8
Case 14	66	1,2-7,8

They can be found in the TIA document TIA-568-C.2, p.136.

5.17.7. Button: ReNext Charts

5.17.8. Button: Mated Charts

5.18. RETURN LOSS DEEMBEDDING

5.19. NEXT/FEXT DEEMBEDDING

5.20. PORT EXTENSION

This is for changing the reference plane of the DUT. You can refer to the TIA standard for further explaining.

5.21. AGILENT VEE PRO RUNTIME

This software is used to run the project without having a Agilent VEE Pro license.

5.22. HOW TO USE IT

5.22.1. Having admin privileges

You can download [Agilent VEE Runtime for free here](). To open the VNA project using the run time, you need to right click on it> open with> Use the standard Windows service

If the runtime isn't present, you need to browse and select it manually.

5.22.2. Issues

5.22.2.1. AGILENT COMMAND EXPERT

Makes the software not launch when using the runtime.

5.23. SETTINGS

The settings are the predefined values in the user interface. It also includes the port configuration in the port settings. When the program launches, it automatically loads the default settings from ..//Settings/defaultset.txt

5.23.1. Load

With this button, you can either reload the default settings, or load custom settings from your folder.

5.23.2. Save

If you want to have different settings for different DUTs, you can save the settings without having it load each time you relaunch the software.

5.23.3. Set as default

With this option, you can overwrite the default settings with the current settings you are using. The next time you will launch the software; those settings will be used and shown.

5.24. INFORMATION BAR

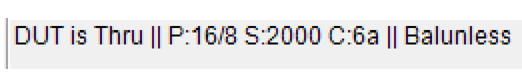


Figure 22: Information bar located at the bottom of the form

DUT is **<Type of DUT>** || P:**<Number of Physical Ports>/<Number of Logical Ports>** S:**<Number of Sweep points>** C:**<Category>** || **<Type of measurement cables>**

6. CONCLUSION

This software could use a lot of work still. Obviously, the algorithms themselves could see some rework. I tried mostly to make it as fast as possible, but the development environment, Agilent Vee, is quite limited for what I wanted to use it for. While it is incredibly useful for doing small functions with the PXI, it's not really made for an user interface. Trying to encapsulate any data with a project this big is a really big problem, and design pattern can hardly be applied if you can't really make objects. If you plan on making a software similar, I would probably use another IDE that can support features like that.

I made this documentation to mainly explain the function and their algorithms. There's almost no documentation in the program itself because the main way of documentation is by blocks. When there's a lot of other block already there, it's hard to follow with the function since you can't really tell by looking which block is going to execute first unless you actually worked a little bit with Agilent Vee.

Another problem with Agilent Vee is the use of the runtime. Running this program outside of the IDE is really hard and has a lot of display bugs. So I recommend using the IDE while trying to run this program.

Finally, while there's a forum about this IDE, information in general is really hard to come by. The user guide in the IDE is probably your best bet. If you can't figure out something, trial and error can mostly get you out of those problems.

7. ANNEXES

7.1. GLOSSARY

Batch File: "A batch file is an unformatted text file that contains one or more commands and has a .bat or .cmd file name extension. When you type the file name at the command prompt, Cmd.exe runs the commands sequentially as they appear in the file."[2]

DUT: Device Under Test.

Firmware: "Permanent software programmed into a read-only memory."[3]

Software: "*Software* refers to computer programs that are designed by a computer programmer or, more likely, a team of computer programmers, to perform a particular function. The software is either embedded in a device, such as a hand-held device or appliance, or installed on a computer."[4]

[2] (Microsoft Corporation)
[3] (Firmware, 2016)

Touchstone File: "A Touchstone file (also known as an S*n*P file after its set of file extensions[3]) is an ASCII text file used for documenting the *n*-port network parameter data and noise data of linear active devices, passive filters, passive devices, or interconnect networks."[5]

VNA: Commonly known as a Vector Network Analyzer. A VNA is an electrical that measures network parameters such as s-parameters. S-parameters are useful for having the magnitude and amplitude of a DUT from a frequency range.

