LTspice IV Presentation

Why Use LTspice?

- Stable SPICE circuit simulation with
  - Unlimited number of nodes
  - Schematic/symbol editor
  - Waveform viewer
  - Library of passive devices
- Fast simulation of switch mode power supplies
  - Steady state detection
  - Turn on transient
  - Step response
  - Efficiency / power computations
- Advanced analysis and simulation options
  - Not all covered in this presentation
- Outperforms or as powerful as pay-for tools
  - In other words LTspice is free!
- Automatically builds syntax for common tasks

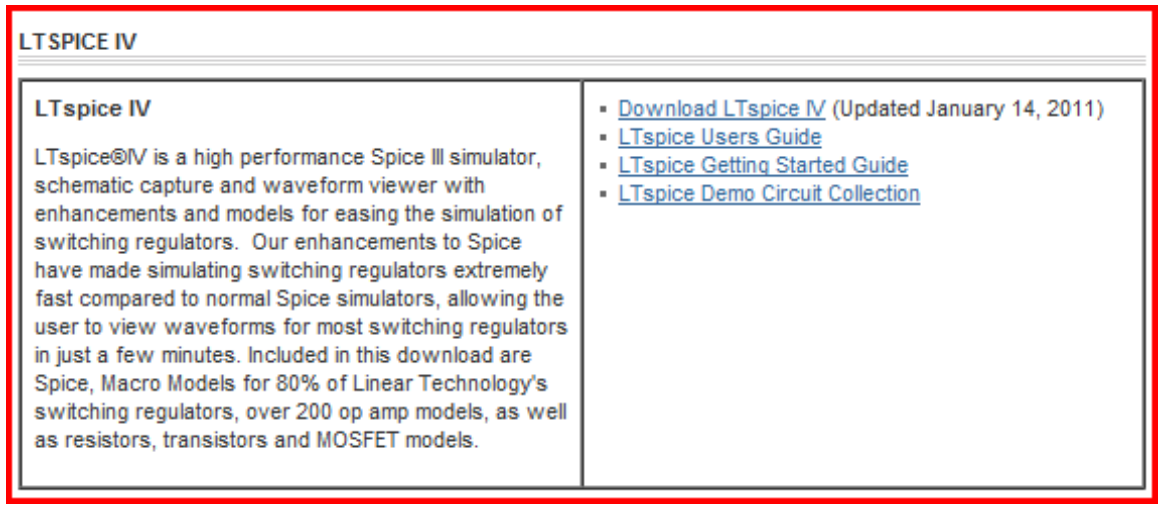
- 1920 macromodels of Linear Technology products
- 1390 Power products

SPICE = Simulation Program with Integrated Circuit Emphasis

LTspice is also a great schematic capture / BOM tool

How Do I Get LTspice and Documentation?

- Go to http://www.linear.com/software
- Left-Click on Download LTspice IV
- Follow the instructions to install



The Basics:
How Do I Get Started using LTspice?

How Do I Get Started Using LTspice?

- Use one of the 100's demo circuit available on linear.com
  - Designed and Reviewed by Factory Apps Group
  - Go to http://www.linear.com/software
- Use a pre-drafted test fixture (JIG)
  - Provides a good starting point, but is not production-ready
  - Used to prove out part models, and are not complete designs.
  - Components are typically "ideal" components and will need to be modified based on your operating conditions
- Use the schematic editor to create your own design
  - LTspice contains models for most LTC power devices and many more
- Use simulation circuits posted on the LTspice Yahoo! User's Group. tech.groups.yahoo.com/group/LTspice
  - Also contains many very helpful discussion threads

You can also check out LTspice capabilities using the education examples available on C:\Program Files\LTC\LTspiceIV\examples\Educational

How Do I Get Started Using LTspice?

- Use one of the 100's demo circuit available on linear.com
  - Designed and Reviewed by Factory Apps Group
  - Go to http://www.linear.com/software
- Use a pre-drafted test fixture (JIG)
  - Provides a good starting point, but is not production-ready
  - Used to prove out part models, and are not complete designs.
  - Components are typically "ideal" components and will need to be modified based on your operating conditions
- Use the schematic editor to create your own design
  - LTspice contains models for most LTC power devices and many more
- Use simulation circuits posted on the LTspice Yahoo! User's Group. tech.groups.yahoo.com/group/LTspice
  - Also contains many very helpful discussion threads

