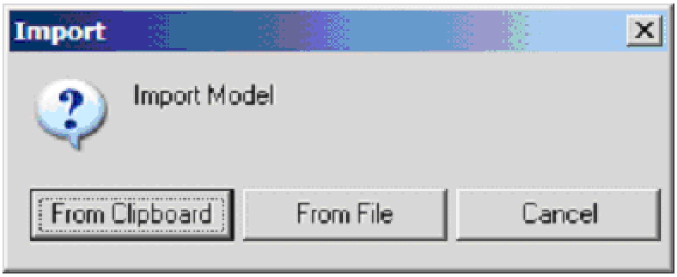
Appendix D: Adding a SPICE Model

Importing SPICE model with Library Manager

Adding a SPICE model to ICAP/4's Part Browser is easy using Library Manager. It automatically adds the appropriate *SRC= and *SYM= lines to the Spice model netlist. Start ICAPS and then choose *Import Spice Model…* under the *File* menu. This brings up a dialog that allows you to import spice model netlist from the clipboard or a text file.



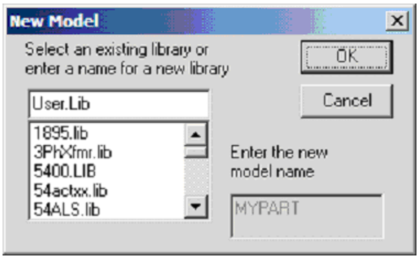
Note: Comment lines start with *. ICAP/4 software uses 5 asterisks as Spice model netlists delimiter.

If the text file contains only one device model, then just import the model "From File." If you have multiple Spice models in the text file, then each model must be imported one at a time. Library Manager will treat any text read-in as a complete model, so only highlight the specific model, copy to clipboard (<Ctrl>+C), and import model "From Clipboard."

- Open up the text file or view webpage with model text.
- Highlight everything from .subckt line to the .ends line of one specific model you want to add, and then copy selected text to the clipboard.
- Press the "From Clipboard" button to import model based only on what is currently stored on the clipboard.

If the model name already exists in your part database, Library Manager will replace your existing model netlist with the text on your clipboard. You will see a text difference between existing and imported spice model netlist. If you don't want the existing model in the part database replaced with your imported model, then exit Library Manager without saving changes, and re-import the model with a modified model name on the .subckt line.

Save New Model



Select an existing library or enter a name for a new library

User.Lib

1895.lib
3PhXfmr.lib
5400.LIB
54actxx.lib
54ALS.lib

Enter the new model name

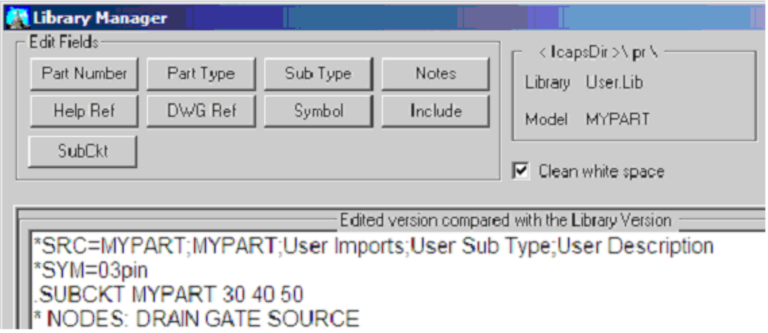
MYPART

The New Model dialog will only come to view if the model name on the .subckt line does NOT already exist in the part database. The unique model name will be shown in the bottom right field under "Enter the new model name."

Make sure that you save the model to a library file that is NOT included with our installation. If you save to an existing library that Intusoft provides, then it will be overwritten when you install an ICAP/4 product update. All current library files in the part database are shown in the list on the left of the New Model dialog. To save to a new library file, type a unique library name, then press the OK button. You can select an existing library file and keep on adding new models to the same library file. Just make sure that this file is not one of Intusoft's provided library files. An example of a safe library file you could save your models to is "User.lib." This file is not included in the installation and is the default file for imported models.