[4] (Smith, 2007)
[5] (Wikipedia, 2015)

7.2. HOW TO FIND YOUR PC'S IP ADDRESS

Press the **windows key + R** and write "**cmd**".

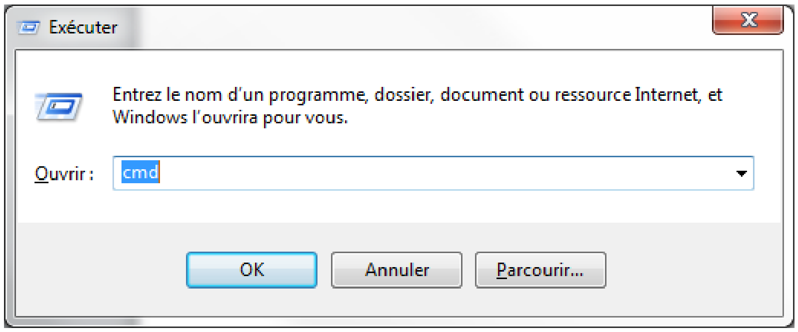


Figure 23: Opening the command prompt

A command prompt should appear. Write ipconfig and you should have info about your Ethernet card.

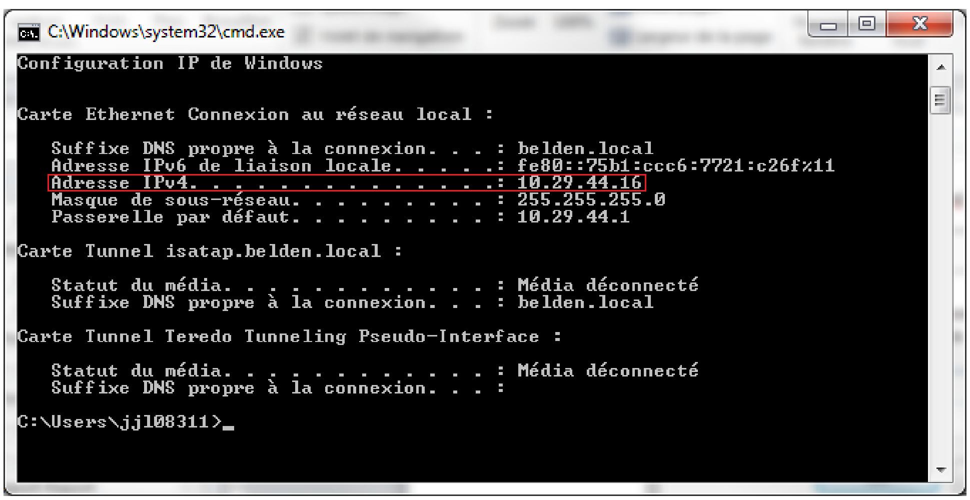


Figure 24: Client's IP Address

7.3. X VS Y PLOT

1. How to give it the data

The graph Plot has two type of data that it can receives; Xdata and Ydata. However, the data is not arranged the same way as the excel library would need it to be. Let's say we want to show 5 points of the linear function "x".

Table 28: function "f(x) = x"

x	y

0	0
1	1
2	2
3	3
4	4

For the excel library, we can only give it an array exactly like shown before, however for the Agilent Vee's graph, the method is different. Using only one trace, the method is simple. You need to have one array for the Y data and one array for the X data, like this:

