Qspice - General Reference Guide by KSKelvin

KSKelvin Kelvin Leung

Created on 8-4-2023
Last update on 1-10-2023

QSPICE

- QSPICE
  - Arthor : Mike Engelhardt
  - Download : https://www.qorvo.com/design-hub/design-tools/interactive/qspice
- Topic Included in this guideline
  - Shortcut Key
  - Hierarchical and Sub-circuit
  - Waveform Viewer
  - Simulation Technique

Part 1
Shortcut Key

Schematic Editor Keyboard Shortcuts

HELP > Schematic Capture > Schematic Editor > Keyboard Shortcuts

Key	Command
(spacebar)	Zoom to fit
B[1]	Behavioral source
C[1]	Capacitor
D[1]	Diode
E[1]	E-source
F	F-source
G[2]	Ground, G-source
H	H-source
I	Current Source
J[1]	JFET
L[1]	Inductor
M[1]	MOSFET
Q[1]	Bipolar Transistor
R[1]	Resistor
S[1]	Voltage Controlled Switch
T[3]	Place Text
V[1]	Voltage Source
W	Start a wire
Y	Piezoelectric Crystal
Z[1]	MESFET

Ctrl-A	Draw an arc(graphical annotation)
Ctrl-B	Draw a box(graphical annotation)
Ctrl-C	Copy selected object(s) to clipboard
Ctrl-F	Find
Ctrl-G	Toggle display of grid dots
Ctrl-L	Draw a line(graphical annotation)
Ctrl-M	Mirror selected object(s)
Ctrl-R	Rotate selected object(s)
Alt-Ctrl-R	Rotate in 45° increments
Ctrl-V	Paste
Ctrl-X	Cut
Ctrl-Y	Redo
Ctrl-Z	Undo
Ctrl-3	Draw a triangle(graphical annotation)
;	Toggle a text graphic's comment status
F2	Toggle visibility of the Symbol and IP Browser pane.
F3	Toggle visibility of the Symbol Properties pane.
F4	Toggle visibility of the output console.
F5	Run the simulation.

[1] Repeated depressions of the key cycles through different versions of the symbol.
[2] Repeated depressions of 'G' cycles through different versions of the ground symbol and then G-source symbols.
[3] The period key, '.', is accepted as a synonym for 'T'.

Symbol Editor and Waveform Viewer Keyboard Shortcuts

HELP > Waveform Viewer > Keyboard Shortcuts / HELP > Schematic Capture > Symbol Editor > Keyboard Shortcuts

Waveform Viewer Keyboard Shortcuts

Key	Command
Delete	Delete attached cursor if pointing to a readout or delete selected plot labels
F	Zoom to fit(all panes)
F4	Toggle visiblity of the console display
F5	Rerun the simulation
␣	Reload Plot configuration file
←	Move attached cursor left
→	Move attached cursor right
↑	Move attached cursor to next step } for .step
↓	Move attached cursor to previous step } for .step
Ctrl-A	Add a trace
Ctrl-C	Copy
Ctrl-D	Delete a plotting pane
Ctrl-F	Find
Ctrl-G	Turn Grid On/Off
Ctrl-V	Paste
Ctrl-P	Print
Ctrl-W	Add a plotting pane
Ctrl-X	Cut
Ctrl-Y	Redo
Ctrl-Z	Undo

** Probe Differentiate Voltage : Hold Alt and click differentiate nodes

Symbol Editor Keyboard Shortcuts

Key	Command
Ctrl-A	Draw an arc defined by three points
Shift-Ctrl-A	Draw an arc defined by four points
Ctrl-B	Draw a box(or a box for an image)
Ctrl-C	Copy selection(s) to clipboard
E	Draw an Ellipse
Ctrl-F	Find
Ctrl-L	Draw a line
Ctrl-M	Mirror selected objects
P	Place a pin
Ctrl-R	Rotate selected objects
T	Place a text attribute
Ctrl-V	Paste
Ctrl-X	Cut
Ctrl-Y	Redo
Ctrl-Z	Undo
Ctrl-3	Draw a triangle
F3	Toggle visibility of the Properties pane.

Waveform Viewer Functions and Keywords (.func , .meas)

HELP > Waveform Viewer > Waveform Expressions

The following functions, constants, and keywords are recognized in expressions of waveform data.

Waveform Viewer Functions and Keywords

Syntax	Description
ABS(x)	Absolute value of x
ACOS(x)	Inverse cosine of x
ACOSH(x)	Inverse hyperbolic cosine of x
ARCCOS(x)	Inverse cosine of x
ARCCOSH(x)	Inverse hyperbolic cosine of x
ARCSIN(x)	Inverse sine of x
ARCSINH(x)	Inverse hyperbolic sine of x
ARCTAN(x)	Inverse tangent of x
ARCTANH(x)	Inverse hyperbolic tangent of x
ASIN(x)	Inverse sine of x
ASINH(x)	Inverse hyperbolic sine of x
ATAN(x)	Inverse tangent of x
ATAN2(x,y)[1]	Four quadrant inverse tangent of x
ATANH(x)	Inverse hyperbolic tangent of x
BUF(x)	x > .5 ? 1 : 0
CBRT(x)	Cube root of x
CEIL(x)	x rounded up to nearest integer
COS(x)	Cosine of x
COSH(x)	Hyperbolic cosine of x
COT(x)	Cotangent of x
D(x)	Derivative of x
DD(x)	Second derivative of x
D²(x)	Second derivative of x
E	2.7182818284590452354
ERF(x)	Error function of x
ERFC(x)	Complementary error function of x
EXP(x)	e raised to the x power
EXP10(x)	10 raised to the x power
FABS(x)	Absolute value of x
FLOOR(x)	x rounded down to nearest integer
FREQ	Frequency
FREQUENCY	Frequency
GAMMA(x)	Gamma function of x
HYPOT(x,y)	√(x²+y²)
IF(x,y,z)	(x > .5) ? y : z
ILOGB(x)	Unbiased exponent of x
IM(x)	Imaginary part of x
IMAG(x)	Imaginary part of x
INT(x)	x rounded to nearest integer
INV(x)	x > .5 ? 0 : 1
INVSQRT(x)	1÷√x
ISNAN(x)	One if x is not a number, otherwise zero
J	√-1
J0(x)	Zero order Bessel function of the first kind at x
J1(x)	First order Bessel function of the first kind at x
JN(x,n)	N^{th} order Bessel function of the first kind at x
K	1.380649e-23 J/°K
LGAMMA(x)	Log-gamma function of x
LIMIT(x,y,z)	Mutually intermediate value of x,y, and z
LN(x)	Natural logarithm of x
LOG(x)	Natural logarithm of x
LOG10(x)	Logarithm of x in base 10
LOG1P(x)	Natural logarithm of (x + 1)
LOG2(x)	Logarithm of x in base 2
LOGB(x)	LOG2(ABS(x))
MAG(x)	Absolute value of x
MAX(x,y)	Maximum of x and y
MAXMAG(x,y)	x or y with maximum magnitude
NAN	A value guaranteed to be not a number
MIN(x,y)	Minimum of x and y
MINMAG(x,y)	x or y with minimum magnitude
PH(x)	Phase of x
PHASE(x)	Phase of x
PI	3.14159265358979323846
POW(x,y)	x raised to the y power
POWN(x,y)	x raised to the nearest integer value of y
PWR(x,y)	Absolute value of x raised to the y power
PWRS(x,y)	x >= 0 ? x^y : $-x^y$
Q	1.602176487e-19 Coulomb
RE(x)	Real part of x
REAL(x)	Real part of x
RINT(x)	x rounded to the nearest integer
ROUND(x)	x rounded to the nearest integer
SGN(x)	Sign of x
SIGN(x)	Sign of x
SIN(x)	Sine of x
SINH(x)	Hyperbolic sine of x
SQRT(x)	Square root of x
TABLE(x,x1,y1,...)	Interpolate the table given as x1,y1, x2,y2,... at point x
TAN(x)	Tangent of x
TANH(x)	Hyperbolic tangent of x
TAUGRP(x)	Group delay of x
TBL(x,x1,y1,...)	Interpolate the table given as x1,y1, x2,y2,... at point x
TEMP	Circuit temperature
TG(x)	Group delay of x
TIME	Time
TRUNC(x)	Integer part of s
URAMP(x)	x > 0 ? x : 0
USTEP(x)	x > 0 ? 1 : 0
Y0(x)	Zero order Bessel function of the second kind at x
Y1(x)	First order Bessel function of the second kind at x
YN(x)	N^{th} order Bessel function of the second kind at x

[1] For complex data, the syntax is ATAN2(z). The meaning is ATAN2(IMAG(z),REAL(z)).

simulation variable

important constant

Function and Operators for Behavioral V and I Sources

HELP > Simulator > Device Reference > B. Behavioral Sources

Functions

Name	Description
abs(x)	Absolute value of x
acos(x)	arc cosine of x
arccos(x)	Synonym for acos()
acosh(x)	arc hyperbolic cosine of x
asin(x)	arc sine of x
arcsin(x)	Synonym for asin()
asinh(x)	Arc hyperbolic sine
atan(x)	Arc tangent of x
arctan(x)	Synonym for atan()
atan2(y,x)	Four quadrant arc tangent of y/x
atanh(x)	Arc hyperbolic tangent
buf(x)	1 if x > .5, else 0
ceil(x)	Integer equal or greater than x
cos(x)	Cosine of x
cosh(x)	Hyperbolic cosine of x
ddt(x)	Time derivitive x
delay(x,y)	x delayed by y
delay(x,y,z)[1]	x delayed by y, but store no more than z history
dlim(x,y,z)	x bounded by y which it asymptotically starts to approach at y+z as a first inverse order Laurent series
exp(x)	e to the x
floor(x)	Integer equal to or less than x
hypot(x,y)	sqrt(x^2 + y^2)
idt(x,y,z)	Time integral of x with initial condtion of y reset when z > .5
if(x,y,z)	If x > .5, then y else z
int(x)	Convert x to integer
inv(x)	0. if x > .5, else 1.
limit(x,y,z)	Intermediate value of x, y, and z
ln(x)	Natural logarithm of x
log(x)	Alternate syntax for ln()
log10(x)	Base 10 logarithm
max(x,y)	The greater of x or y
min(x,y)	The smaller of x or y
pow(x,y)	x^y
pwr(x,y)	abs(x)^y
pwrs(x,y)	sgn(x)*abs(x)^y
random(x)	Random number from 0. to 1. depending on the integer value of x. Interpolation between random numbers is linear for non-integer x.
sin(x)	Sine of x
sinh(x)	Hyperbolic sine of x
sqrt(x)	Square root of x
table (x,a,b,c,d,...)	Interpolate x from the look-up table given as a set of pairs of constant values.
tan(x)	Tangent of x.
tanh(x)	Hyperbolic tangent of x
ulim(x,y,z)	x bounded by y which it asymptotically starts to approach at y-z as a first inverse order Laurent series

Operators grouped in reverse order of precedence of evaluation

Operand	Description
&	Boolean AND
|	Boolean OR
>	True if expression on the left is greater than the expression on the right.
<	True if expression on the left is less than the expression on the right.
>=	True if expression on the left is greater than or equal the expression on the right.
<=	True if expression on the left is less than or equal the expression on the right.
+	Addition
-	Subtraction
*	Multiplication
/	Division
**	** / ^ Raise left hand side to power of right hand side. Same as '^'.
!	Boolean not the following expression.

Available Function in B source not listed

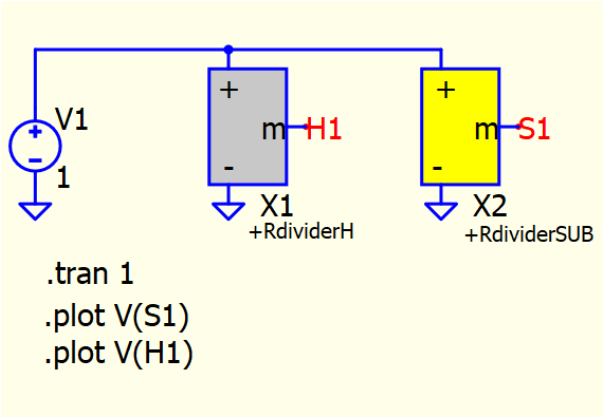
- Trunc(x) ; floor(x) ; int(x) : rounded down integer
- Rint(x) ; round(x) : rounded to nearest integer
- Ceil(x) : rounded up integer
- Ustep(x) : x > 0 ? 1 : 0
- Uramp(x) : x > 0 ? x : 0

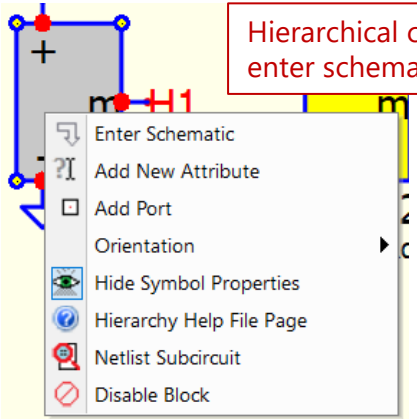
Part 2
Hierarchical and Sub-circuit

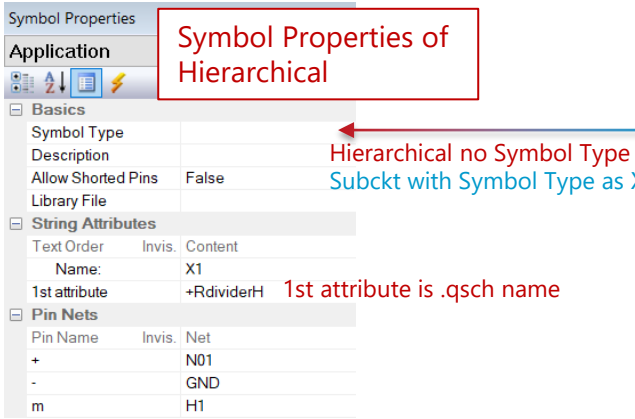
Hierarchical and Sub-circuit : Comparison

Qspice : parent - hierarchical and subckt.qsch | +RdividerH.qsch

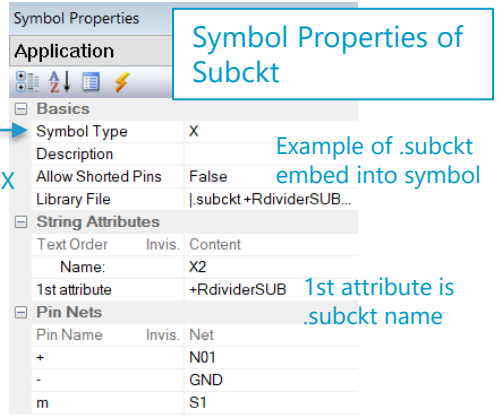
- Hierarchical and Sub-circuit
  - They are similar and both support by .qsym symbol, but two different concepts
  - Hierarchical
    - Call a child schematic (.qsch) for simulation
      - Circuit in child schematic (.qsch)
    - Waveform viewer can probe simulation result in daughter schematic
  - Sub-circuit (.subckt)
    - Call a sub-circuit (.subckt) for simulation
      - Circuit in .subckt model
    - Waveform viewer cannot probe simulation result in subckt
      - Result is calculated and stored, just not able to directly probe it.
    - In Qspice, .subckt syntax can embed into .qsym in library file properties (i.e. can share a single .qsym file for simulation)







Hierarchical no Symbol Type
Subckt with Symbol Type as X



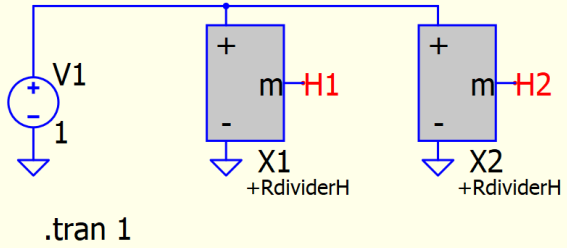
Example of .subckt embed into symbol

Hierarchical and Sub-circuit : Comparison

Qspice : parent - hierarchical and subckt (dual hierarchical/subckt).qsch

- Hierarchical and Sub-circuit
  - In netlist, both Hierarchical and Sub-circuit call .subckt syntax
  - Hierarchical
    - Child schematic is a .subckt in Partent netlist
    - Symbol calls this child schematic name
  - Sub-circuit (.subckt)
    - Each symbol calls an individual .subckt by naming its by add prefix as Xnnn•<subckt name>

Hierarchical Block



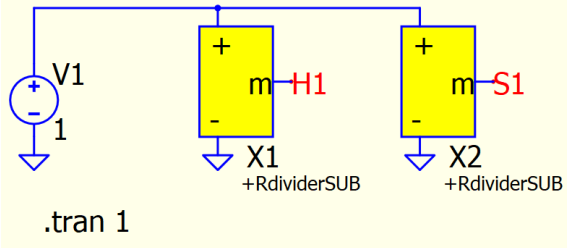

```
* C:\QspiceKSKelvin\01 User Guide and Script\02
V1 N01 0 1
X1 N01 0 H1 +RdividerH
X2 N01 0 H2 +RdividerH

.subckt +RdividerH + - m
R1 + m 1K
R2 m - 1K
.ends +RdividerH

.tran 1
.end
```

.subckt for X1 and X2
<child schematic name>

Sub-Circuit (.subckt)




```
* C:\QspiceKSKelvin\01 User Guide and Script\02
V1 N01 0 1
.subckt X1•+RdividerSUB + - m
R1 + m 1K
R2 m - 1K
.ends +Rdivider
X1 N01 0 H1 X1•+RdividerSUB
.subckt X2•+RdividerSUB + - m
R1 + m 1K
R2 m - 1K
.ends +Rdivider
X2 N01 0 S1 X2•+RdividerSUB
.tran 1
.end
```

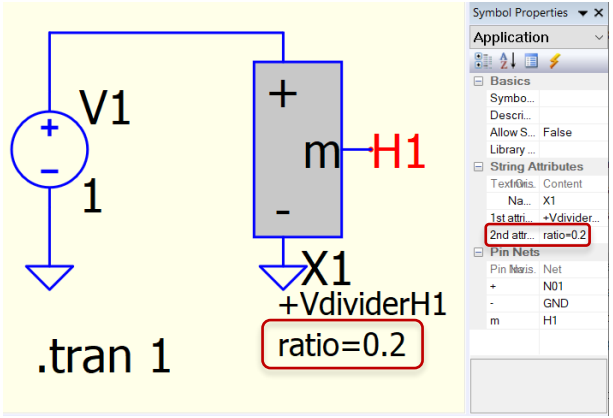
.subckt for X1
X1•<subckt name>

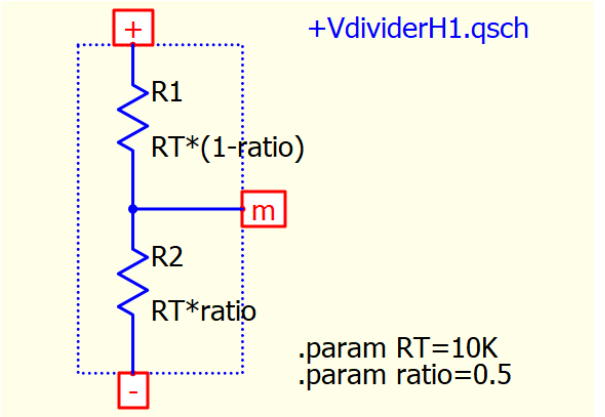
.subckt for X2
X2•<subckt name>

Hierarchical and Sub-circuit : Parameter Passing

Qspice : parent-PassParamHierarchical.qsch | +VdividerH1.qsch

- Parameter Passing
  - Hierarchical and Sub-circuit works in same way
  - As default, .subckt or child schematic load its .param
  - In parent schematic, if string attribute in symbol contains parameters, they will override .param within .subckt or child schematic

