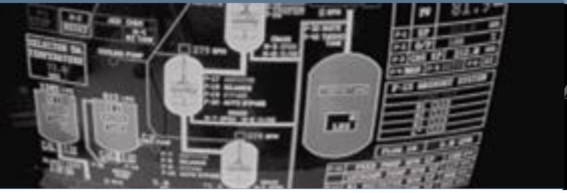
Creating a Schematic Symbol



Creating Symbols – Notice to Windows OS users

- For “*Win OS with User Account Control*” users (ex. Vista, Win7 and Win8), either turn-off UAC or start LTspice using “Run as Administrator” even if you are logged as an administrator. If not done:
  - Manual modifications to the elements within the /LTspice/sym/ folder will not be saved. That includes modifications to newly created symbols, automatically generated or not.
  - Prevent the automatic creation of .subckt symbols

Automatic Symbol Creation

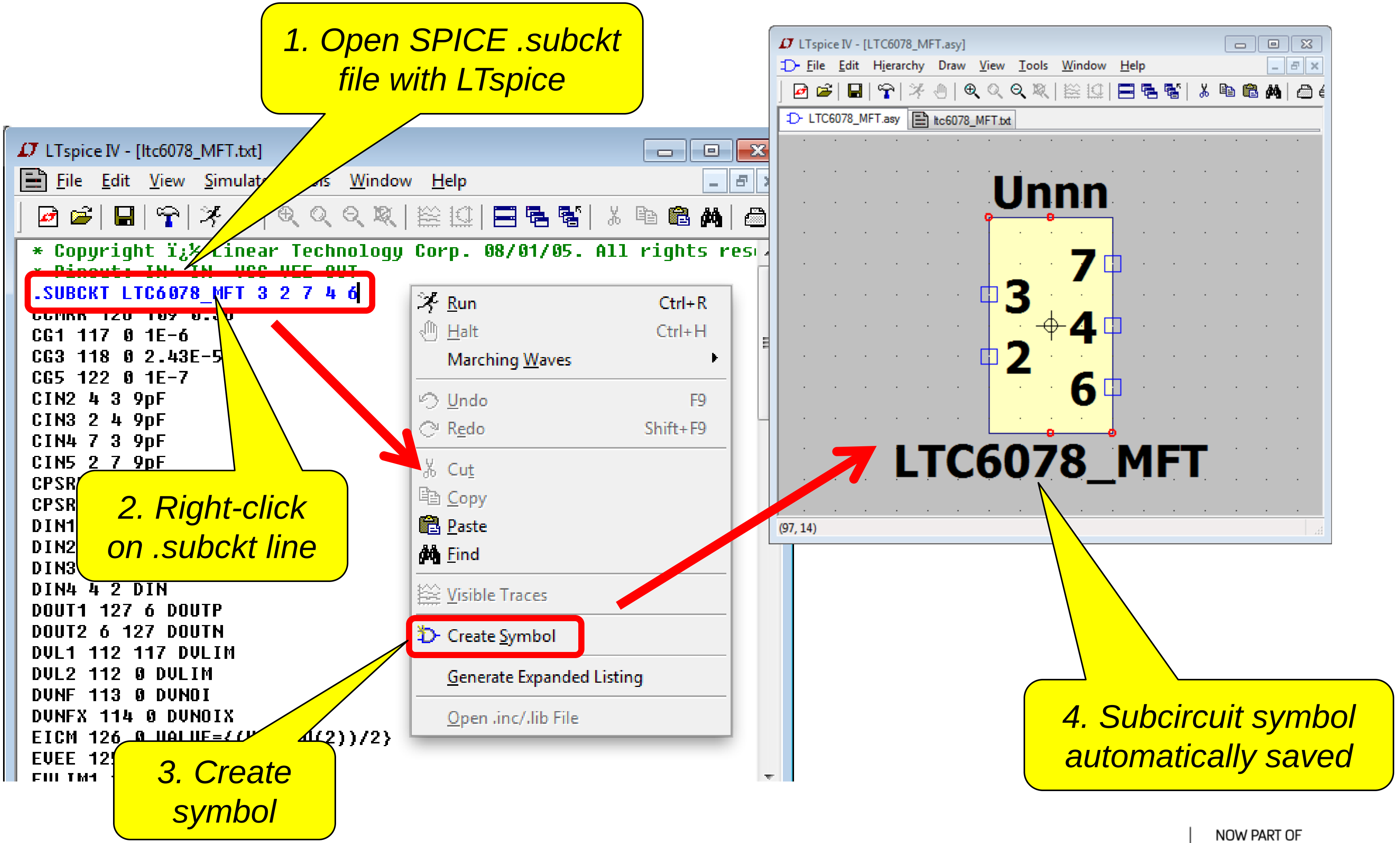
A symbol can be automatically generated in two situations.

- Subcircuit models: When editing an ASCII netlist that contains subcircuit definitions, you place the cursor on the line containing the name of the subcircuit, right click, and execute context menu item "Create Symbol."
- Hierarchical schematic: When editing a schematic, you can execute menu item Hierarchy➔Open this Sheet's Symbol. When no symbol is found, LTspice will ask if you would like one automatically generated.

Important: The symbols are automatically saved – no need to manually save. They are saved in folder:

- **The "working" folder for hierarchical symbols**
- **\LTspice\Lib\Sym\AutoGenerated for subcircuit symbols**

Automatic Symbol Creation - Subcircuits



1. Open SPICE .subckt file with LTspice

2. Right-click on .subckt line

3. Create symbol

4. Subcircuit symbol automatically saved

```
* Copyright ï¿½ Linear Technology Corp. 08/01/05. All rights rese
* Pinout: IN+ IN- VCC VEE OUT
.SUBCKT LTC6078_MFT 3 2 7 4 6
CCMRR 120 109 0.5
CG1 117 0 1E-6
CG3 118 0 2.43E-5
CG5 122 0 1E-7
CIN2 4 3 9pF
CIN3 2 4 9pF
CIN4 7 3 9pF
CIN5 2 7 9pF
CPSR
CPSR
DIN1
DIN2
DIN3
DIN4 4 2 DIN
DOUT1 127 6 DOUTP
DOUT2 6 127 DOUTN
DVL1 112 117 DVLIM
DVL2 112 0 DVLIM
DVNF 113 0 DVNOI
DVNFX 114 0 DVNOIX
EICM 126 0 VALUE={(V      V(2))/2}
EVEE 12
EVLIM1
```

