Hints, Tips and Solutions

Colin Shaw, Applications and Support Engineer

1. Customers are starting to use 64bit machines with large hard drives and Gigs of RAM and tend to forget the limitations imposed by a 32bit operating system. If you have more than 2Gigs of RAM in a system the extra is not used by the 32bit compiled program as it is not possible to address a contigeous memory space bigger than this when you have 1 parity bit and 31 data bits (2^31). If you are however using our 64bit compiled version of *SmartSpice* then you can use as much memory (RAM) as you care to put into the system (2^63 is a huge number compared to the 32 bit case).

2. Some customer have experienced SFLM 6 License problems. The first clue you have a License problem is the "splash screen" of *SmartSpice* comes up and eventually disappers without any other window being generated. In this case *SmartSpice* has not retrieved a valid License. You will notice on the "splash screen" we write some status messages like "Initilizing" etc. which gives some clue to what is happening in the background. The best way to check your License server is to run a browser window on the License machine using the URL *"http://localhost:3162"*. You can use the IP address or the server name in place of the word "localhost". Since SFLM 6 encompasses both a local License and an EECAD pay by use License it is also usefull to run "sflm_access" to check that your primary server (first one on list is the right one). Sometimes customers have a local Licence but are only pointing at the EECAD server on the web causeing the software not to start.

3. A common problem is to have the wrong syntax in a deck. For example Pspice uses the symbol "$" as part of the node name or sometimes part of a variable name. SmartSpice by default uses the "$" as an in line comment character. This means any text after this point is ignored by the "parser" which is interpreting the deck syntax into a set of mathmatical conditions to be solved. *SmartSpice* allows the change of in line comment character by specifying in the deck (.OPTION inlinecc="#") which in this case would set the character used to a "#" and therfore node names are interpreted correctly. The other way is to invoke *SmartSpice* from the command line in the right way eg (SmartSpice - pspice).

Sometimes the syntax is less obvious but attention to detail can save a lot of confusion. Unix for example has different types of the quote mark one is straight (') and another is like a backslash (`) and it matters what type is used and that they are paired up if the parser is to interpret the meaning correctly. (You will have the same issue if you ever write scripts in a C shell for example).

```
e.g. [set i = `expr $1 + 1`] in a control loop
```
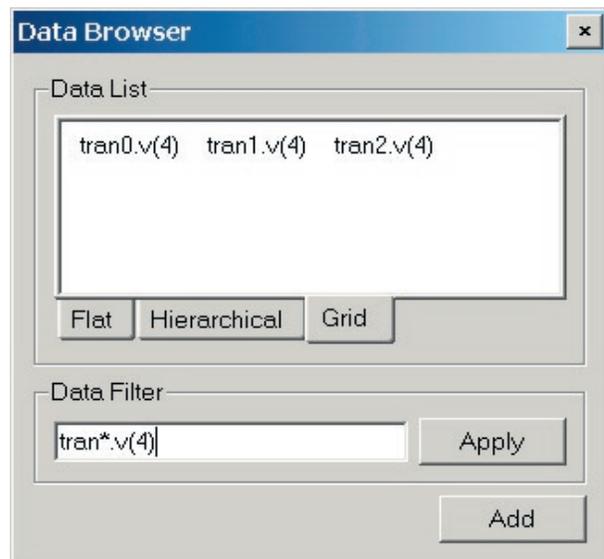

Figure 1. Data Browser window.

4. When you get as far as plotting out a set of vectors in *SmartView* people have often wanted to plot the same named vector from a number of analysis runs. This is best done by opening *SmartView* and from the menu bar go to "File - Open" and select the RAW data file (data generated by a *SmartSpice* simulation run). A "Data Browser" window is opened showing the anlyses runs performed and that can be opened up to reveal the associated individual vectors. A set of associated tabs are along the bottom of this sub-window select the "Grid" tab. You will now see an array of vector names in a winsdow that is sizeable. By use of this window and the filter section below you can display the vectors of interest. Now do an area select and you have the vectors to be plottted by pressing the "Add" button. A simple case is shown in the attached picture but illustrates the power of this feature. Without this you would need to select and open each anlysis and the select the vector and repeat this many times through the contents of the file.