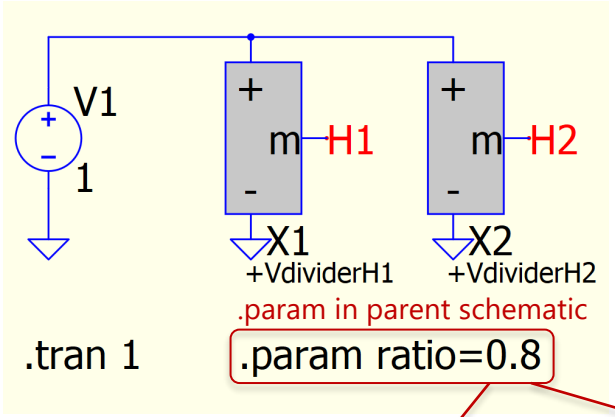


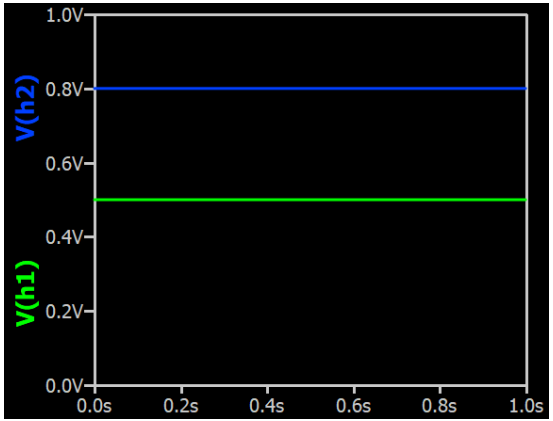


Parameter Passing with Global .param from parent

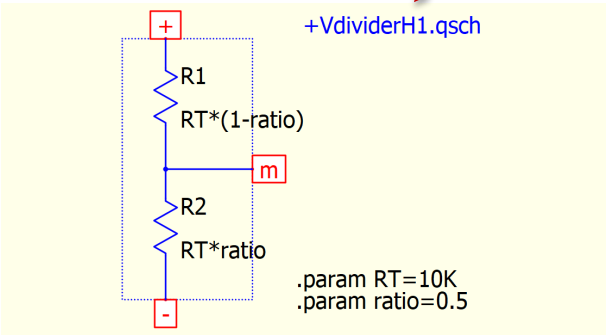
Qspice : parent-PassGlobalParam.qsch | +VdividerH1.qsch | +VdividerH2.qsch

- .param from Parent
  - Global .param passing from parent into a child schematic depends whether this child schematic has the parameter defined
  - If no such parameter is defined in child schematic, global .param override
  - If parameter is defined in child, global .param is ignored. Only string attribute in symbol has ability to override child schematic defined parameter

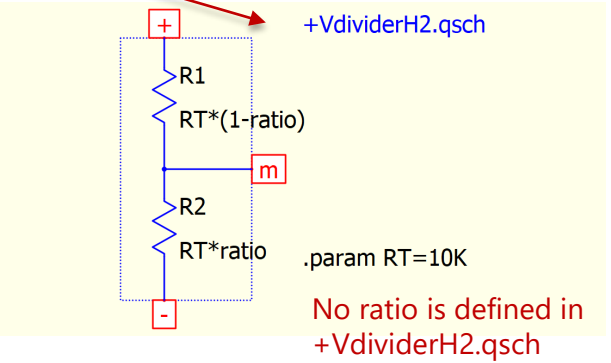




No effect



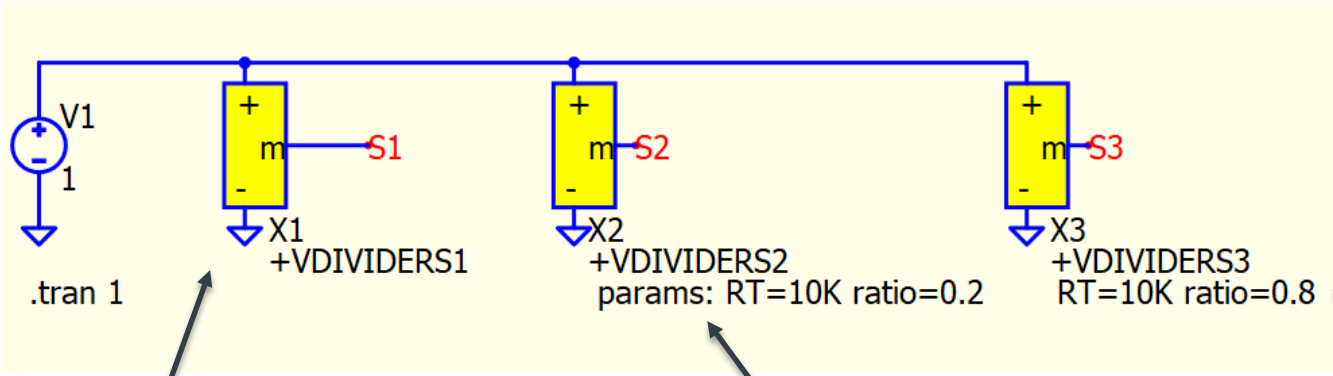
In effect



Three Way to Define Default Parameters in .subckt

Qspice : parent-PassParamSubckt.qsch | +VdividerS.txt

- Three Way to Define Default Parameters in .subckt




```
.subckt +VdividerS1 + - m
R1 + m RT*(1-ratio)
R2 m - RT*ratio
.param RT=10K
.param ratio=0.5
.ends +VdividerS1
```

```
.subckt +VdividerS2 + - m params: RT=10K ratio=0.5
R1 + m RT*(1-ratio)
R2 m - RT*ratio
.ends +VdividerS2
```

```
.subckt +VdividerS3 + - m RT=10K ratio=0.5
R1 + m RT*(1-ratio)
R2 m - RT*ratio
.ends +VdividerS3
```

For this version, if removes .param lines, param can be added in symbol in string attribute
Right Click on symbol > Add New Attribute

Three Type of Sub-Circuit (.subckt) Symbol (.qsym)

Qspice : parent - 3 type subckt symbol.qsch

Embedded SUBCKT
[Easy to share, just one .qsym]

Symbol Properties

Application

Basics		
Symbol Type	X	
Description		
Allow Shorted Pins	False	
Library File	|.subckt +RdividerH1 + - m\nR...	
String Attributes		
Text Order	Invis.	Content
Name:		X1
1st attribute		+RDIVIDERH1

.subckt is in library file properties

Link to Library
[Recommend for complex .subckt]

Symbol Properties

Application

Basics		
Symbol Type	X	
Description		
Allow Shorted Pins	False	
Library File	subckt_library.txt	
String Attributes		
Text Order	Invis.	Content
Name:		X2
1st attribute		+RDIVIDERH2

Library is in library file properties

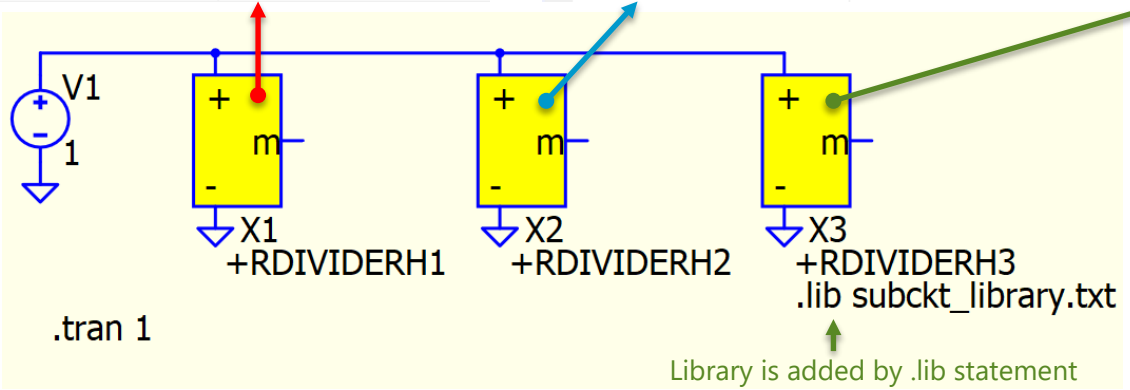
No Embed or Link
[For universal symbol]

Symbol Properties

Application

Basics		
Symbol Type	X	
Description		
Allow Shorted Pins	False	
Library File		
String Attributes		
Text Order	Invis.	Content
Name:		X3
1st attribute		+RDIVIDERH3

Nothing is in library file properties



subckt_library.txt

```
1  .subckt +RdividerH2 + - m
2  R1 + m 1K
3  R2 m - 2K
4  .ends +RdividerH2
5
6  .subckt +RdividerH3 + - m
7  R1 + m 1K
8  R2 m - 3K
9  .ends +RdividerH3
```

Hierarchical and Sub-circuit Sub-Topics

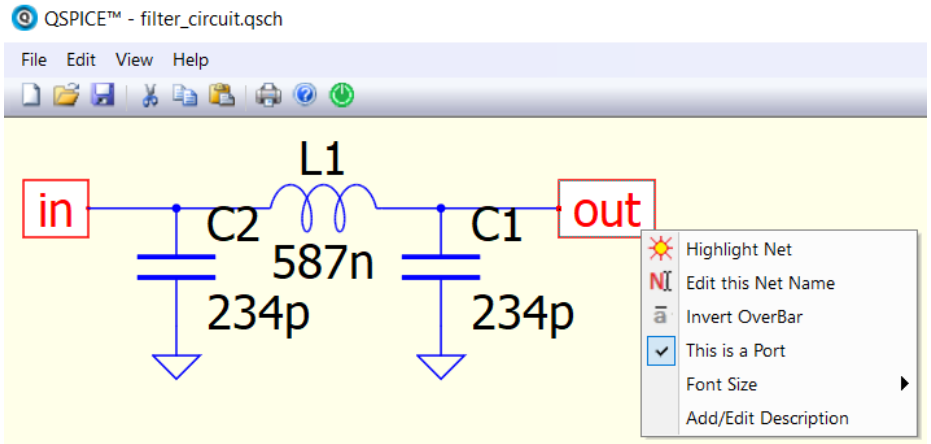
- Part 2A : Hierarchical Block
  - Create hierarchical block from child to parent or parent to child schematic
  - Create symbol for hierarchical block
  - Get .subckt from hierarchical block to convert into an embedded subckt symbol
- Part 2B : Symbol for Subckt [Embedded Subckt]
- Part 2C : Symbol for Subckt [Link to Library]
- Part 2D : Convert MOSFET M to subckt Symbol
  - Demonstrate how to convert a MOSFET M symbol into subckt to save effort in creating a MOSFET symbol for .subckt MOSFET model from 3rd party vendor
- Part 2E : Bus and Hierarchical Block

Part 2A
Hierarchical Block

Hierarchical Block : From Child to Parent

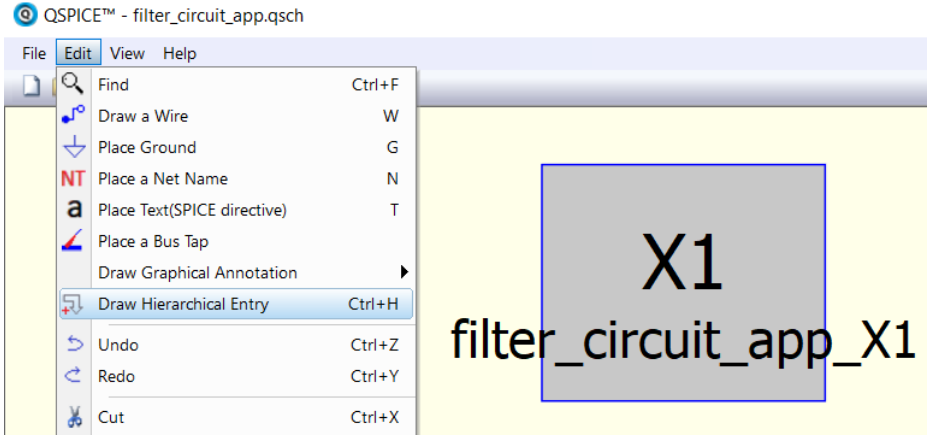
Qspice : filter_circuit.qsch | filter_circuit_app.qsch

[1] Create a child schematic (.qsch) with circuit and net label
[2] Right click on net label and select “This is a Port”



Method #1
[3] Create a new schematic which will call to use hierarchical
[4] Edit → Draw Hierarchical Entry

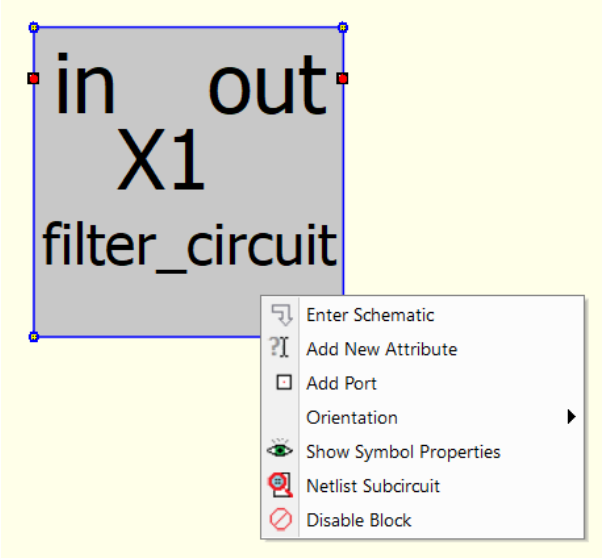
Method #2
[3] In child schematic, Right click > Open Parent Schematic
This will automatically create a parent schematic contains hierarchical symbol, with all Port automatically created



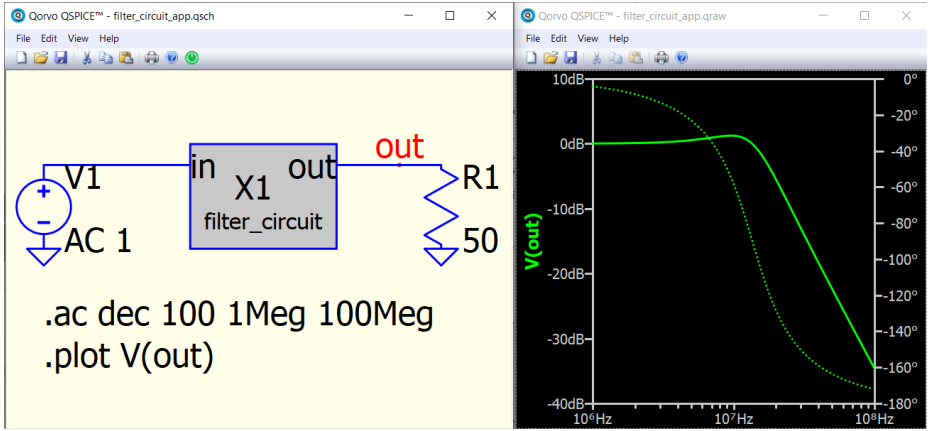
Hierarchical Block : From Child to Parent

Qspice : filter_circuit.qsch | filter_circuit_app.qsch

[5] Change component text (1st attribute) to match child schematic name
[6] Right click hierarchy component and “Add Port”
[7] Name ports as port name defined in child schematic
[8] Right click hierarchy component and “Enter Schematic” should open child schematic



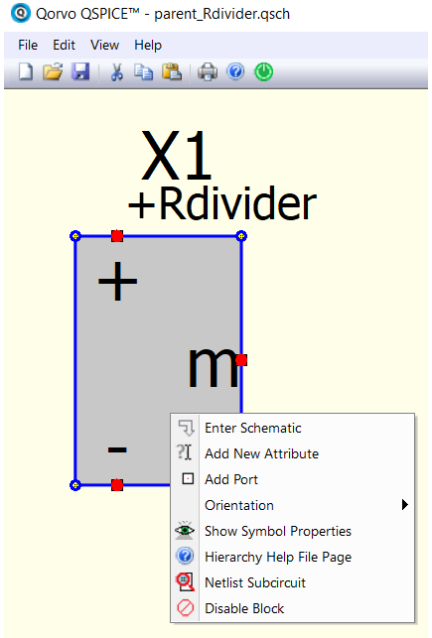
A completed example

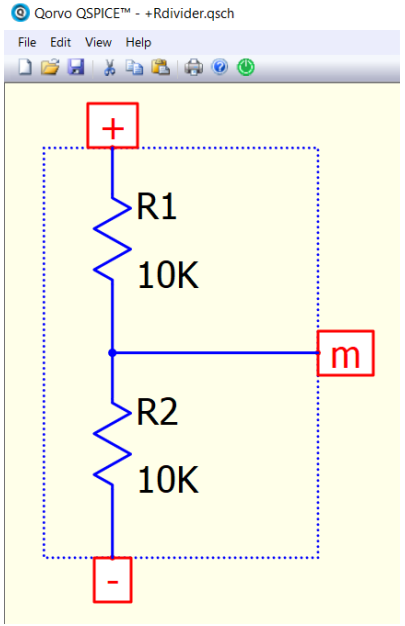


Hierarchical Block : From Parent to Child

[1] Right click > Draw Hierarchical Entry
[2] Rename component text (1st attribute) to child schematic name
** Child schematic will be created later
[3] Right click within Hierarchical Block > Add Port
[4] Right click > Enter Schematic, it will create a child .qsch

[5] Create circuit in child schematic

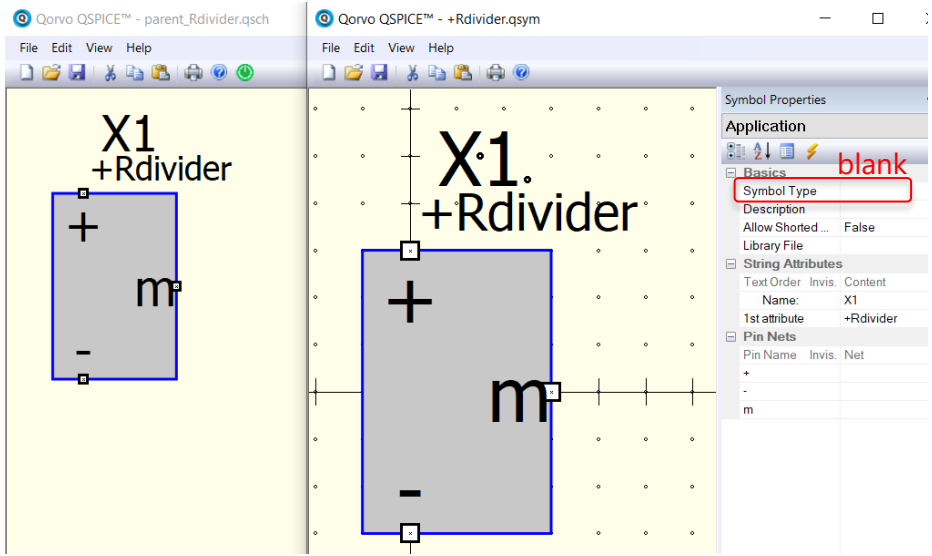




Hierarchical Block : Create Symbol (.qsym) for Hierarchical

Qspice : parent_Rdivider.qsch | +Rdivider.qsym

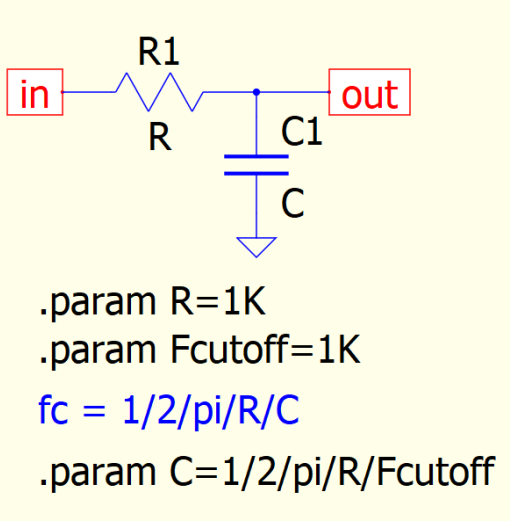
[1] In a parent schematic which contains a hierarchical block
[2] Hold Shift, draw a selection box to select Hierarchical
[3] Press Ctrl-C to copy
[4] File > New > New Symbol
[5] In New Symbol window, Press Ctrl-V to paste
[6] A symbol for hierarchical block is created, now, you can edit this symbol. Just bear in mind that "Symbol Type" must be blank for hierarchical symbol