Define Where to Find Model in Part Browser

Notice that *SRC= and *SYM= have been added to the top of your imported model. These lines link to our Part Browser and symbol.



< IcapsDir >\ pr \

Library User.Lib

Model MYPART

☑ Clean white space

```
*SRC=MYPART;MYPART;User Imports;User Sub Type;User Description
*SYM=03pin
.SUBCKT MYPART 30 40 50
* NODES: DRAIN GATE SOURCE
```

The *SRC= line contains the information used by SpiceNet's Part Browser for part selection. It will be given the above default text except for where you see MYPART. This text is based on the model name assigned on the .subckt line. 03pin is based on the number of nodes on the .subckt line.

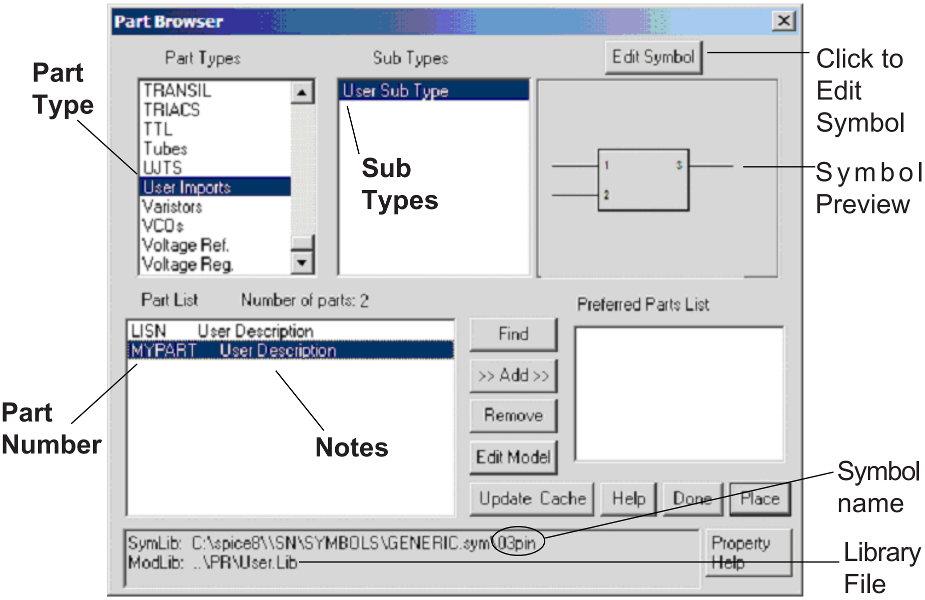
You can modify the text shown in the Part Browser by using the Edit Fields buttons. Each Edit Field is separated by a semicolon. There MUST be four semicolons on the *SRC= line.

*SRC=Part Number;SubCkt;Part Type;Sub Type;Notes

Note: The length of the Part Number and Notes field combined should be no longer than 30 characters. The Part Type and Sub Type each should be no longer than 13 characters.The SubCkt field MUST be unique and match the model name on the .subckt line below. Spaces or !@#$%^&* characters are NOT allowed for the SubCkt field.

After you are done modifying the source line *SRC= and symbol line *SYM=, you MUST save your changes and update the part database.

Validating That Your Part Was Added



SymLib: C:\spice8\\SN\SYMBOLS\GENERIC.sym\03pin
ModLib: ..\PR\User.Lib

Note: SymLib and ModLib at the bottom of the Part Browser dialog reveals the location of the part symbol and netlist.