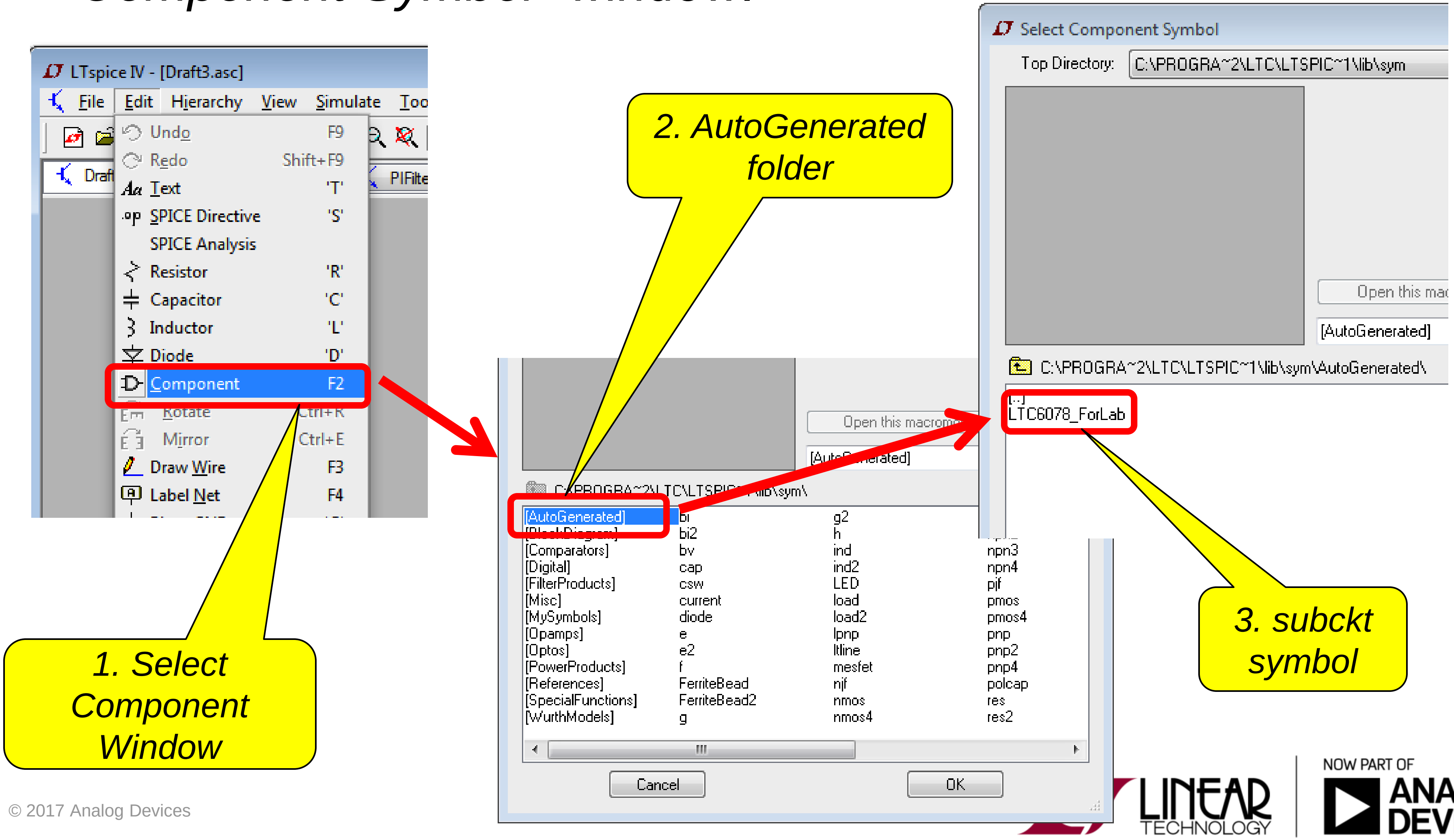
Automatic Symbol Creation - Subcircuits

- Without modifying the symbol newly created, the automatic symbol creation process requires that the spice model file remain in the same directory as when it was use to create the symbol.
  - The symbol points to the absolute path for the model file. As an example:

    “C:\LTspiceLab\CreatingSchematicSymbols\LTC6078ModelForLab.txt”
  - Symbol editing is discussed in details later on in this presentation (ex. changing the path).

Automatic Symbol Creation - Subcircuits

- The symbol can be placed on a schematic via the “*Select Component Symbol*” window.



1. *Select Component Window*

2. *AutoGenerated folder*

3. *subckt symbol*

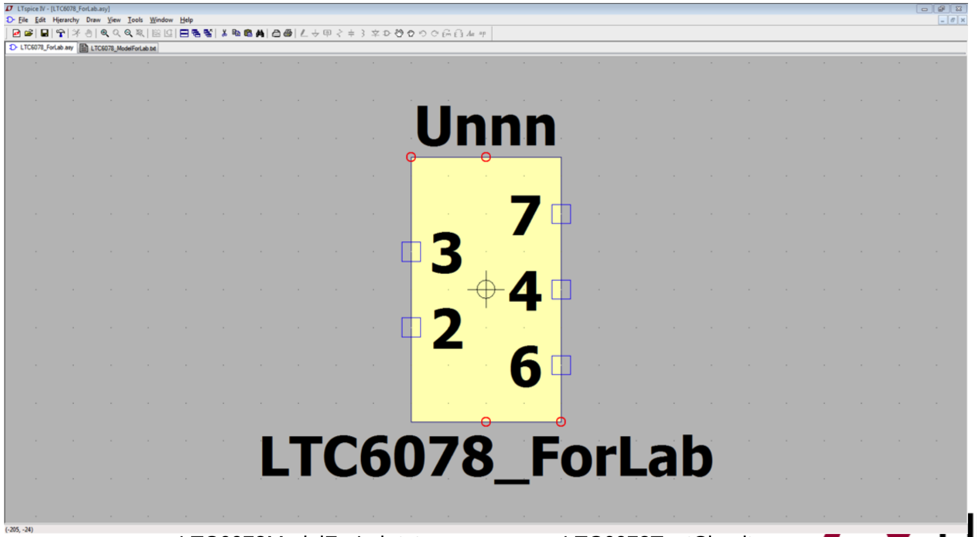
Top Directory: C:\PROGRA~2\LTC\LTSPIC~1\lib\sym

C:\PROGRA~2\LTC\LTSPIC~1\lib\sym\AutoGenerated\

Automatic Symbol Creation - Subcircuits

Hands-on Exercise:

- Create the symbol for the “LTC6078ForLab” model using the automatic symbol creation feature.
- The model file, “LTC6078ModelForLab.txt”, is located in <folder link>
- Once generated, use the “LTC6078TestCircuit.asc” circuit to verify the newly created model.