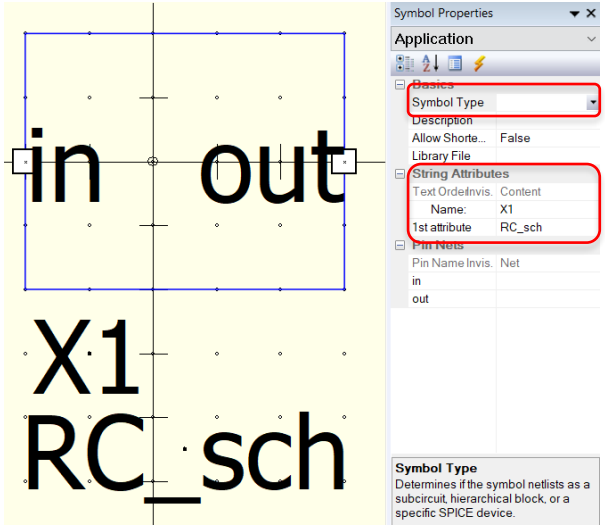
Create Symbol (.qsym) for Hierarchical : Demonstration #1

Qspice : RC_sch.qsym ; RC_sch.qsch

[1] Draw a schematic
This example has
- Two ports : in and out



[2] Create a symbol
- Pin name needs to match schematic ports (order not important)
- Use Text to assign
  - Name : X1
  - 1st attribute : [schematic name]
- Symbol Type : Blank (nothing)
  - Don't assign a X (X for subckt), hierarchical entry no symbol type



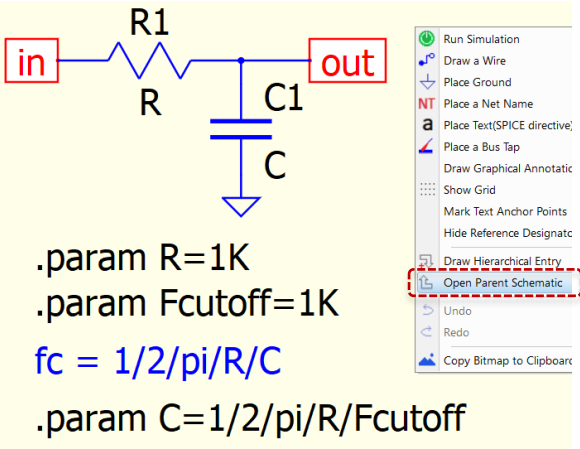
Remark :
Major Different for Symbol to call schematic (hierarchical entry) and subckt
- To call schematic (hierarchical entry)
  - Symbol Type : Blank
  - Name : X1
  - 1st attribute : schematic name
- To call subckt
  - Symbol Type : X
  - Name : X1
  - 1st attribute : subckt name

Create Symbol (.qsym) for Hierarchical : Demonstration #2

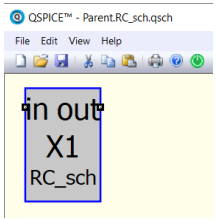
Qspice : RC_sch.qsym ; RC_sch.qsch

[1] Draw a schematic
This example has
- Two ports : in and out
- Right click these nets and select "This is a port" (only these ports will auto generate hierarchical entry)

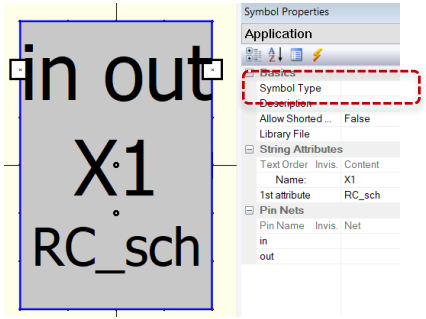
[2] Right Click > Open Parent Schematic
It will ask to automatically generate a parent schematic



[3] Copy hierarchical block in parent with Ctrl-C



[4] File > New > New Symbol, paste with Ctrl-V
** Symbol Type : Blank (nothing)
- Don't assign a X (X for subckt), hierarchical no symbol type. If you re-open a hierarchical symbol, please pay attention in here as it may auto assign an X into Symbol Type



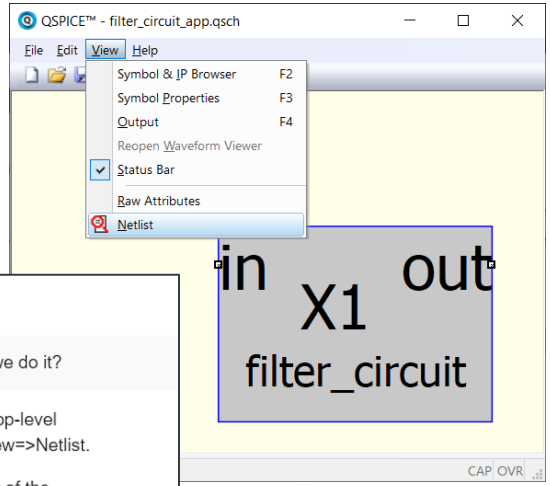
[5] Save as a .qsym symbol

Remark :
Major Different for Symbol to call schematic (hierarchical entry) and subckt
- To call schematic (hierarchical)
  - Symbol Type : Blank
  - Name : X1
  - 1st attribute : schematic name
- To call subckt
  - Symbol Type : X
  - Name : X1
  - 1st attribute : subckt name

Qspice HELP Reference
Help > Schematic Capture > Schematic Hierarchy

Hierarchical Block : Get subckt with Hierarchical Block method

[1] If a hierarchical block is created, in top-level schematic View > Netlist



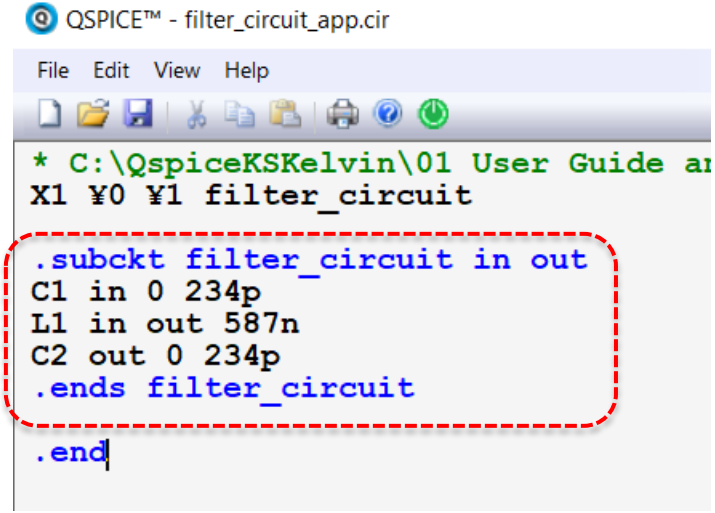
Engelhardt

Could we hear some detail how can we do it?

1. In the higher-level schematic, do top-level menu(not right click) command View=>Netlist.
2. In the netlist, identity the subcircuit of the hierarchal block you wish to library as widely usable IP.
3. Select that block of text and copy it to the clipboard with Ctrl-C.
4. Close the netlist and paste(Ctrl-V) the text into a schematic(or a blank symbol). That will invoke the 3rd party import routine.
5. You'll now have a symbol that contains the circuitry that you can use in any schematic in any directory.
6. Enjoy.

–Mike

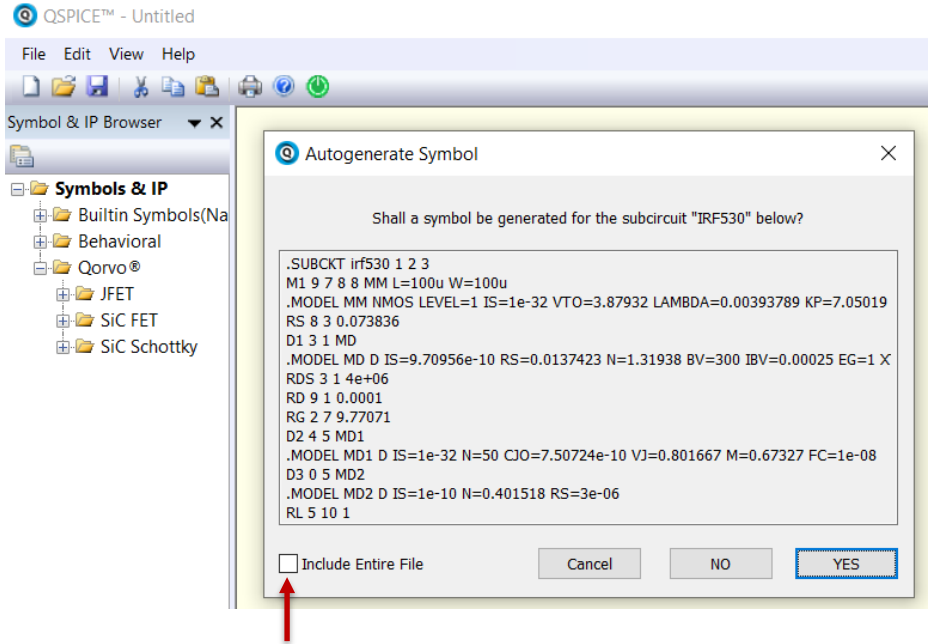
[2] In the netlist, identify the subcircuit of the hierarchal block
[3] Select that block of text and copy it to the clipboard with Ctrl-C
[4] Close the netlist and paste (Ctrl-V) the text into a schematic (or a blank symbol) to invoke the 3rd party import routine
[5] You'll now have a symbol that contains the circuitry that you can use in any schematic in any directory.



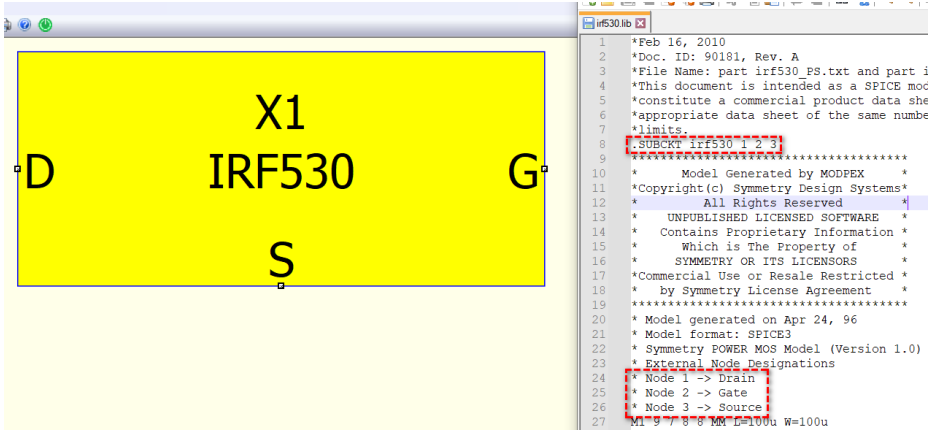
Part 2B
Symbol for Subckt
[Embedded Subckt]

Symbol for Subckt [Embedded Subckt]

[1] Assume user has a sub-circuit .subckt in text format library file
[2] Use a text editor to open library file, copy text for sub-circuit
[3] In Qspice schematic, paste the text (Ctrl-V)



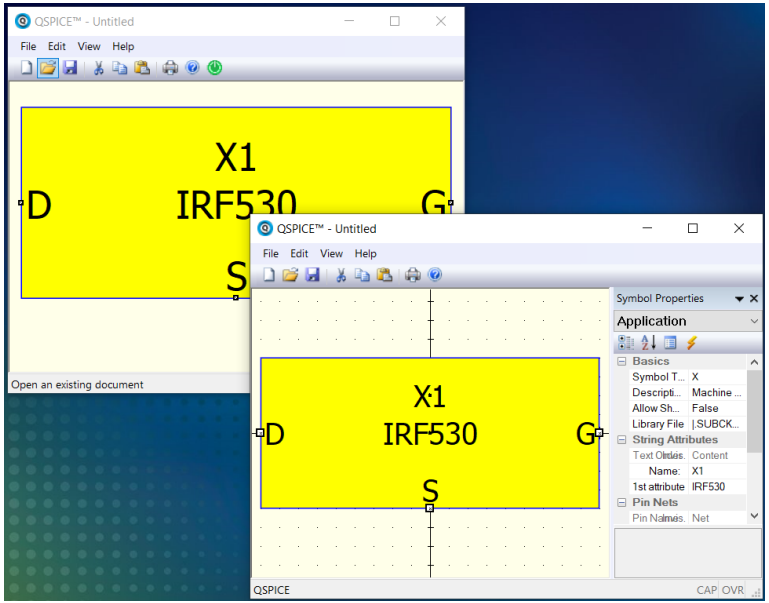
[4] Rename pins by referring to description in model file



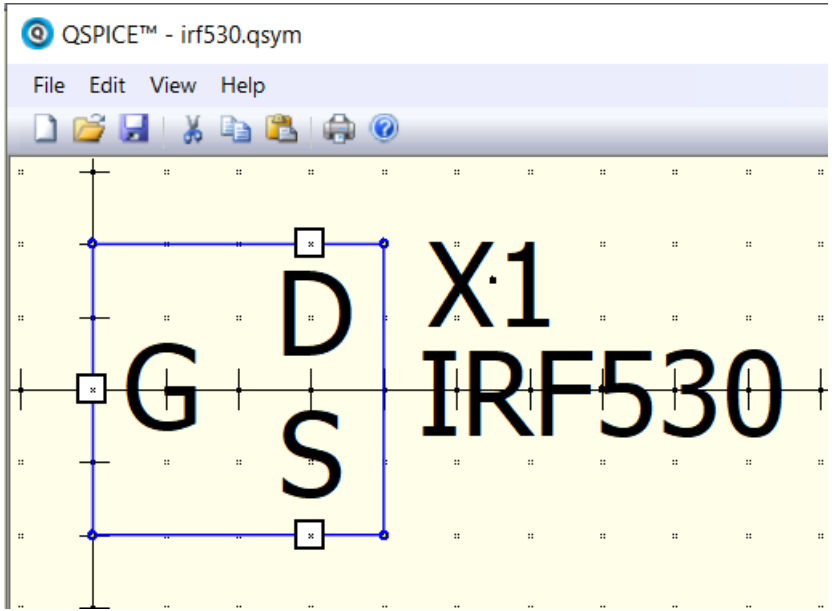
** If sub-circuit consist of other .subckt, click this block to include entire file

Symbol for Subckt [Embedded Subckt]

[5] File > New > New Symbol to open a Symbol Window
[6] In schematic, Ctrl-C to copy Component X1
[7] Goto Symbol Window, Ctrl-V to paste component

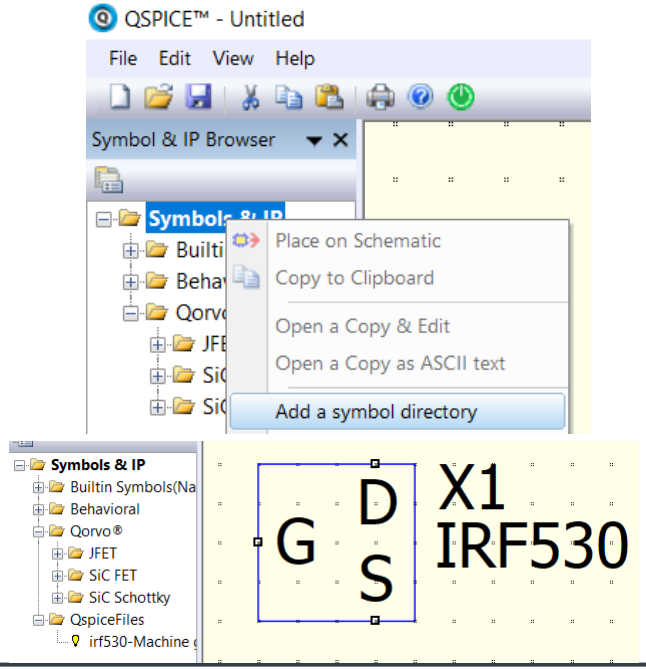


[8] Delete the box, rearrange pins location, and draw the symbol
[9] Save into a .qsym symbol format

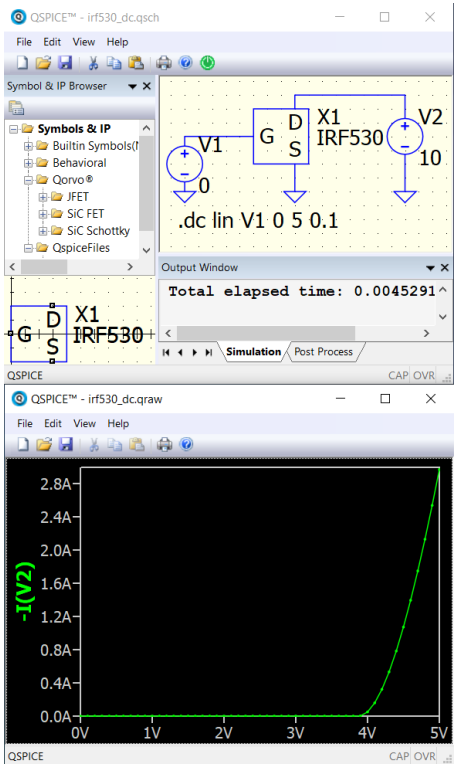


Symbol for Subckt [Embedded Subckt]

[10] In Schematic, Symbol & IP Browser, Right Click to "Add a symbol directory"
[11] Drag created component to schematic
[12] ** text library is no longer required as .subckt is integrated into symbol



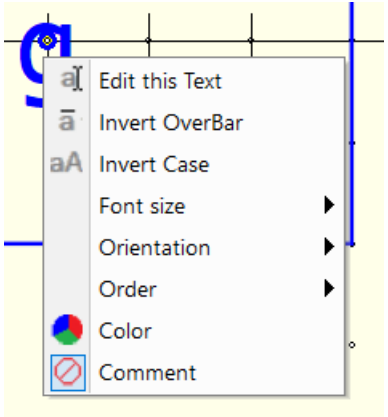
A completed example

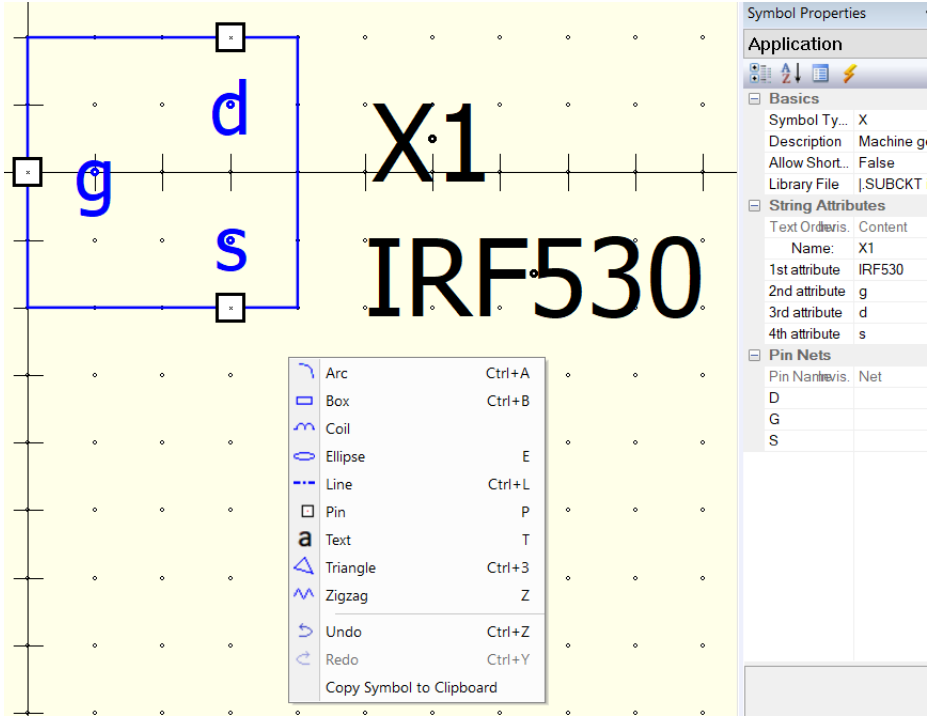


Symbol for Subckt [Embedded Subckt] : Label with Text

Qspice : irf530 with text.qsym

- Text can be used in label
  - For example, instead of changing net name, you can
    - Right click > Text
    - Right click on text > Select “Comment”
      - Text not comment will become valid item in netlist
      - Can change font size and color
    - Be careful 1st attribute is device name (e.g. IRF530 in example), and doesn’t comment it