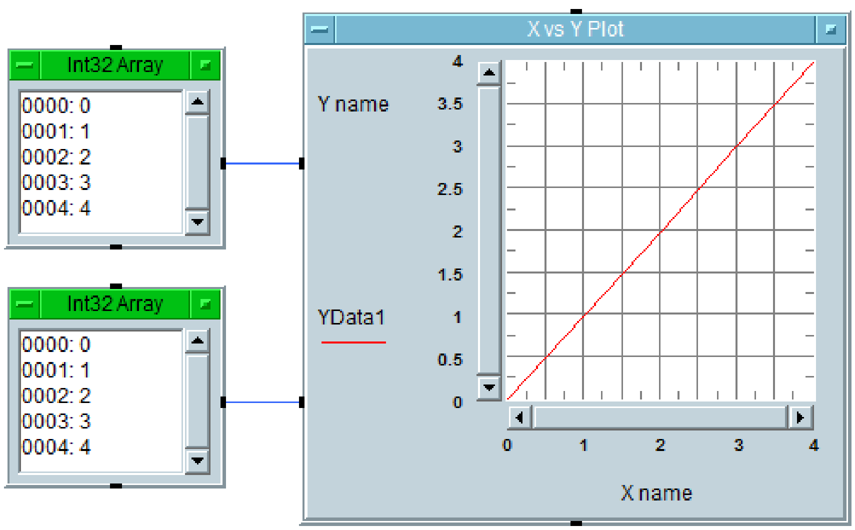


Figure 25: X vs Y Plot using 1D arrays

However, if you want to have more than one trace, things get a little more complicated. First you need to set the property "MatrixInput" to true. Then, you need a 2D array for both the Ydata and Xdata. Every row, in those arrays, represents a trace on the graph and every column represents a point.

Let's say we want to display 10 points of the line "x" and 10 points of the parabola "x^2" in the domain [-5,4], we would have to have these arrays:

Table 29: X Data for 2 traces

-5	-4	-3	-2	-1	0	1	2	3	4

-5	-4	-3	-2	-1	0	1	2	3	4

Table 30: Y data for 2 traces

-5	-4	-3	-2	-1	0	1	2	3	4
25	16	9	4	1	0	1	4	9	16

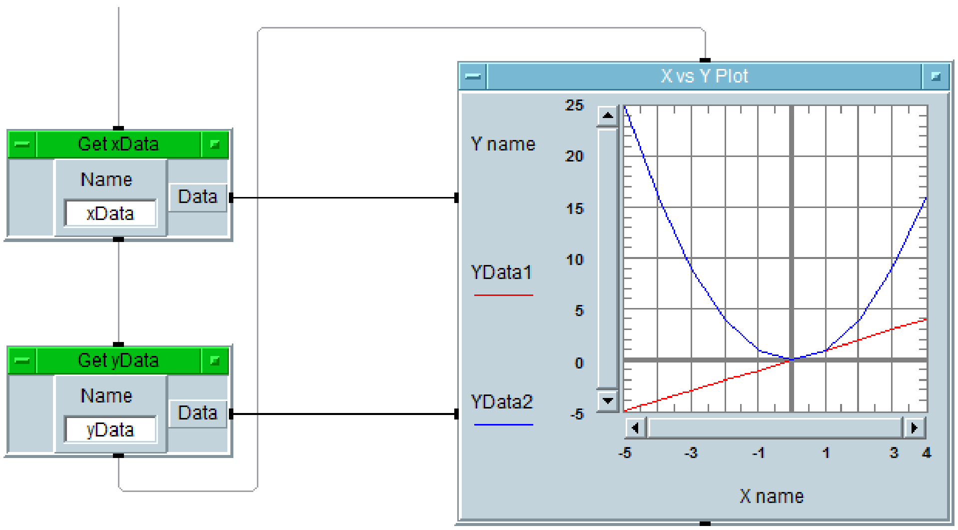


Figure 26: X vs Y Plot using matrixes

7.4. DECIBELS, LINEAR AND COMPLEX NUMBER CONVERSION

Magnitude/Phase to Linear/Phase:

$$r = 10^{dB/20} \tag{85}$$

$$\theta = \theta \tag{86}$$

Linear/Phase to Real/Imaginary:

$$real = r * \cos(\theta) \tag{87}$$

$$imaginary = r * \sin(\theta) \tag{88}$$

Real/Imaginary to Linear/Phase:

$$r = \sqrt{real^2 + imaginary^2} \tag{89}$$

$$\theta = \operatorname{atan}\left(\frac{imaginary}{real}\right) \tag{90}$$

PS: Since atan only calculates the first quadrant, you need to manually fix the phase. Of course, we in the software we use atan2, which automatically does it for you. Be careful if you want to use Excel, since the function inverse the parameters (atan(x/y) is atan2(y,x)).