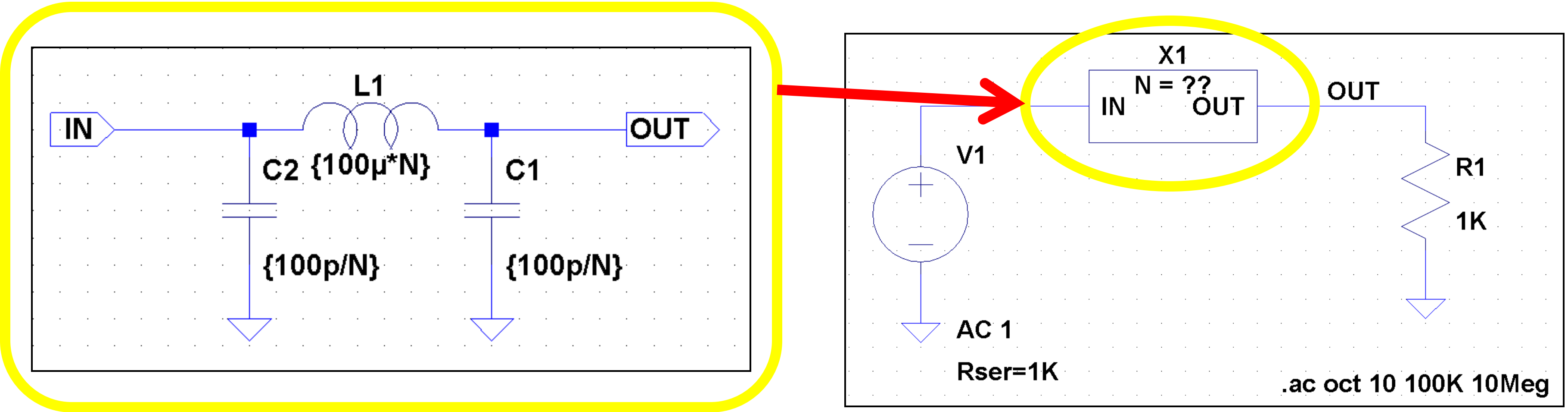


LTC6078ModelForLab.txt

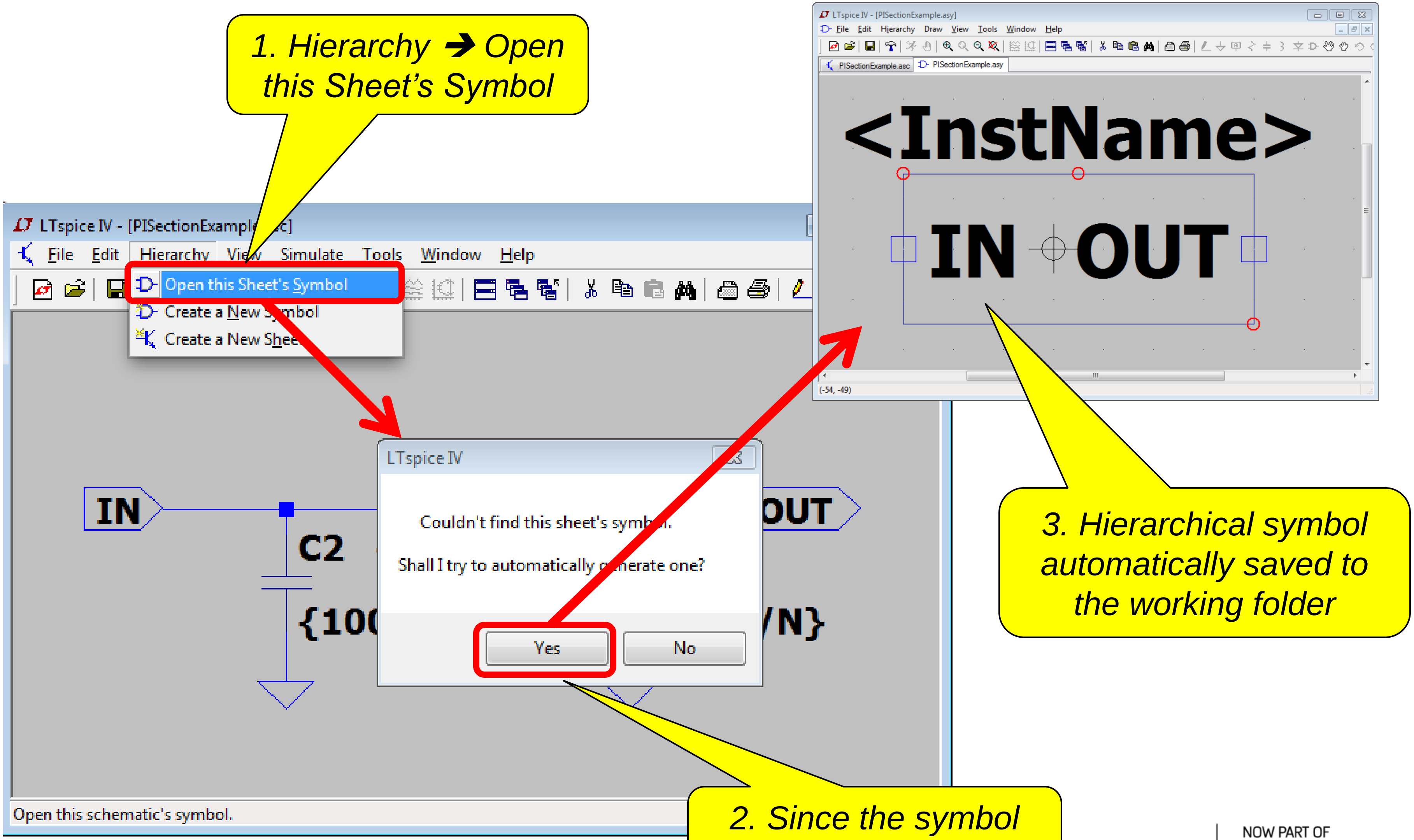
LTC6078TestCircuit.asc

Hierarchical Schematics

- Hierarchical schematic drafting has powerful advantages:
  - Much larger circuits can be drafted than can fit onto a one sheet schematic while retaining the clarity of the smaller schematics
  - Repeated circuitry can be easily handled in an abstract manner, i.e. “black boxes” with full functionality
  - Can be re-used across several schematics



Automatic Symbol Creation - Hierarchical



1. Hierarchy → Open this Sheet's Symbol

2. Since the symbol doesn't exist…

3. Hierarchical symbol automatically saved to the working folder

LTspice IV

Couldn't find this sheet's symbol.