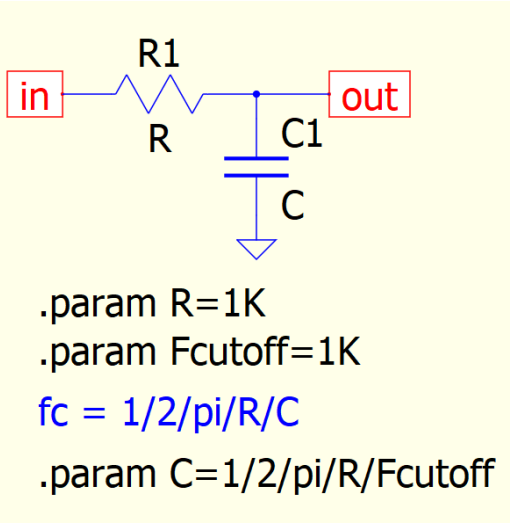




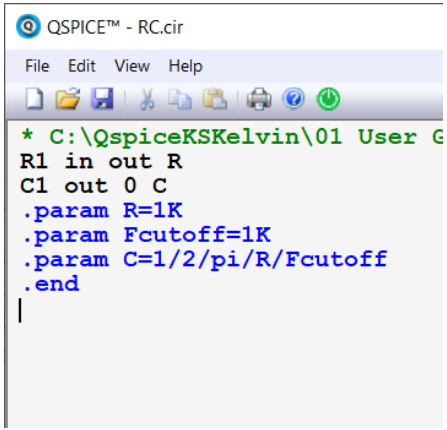
** Technique to Create embedded SUBCKT Script from Schematic

Qspice : RC_Params.qsym ; RC_sch.qsch

[1] Draw a schematic
This example has
- Two ports : in and out
- Two input params : R and Fcutoff
- One calculated parameter : C



[2] View > Netlist, copy this netlist



Subcircuit Definition

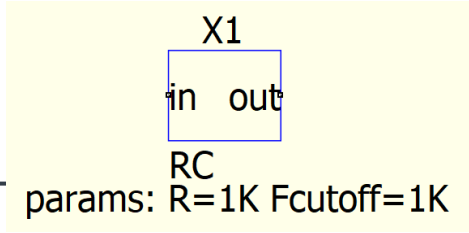
Syntax: .subckt NAME N1 N2 N3 ...
...
...
...
.ends NAME

[3 : Method#1] Convert netlist to subckt
- First line add .subckt
- RC is NAME of subckt determined by user
- Follow with ports (net) : in out
- Follow with params : R=1K Fcutoff=1K
- Remove .param R=1K and .param Fcutoff=1K
- Last line add .ends RC

This is the finished version

```
.subckt RC in out params: R=1K Fcutoff=1K
R1 in out R
C1 out 0 C
.param C=1/2/pi/R/Fcutoff
.ends RC
```

Copy and paste .subckt script to schematic, then follow standard symbol creation procedure for embedded SUBCKT symbol creation



** Technique to Create embedded SUBCKT Script from Schematic

Qspice : RC_noParams.qsym

[3 : Method#2] Convert netlist to subckt

- First line add .subckt
- RC is NAME of subckt determined by user
- Follow with ports (net) : in out
- Remove .param R=1K and .param Fcutoff=1K
- Last line add .ends RC

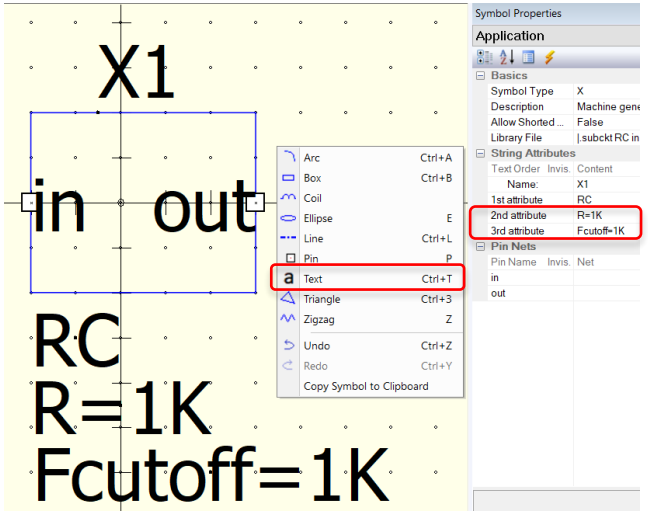
This is the finished version

```
.subckt RC in out
R1 in out R
C1 out 0 C
.param C=1/2/pi/R/Fcutoff
.ends RC
```

Copy and paste .subckt script to schematic, then follow standard symbol creation procedure for embedded SUBCKT symbol creation

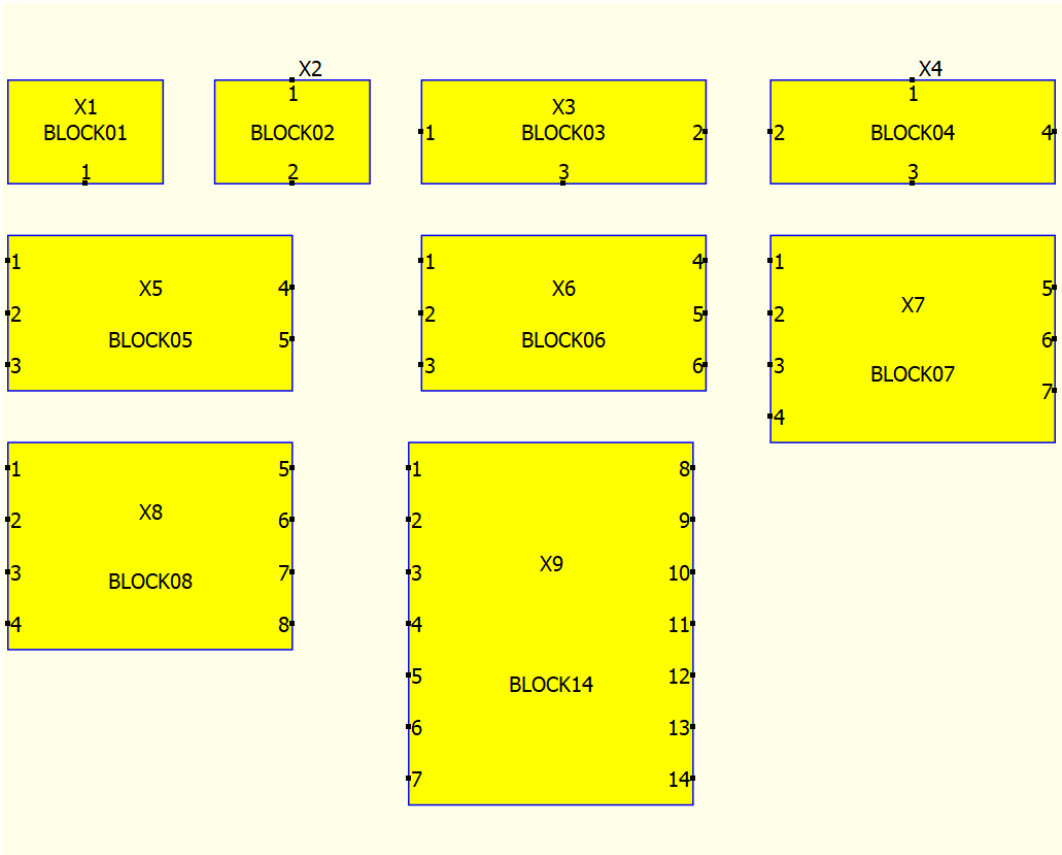
[4 : Method#2] Add input param into symbol

- Two input params : R=1K and Fcutoff=1K
- Type T or Right Click > Text, input
  - R=1K
  - Fcutoff=1K
- This will create 2nd and 3rd attribute in String Attributes, where you can select to visible or not in symbol



Autogenerate Symbol Pin Assignment

Qspice : Autogenerate Symbol Pin Assignment.qsch




```
.subckt Block01 1
.ends Block01

.subckt Block02 1 2
.ends Block02

.subckt Block03 1 2 3
.ends Block03

.subckt Block04 1 2 3 4
.ends Block04

.subckt Block05 1 2 3 4 5
.ends Block05

.subckt Block06 1 2 3 4 5 6
.ends Block06

.subckt Block07 1 2 3 4 5 6 7
.ends Block07

.subckt Block08 1 2 3 4 5 6 7 8
.ends Block08

.subckt Block14 1 2 3 4 5 6 7 8 9 10 11 12 13 14
.ends Block14
```

Part 2C
Symbol for Subckt
[Link to Library]

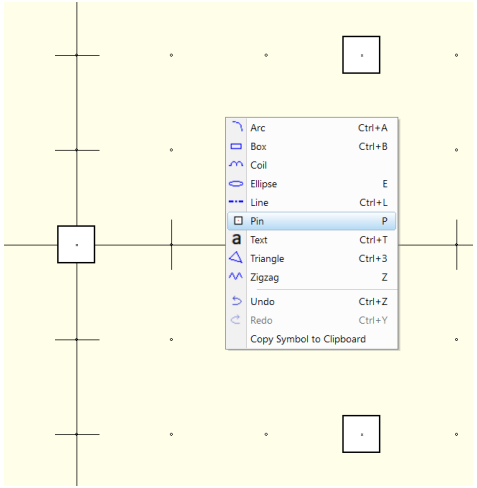
Symbol for Subckt [Create Symbol and Link to Library]

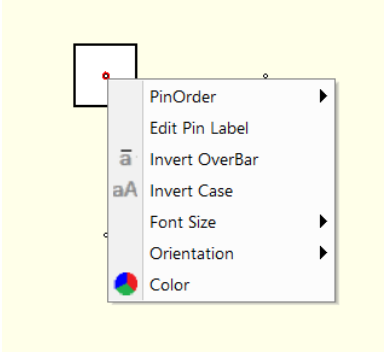
Example to create subckt symbol for irf530

[1] File → New → New Symbol

[2] Right Click → Pin (to add 3 pins with order D, G, S)

[3] Right Click at center of Pin to review PinOrder and PinLabel



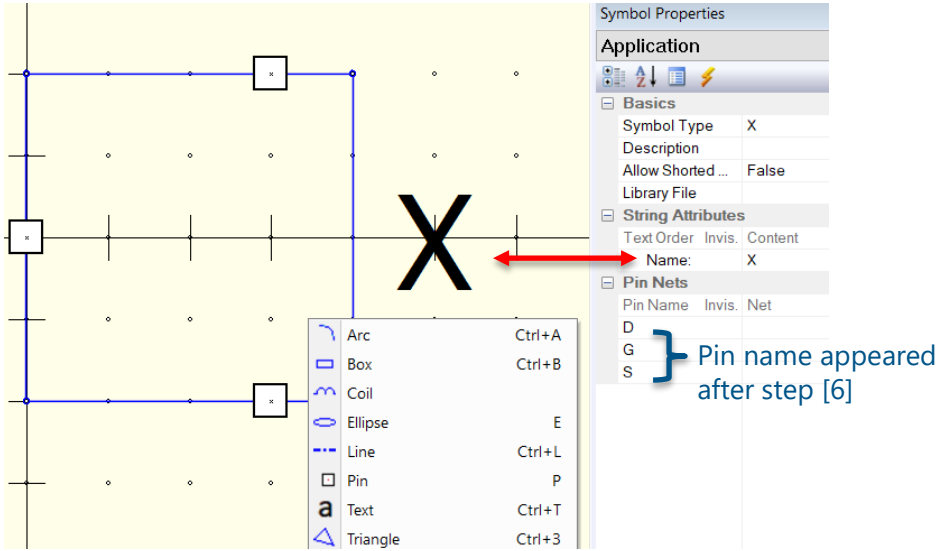


irf530.lib

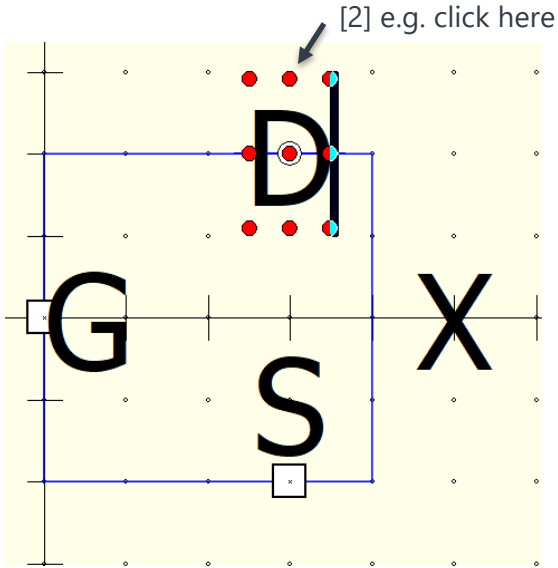
```
1  *Feb 16, 2010
2  *Doc. ID: 90181, Rev. A
3  *File Name: part irf530_PS.txt and part irf
4  *This document is intended as a SPICE model
5  *constitute a commercial product data sheet
6  *appropriate data sheet of the same number
7  *limits.
8  .SUBCKT irf530 1 2 3
9  ************************************
10 *      Model Generated by MODPEX     *
11 *Copyright(c) Symmetry Design Systems*
12 *         All Rights Reserved        *
13 *    UNPUBLISHED LICENSED SOFTWARE   *
14 *   Contains Proprietary Information *
15 *      Which is The Property of      *
16 *     SYMMETRY OR ITS LICENSORS      *
17 *Commercial Use or Resale Restricted *
18 *   by Symmetry License Agreement    *
19 ************************************
20 * Model generated on Apr 24, 96
21 * Model format: SPICE3
22 * Symmetry POWER MOS Model (Version 1.0)
23 * External Node Designations
24 * Node 1 -> Drain
25 * Node 2 -> Gate
26 * Node 3 -> Source
27 M1 9 7 8 8 MM L=100u W=100u
28 * Default values used in MM:
```

Symbol for Subckt [Create Symbol and Link to Library]

[4] Draw a box for outline
[5] Put an "X" in Symbol Type in Symbol Properties
[6] Right Click → Text → Put an "X"

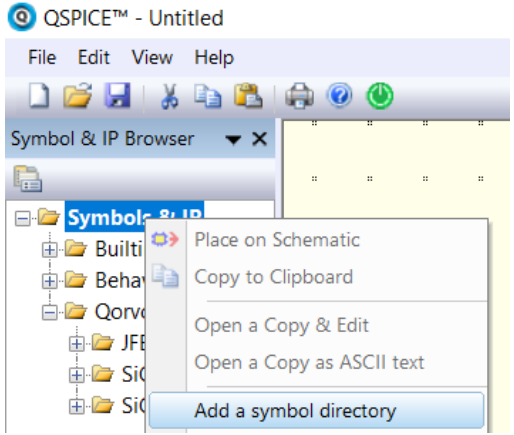


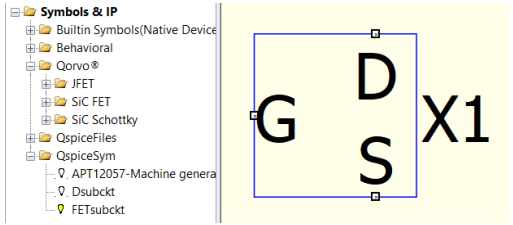
[7] To justify Pin label, double click center of Pin
[8] Click red dot other than its centered justification
[9] Save symbol file as .qsym



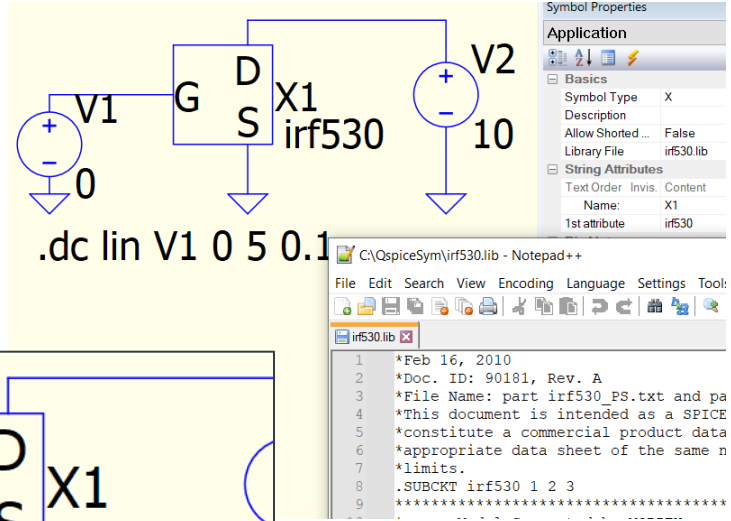
Symbol for Subckt [Create Symbol and Link to Library]

[10] In Schematic, Symbol & IP Browser, Right Click to "Add a symbol directory"
[11] Drag created component to schematic





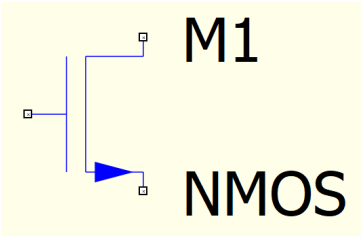
[12] Right Click on symbol, "Add New Attribute" as irf530
[13] In Symbol Properties, add "Library File" as irf530.lib
** library file is required to be put in schematic directory



Part 2D
Convert MOSFET M to subckt Symbol

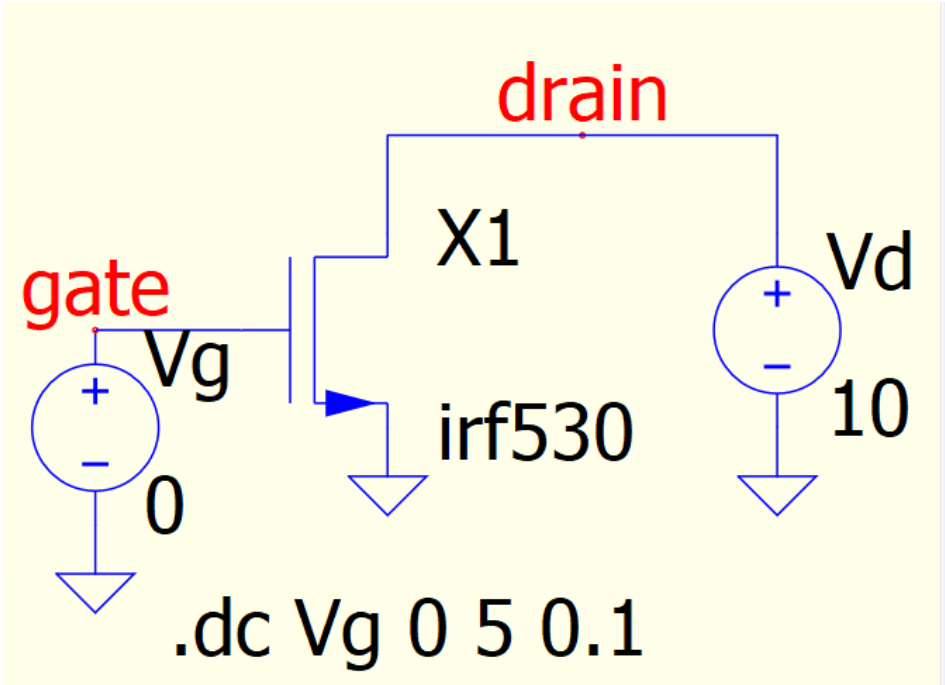
Convert MOSFET M to subckt Symbol

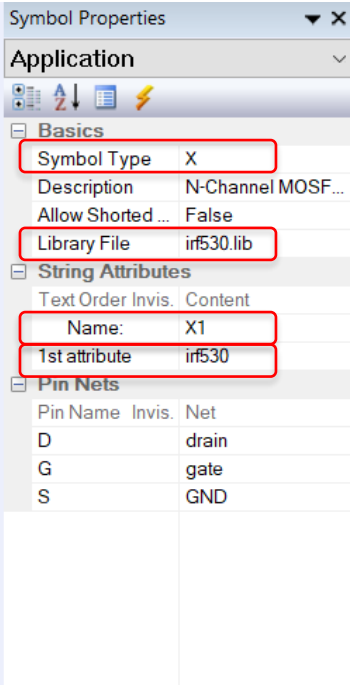
Qspice : Call Lib from M.qsch

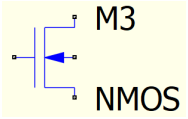


It is possible to convert MOSFET symbol M into a sub-circuit symbol by

1. Change Symbol Type from MN to X
2. Library File as subckt library file
3. 1st attribute as subckt name
4. [Optional] Change Symbol Name to X?