Demo Circuits on linear.com

Go to http://www.linear.com/software

LTspice IV

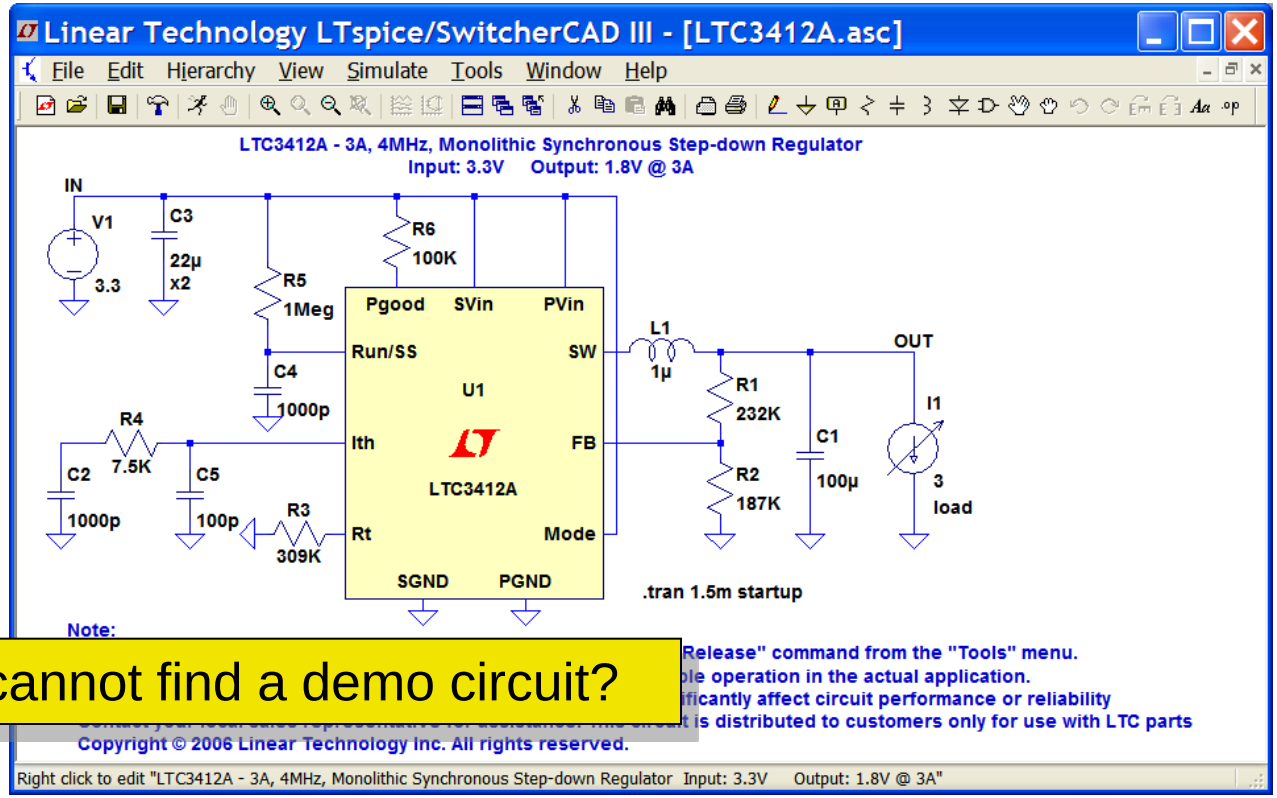
LTspice IV is a high performance Spice III simulator, schematic capture and waveform viewer with enhancements and models for easing the simulation of switching regulators. Our enhancements to Spice have made simulating switching regulators extremely fast compared to normal Spice simulators, allowing the user to view waveforms for most switching regulators in just a few minutes. Included in this download are Spice, Macro Models for 80% of Linear Technology's switching regulators, over 200 op amp models, as well as resistors, transistors and MOSFET models.

- Download LTspice IV (Updated May 5, 2009)
- LTspice Users Guide
- LTspice Getting Started Guide
- LTspice Demo Circuit Collection

Part Number	Updated	Download
LT1071HV - 5A and 2.5A High Efficiency Switching Regulators	May 5th, 2006	LT1071HV.asc
LT1072HV - 1.25A High Efficiency Switching Regulator	May 5th, 2006	LT1072HV.asc
LT1076HV - Step-Down Switching Regulator	May 5th, 2006	LT1076HV.asc
LT1111 - Micropower DC/DC Converter Adjustable and Fixed 5V, 12V	May 26th, 2006	LT1111.asc
LT1172HV - 100kHz, 5A, 2.5A and 1.25A High Efficiency Switching Regulators	May 5th, 2006	LT1172HV.asc
LT1173 - Micropower DC/DC Converter Adjustable and Fixed 5V, 12V	Jun 12th, 2006	LT1173.asc
LT1308B - Single Cell High CurrentMicropower 600kHz Boost DC/DC Converter	May 26th, 2006	LT1308B.asc
LT1370HV - 500kHz High Efficiency 6A Switching Regulator	May 26th, 2006	LT1370HV.asc

Demo Circuits

- ✓ Designed and reviewed by factory apps group
  - It remains the customer's responsibility to verify proper and reliable operation in the actual application
  - Component substitution and printed circuit board layout may significantly affect circuit performance or reliability



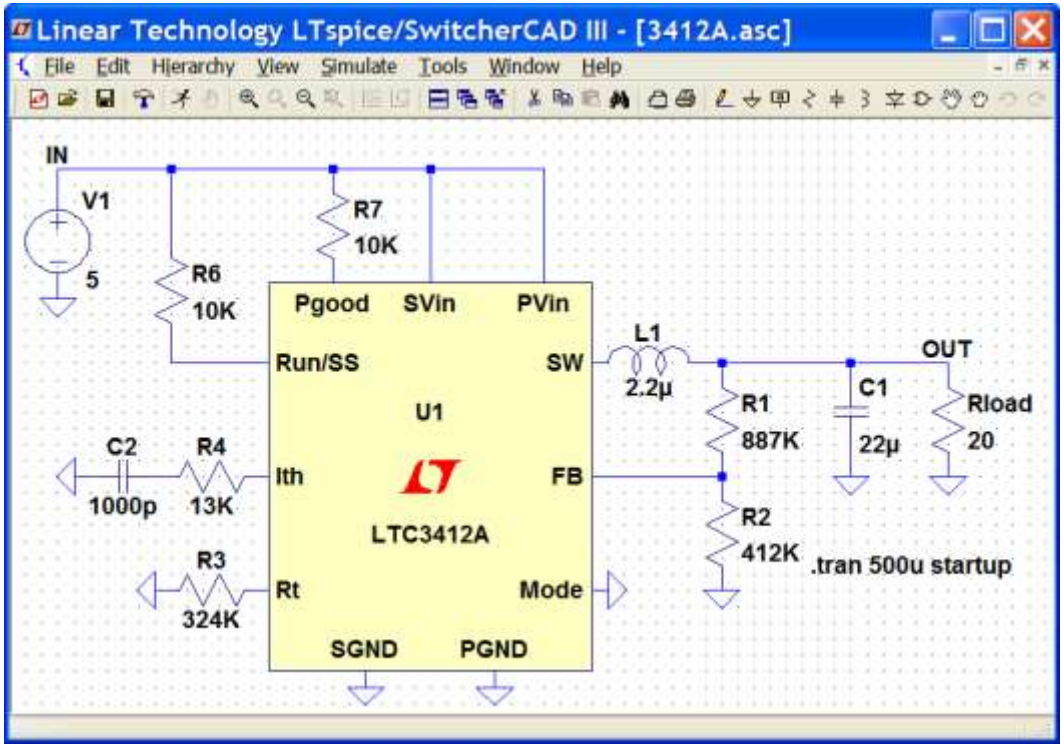
What if I cannot find a demo circuit?