Shall I try to automatically generate one?

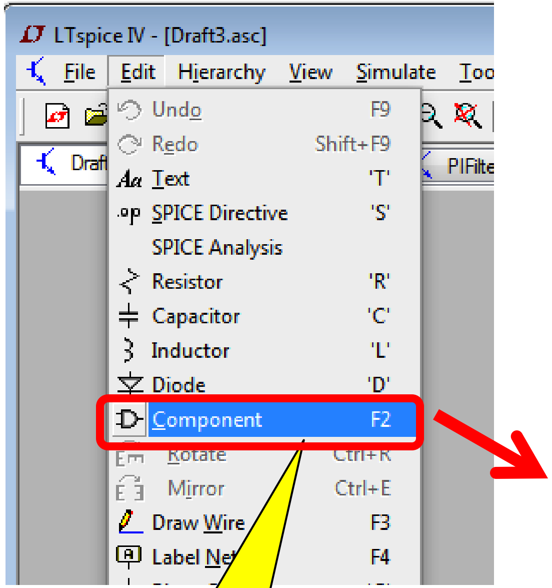
Yes No

Open this schematic's symbol.

LTspice IV - [PISectionExample.asc]
File Edit Hierarchy View Simulate Tools Window Help
Open this Sheet's Symbol
Create a New Symbol
Create a New Sheet
IN
OUT
C2
{100
/N}
LTspice IV - [PISectionExample.asy]
File Edit Hierarchy Draw View Tools Window Help
PISectionExample.asc
PISectionExample.asy
<InstName>
IN
OUT
(-54, -49)

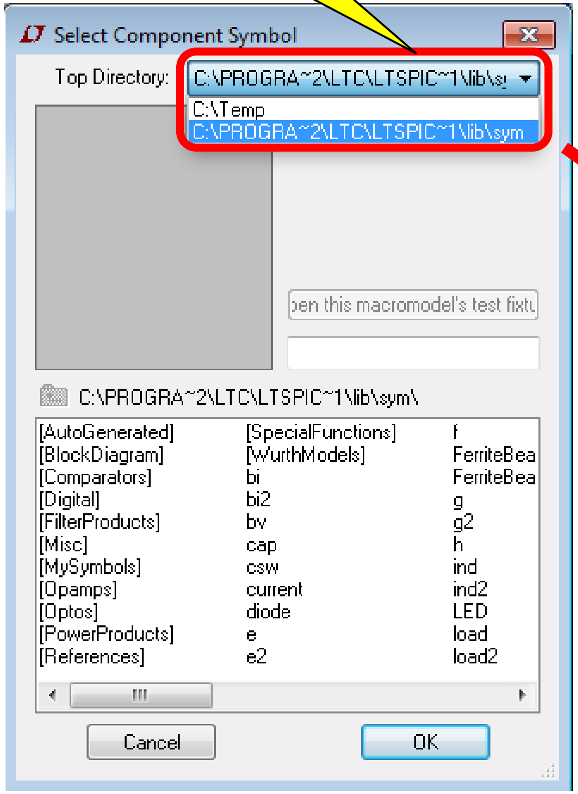
Automatic Symbol Creation – Hierarchical symbols

- To place a hierarchical block on a schematic:



1. Edit ➔ Component

2. Change Top Directory

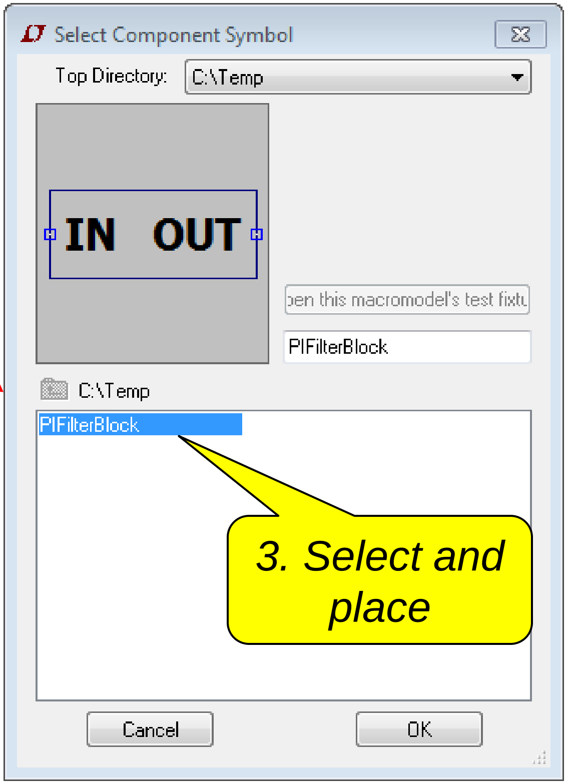


Top Directory: C:\PROGRA~2\LTC\LTSPIC~1\lib\s

C:\Temp
C:\PROGRA~2\LTC\LTSPIC~1\lib\sym

C:\PROGRA~2\LTC\LTSPIC~1\lib\sym\

[AutoGenerated]
[BlockDiagram]
[Comparators]
[Digital]
[FilterProducts]
[Misc]
[MySymbols]
[Opamps]
[Optos]
[PowerProducts]
[References]
[SpecialFunctions]
[WurthModels]
bi
bi2
bv
cap
csw
current
diode
e
e2
f
FerriteBea
FerriteBea
g
g2
h
ind
ind2
LED
load
load2