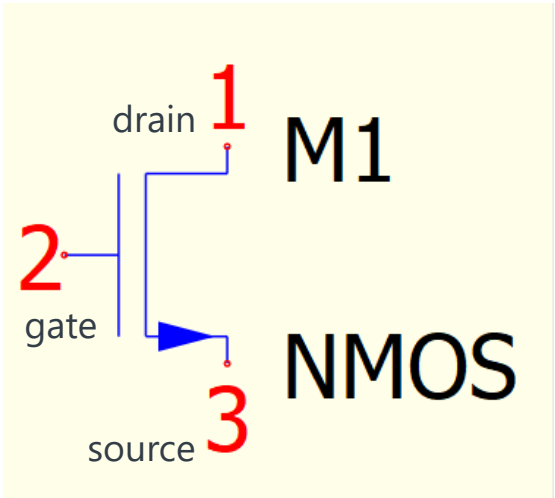




** this alternative symbol is 4 pins (+ base), which cannot support 3 pin subckt

Pin Order in Symbol MN and MP

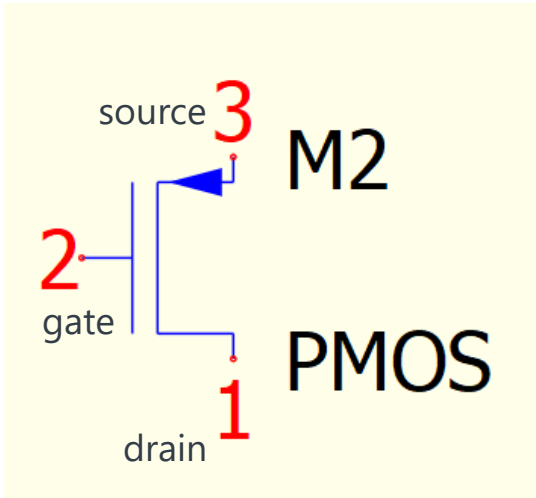
Pin Order for Symbol MN (NMOS)



Application

Basics	
Symbol Type	MN
Description	N-Channel MOSF...
Allow Shorted ...	False
Library File	NMOS.txt
String Attributes	
Text Order Invis.	Content
Name:	M1
1st attribute	NMOS
Pin Nets	
Pin Name Invis.	Net
D	1
G	2
S	3

Pin Order for Symbol MP (PMOS)



Application

Basics	
Symbol Type	MP
Description	P-Channel MOSF...
Allow Shorted ...	False
Library File	PMOS.txt
String Attributes	
Text Order Invis.	Content
Name:	M2
1st attribute	PMOS
Pin Nets	
Pin Name Invis.	Net
D	1
G	2
S	3

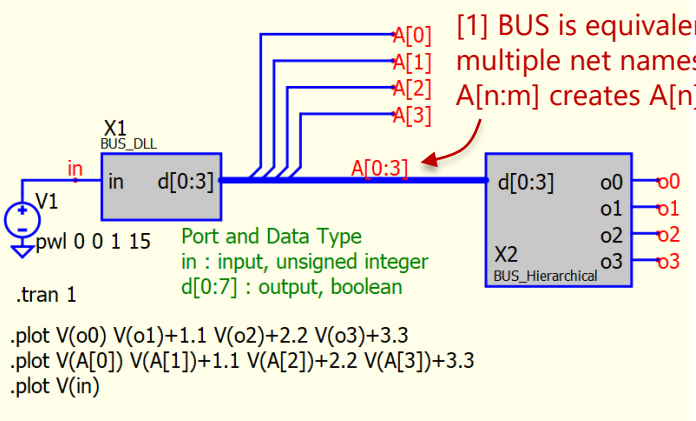
Part 2E
Bus and Hierarchical Block

Bus and Hierarchical Block

- Bus and Hierarchical Block
  - With Bus, data is defined as Data[n:m]
  - In Qspice, this net name format create a series of net names from Data[n] to Data[m]
    - If n<m, net name are Data[n], Data[n+1], Data[n+2], … , Data[m]
    - If n>m, net name are Data[n], Data[n-1], Data[n-2], … , Data[m]
  - For hierarchical block, subckt bus net names are assigned according to index sequence
  - To use data bus, it is recommending bus, hierarchical block and subckt with same data bus index, which can prevent unexpected behavior in net assignment

Bus and Hierarchical Block

Qspice : Parent-BUS.qsch / BUS_Hierarchical.qsch / bus_dll.cpp



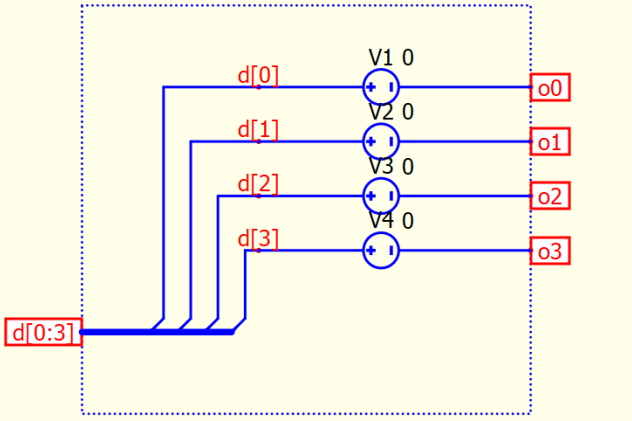
[1] BUS is equivalent to create multiple net names with []
A[n:m] creates A[n], A[n+1], ... A[m]

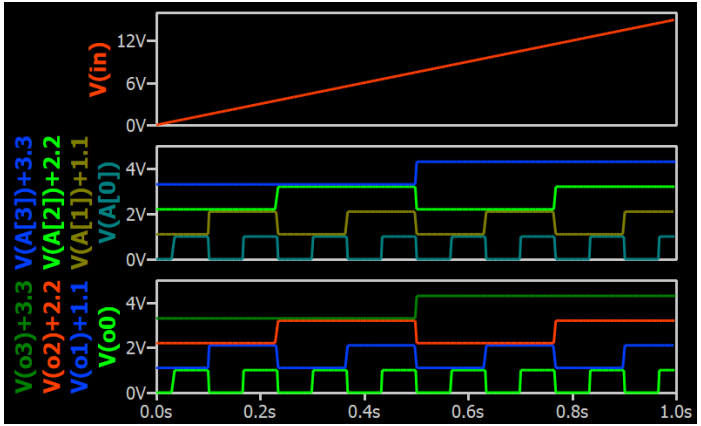
```
* C:\QspiceKSKelvin\01 User Guide and Script\01 Qspice Reference (
Ø†X1 «in´ui» «A[0]´b A[1]´b A[2]´b A[3]´b» «»  BUS_DLL
V1 in 0 pwl 0 0 1 15
X2 A[0] A[1] A[2] A[3] o0 o1 o2 o3 BUS_Hierarchical

.subckt BUS_Hierarchical d[0] d[1] d[2] d[3] o0 o1 o2 o3
V1 d[0] o0 0
V2 d[1] o1 0
V3 d[2] o2 0
V4 d[3] o3 0
.ends BUS_Hierarchical

.tran 1
.plot V(o0) V(o1)+1.1 V(o2)+2.2 V(o3)+3.3
.plot V(A[0]) V(A[1])+1.1 V(A[2])+2.2 V(A[3])+3.3
.plot V(in)
.end
```

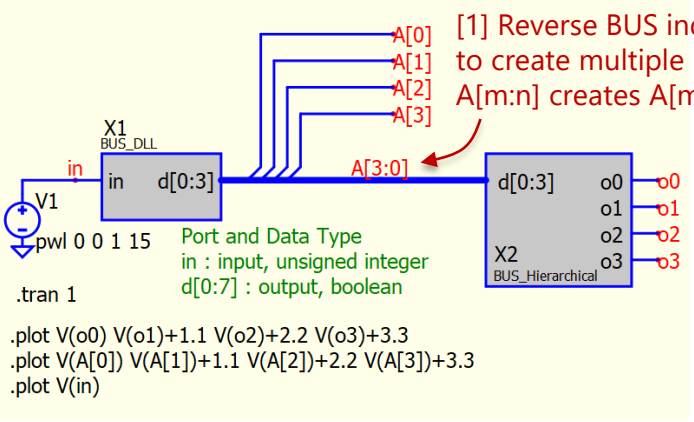
[2] Subckt X1, hierarchical X2 are all feed in same name and order





Bus and Hierarchical Block : Change BUS name order

Qspice : Parent-BUS.qsch / BUS_Hierarchical.qsch / bus_dll.cpp



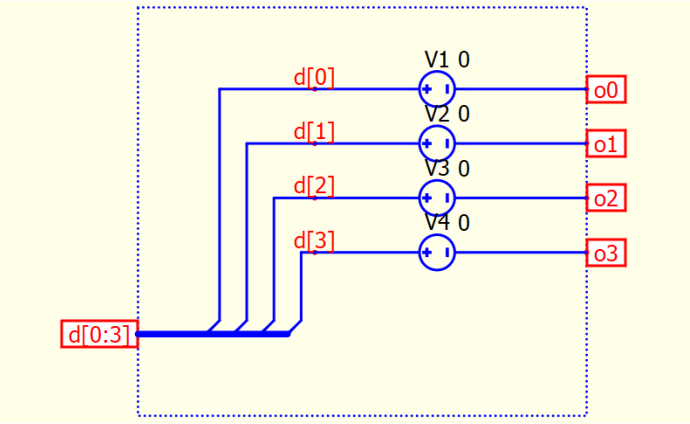
[1] Reverse BUS index is equivalent to create multiple net names with [] A[m:n] creates A[m], A[m-1], ... A[n]

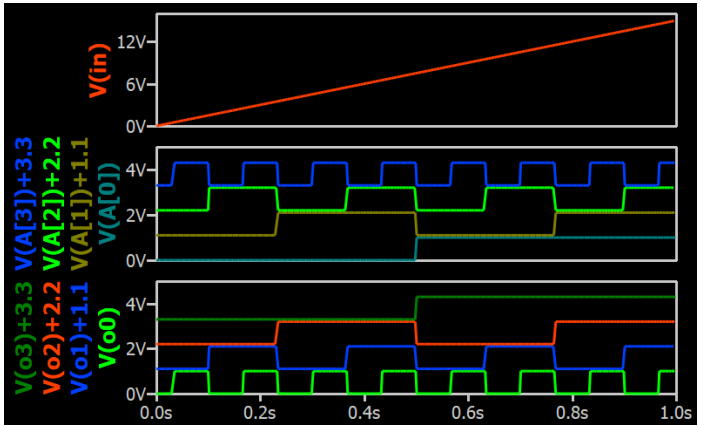
```
* C:\QspiceKSKelvin\01 User Guide and Script\01 Qspice Reference (
ØtX1 «in´ui» «A[3]´b A[2]´b A[1]´b A[0]´b» «» BUS_DLL
V1 in 0 pwl 0 0 1 15
X2 A[3] A[2] A[1] A[0] o0 o1 o2 o3 BUS_Hierarchical

.subckt BUS_Hierarchical d[0] d[1] d[2] d[3] o0 o1 o2 o3
V1 d[0] o0 0
V2 d[1] o1 0
V3 d[2] o2 0
V4 d[3] o3 0
.ends BUS_Hierarchical

.tran 1
.plot V(o0) V(o1)+1.1 V(o2)+2.2 V(o3)+3.3
.plot V(A[0]) V(A[1])+1.1 V(A[2])+2.2 V(A[3])+3.3
.plot V(in)
.end
```

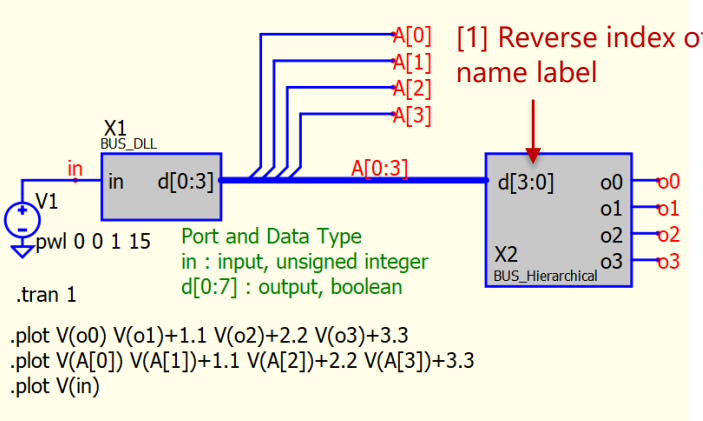
[2] Subckt X1-d[0] is connected to A[3]
[3] This order is feed into hierarchical block X2, e.g. A[3] is feed into X2-d[0]





Bus and Hierarchical Block : Change Hierarchical net label order

Qspice : Parent-BUS.qsch / BUS_Hierarchical.qsch / bus_dll.cpp

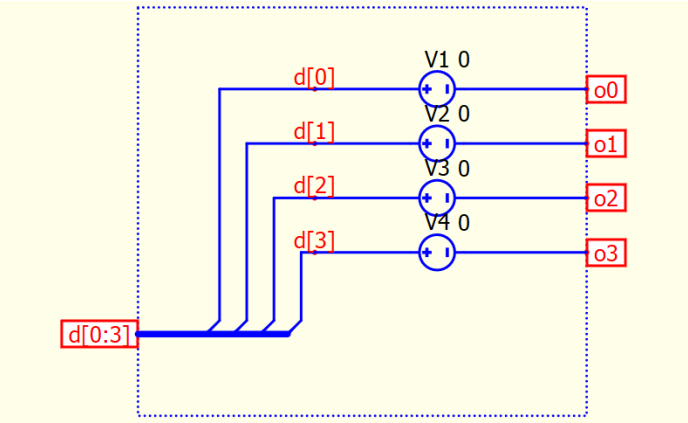


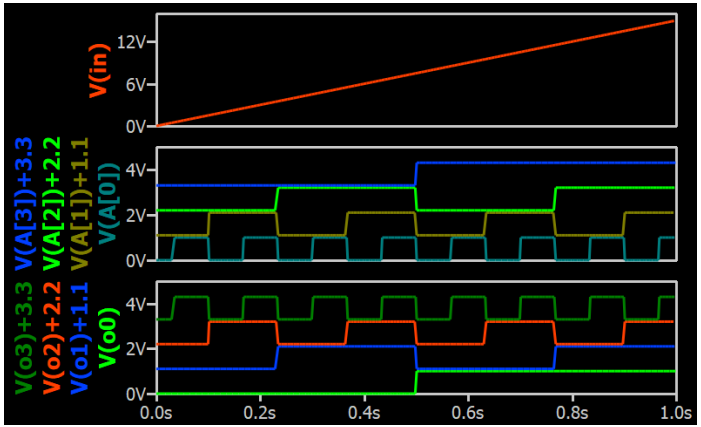

```
* C:\QspiceKSKelvin\01 User Guide and Script\01 Qspice Reference (
Ø†X1 «in´ui» «A[0]´b A[1]´b A[2]´b A[3]´b» «»  BUS_DLL
V1 in 0 pwl 0 0 1 15
X2 A[0] A[1] A[2] A[3] o0 o1 o2 o3 BUS_Hierarchical

.subckt BUS_Hierarchical d[3] d[2] d[1] d[0] o0 o1 o2 o3
V1 d[0] o0 0
V2 d[1] o1 0
V3 d[2] o2 0
V4 d[3] o3 0
.ends BUS_Hierarchical

.tran 1
.plot V(o0) V(o1)+1.1 V(o2)+2.2 V(o3)+3.3
.plot V(A[0]) V(A[1])+1.1 V(A[2])+2.2 V(A[3])+3.3
.plot V(in)
.end
```

[2] Hierarchical block X2 name is reversed, but Hierarchical / Subckt net assignment is based on order, therefore, A[0] is feed to hierarachical subckt d[3] in this case





Waveform Viewer

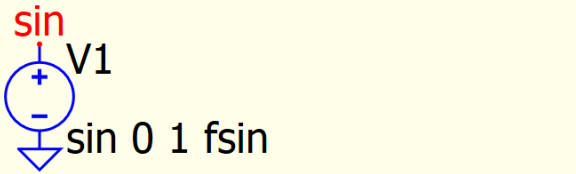
Waveform Viewer Plot Config File (*.pfg) and .plot directive

- Waveform Viewer Config File (*.pfg) and .plot
  - In waveform viewer, plot config can be saved with File > Save Config : [qschname].pfg
    - This config file save windows, traces and axis setting
    - Press spacebar in waveform viewer can re-load config file [qschname].pfg
  - Two unique feature [qschname].pfg can provide but not support by .plot
    - Pre-define x-axis Quantity
    - Pre-define x and y-axis range

Waveform Viewer	Plot Config File [1] [qschname].pfg	.plot command in schematic	Outcome
Closed before Simulation	No	No	A blank waveform viewer
	No	Yes	Plot according to .plot command
	Yes	[ignore]	Plot according to [qschname].pfg config
Opened before Simulation	[ignore]	[ignore]	Keep windows and traces setting from last plot, reset x and y-axis
[1] Save plot config in Waveform Viewer : File > Save Config			
** In Waveform Viewer, press Spacebar to reload [qschname].pfg plot config file			
** to use .plot, delete [qschname].pfg and close waveform viewer before run of simulation			

Data Export in Waveform Viewer – with @ in expression for .step

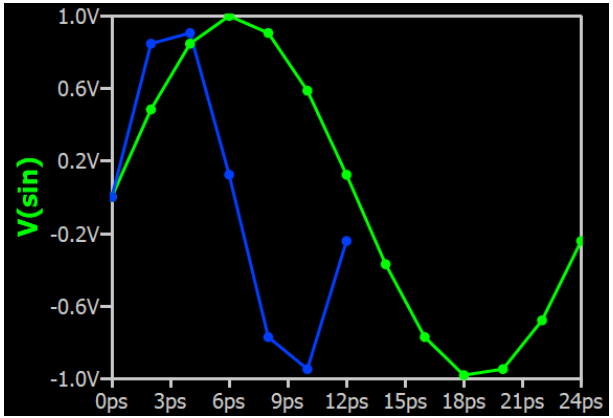
Qspice : waveform - with @ for step.qsch




```
.step param fsin list 40G 80G
.tran 1/fsin
.plot V(sin)
```