Linear/Phase to Magnitude/Phase:

$$dB = 20 * log10(r) \tag{91}$$

$$\theta = \theta \tag{92}$$

7.5. BELDEN FUNCTIONS

7.5.1. BuildLimitArray

This function creates an array for the limits depending on the frequency points and the formula in its input.

- Inputs:
  Formula: A 2 dimensional text array with the formula and its domain. Let's take TIA's 568-C.2 Channel return loss limit. The array for the function would need to be:

1 <= f < 10	19
10 <= f < 40	24 – 5*log10(f)
40 <= f < 398.1	32-10*log10(f)
398.1 <= f <= 500	6

  You must follow the standard formula block syntax for it to work.

  FreqArray: A 1 dimensional real64 array with the frequency points in hertz.

- Output:
  LimitArray: A 1 dimensional real64 array with the limits value at the corresponding index of the freqArray input. If there's not limit at a certain point, the value at the index of the limitArray will be zero.

7.5.2. horzcat

This function horizontally concatenates two arrays.

- Inputs:
  a1, a2: 1 or 2 real64 dimensional arrays with the same dimension 1 size.

- Outputs:
  array: 2 dimensional real64 array.

Example:

$$a1 = \{1,2,3,4,5\} \tag{93}$$

$$a2 = \begin{pmatrix} 6 & 11 \\ 7 & 12 \\ 8 & 13 \\ 9 & 14 \\ 10 & 15 \end{pmatrix} \tag{94}$$

$$array = \begin{pmatrix} 1 & 6 & 11 \\ 2 & 7 & 12 \\ 3 & 8 & 13 \\ 4 & 9 & 14 \\ 5 & 10 & 15 \end{pmatrix} \tag{95}$$

7.5.3. vertcat

This function horizontally concatenates two arrays.

- Inputs:
  a1, a2: 1 or 2 real64 dimensional arrays with the same dimension 2 size. (or dimension 1 size for 1D arrays)

- Outputs:
  array: 2 dimensional real64 array.

Example:

$$a1 = \{1,2,3,4,5\} \tag{96}$$

$$a2 = \begin{pmatrix} 6 & 7 & 8 & 9 & 10 \\ 11 & 12 & 13 & 14 & 15 \\ 16 & 17 & 18 & 19 & 20 \end{pmatrix} \tag{97}$$

$$array = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \\ 11 & 12 & 13 & 14 & 15 \\ 16 & 17 & 18 & 19 & 20 \end{pmatrix} \tag{98}$$

7.5.4. Formulas

This functions will give you the domain and formula for a specified system, parameter and category.

- Inputs:

System $$\{Backbone\ Cable, Channel, Connecting\ Hardware, Cord\ Cable, Cords\ and\ Jumpers, Horizontal\ Cable, Permanent\ Link\} \tag{99}$$

Parameter $$\{Return\ Loss, Insertion\ Loss, NEXT\ Loss, FEXT\ Loss, ACR, ACRF, PSNEXT\ Loss, PSFEXT\ Loss, PSANEXT\ Loss, PSAFEXT\ Loss, PSACR, PSACRF, Propagation\ Delay, Coupling\ Attenuation, TCL, ELTCTL\} \tag{100}$$

Category $$\{3, 5e, 6, 6a\} \tag{101}$$

- Outputs:

  An array of the domain and formula used to build the limits array for the function "BuildLimitArray".