Top Directory: C:\Temp

PIFilterBlock

C:\Temp

PIFilterBlock

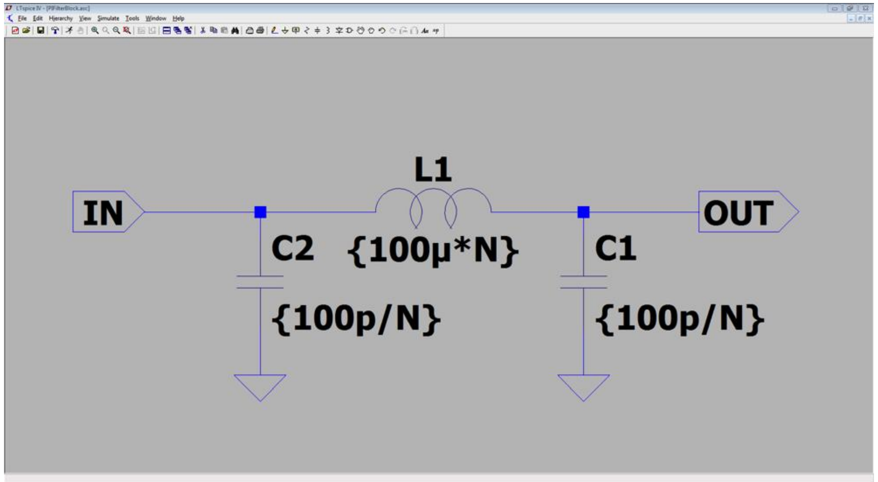
3. Select and place

© 2017 Analog Devices

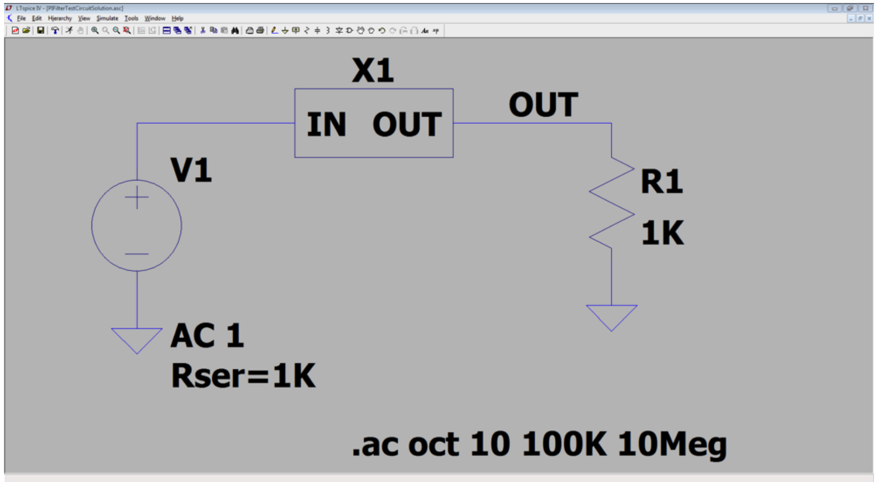
Automatic Symbol Creation - Hierarchical

Hands-on Exercise:

- Using the automatic symbol creation feature, create a hierarchical symbol for the circuit “PIFilterBlock.asc”.
- Once the symbol is created, verify its functionality by using the test jig “PIFilterTestCircuit.asc”.



PIFilterBlock.asc



PIFilterTestCircuit.asc

Creating/Editing schematic symbol

- To launch the symbol editor:
  - *Ctrl+RightClick* on the symbol then “Open Symbol”
  - *File* pull-down menu ➔ *New symbol*,
  - Open up a known symbol (.asy). *File* menu ➔ *Open*
- A few reasons to use the symbol editor:
  - When it comes to schematic entry, the automatically generated symbols, especially the .subckt, might not be effective and easy to wire-up. Editing the symbols can significantly improve its ease-of-use;
  - Not all symbols can be automatically generated (ex. .model);
  - ….

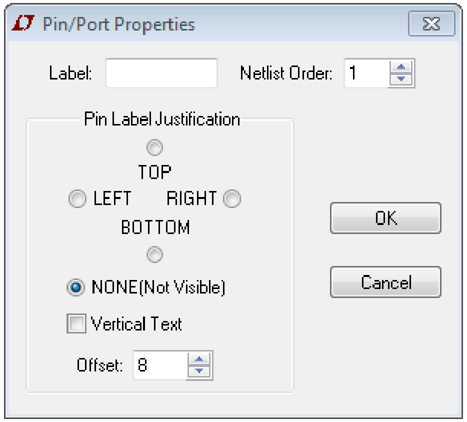
Symbol Editing – Drafting and Adding Pins

Drafting [*Draw* menu]:

- Use Draw to sketch the symbol

Add Pin/Port [*Edit* menu]:

- The "Pin Label Position" determines how the pin label is presented (text justification).
- Label: If the symbol represents …
  - a SPICE primitive element or a subcircuit from a library, then the pin label has no direct electrical impact on the circuit.
  - the lower-level schematic of a hierarchical schematic, then the pin name is significant as the name of a net in the lower level schematic.
- The "Netlist Order" determines the order this pin is netlisted for SPICE



Pin/Port Properties

Label:

Netlist Order: 1

Pin Label Justification

○ TOP

○ LEFT

RIGHT ○

BOTTOM ○

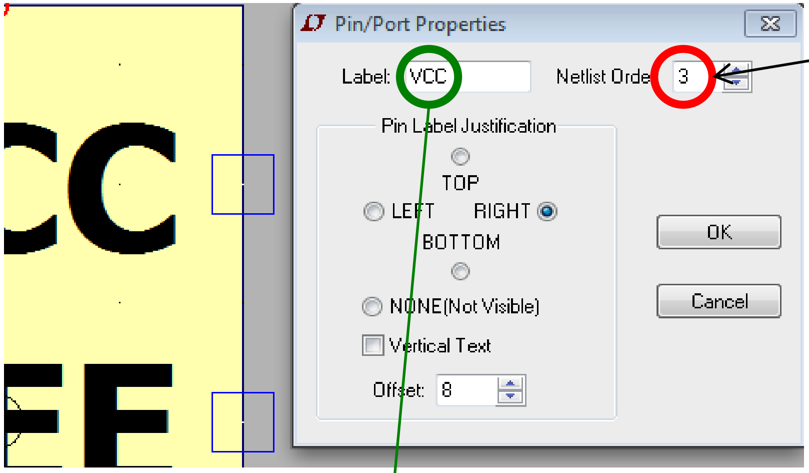
● NONE(Not Visible)

□ Vertical Text

Offset: 8

Symbol Editing – Pins and Netlist order

- "*Netlist Order*" determines the order this pin is netlisted for SPICE
  - It links a pin to a node of a SPICE model
  - The left most node in the model definition is associated to the 1st position of the netlist order and increments as such until the right most node corresponds to the last netlist order position.