- Data Export
- Setup Data Export
  - File > Export Data
  - Number Points : All
  - Expression(s) : V(sin),FSIN

- Data Export with @
- Setup Data Export
  - File > Export Data
  - Number Points : All
  - Expression(s) : V(sin)@1,V(sin)@2



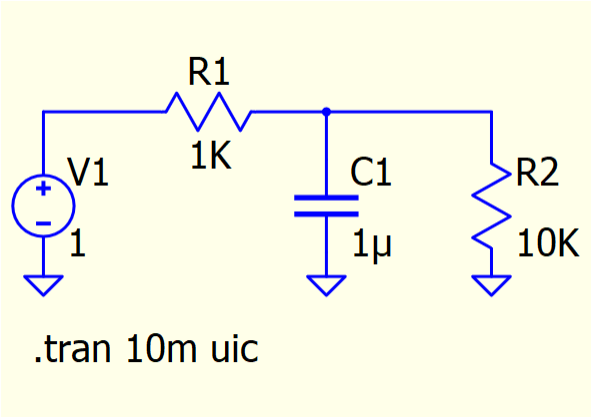

```
Time,V(sin),FSIN
0,0,40000000000
2.001953125e-12,0.482183772079123,40000000000
4.00390624999999e-12,0.844853565249706,40000000000
6.00585937500001e-12,0.998118112900149,40000000000
8.00781250000003e-12,0.903989293123441,40000000000
1.0009765625e-11,0.58579785745643,40000000000
1.20117187500001e-11,0.122410675199201,40000000000
1.40136718750001e-11,-0.371317193951856,40000000000
1.6015625e-11,-0.773010453362737,40000000000
1.80175781249999e-11,-0.983105487431211,40000000000
2.00195312499998e-11,-0.949528180593055,40000000000
2.20214843749997e-11,-0.680600997795516,40000000000
2.40234374999995e-11,-0.242980179903377,40000000000
0,0,80000000000
2.001953125e-12,0.844853565249706,80000000000
4.00390625000001e-12,0.903989293123441,80000000000
6.00585937500003e-12,0.122410675199201,80000000000
8.0078125e-12,-0.773010453362736,80000000000
1.00097656249999e-11,-0.949528180593055,80000000000
1.20117187499998e-11,-0.242980179903377,80000000000
```

```
Time,V(sin)@1,V(sin)@2
0,0,0
2.001953125e-12,0.482183772079123,0.844853565249706
4.00390624999999e-12,0.844853565249706,0.903989293123441
6.00585937500001e-12,0.998118112900149,0.122410675199201
8.00781250000003e-12,0.903989293123441,-0.773010453362739
1.0009765625e-11,0.58579785745643,-0.949528180593
1.20117187500001e-11,0.122410675199201,-0.596254180248215
1.40136718750001e-11,-0.371317193951856,-0.596254180248215
1.6015625e-11,-0.773010453362737,-0.596254180248215
1.80175781249999e-11,-0.983105487431211,-0.596254180248215
2.00195312499998e-11,-0.949528180593055,-0.596254180248215
2.20214843749997e-11,-0.680600997795516,-0.596254180248215
2.40234374999995e-11,-0.242980179903377,-0.596254180248215
0,0,0
2.001953125e-12,0.482183772079122,0.844853565249706
4.00390625000001e-12,0.844853565249707,0.903989293123441
6.00585937500003e-12,0.998118112900148,0.122410675199201
8.0078125e-12,0.903989293123442,-0.773010453362736
1.00097656249999e-11,0.585797857456455,-0.949528180593055
1.20117187499998e-11,0.122410675199269,-0.242980179903377
```

Snapshot Data Method – Export Data with Single Number Points

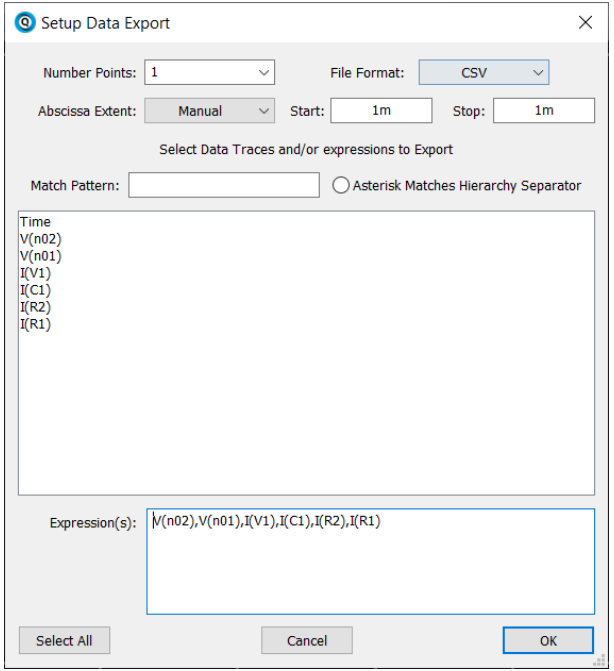
Qspice : waveform - time snapshot.qsch

- Snapshot Data
  - To create a snapshot dataset (e.g. all calculated results at particular time)
  - This example demonstrate a snapshot data in csv format with export data method
  - Idea is to force number points in data export to 1
    - Output two row but both are identical if start and stop are same
    - If start and stop are not same, output two row with time=start and time=stop



In waveform viewer

1. File > Export Data
2. Change Number Points to 1
3. File Format : CSV
4. Abscissa Extent : 1, Start = Stop
5. Select All

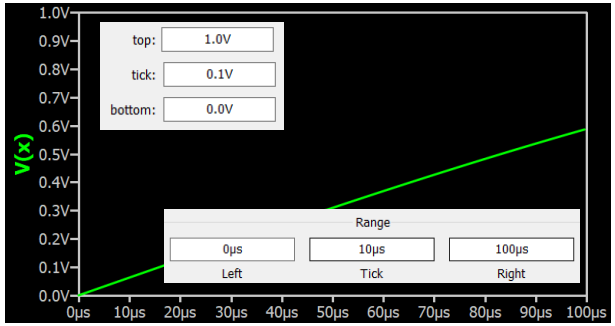


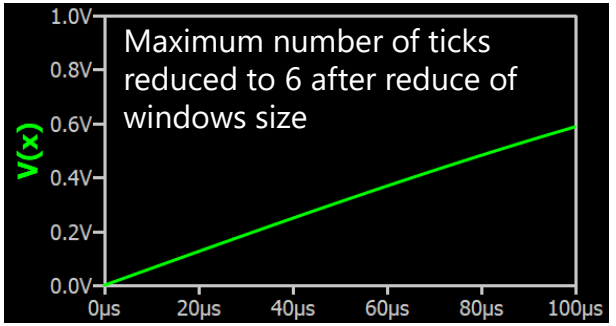
Result in exported csv

Time	V(n02)	V(n01)	I(V1)	I(C1)	I(R2)	I(R1)
0.001	1	0.606398	-0.00039	0.000333	6.06E-05	0.000394
0.001	1	0.606398	-0.00039	0.000333	6.06E-05	0.000394

Waveform Viewer – Minimum Tick

- Minimum Tick
  - Maximum number of ticks in x- and y-axis are 11
  - Depends on windows size, maximum number of ticks can reduce to 6
  - Therefore, minimum allowable tick is $\frac{\text{Right}-\text{Left}}{10}$ or $\frac{\text{top}-\text{bottom}}{10}$

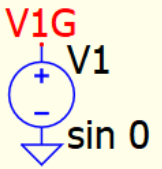




Simulation Technique

Max Time Step in .tran (and .bode) : Two methods

Qspice : MaxTimeStep.qsch



sin 0 1 1G .plot V(V1G)

Method 1 : Traditional Berkeley Syntax

.tran IGNORED TSTOP [TSTART [MAXSTEP]] [UIC]

.tran 0 10n 0 10p

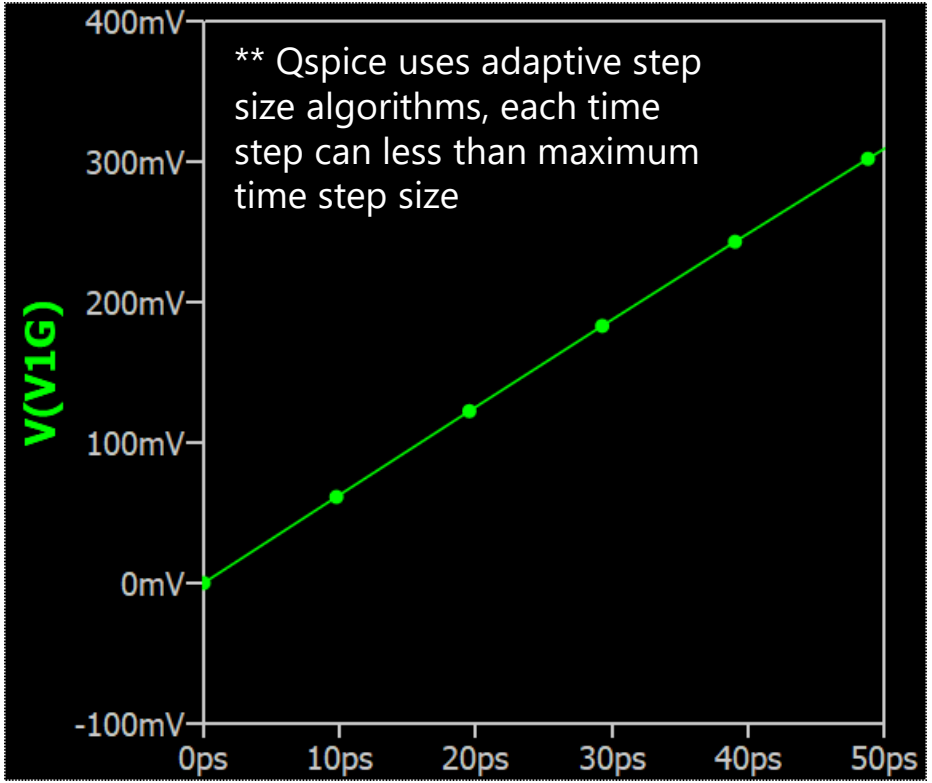
↑ Max Time Step

Method 2 : MAXSTEP in Simulator Option

.tran 10n

.options MAXSTEP=10p

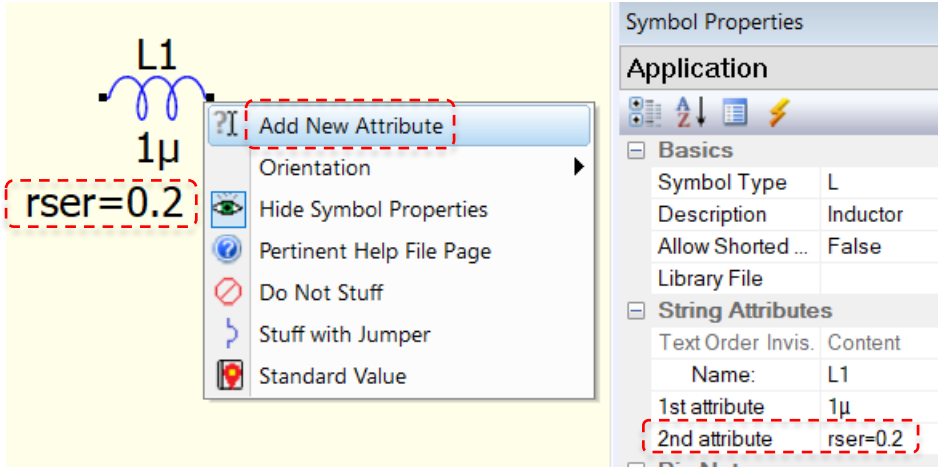
↑ Max Time Step
(this method apply to .tran and .bode)



Add [additional instance parameters]

1. Right Click on Component
2. Select “Add New Attribute”
3. Type parameter name and value [refer to help for full list of instance parameters]

This is an example to assign 0.2 ohms series resistance to inductor L1



Inductor Instance Parameters

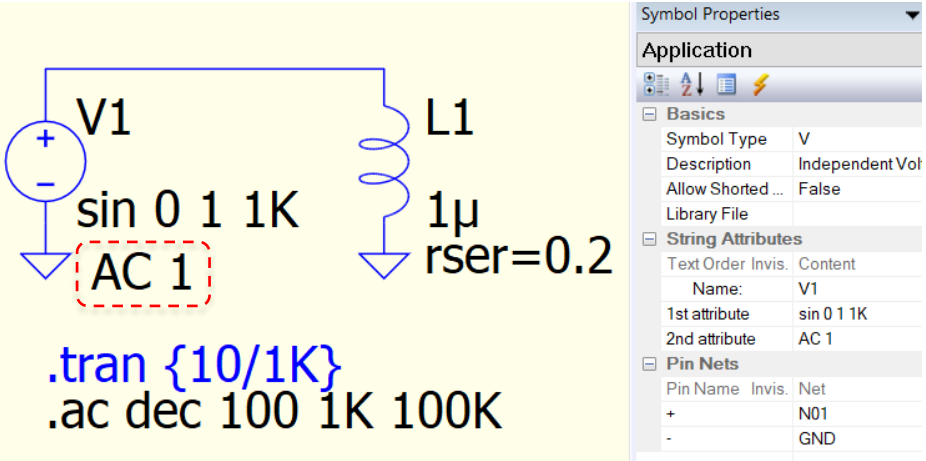
Name	Description	Units	Default
AG	Wire or stripline is made of gold		
AL	Wire or stripline is made of aluminum		see below
AU	Wire or stripline is made of silver		
BEND	Fractional inductance correction for wire bend or proximity effects		1.
CPAR	Parallel capacitance	F	0.
CU	Wire or stripline is made of copper		see below
DIAMETER	Diameter of wire or air coil	m	
FREQUENCY	Frequency at Q. Also used to compute Rser due to skin effect		
HEIGHT	Height of PCB stripline above ground plane	m	
IC	Initial current if uic is specified on .tran statement	A	none
INDUCTANCE	Inductance of inductor	H	0.0
ISAT	Current causing inductance to drop to SATFRAC×INDUCTANCE	A	Infinite
LENGTH	Length of wire, stripline, or air coil	m	
LSAT	Inductance asymptotically approached in saturation	H	10% of INDUCTANCE
M	Number of parallel inductors		1.0
NI	Wire is made of nickel		
Q	Quality factor at FREQUENCY		
RPAR	Equivalent parallel resistance	Ω	Infinite
RSER	Equivalent series resistance	Ω	0.0
SATFRAC	Fractional drop in inductance at ISAT		0.7
THICK	Thickness of stripline on top of a PCB	m	0.0
TURNS	Number of turns of an air coil		
VERBOSE	Print wire L, Rser, Rpar results on the console		(not set)
WIDTH	Width of stripline on top of a PCB	m	

AC and DC Attribute in Source

Qspice : AC with Transient Source.qsch ; AC with Bias.qsch

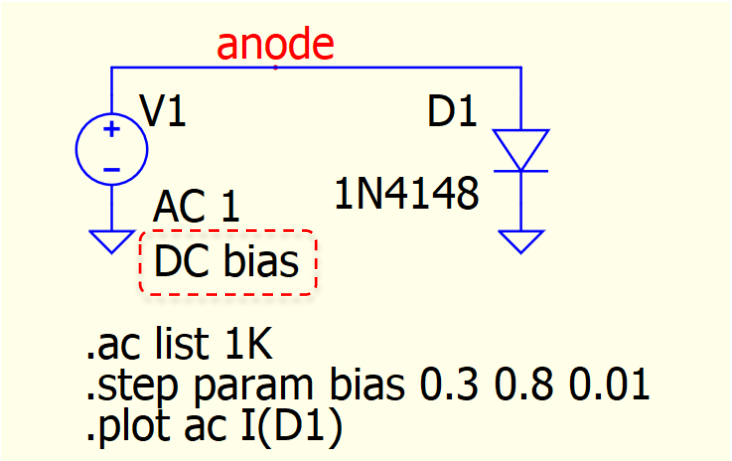
Technique to perform AC analysis with a transient source

1. Right Click on Voltage/Current source
2. Select “Add New Attribute”
3. Type “AC 1” to define a 1V source for AC sweep
4. Add a .ac analysis statement, and comment transient analysis



Technique to perform AC analysis with DC in source

1. Right Click on Voltage/Current Sourec, Add New Attribute
2. To add DC source, type “DC …”
   - If without DC, simulator may not interpret the DC voltage during simulation. Best practice is to add DC



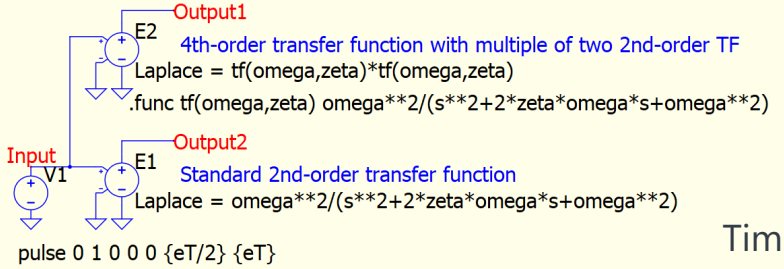
Laplace Time and Frequency Domain Simulation

Qspice : Laplace Simulation - Fdomain.qsch ; Laplace Simulation - Tdomain.qsch

2nd-order system step response
transfer function : omega^2/(s^2+2*zeta*omega+omega^2)