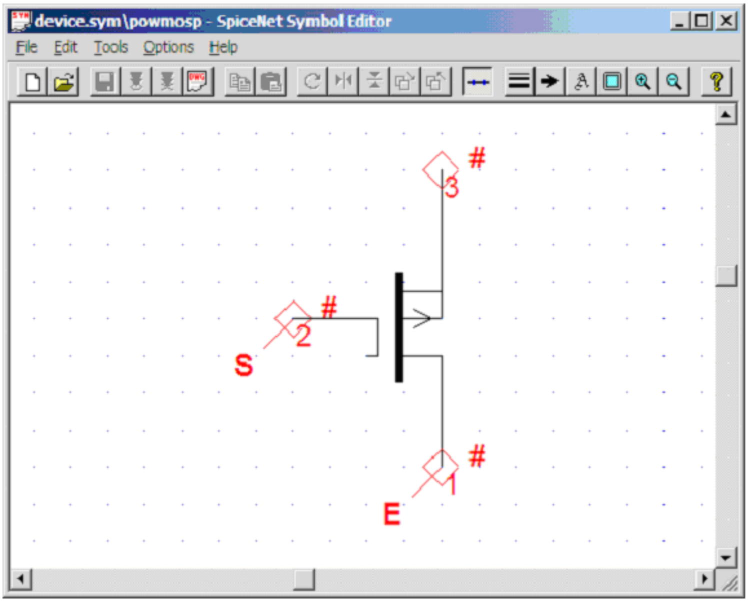
Bring to view your imported model in the part browser to make sure everything is working properly. This step validates that your *SRC= and *SYM= lines were correct.

- Start ICAPS.
- Select *Part Browser* from the *Part* menu or type "x" on the keyboard.
- Click "Find" button and search for your part.

Use Existing Symbol

The symbol name is shown on the SymLib line after the compound symbol file, which ends with .sym extension. If you want to use an existing symbol then select a similar part with the correct pins. Click the "Edit Symbol" button on the Part Browser to open the Symbol Editor and make sure the pin order matches the order of pins on the .subckt line of your added model. If it does, then click “Edit Model” button in the Part Browser to bring up Library Manager. Then, click on the “Symbol” button and type the symbol name.

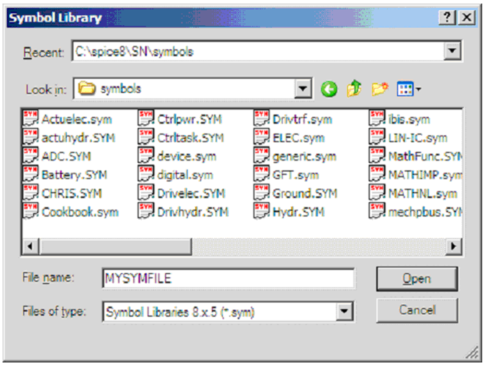
Example: We imported a PNP Power Mosfet model. Notice that in the Parts Browser for existing PNP Power Mosfets, they use a powmosp symbol in the device.sym compound symbol file. If the edit symbol button is clicked, pin1 is on drain, pin2 is on the gate, and pin3 is on the source. This is the desired symbol, so the symbol name powmosp can be used for the *SYM= line. If the pins didn’t match, one can either re-order the nodes on the subckt line or modify the pins on the symbol.



PIN ORDER #1 #2 #3
.SUBCKT MYPART 30 40 50
* NODES: DRAIN GATE SOURCE

Create New Symbol or Modify Existing Symbol

You can create a new symbol or modify an existing symbol. Just make sure that it is NOT saved in an Intusoft provided compound symbol file. If you do, the file will be overwritten when you update or re-install the software. Many models link to the same symbol. Remember to check the pin order and number of pins.

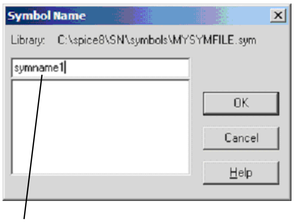


Recent: C:\spice8\SN\symbols

Look in: symbols

File name: MYSYMFILE

Files of type: Symbol Libraries 8.x.5 (*.sym)



Library: C:\spice8\SN\symbols\MYSYMFILE.sym

symname1

This is the symbol name you specify on the *SYM= line. Spaces or any of the characters !@#$%^&* in the name are NOT allowed.