Pin/Port Properties

Label: VCC

Netlist Orde 3

Pin Label Justification

○ TOP

○ LEFT

● RIGHT

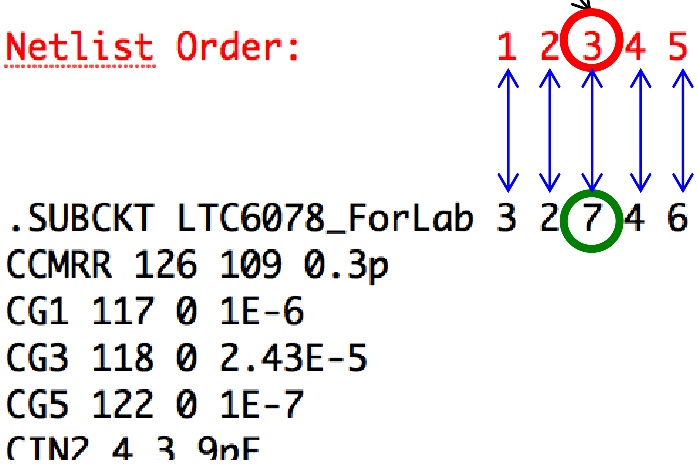
○ BOTTOM

○ NONE(Not Visible)

□ Vertical Text

Offset: 8

Netlist order



```
Netlist Order:                   1 2 3 4 5

.SUBCKT LTC6078_ForLab 3 2 7 4 6
CCMRR 126 109 0.3p
CG1 117 0 1E-6
CG3 118 0 2.43E-5
CG5 122 0 1E-7
CTN2 4 3 9nF
```

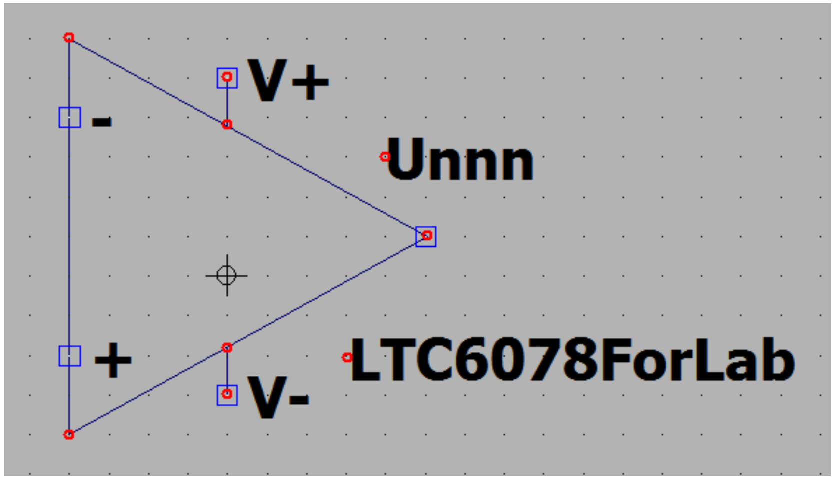
Setting Netlist Order 3 to pin VCC links VCC to Node7 of the SPICE model

Symbol Editing

Hands-On Exercise:

Modify the previously generated LTC6078ForLab symbol so it looks as follow:

S



- Once completed, reuse the “LTC6078TestCircuit.asc” circuit to verify the symbol is still functional.

C S

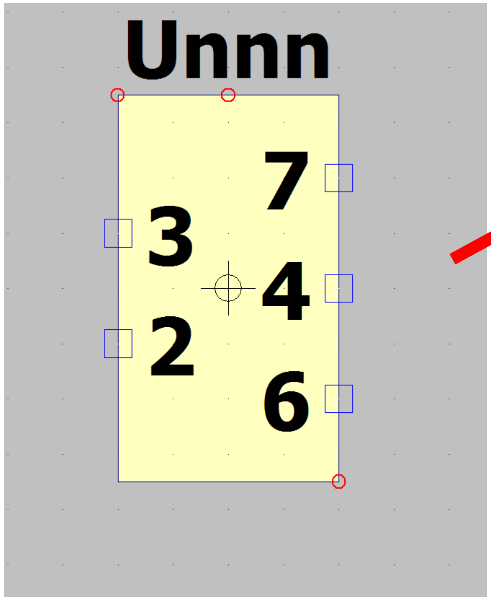
ADVANCED TOPICS

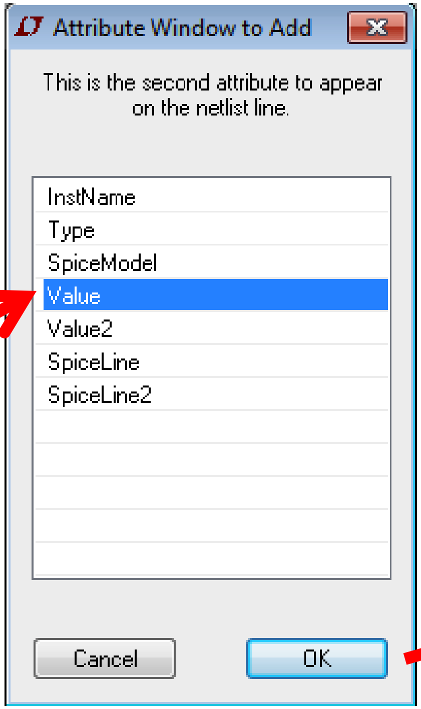
To consult at your leisure.

Symbol Editing – Attributes

Attribute Window [*Edit* ➔ *Attributes*]:

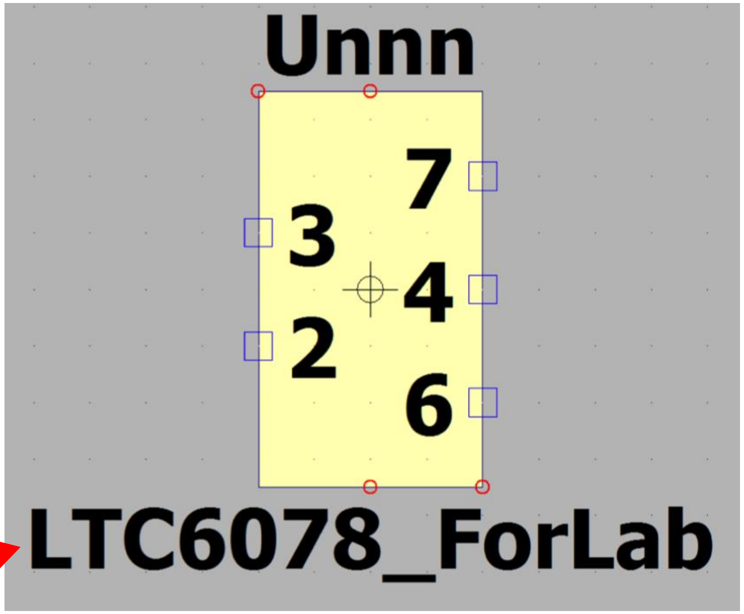
- To edit the visibility of attributes
- Select the attribute you desire to place on the symbol and click OK to place it.