How Do I Get Started Using LTspice?

- Use one of the 100's demo circuit available on linear.com
  - Designed and Reviewed by Factory Apps Group
  - Go to http://www.linear.com/software
- Use a pre-drafted test fixture (JIG)
  - Provides a good starting point, but is not production-ready
  - Used to prove out part models, and are not complete designs.
  - Components are typically "ideal" components and will need to be modified based on your operating conditions
- Use the schematic editor to create your own design
  - LTspice contains models for most LTC power devices and many more
- Use simulation circuits posted on the LTspice Yahoo! User's Group tech.groups.yahoo.com/group/LTspice
  - Also contains many very helpful discussion threads

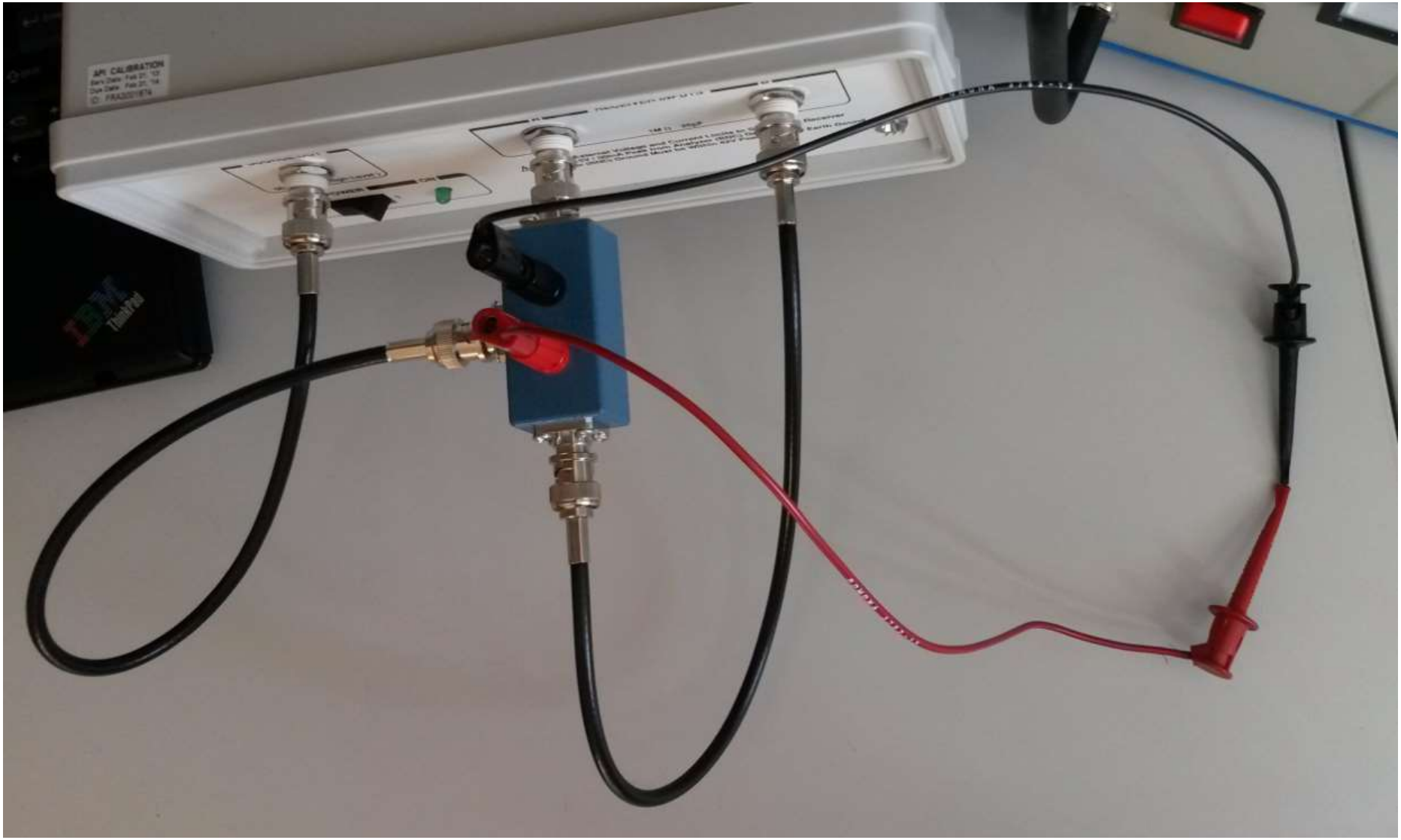
Pre-drafted Test Fixture

- Provides a good starting point
  - These simulations / designs are not production-ready
  - Used to prove out part models, and are not complete designs.