- Choose *Save copy as…* from the *File* menu in Symbol Editor.
- Enter the compound symbol file name you want to save your new symbol to, and press the Open button.
- Now type your symbol name and press OK.

If you saved this symbol to a new compound symbol file, then you need to edit the Sym.@@@ file located in the Spice8\SN directory.

For example, you would add the line **..\sn\symbols\MYSYMFILE.SYM** to the SYM.@@@ file as shown below if your compound symbol file name was MYSYMFILE.SYM. Note: The existing paths are relative but you can specify an absolute path if you want.

```
..\sn
..\sn\symbols
..\sn\symbols\device.SYM
..\sn\symbols\digital.SYM
..\sn\symbols\lin-ic.SYM
..\sn\symbols\special.SYM
..\sn\symbols\system.SYM
..\sn\symbols\ttl74xx.SYM

..\sn\symbols\MYSYMFILE.SYM
```

You can save all your new symbols in one compound symbol file so you don't have to repeat the above step each time.

Create a Folder to Contain Your Own Models

If you want to separate your libraries from the libraries provided by Intusoft, you must modify the LIB.@@@ file located in the Spice8\SN directory.

For example, you would add the line C:\MYLIB to the LIB.@@@ file as shown below, if your model library file was placed in the MYLIB directory. Only .LIB files in the specified directories will be added to the part database.

```
..\pr
C:\MYLIB
```

Eliminating Duplicate Parts Errors

To compile the new part database, Start ICAPS, select the *Update Part Database…* from SpiceNet *File* menu, or by selecting MakeDB in ICAP_4 program group. You must resolve all duplicate part errors in the libraries after MakeDB is done compiling.

```
Compiling User.LIB
-- Warning: Duplicate .MODEL or .SUBCKT name MYPART in POWMOS.LIB
```

In order to correct this use IsEd to find and modify one of the duplicate subckt part names. If you find a large number of duplicate parts within a single library, you can change the .LIB extension to .LBK so MakeDB will ignore it.

What MakeDB does in more detail

To access parts from the Part Browser dialog, SpiceNet must be able to access two database files, dbase.@@@ and index.@@@. The source files from which these two files are created, and the utility program (MakeDB.exe) used to update the database files, are located in the C:\spice8\sn directory.
Editing of the source files and recompilation of the database files is necessary if you want to:

- Add your own IsSpice4 models or subcircuits to SpiceNet to be able to place them with the Part Browser.
- Add a new symbol to represent a new or existing subcircuit.

The files that tell MakeDB.exe which libraries and symbols to compile into the part database are:

- Lib.@@@ This file contains the path(s) to the library directories to be included in the index.@@@ file. All files with a .LIB extension in the directories listed (NOT subdirectories) will be compiled. By default this contains only ..\PR. Relative path is taken to be relative to MakeDB.exe. Explicit path is used as is.
- Sym.@@@ This file contains a list of all of the compound symbol files to be used when compiling the SpiceNet database. The first entry must be "..\SN." Each remaining entry is a compound symbol file. The compound symbol file must have a .SYM extension. Any entry using an implicit path is considered relative to MakeDB.exe. Any entry using an explicit path is used as is.

The following entities are created after compiling the symbol/ library database:

- dbase.@@@ This is a compiled database file made from *SRC lines in the library files. The information in this file is used by SpiceNet's Part Browser dialog and component placement by part number.
- index.@@@ This is a compiled index file containing the name of every model and subcircuit, the library file containing the model/subcircuit, and its corresponding SpiceNet symbol (*SYM).

Network Version Notes

Environment variable IS@@@ is used to specify the directory location of the *.@@@ files. SpiceNet uses this environment variable to locate DBASE.@@@ and INDEX.@@@. MakeDB uses this environment variable to locate SYM.@@@ and LIB.@@@, and also a directory location to write DBASE.@@@ and INDEX.@@@.

Each client has the option of using libraries and symbols on the server, local client machine, or both. In addition to the libraries stored on the server in the PR directory, other library locations can be specified. The contents of the LIB.@@@ and SYM.@@@ files determine which libraries and symbols, respectively, will be compiled into the SpiceNet part database.