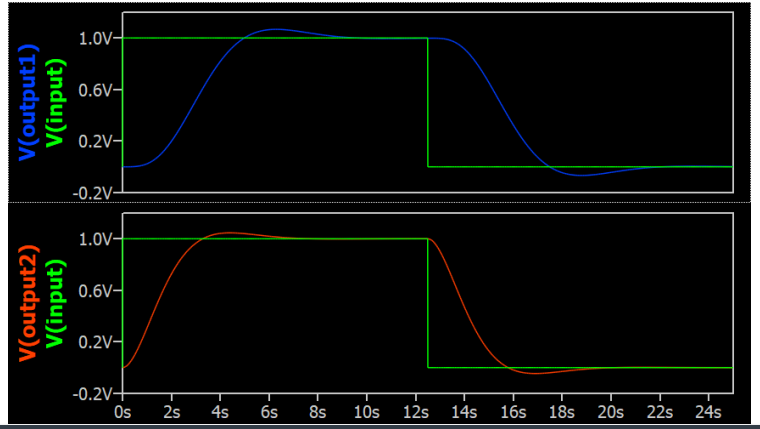
.param zeta = 0.7
.param omega = 1

.tran {eT}
.param eT=25

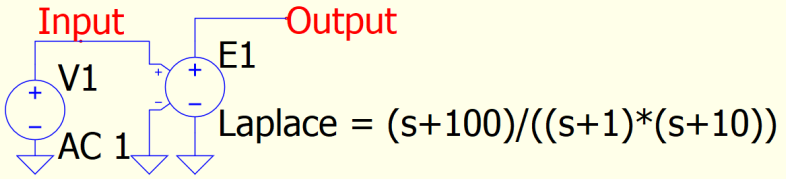


pulse 0 1 0 0 0 {eT/2} {eT}

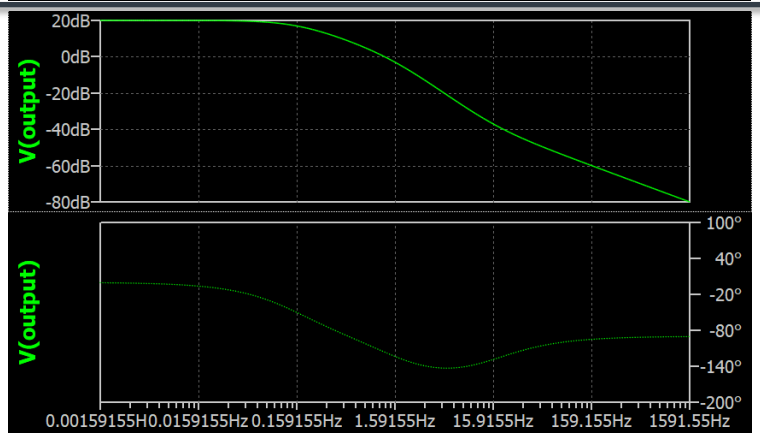
Time Domain



Frequency Domain



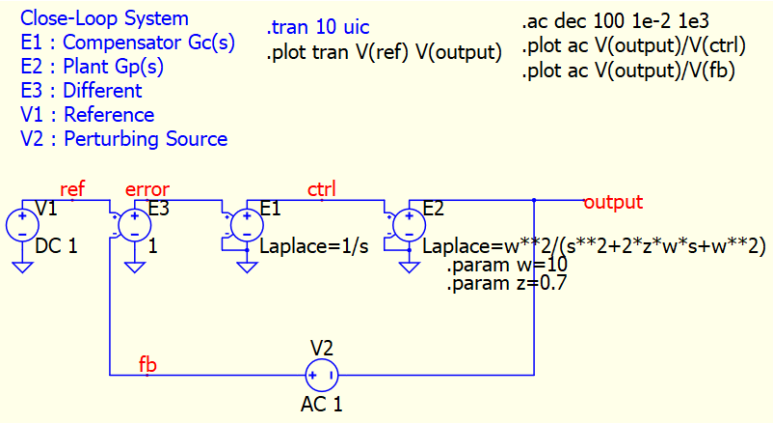
.ac dec 100 {1e-2/2/pi} {1e4/2/pi}

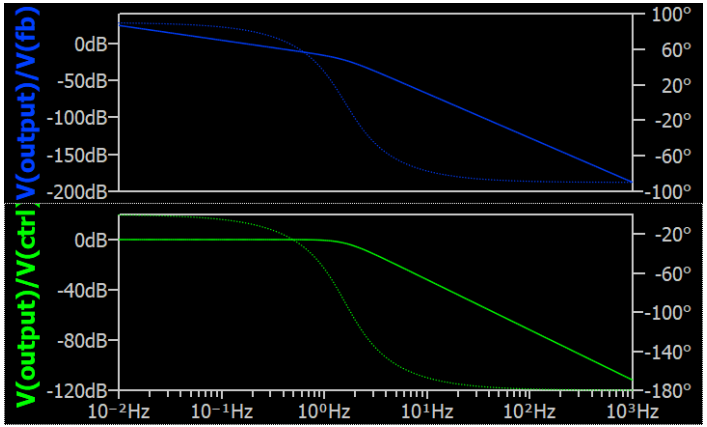


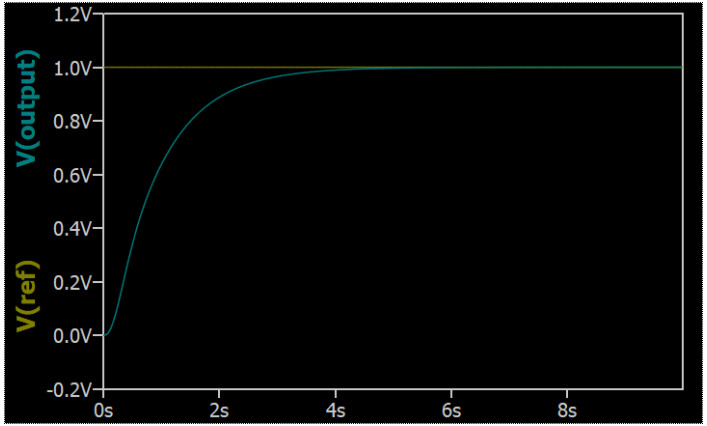
Laplace Time and Frequency Domain Simulation

Qspice : Laplace Close Loop.qsch

- Close Loop System Time and Bode
  - A technique to get Gp(s) and GH(s) is to add a perturbing source between output and feedback and perform ac analysis
  - In this example, Laplace function can collect in series for both .tran and .ac directive

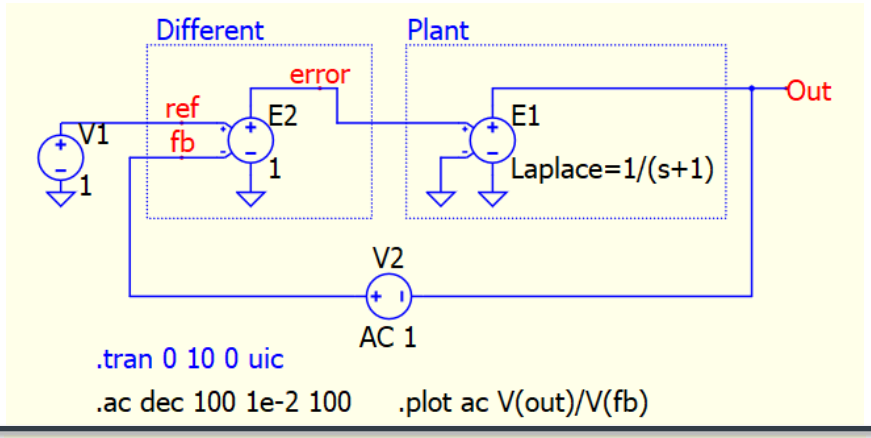


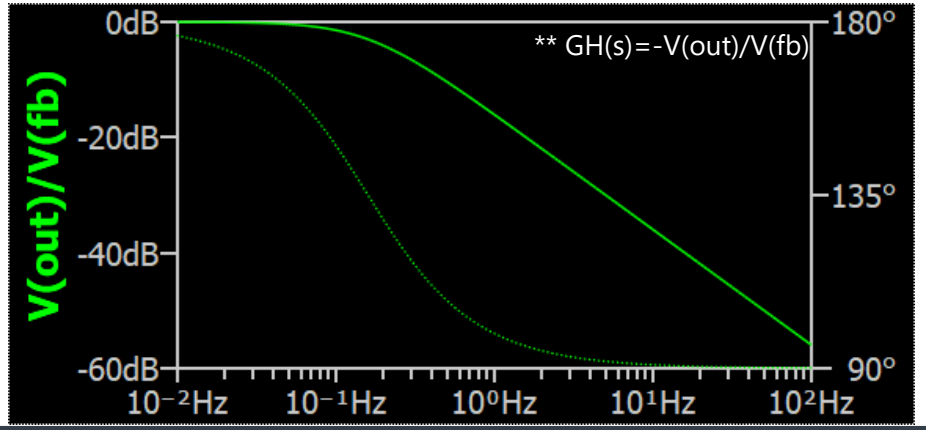


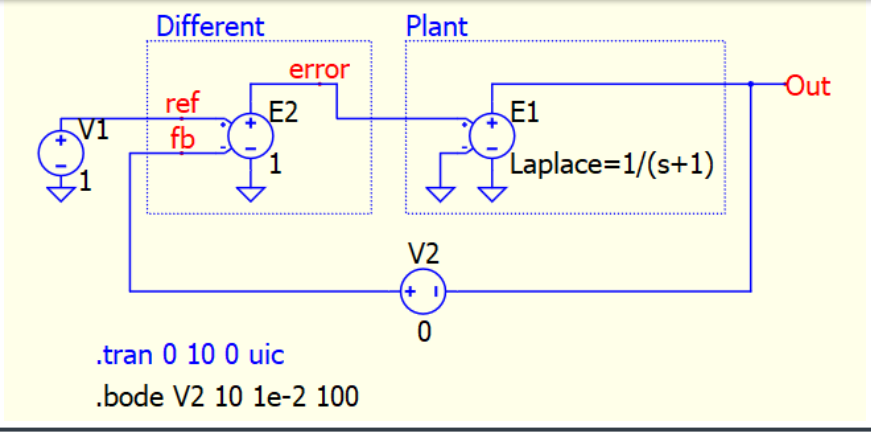


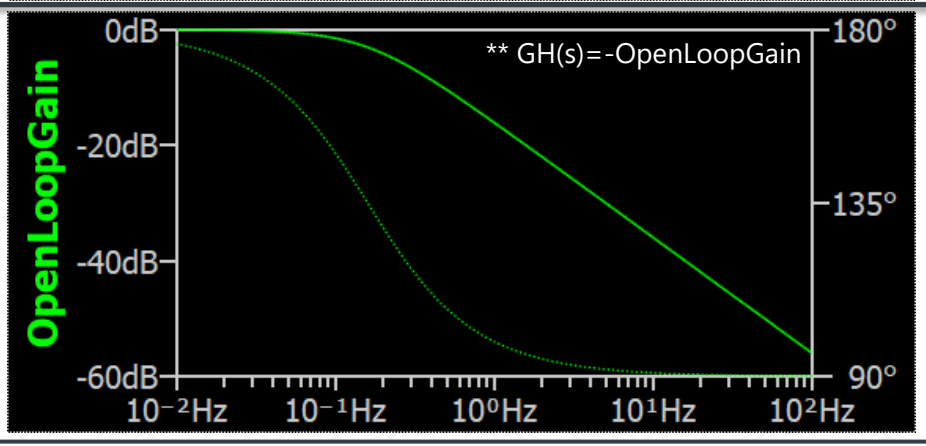
AC (.ac) and Frequency Response Analysis (.bode)

Qspice : ACmethod.qsch ; BODEmethod.qsch



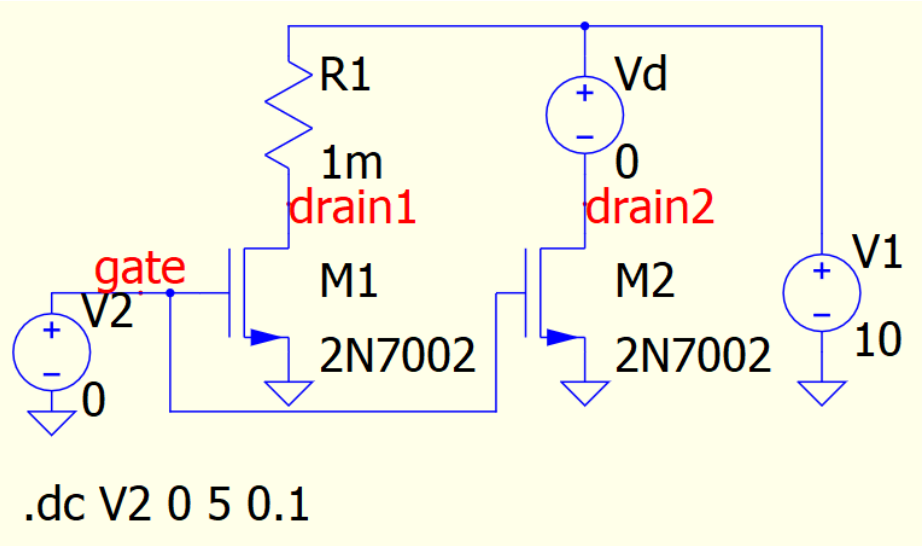


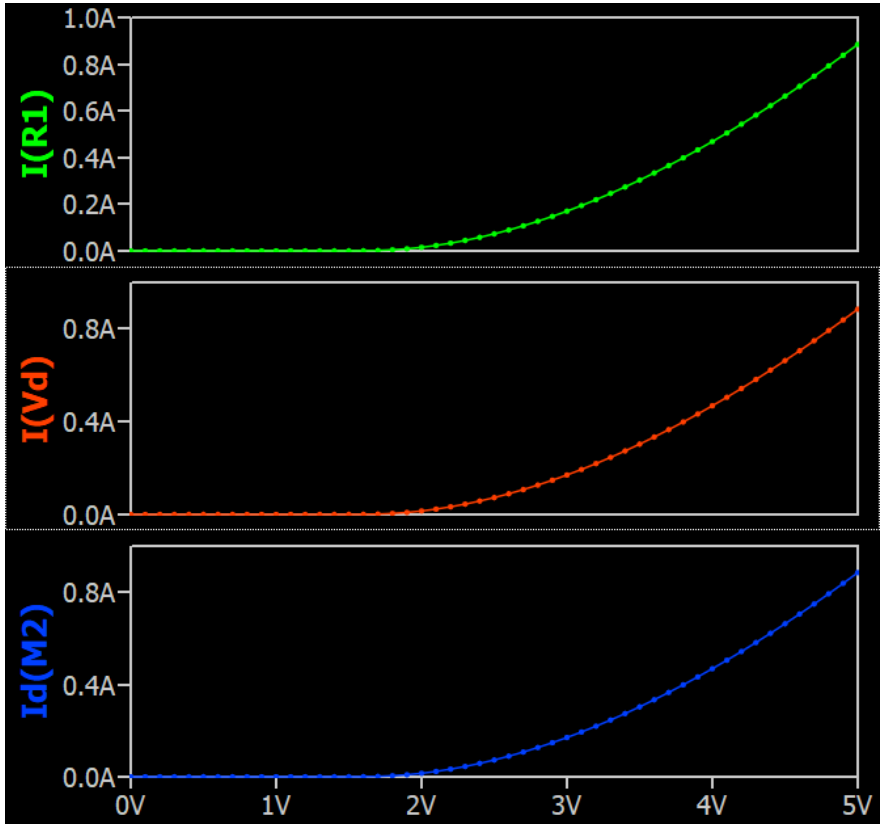




Technique to Probe Drain Current / General Current Probe

Qspice : Current_Probe_Method.qsch





3 common current probe method

1. Add a series resistor and probe R current
2. Add 0V voltage source and probe current of this voltage source. +ve represent current flow from + to – direction within symbol (i.e. current flow downward in above example)
3. Ctrl-A (Add Plot) in waveform viewer and select Id(Mnnn)

Selection Guide option for Circuit Elements with 3rd Party Library

- Purpose
  - Use Q transistor as an example of how to have selection guide from 3rd party library
- Procedure
  - In C:\Program Files\Qspice, create a .txt file
    - e.g. My_NPN.txt
    - May require admin access
  - Copy and paste .model context into .txt and save
    - https://ltwiki.org/index.php?title=Standard.bjt
    - This link contains a list of BJT model
  - In Qspice schematic, add a NPN transistor with shortcut Q
  - Right click transistor, open symbol properties and change the library file from NPN.txt to My_NPN.txt
  - Right click transistor and Selection Guide is available now
- Reference
  - https://forum.qorvo.com/t/adding-model-files-to-qspice/14963/7

stevenbennett 5h

That's very helpful thanks. For anyone wanting more detail, this is what worked for me:

1: Create a custom .model containing text file in C:\Program Files\QSPICE e.g. My_NPN.txt

2: Paste in single, or multiple, .model statements e.g. from Standard.bjt - LTwiki-Wiki for LTspice 1 and save.

3: Add an NPN transistor from the "Q" folder in the Symbols & IP folder list in QSPICE.

4: Open the symbol properties for the NPN transistor by double clicking and change the Library File from NPN.txt to My_NPN.txt

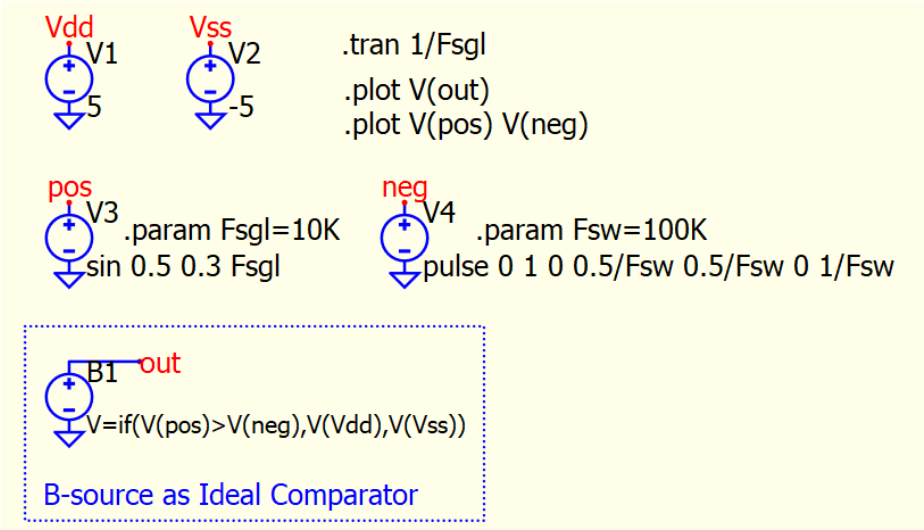
5: Right click the NPN symbol and choose Selection Guide, which will now display all the added models.

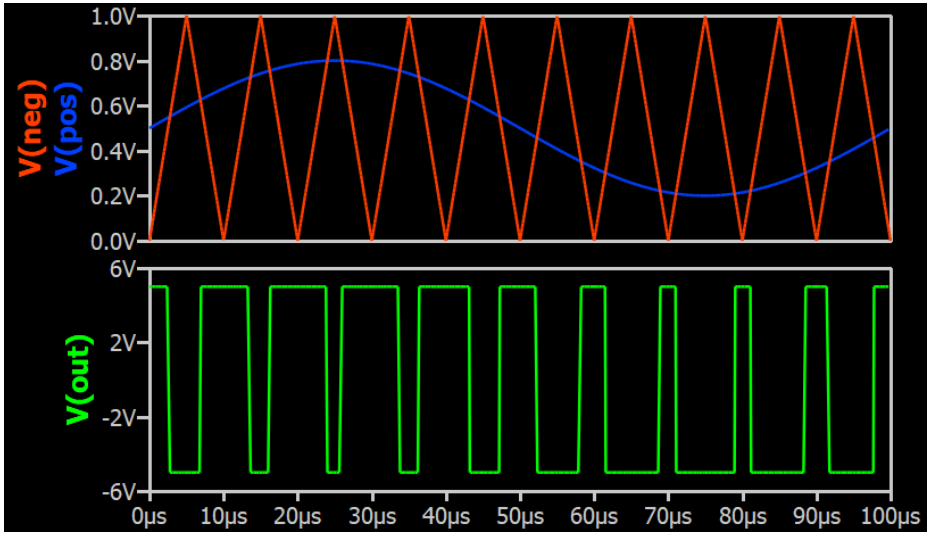
6: The file My_NPN.txt will survive any of the frequent QSPICE updates.