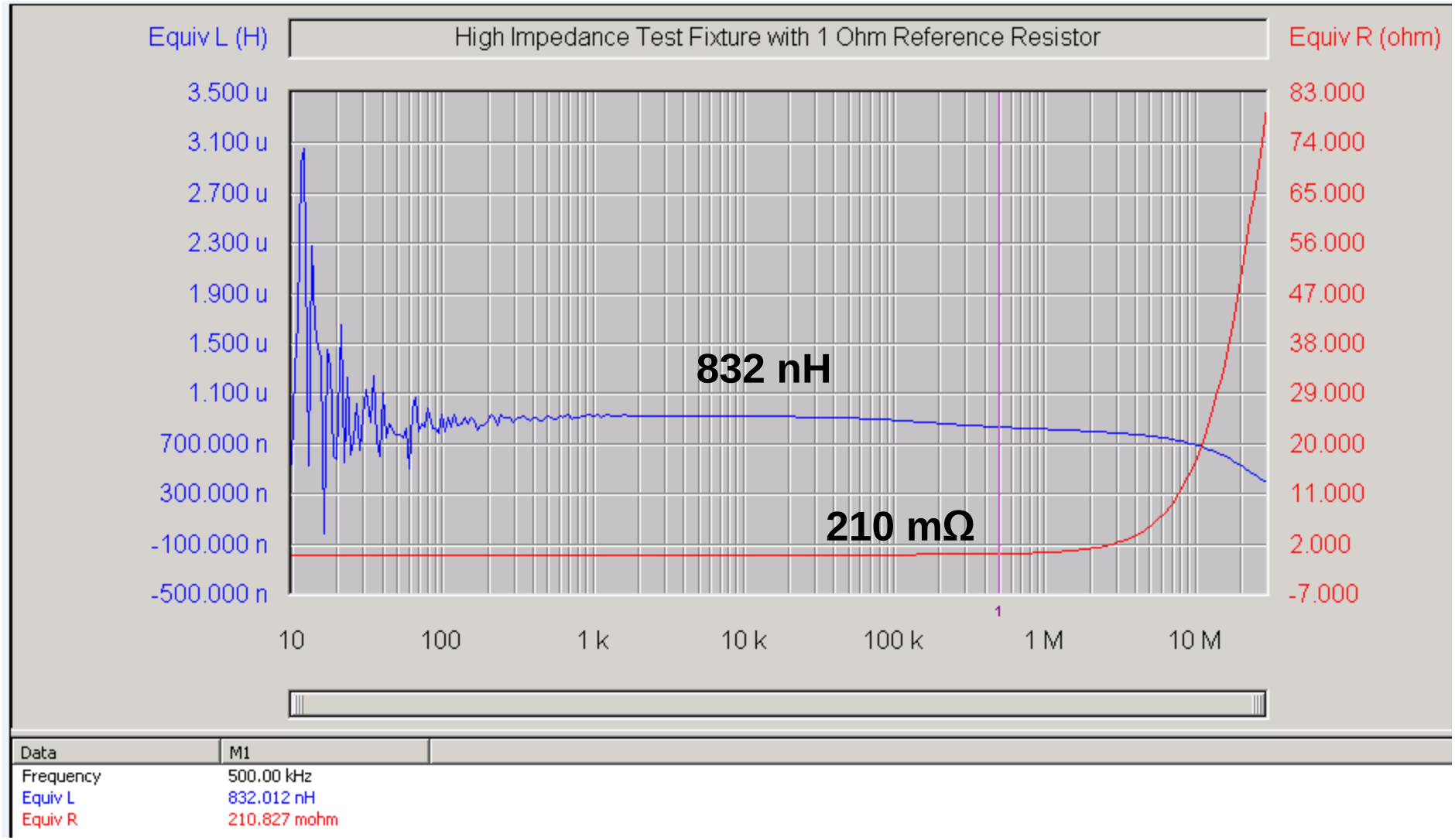


- It remains the customer's responsibility to verify proper and reliable operation in the actual application
- Printed circuit board layout may significantly affect circuit performance and reliability