The compiled DBASE.@@@ and INDEX.@@@ files will determine what information is accessible to the client.

To add new symbols or libraries for all ICAP/4 Client users, add the .LIB file(s) to the PR directory on the server. Add the required compound symbol files to the SN\Symbols directory on the server. Recompile the SpiceNet database by running the MakeDB program in the SN directory. Click the "Compile" button. Copy the LIB.@@@, SYM.@@@, INDEX.@@@ and the DBASE.@@@ files to each client's IS@@@ directory (C:\spice8client by default).

IMPORTANT NOTE: If the client has created their own DBASE.@@@ and INDEX.@@@ files in an effort to have their own local (private) libraries, then you will have to update the database from the Client side. To add libraries and/or symbols to a local client only, you must specify the path to the local .LIB file(s) in the LIB.@@@ file.

For example,

```
  ..\PR
  C:\MYLIBS     <-- example of a new library path entry
```

Specify the path to the local compound symbol files (*.SYM). SYM.@@@ is located in the directory pointed to by the IS@@@ environment variable.

For example,

```
  ..\sn
  ..\sn\symbols
  ..\sn\symbols\device

  C:\spice8client\MYSYM.SYM   <-- example of a new
                                  symbol file entry
```

In these @@@ files, relative paths, e.g., ..\PR, are relative to the server. Explicit paths, e.g., the symbol file entry C:\spice8client\MYSYM.SYM, are always relative to the client. Recompile the SpiceNet database by selecting *Update Part Database...* from the *File* menu. This operation can take a long time depending on network access and performance.

Duplicate library names or library part names should be avoided when possible. Duplicates are treated on a "first-come, first-serve" basis, whereby the search order is defined by the entries in the LIB.@@@ and SYM.@@@ files. If you want your client libraries to take precedence over those provided on the server, then list the client entries before the server entries. If you remove the server library path entries, then only the client libraries will be compiled. The same is true for symbols. The MakeDB program creates the DBASE.@@@ and INDEX.@@@ files in the client's IS@@@ directory. SpiceNet will use these database files rather than those on the server.

Appendix E: Export Schematic of Model as Subckt

Make Configuration For Export

Prepare a circuit configuration with only the circuitry that you want to include in the subcircuit. This will most likely require that you remove any test circuitry or stimulus sources. Along this line, be sure to attach wires, test points or continuation symbols on any nodes that are unconnected. In this way, unconnected parts will have their pins resolved with node numbers or names.

Let’s cover the basic steps to accomplish this task. We will start with a set of circuitry that is on one layer in one configuration in SpiceNet. We'll then split the circuitry into two parts: one part, which is not needed for the subckt export, will be on a layer called "Test Circuitry." The second, which contains the circuitry destined for subckt export, will be on a second layer called "Circuit Under Test." Then we'll create two configurations called, "For Simulation" and "For Export." The "For Simulation" configuration will contain both layers. The "For Export" configuration will only contain the "Circuit Under Test" layer. Note that items on a layer that is unique to a configuration, will not affect other configurations.

To create a new circuit configuration that uses a portion of your existing circuitry

- Select Options > Layers...
- Click the Rename… button and type "Circuit Under Test.” Select OK.
- Click the New… button to create a new layer. Name the new layer "Test Circuitry.” Select OK.
- Select OK to close the Layers dialog. All of the circuitry is now on the “Circuit Under Test” layer. We can see this by pressing the eye icon at the bottom left of the schematic windows next to the layers drop-down list. Only parts on the selected layer should be highlighted.
- Select ONLY the circuitry that you do NOT want represented in your exported subckt. Remove all test circuitry and stimulus sources by holding down the <Shift> key, and select each component that you want to exclude.