Attribute Window to Add

This is the second attribute to appear on the netlist line.

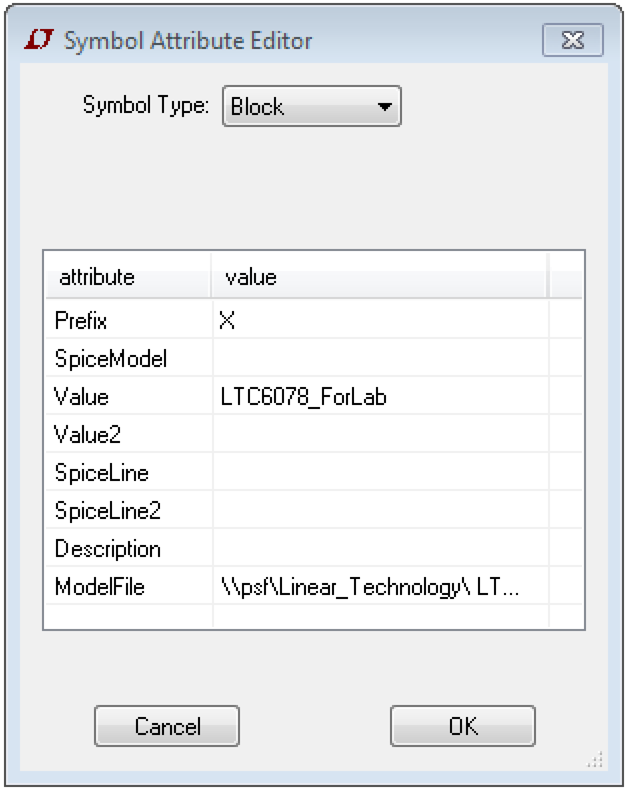
- InstName
- Type
- SpiceModel
- Value
- Value2
- SpiceLine
- SpiceLine2



Symbol Editing – Attributes

Edit Attributes [*Edit ➔ Attributes*]:

- Define default attributes for a symbol
- The default attributes are:
  - Prefix
  - SpiceModel
  - Value
  - Value2
  - SpiceLine
  - SpiceLine2
  - Description
  - ModelFile
  - Symbol Type



Symbol Type: Block

attribute	value
Prefix	X
SpiceModel	
Value	LTC6078_ForLab
Value2	
SpiceLine	
SpiceLine2	
Description	
ModelFile	\\psf\Linear_Technology\ LT...

Symbol Editing – Attributes

Prefix:

- The attribute “Prefix” is the most important.
- It determines the basic type of symbol:
  - If the symbol is intended to represent a SPICE primitive, the symbol should have the appropriate prefix: R for resistor, C or capacitor, M for MOSFET, etc.
    - See the “Circuit Element Quick Reference” (Help Menu) for a complete set of SPICE primitives available.
  - The Prefix should be 'X' if you want to use the symbol to represent a subcircuit.

Symbol Editing – Attributes

Symbol Type:

- If you wish the symbol to represent another page of a hierarchical schematic, all attributes should be left blank and the symbol type should be changed from "Cell" to "Block". No attribute values need be set

Symbol Editing – Attributes

Value, Value2, Spiceline and Spiceline2:

- Along with Spicemodel, these attributes are all part of the overall value of the component. A component with these attributes defined is usually netlisted as:

  <name> node1 node2 [...] <SpiceModel>

  + <Value> <Value2> <SpiceLine> <SpiceLine2>

- Can be used to enter fixed or parametric values or spice directives
  - The main value for the component, such as a FET name or a capacitor value, is typically defined via the *Value* attribute.
    - This attribute is usually made visible on the schematic
  - Value2, Spiceline and Spiceline2 are secondary values
    - For an intrinsic model, these attributes, at the schematic level, are commonly used to pass instance parameters

Symbol Editing – Attributes

Value, Value2, Spiceline and Spiceline2:

- As for intrinsic models (caps, inductors, etc), instance specific parameters can be created for subcircuits
  - The *Value2*, *Spiceline* and *Spiceline2* attributes are often use to define the parameters at the symbol and component (schematic) level
  - The parameters have to exist inside the subcircuit definition in order to be modified on the schematic :

```
.subckt divider n1 n2 n3
r1 n1 n2 {top}
r2 n2 n3 {bot}
.ends
```

    - All instances of a symbol calling such a .subckt definition could see their parameters being set by a single .param statement defined on the schematic (ex. ".param top=20k").

DividerTestCircuit.asc

Symbol Editing – Attributes

Value, Value2, Spiceline and Spiceline2:

- To limit the scope of a subckt parameter to a given instance of a symbol calling that subcircuit model, the parameter must be set by using these methods:
  - A *.param* statement can be added to the subcircuit.