B-Source as Comparator

Qspice : B-Source as Comparator.qsch

- Concept of Ideal Comparator with Behavioral Voltage Source
  - Formula of B-source is : if(V(pos)>V(neg),V(Vdd),V(Vss))
  - Practical comparator output normally is open-drain configuration, this is just for simulation purpose

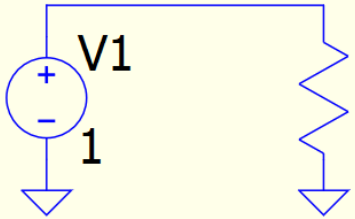




Monte Carlo

Qspice : Monte Carlo.qsch

Random number from 0 to 1 depending on the seed



1K*(1+.1*{random()})

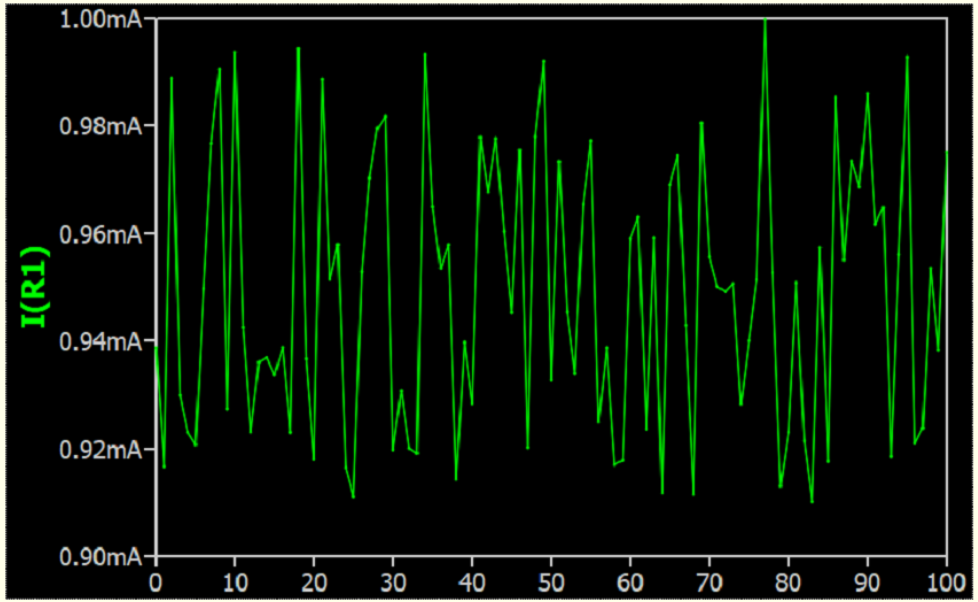
This example .1*{random()}
equivalent +0% to +10% change

```
.op
.step param dummy 0 100 1
.plot I(R1)

.options seedclock
.options seed=5
```

This enable random seed to be generated

This assign manual seed



Engelhardt

9-4-2023

OK, I just implemented

.options seedclock

It convolutes a 10MHz system clock with the simulation process ID to generate a physically random integer to seed the Mersenne Twister.

Flat(x) and MC(x,y) functions equivalent to Ltspice

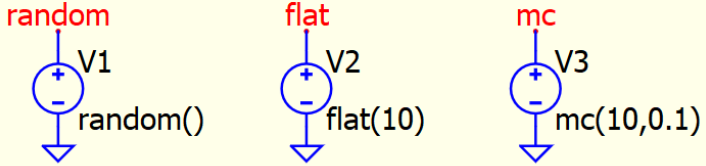
Qspice : Flat and MC Function.qsch

- Uniform random distribution
  - LTspice offers flat(x) and mc(x,y) functions, but not in Qspice (last check 10-3-2023)

- Function for flat(x) and mc(x,y)
  - .func flat(x) x*((random()*2)-1) ← Generate random [-x, x]
  - .func mc(x,y) x*(1+y*(random()*2-1)) ← Generate random [x*(1-y), x*(1+y)]

```
.step param x 1 200 1    Dummy For Loop
.op
```

```
.plot V(mc)
.plot V(flat)
.plot V(random)
```

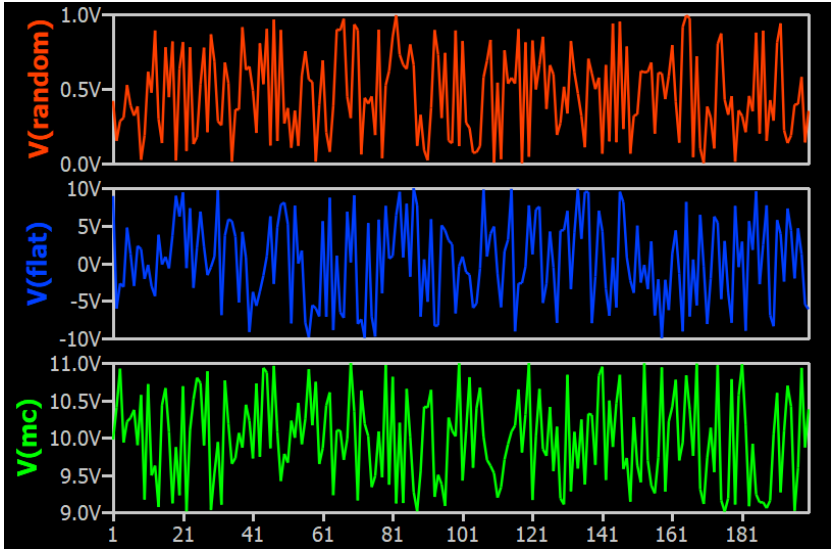


flat(x) : Random number between -x and x with uniform distribution

```
.func flat(x) x*((random()*2)-1)
```

mc(x,y) : A random number between
x*(1+y) and x*(1-y) with uniform distribution

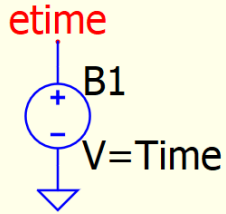
```
.func mc(x,y) x*(1+y*(random()*2-1))
```



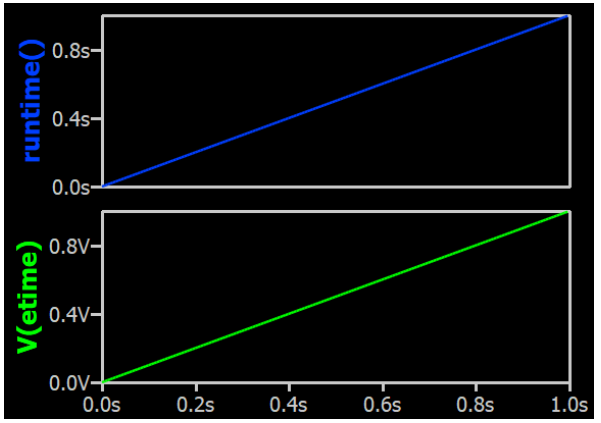
Time in .tran and Logic Diagram in Waveform Viewer with .plot

Qspice : Time in .tran.qsch ; Logic Signal Plot.qsch

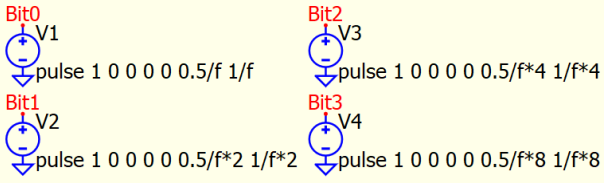
- Time in .tran
  - In .tran, simulation time is stored as a parameter named **Time**
  - Therefore, use a B-source can convert Time into a voltage
  - Time can also be used in function



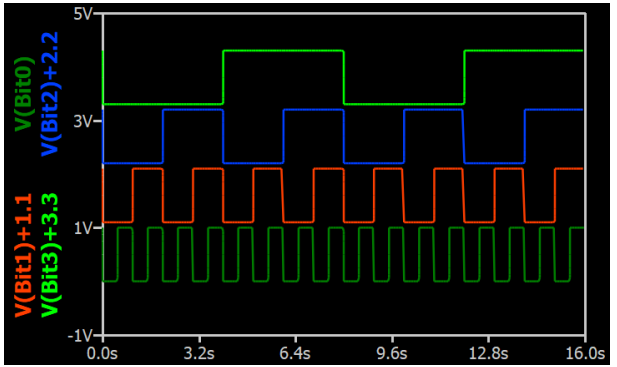

```
.tran 1    .func runtime() Time
.plot V(etime)
.plot runtime()
```



- Logic Diagram
  - A simple idea to plot logic signal into logic diagram format
  - Idea is to add an offset for each logic in .plot



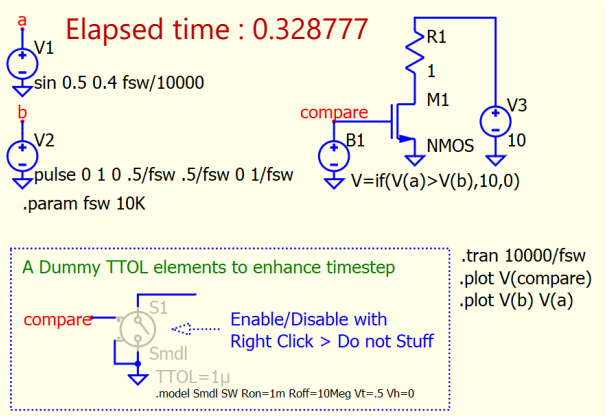

```
.param f=1
.tran 1/f*16
.plot V(Bit3)+3.3 V(Bit2)+2.2 V(Bit1)+1.1 V(Bit0)
```

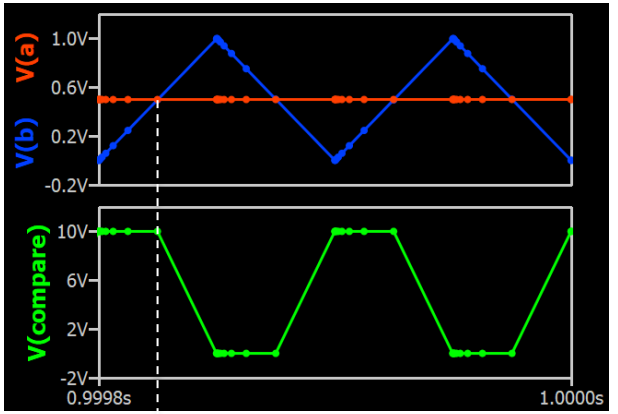


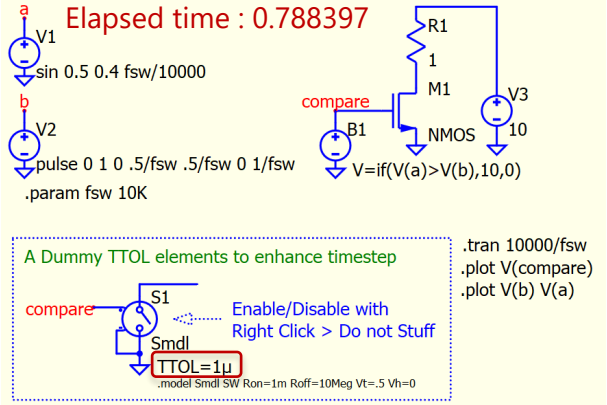
Dummy TTOL device to help in adaptive timestep

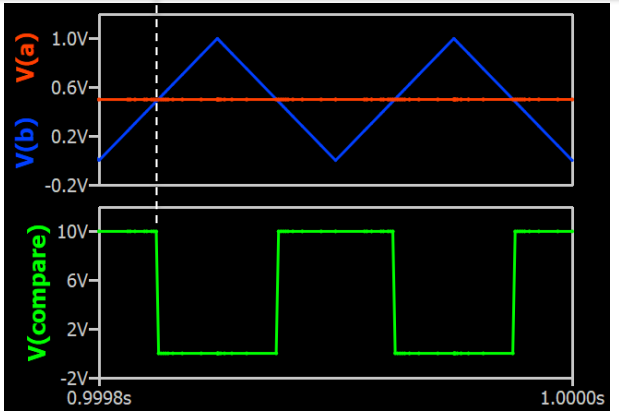
Qspice : TTOL - Dummy TTOL element - Enhance Timestep.qsch

- Dummy TTOL device
  - Qspice uses adaptive timestep
  - If a circuit uses a B-source, if(x,y,z) as a comparator, without TTOL device, its simulation timestep can far from compare instance and output looks weird
    - Example on Top Row
    - Precise time instance at compare action, but as no extra timestep after compare action, output looks like ramping as next timestep is far away (interpolation)
  - To resolve this without using MAXSTEP to limit timestep, a dummy TTOL device can be used (e.g. Switch), with TTOL instance parameters included
    - Example on Bottom Row
    - Extra time steps are added after V(compare) flip the switch, with additional time steps, output looks reasonable
    - Smaller TTOL value can yield a better results but with longer elapsed time





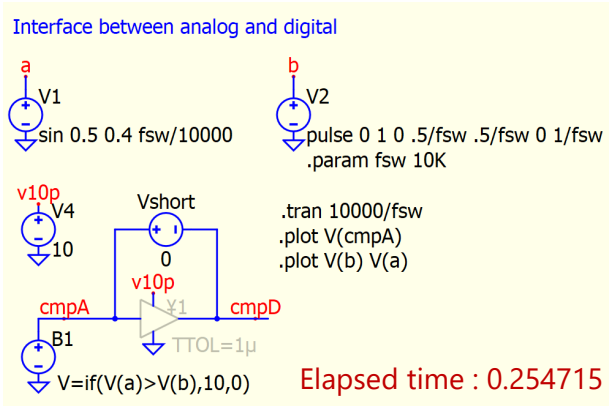


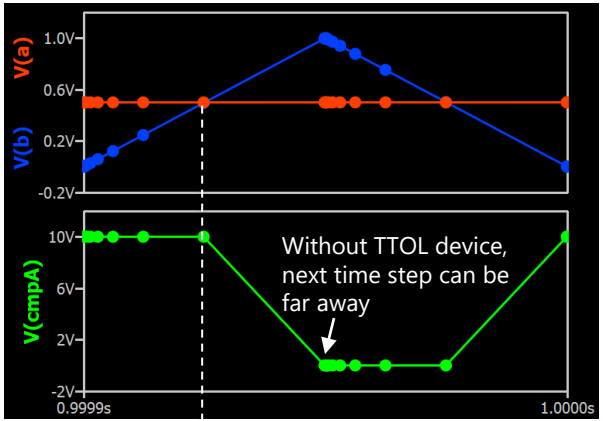


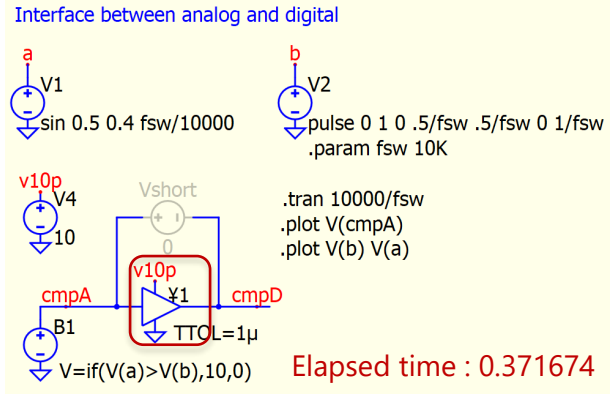
TTOL device to help in adaptive timestep (e.g. function IF)

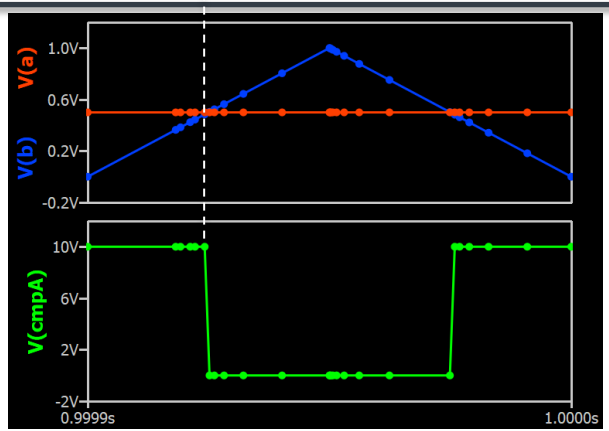
Qspice : TTOL - TTOL device to Interface Analog and Digital.qsch

- TTOL device interface
  - Qspice uses adaptive timestep
  - If a circuit uses a B-source, if(x,y,z) as a comparator, without TTOL device, its simulation timestep can far from compare instance and output looks weird
    - Example in Top Figure
    - Precise time instance at compare action, but as no extra timestep at compare action, output looks like trapezoidal as next timestep is far away
  - To resolve this without using MAXSTEP to limit timestep, a TTOL device can be used (e.g. buffer, with default TTOL=1u)
    - Example in Bottom Figure
    - Extra time steps are added after V(cmpA) flip the buffer, with additional time steps, output looks square waveform
    - Smaller TTOL value can yield a better results but with longer elapsed time





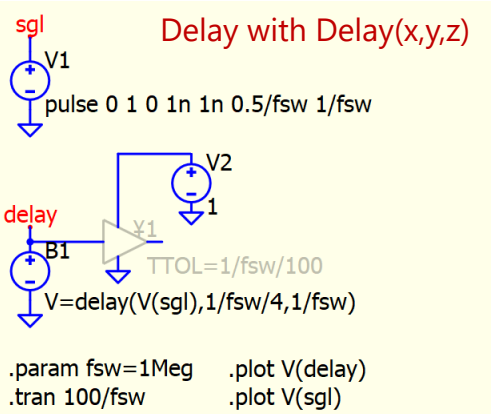


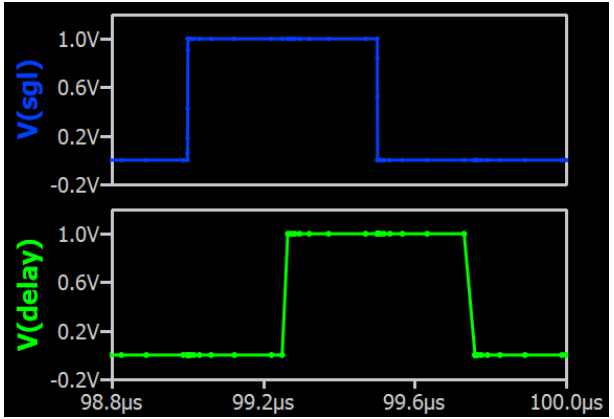


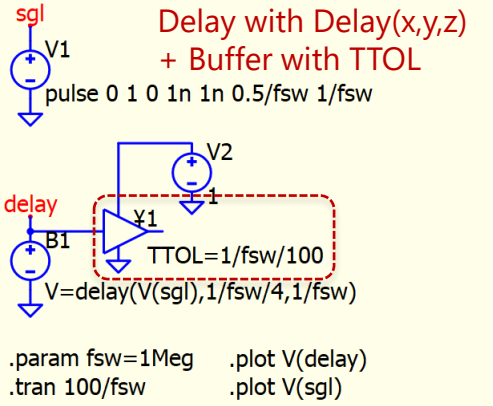
TTOL device to help in adaptive timestep (e.g. function DELAY)

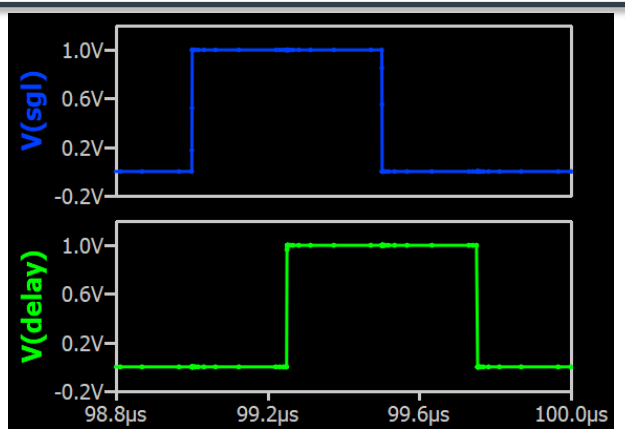
Qspice : TTOL - TTOL for Pulse Delay.qsch

- TTOL device interface
  - Pulse source has instance parameter TimeCtrl to determine timestep at each breakpoint (default is TimeCtrl=Limits), therefore, extra timestep at its rising/falling edge
    - Example in Top Figure
    - But the edge of delayed signal from behavioral source has no information of extra timestep is required, therefore, delayed signal looks like trapezoid
  - A buffer with TTOL is used to improve sharpness of pulse edge
    - Example in Bottom Figure
    - Buffer is triggered when its input cross REF voltage, with TTOL instance parameter, extra timestep is added at such moment









Delay with Transmission Line (alternative way for delay function)

Qspice : Transmission Line for Pulse Delay.qsch

- Delay with Transmission Line
  - Beside of delay(x,y,z) function in behavioral source, delay can be generated with transmission line terminate with Zo
  - However, this approach may generate overshoot/undershoot if maximum time step is not defined, this crux is related to Qspice design as a trade between simulation time and accuracy