- Right mouse click and choose Move Item to Layer > “Test Circuitry” to transfer all selected components to the “Test Circuitry” layer.
- Config 1 now has the circuitry split on two layers. We can confirm this by selecting the “Test Circuitry” layer in the drop-down list at the lower left corner of the schematic, and then pressing the eye icon next to the list.
- Select Options > Configuration > Edit…
- Click the Edit… button and rename "Config 1" to "For Simulation.” Select OK.
- Click the New… button and create a new configuration. Call it "For Export.”
- Remove (deselect) the newly created layer ("Test Circuitry") from the configuration. Select OK.
- Select OK again to close the dialog.
- Change between the two configurations by using the toolbar configuration dropdown (left). Notice the differences. The layers available in the dropdown list at the lower left corner of the schematic are based on which layers are in the selected configuration.

Note: In order to account for unconnected parts, whose connections are via parts on layers that are not active in this configuration, you MUST attach a wire, test point or continuation symbol(s) to the unconnected pins. Every pin on every part must have a node number or name. Pins on the ends of dangling wires are OK.

Define Subckt Parameters

If you use parameters text block to define global parameters, then you need to manually copy and paste your parameters to the “For Export” configuration. Global parameters will become your exported subckt parameters. The Parameters text block takes all parameters listed in a schematic text block if it starts with the keyword “parameters,” and moves them to the selected configuration global parameters list when you simulate.

- Select Actions > Simulation Setup > Parameters Setup…
- Copy and paste parameters from “For Simulation” to “For Export” configuration
- Define default parameters or use three question marks (???) if you want user to define passed parameters

Before exporting this configuration, you need to make a subdrawing to define the subckt nodes.

- Select Subdrawings > Make Subdrawing…
- The Subdrawing name is the same name as the current configuration. You may change it in the Subdrawing Name: field.
- Click on the node from the list of nodes on the left, that you want to expose from inside the subcircuit (external connection, and that will be on the .SUBCKT line)
- Click the Add button to add the selected node to the list of subcircuit pins. Repeat this operation for each node you want to expose.
- You may arrange the node order using the Move Up/Move Down buttons. You can also assign the pins to be hidden. You will then be able to make connections to the hidden pins by using a continuation symbol with the same name as the hidden pin. The hidden pin name is assigned inside of the part’s Properties dialog.
- Select your symbol. “Use Wizard to Create Symbol…” enables you to specify pin arrangement and names. "Use Default Symbol" is just a rectangle with the number of pins you expose. If you choose “Get Symbol From Library,” you need to select an existing symbol.

Wizard Note: If you use the Wizard option you MUST save the symbol you make in a compound symbol library file before selecting the Finish Subdrawing button at the bottom of the Symbol Editor screen. This requires you to enter a symbol file name, plus a name for the individual symbol. If the compound symbol file is new, then you must specify the path and symbol file name in SYM.@@@ before you run MakeDB. Remember the symbol name, and NOT the compound symbol file name. You will need to add this symbol name to the *SYM= line in the subcircuit netlist you generate.

Exporting the Subcircuit Netlist

- Select File > Export…
- Select the SubCkt option from the drop-down list. Select OK.

The subckt part will be added to SPICE8\PR\USER.LIB. Make sure the appended model is not a duplicate model. Modify the *SRC= and *SYM= line so the part is placed in the Part Browser where you desire, and uses the symbol you want. If you have Library Manager, you will be able to specify any library file you desire, and your modifications to the *SRC= and *SYM= line will NOT be overwritten every time you re-export.

Make Your Exported Subckt Model Easier to Use

After you export your subckt model, you will want to add a few modifications to make it easier for other people to use.

- Modify the *SRC= line so you can easily find it in the Part Browser. See Appendix D on how to modify this line.
- Modify the *SYM= line so you are using an easily recognizable symbol. See Appendix D on how to modify this line.
- Add comment lines explaining passed parameters and their units. If a line starts with one asterisk it is considered a comment line. Be careful. Models are separated by a row of at least five asterisks, "*****". If you add rows of asterisks to separate sections of your model, or create a box out of asterisks, then you will be inadvertently cutting your model into pieces.
- Link your subckt model to the schematic you exported it from.. In Library Manager you can press the "DWG Ref" button to browse for your schematic file and add the proper *DWG line. At any time you can bring this up in Library Manager and press the "Test" button to launch your reference schematic. Note: Schematics can't be launched in SpiceNet if the part browser is open.