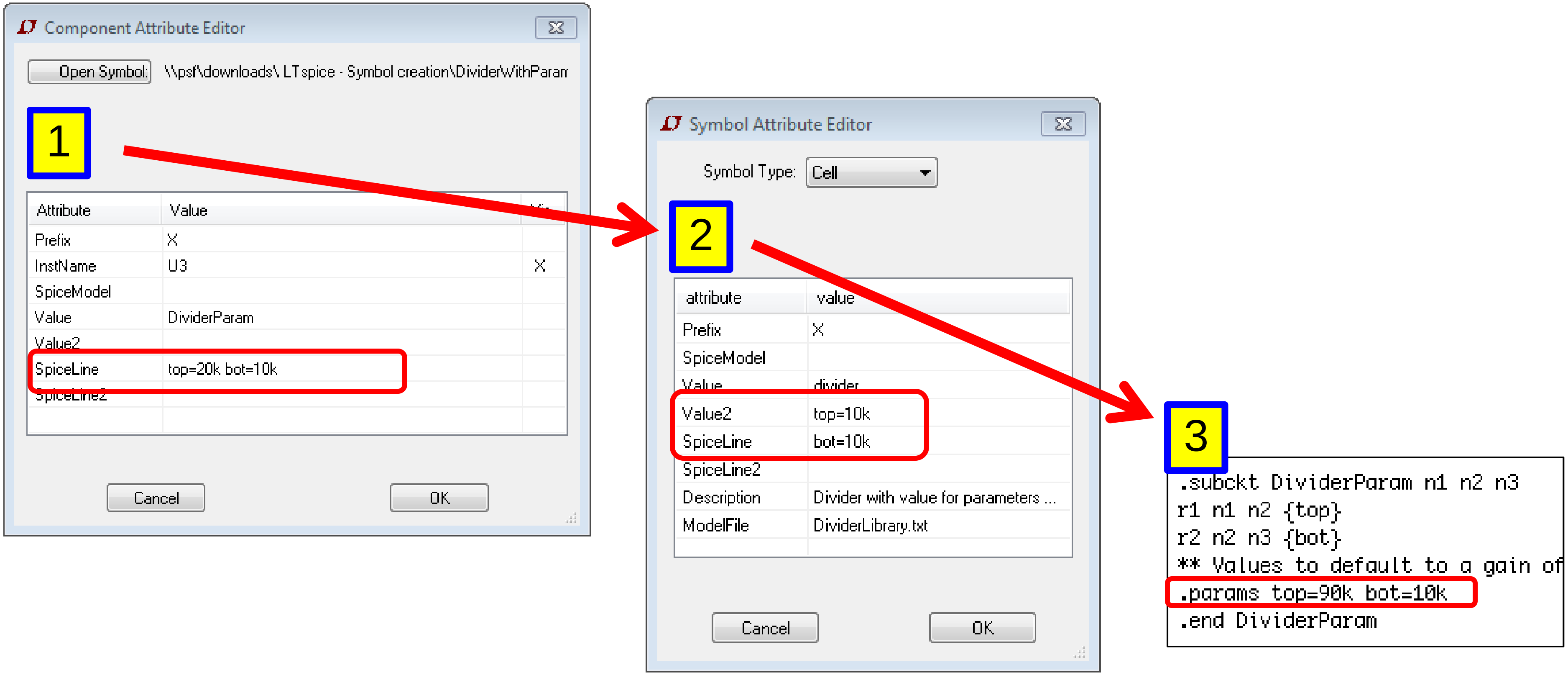
    ```
    .subckt divider n1 n2 n3
    r1 n1 n2 {top}
    r2 n2 n3 {bot}
    .params top=10k  bot=20k   ** subckt instance uses these values by default
    .ends
    ```

  - Using the "Symbol Attribute Editor", define the symbol's *Value2*, *Spiceline* and *Spiceline2* attributes to set the parameter(s).
  - Using the "Component Attribute Editor" window, modify and/or add the attributes *Value2*, *Spiceline* and *Spiceline2* of the component to set the parameter(s).

Symbol Editing – Attributes

Value, Value2, Spiceline and Spiceline2:

- The level of precedence of the parameter definitions listed on the previous slide is shown below:



Component Attribute Editor

\\psf\downloads\ LTspice - Symbol creation\DividerWithParam

Attribute	Value	Vis.
Prefix	X	
InstName	U3	X
SpiceModel		
Value	DividerParam	
Value2		
SpiceLine	top=20k bot=10k	
SpiceLine2		

Symbol Attribute Editor

Symbol Type: Cell

attribute	value
Prefix	X
SpiceModel	
Value	divider
Value2	top=10k
SpiceLine	bot=10k
SpiceLine2	
Description	Divider with value for parameters ...
ModelFile	DividerLibrary.txt

```
.subckt DividerParam n1 n2 n3
r1 n1 n2 {top}
r2 n2 n3 {bot}
** Values to default to a gain of
.params top=90k bot=10k
.end DividerParam
```

Symbol Editing – Attributes

SpiceModel and ModelFile:

- Name of the file including the SPICE model
  - SpiceModel can also be the name of a subcircuit
- An absolute path name (ex. C:\Users\Libraries\...) does not need to be added to the file name if the file is located in either:
  - `<LTspiceIV>` \lib\sub
  - The directory that contains the simulation file

LTC6090TestCircuitSolution.asc

Symbol Editing – Attributes

SpiceModel and ModelFile:

- There are two methods of defining a subcircuit symbol (*Prefix* attribute = 'X' define) so it automatically includes the library that contains the definition of the subcircuit called by the component:
  - Symbol definition #1:
    - Define attributes *Value* and *Value2*.
    - Such a component netlists as two lines of SPICE:

      .lib <SpiceModel>

      <name> node1 node2 [...] <Value2>
    - ***Components defined using this method are not editable on the schematic.***

LTC6090TestCircuitSolution.asc

Symbol Editing – Attributes

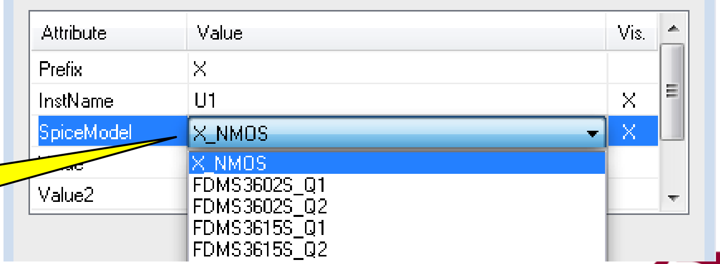
SpiceModel and ModelFile:

- Symbol definition #2:
  - Define a *ModelFile* attribute.
  - Such a component netlists as two lines of SPICE:

    .lib <ModelFile>

    <name> node1 node2 [...] <SpiceModel> <Value> <Value2> <SpiceLine> <SpiceLine2>

  - **An instance of this symbol is editable on a schematic**
  - If *SpiceModel* exists and is the name of a subcircuit in the file <*ModelFile*> then a drop list of all subcircuits names will be available when an instance of the symbol is edited on a schematic.



Attribute	Value	Vis.
Prefix	X	
InstName	U1	X
SpiceModel	X_NMOS	X
[illegible]		
Value2		

X_NMOS
FDMS3602S_Q1
FDMS3602S_Q2
FDMS3615S_Q1
FDMS3615S_Q2

Pull-down Menu !

NmosSubcktDropList.asc