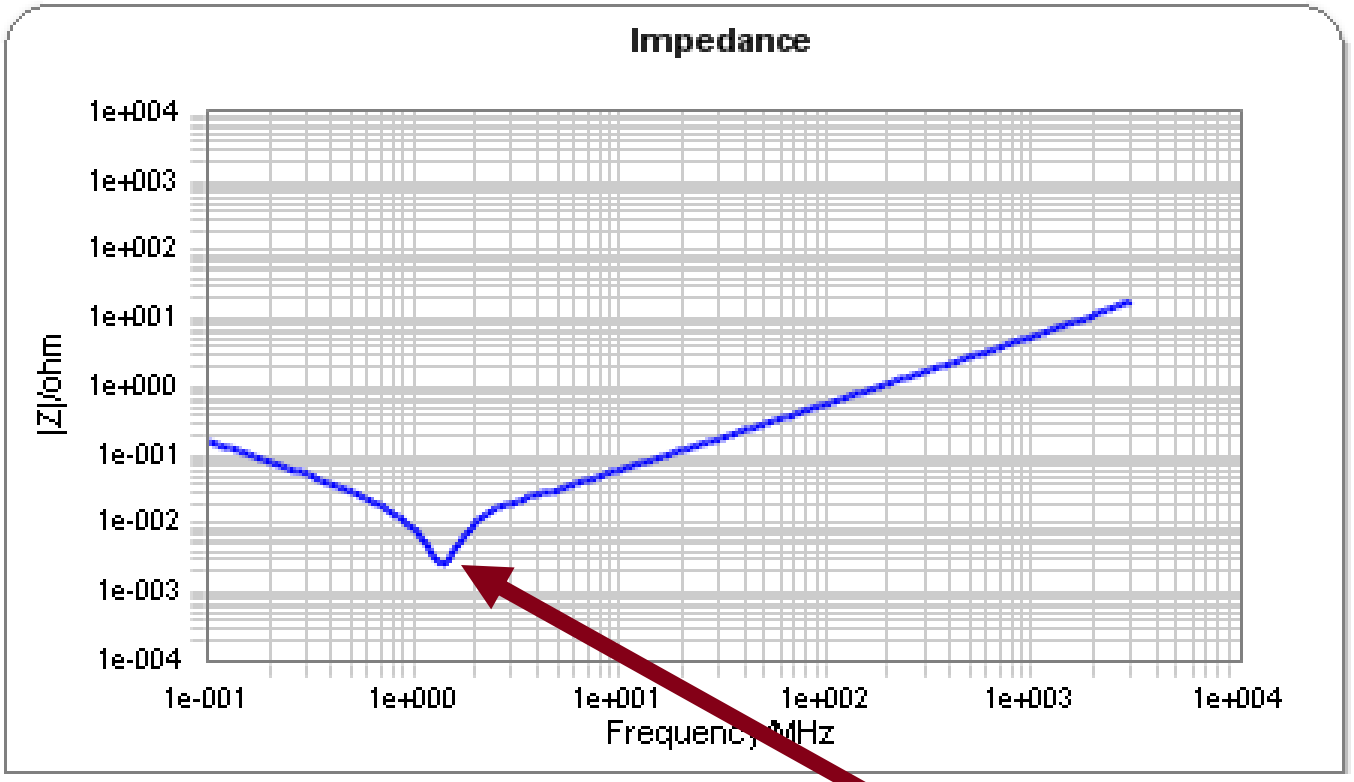
Realistic Source L and R



Realistic Source L and R



Realistic MLCC ESR



~3-4 mΩ

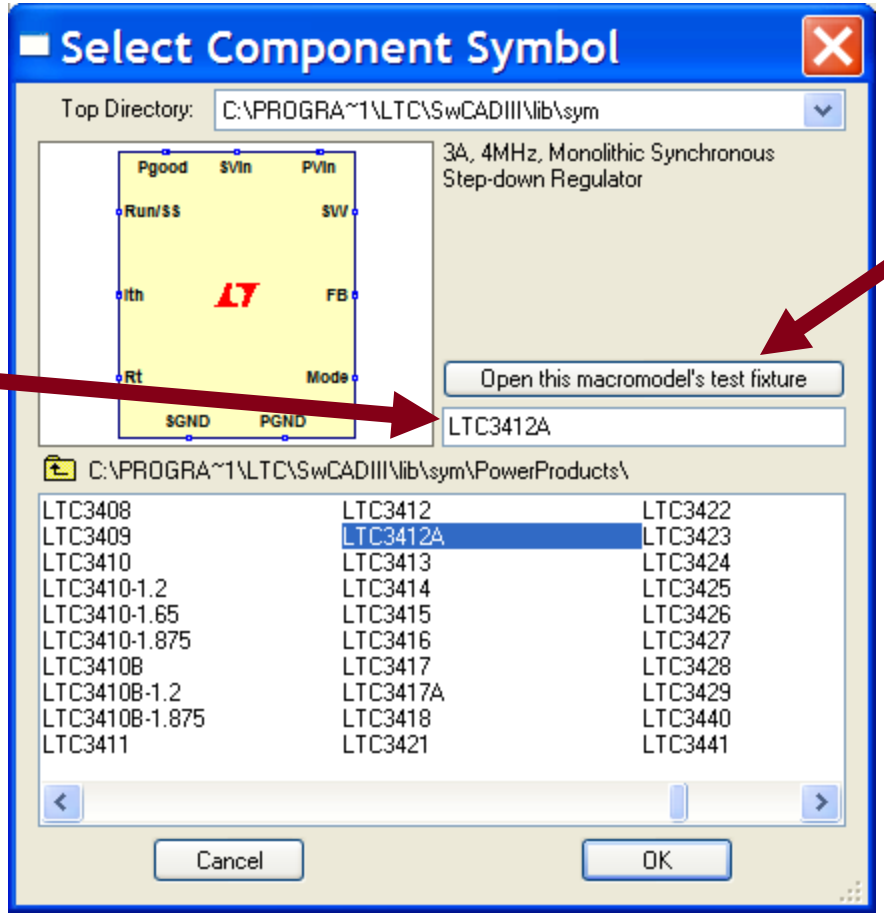
- **10 µF, 25V, 1206, X7R**

Selecting a Model & Opening Test Fixture

- Use the “root” part to search for the model
  - i.e. 3412A
- Select “Open this macromodel’s test fixture”