7.5.5. As2Dhorzarray

Takes a one dimensional array as an input, and outputs a two dimensional array with the number of column equals to the total size of the input array. Since VEE's graphs either use 1D array or not, it's simpler to only use 2D arrays for everything, because changing the graph's inputs cannot be done programmatically.

$$input\ array = \{1,2,3,4,5\} \tag{102}$$

$$output\ array = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \end{pmatrix} \tag{103}$$

$$output_{size} = (1,5) \tag{104}$$

7.5.6. As2Dvertarray

Takes a one dimensional array as an input, and outputs a two dimensional array with the number of column equals to the total size of the input array. Since VEE's graphs either use 1D array or not, it's simpler to only use 2D arrays for everything, because changing the graph's inputs cannot be done programmatically.

$$input\ array = \{1,2,3,4,5\} \tag{105}$$

$$output\ array = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{pmatrix} \tag{106}$$

$$output_{size} = (5,1) \tag{107}$$

7.6. KNOWN ERRORS

7.6.1. Could not load the following .Net reference

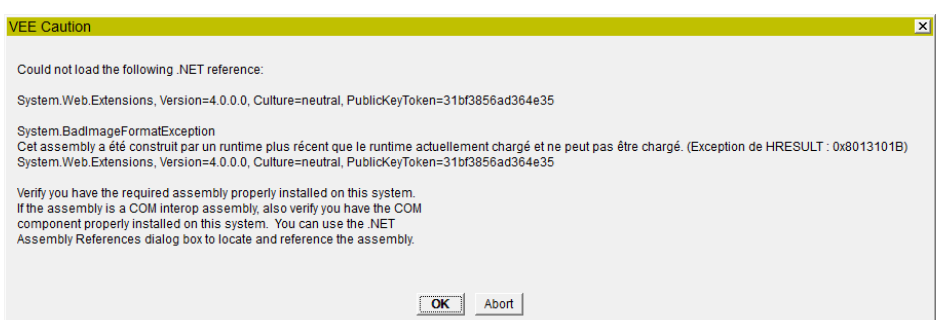


Figure 27: Error happening when .Net path is not set in the config file

Agilent VEE pro originally uses Microsoft .net frameworks v2.0.50727. Therefore, if you want to use a newer framework, like v4.0.3019, you need to change the config file of VEE since there's no option in the IDE. To do so, you need to go VEE's install folder (**C:\Program Files\Agilent\VEE Pro 9.3.**) and find the vee.exe.config file. Save it as .backup and copy this new file in the same folder. You might also want to do the same thing for the run time client, if you play on using it.

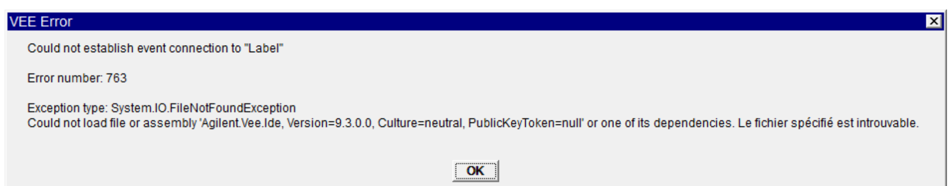
Vee.exe.config

If you open the file using any file editor, you can notice that the only difference is adding:

```
<startup useLegacyV2RuntimeActivationPolicy="true" >
<supportedRuntime version="v4.0.30319"/>
</startup>
```

After the </userSettings> tag.

7.6.2. Could not establish event connect to



This error prevents you from loading the software and can also be very dangerous. The cause of this error is leaving to many panels open when saving the VEE project.

Having to import ".NET" libraries while reloading that many panels makes the project unstable. Normally, reopening it a few times can fix the problem, however the more panels you had open, the higher the chances of it to crash. So if you had too many open, you might not be able to open the project anymore.

So it is highly recommended to always close all the opened panels before saving the project.

8. BIBLIOGRAPHY

Firmware. (2016, January 7). Retrieved January 7, 2016, from The Oxford Pocket Dictionary of Current English: http://www.encyclopedia.com/doc/1O999-firmware.html

Microsoft Corporation. (n.d.). *Windows XP Professional Product Documentation*. Retrieved January 7, 2016, from Microsoft: https://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/batch.mspx?mfr=true

Smith, D. (2007). *Software*. Retrieved January 7, 2016, from Encyclopedia of Business and Finance, 2nd ed..: http://www.encyclopedia.com/doc/1G2-1552100286.html

Wikipedia. (2015, November 16). *Touchstone File*. Retrieved January 2016, 7, from Wikipedia: https://en.wikipedia.org/wiki/Touchstone_file