  The format is: *DWG=Path\Filename

- Create .RTF (Rich Text Format) Help on how to use your subckt model. The Property Help button in the X… Parts Browser dialog, and in each Part Properties dialog, brings up the referenced help. Click on the “Help Ref” button to safely create your own RTF help for your added model.

  The format is: *HELP=pr\<library>\<model>.RTF

- Add *FAMILY line to enable auto-bridging for non-analog pins, or to make hidden pins available. If a subcircuit has no *FAMILY line, SpiceNet will assume that all pins are analog and no bridging will occur. The names used for this line are symbolic names that represent bridging components that will be used when the part is connected in SpiceNet. If a subcircuit that uses a combination of analog and digital connections is constructed, the *FAMILY line is used to make sure that each connection point is properly translated.

  Note: The *FAMILY line is located inside the subcircuit definition (after the .subckt line and before the row of asterisks marking the end of the subcircuit). This is a requirement of the *FAMILY line.

  The following *FAMILY terms are pre-defined. You can control these levels by selecting Options > Mixed Signal Properties… in SpiceNet.

  "Dout" - generic bridge using TTL levels
  "Din" - generic bridge using TTL levels
  "TTLout" - TTL bridge using TTL levels
  "TTLin" - TTL bridge using TTL levels
  "ECLout" - ECL bridge using ECL levels
  "ECLin" - ECL bridge using ECL levels
  "Rout" - Real bridge using R2A bridge
  "Rin" - Real bridge using A2R bridge

  Hidden Pins Using *FAMILY

  The *FAMILY line can also be used to expose a node inside the subcircuit. This is useful for probing and examining voltage values inside the subcircuit from the top level displayed in SpiceNet. To expose a node nested in a subcircuit, add an additional family name at the end of the list preceded by a pound symbol (#).

For example:
.SUBCKT LM714 1 2 3 4 5 10
**FAMILY ANALOG ANALOG ANALOG ANALOG ANALOG #ANALOG*

This example shows the five subcircuit connections. The last #ANALOG will cause a sixth editable node to be exposed inside SpiceNet from the properties dialog of the LM741. In the part's properties dialog you will be able to assign a node name to the exposed node, and thus connect it to any other place in the circuit that has the same node name.

- Add *PINOUT Line for PCB export.
  The format for a *PINOUT line in a library file is:

  **PINOUT package_name pin_number pin_number ; pin_number pin_number: uncommitted pins*

  The package_name is separated by spaces from the *PINOUT and the pin list. Each pin_number in a pin list is separated by a space. If there is more than one component in the package, the second sequence of numbers follows the first, separated by a semicolon. The pin_numbers in a pin list represent the actual pin numbers used by the manufacturer. The order that the pin numbers appear must match the order of the connection on the .subckt line or, in the case of primitive parts, the order that IsSpice4 expects.

  An Example for a National Instruments AD822:
  **pinout SOIC 3 2 8 4 1;5 6 8 4 7*
  **pinout Cerdip 3 2 8 4 1;5 6 8 4 7*

  If the package has pins that are not represented on the subcircuit, any symbolic name can be given on the *Pinout line, along with the pin number. Place all such pin descriptions at the end of the line following a colon. The name will appear in the Footprint Pins Assignment dialog. These pins can be renamed in the Port column. However, hidden pins can't be added in within this dialog. Users first have to modify the *pinout line in .lib files to get the hidden pins shown for renaming. The following is an example of a LS04 digital inverter that has uncommitted pins, no VCC, and GND connections for simulation:

  .SUBCKT LS04 1 2
  **pinout W 1 2;3 4;5 6;9 8;11 10;13 12:VCC=14 GND=7*