1. Enter 3412A

2. Select

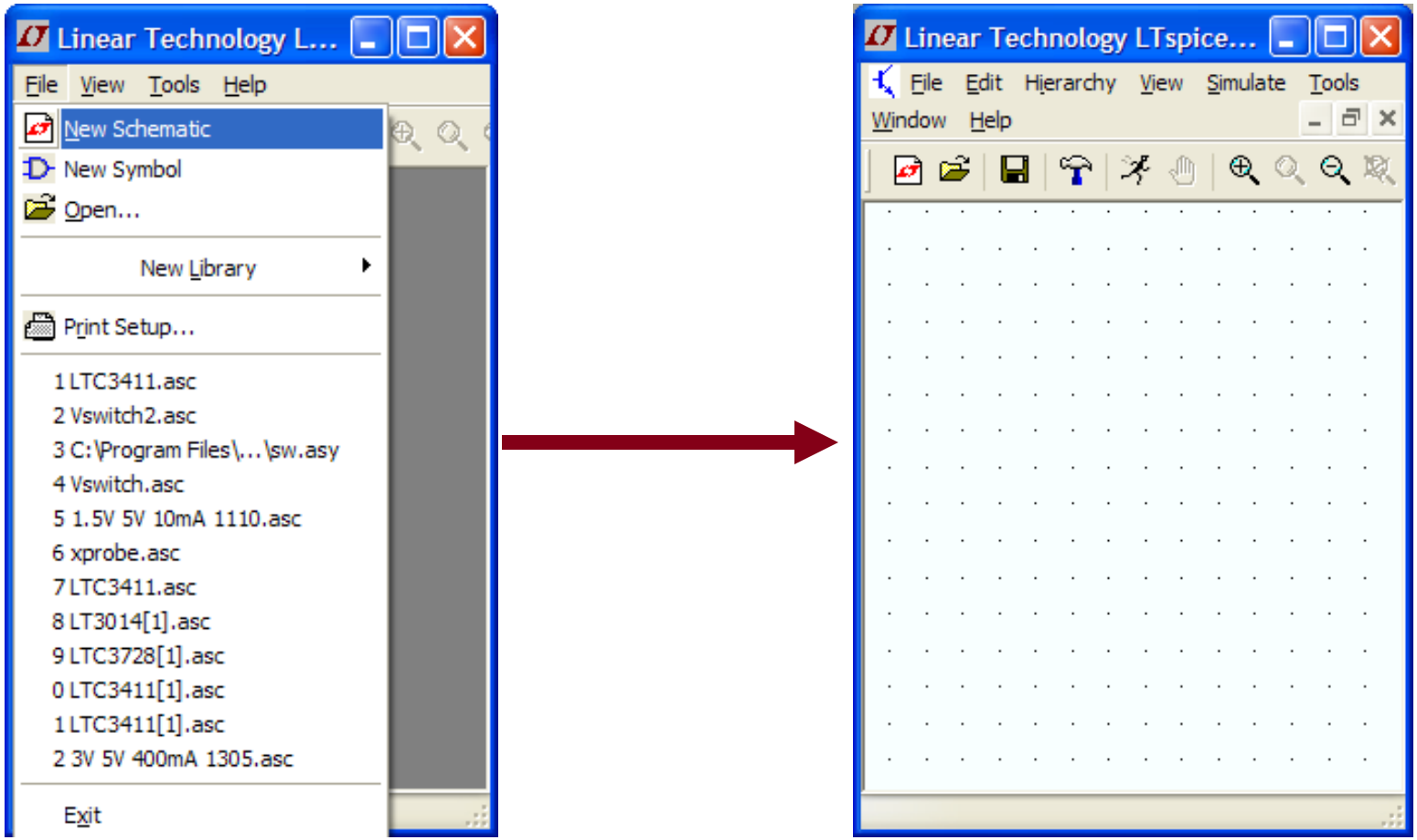


How Do I Get Started Using LTspice?

- Use one of the 100's demo circuit available on linear.com
  - Designed and Reviewed by Factory Apps Group
  - Go to http://www.linear.com/software
- Use a pre-drafted test fixture (JIG)
  - Provides a good starting point, but is not production-ready
  - Used to prove out part models, and are not complete designs.
  - Components are typically "ideal" components and will need to be modified based on your operating conditions
- Use the schematic editor to create your own design
  - LTspice contains models for most LTC power devices and many more
- Use simulation circuits posted on the LTspice Yahoo! User's Group tech.groups.yahoo.com/group/LTspice
  - Also contains many very helpful discussion threads

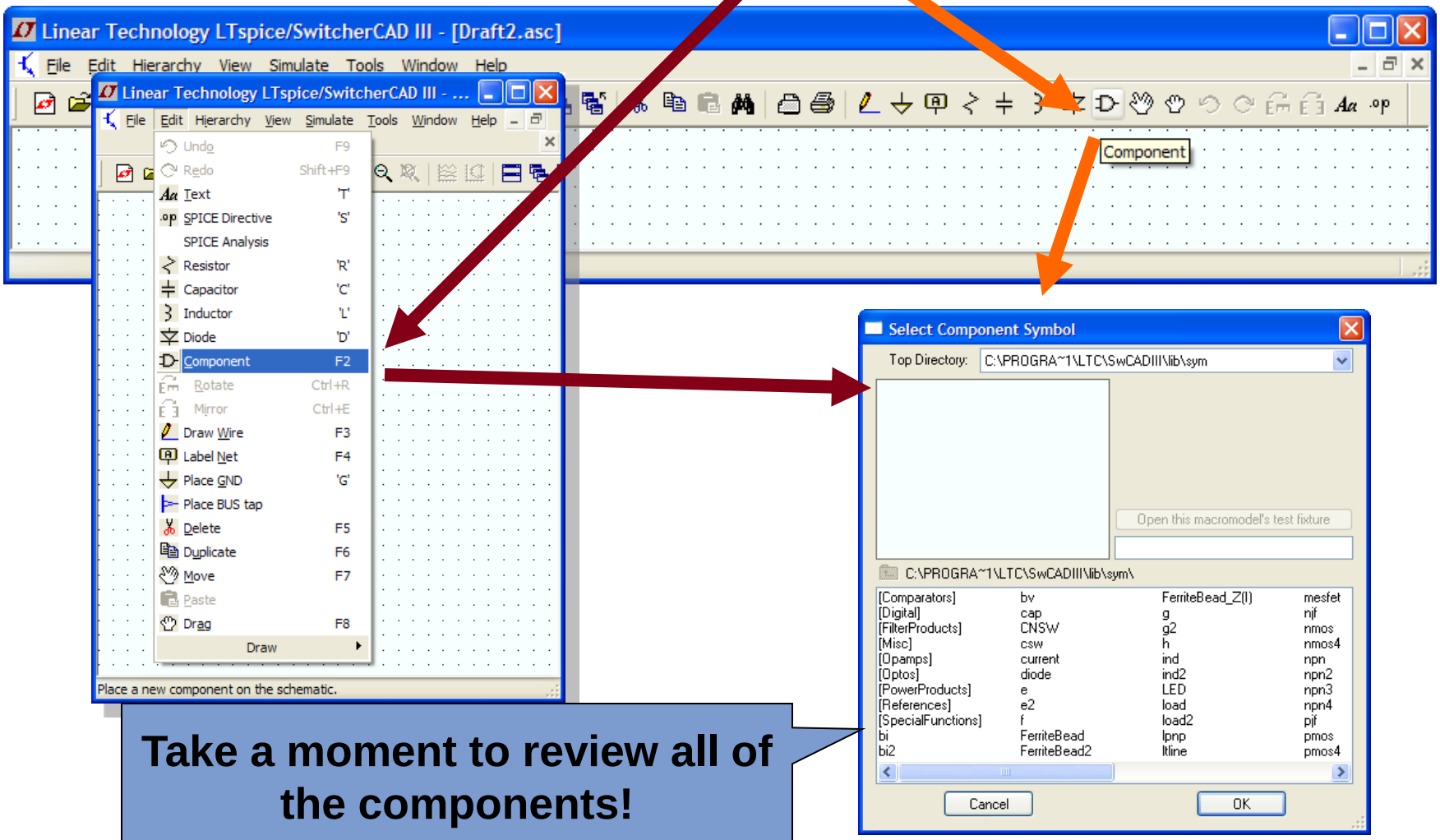
Start With a New Schematic

- Select File and New Schematic
  - Will open up a blank schematic screen



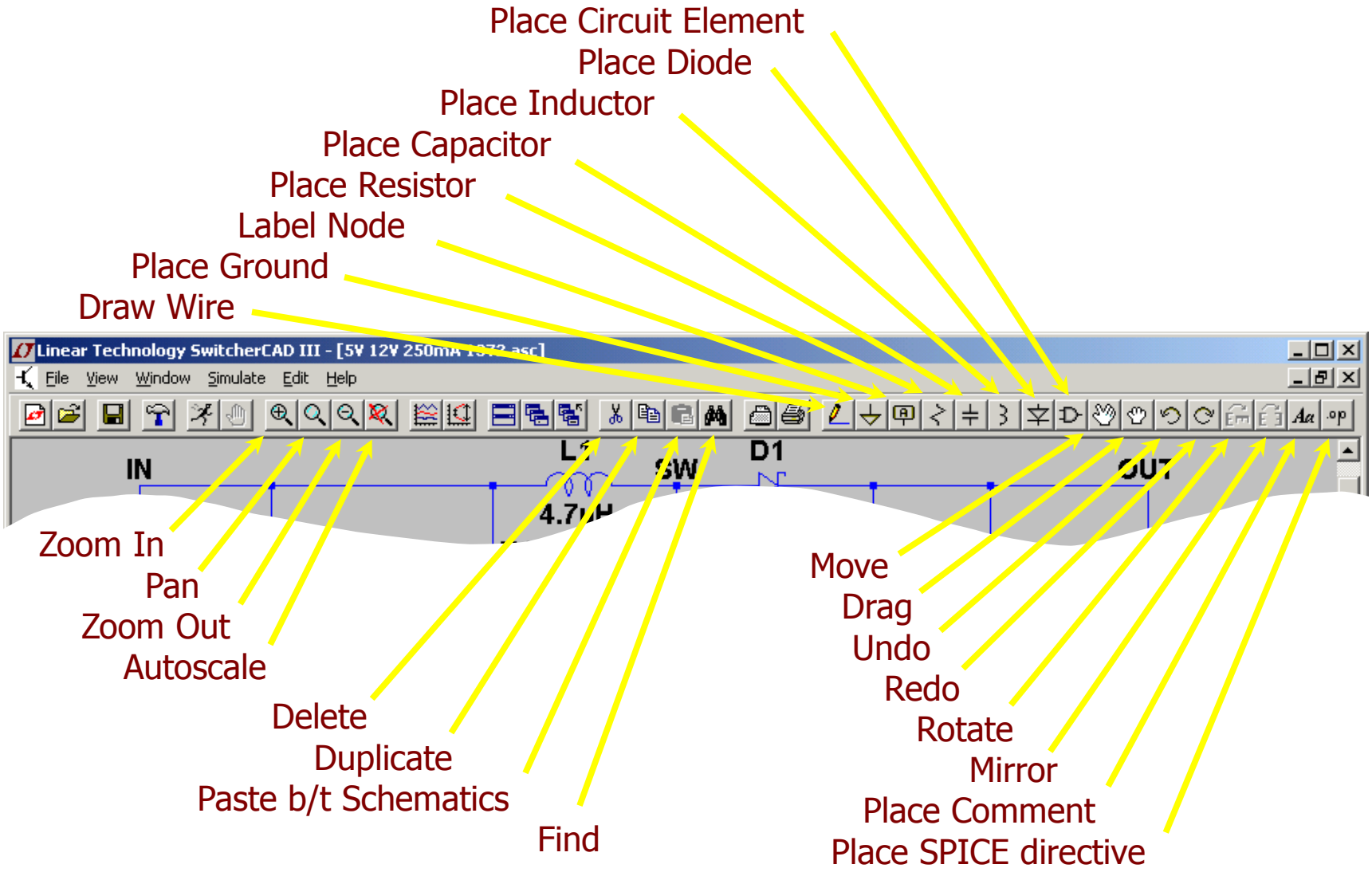
Add a Component

- Use Add a Component or F2



Take a moment to review all of the components!

Schematic Editing



Using Labels to Specify Units for Component Attributes

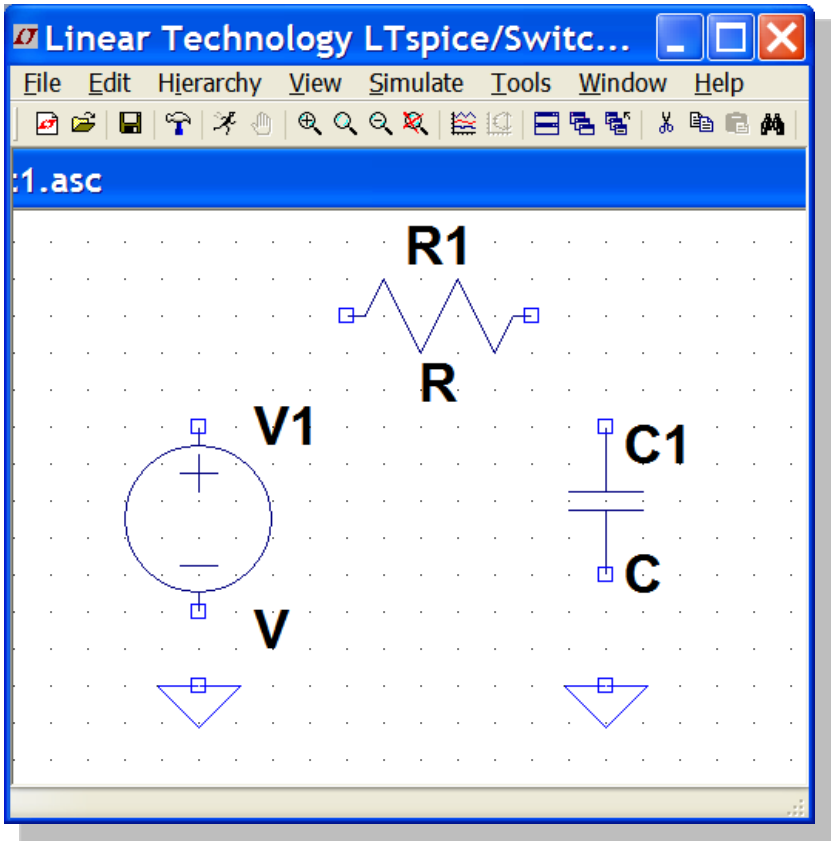
- K = k = kilo = 10^3
- MEG = meg = 10^6
- G = g = giga = 10^9
- T = t = tera = 10^{12}
- M = m = milli = 10^{-3}
- U = u = micro = 10^{-6}
- N = n = nano = 10^{-9}
- P = p = pico = 10^{-12}
- F = f = femto = 10^{-15}

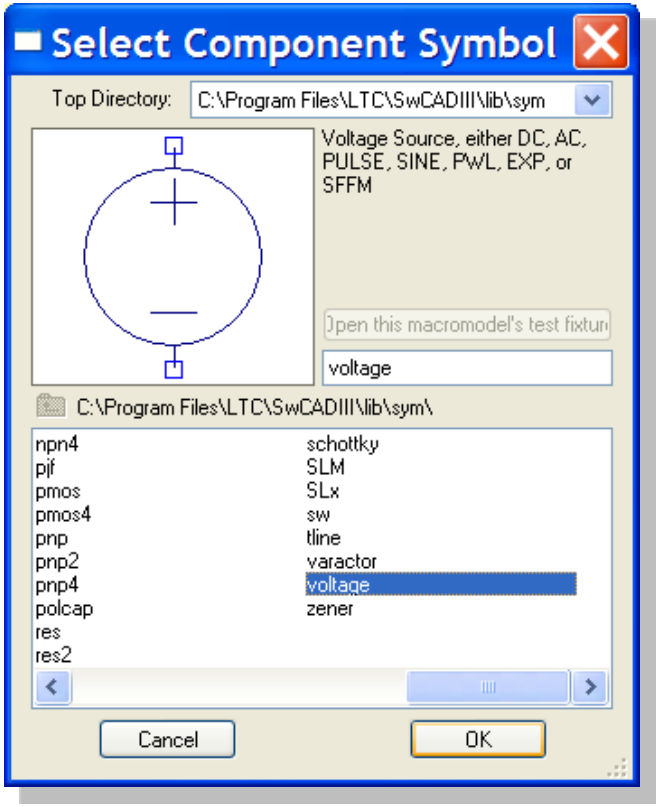
Hints

- Use ***MEG (or meg)*** to specify 10^6, not *M*
- Enter ***1*** for 1 Farad, not *1F*

Wiring up a Simple RC Circuit

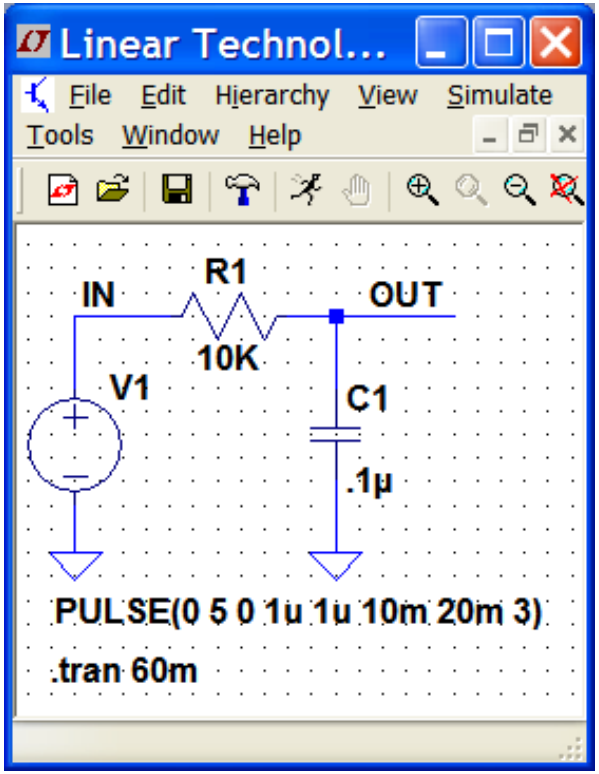
- Using the toolbar, select New Schematic
- Using the toolbar, select a Resistor, Capacitor and Ground. Place these on the schematic as shown below. Use **Ctrl R** to rotate before placement
- Using the toolbar, select Component. From the component window, type “voltage” in the dialog box, and click “OK” to place a voltage source

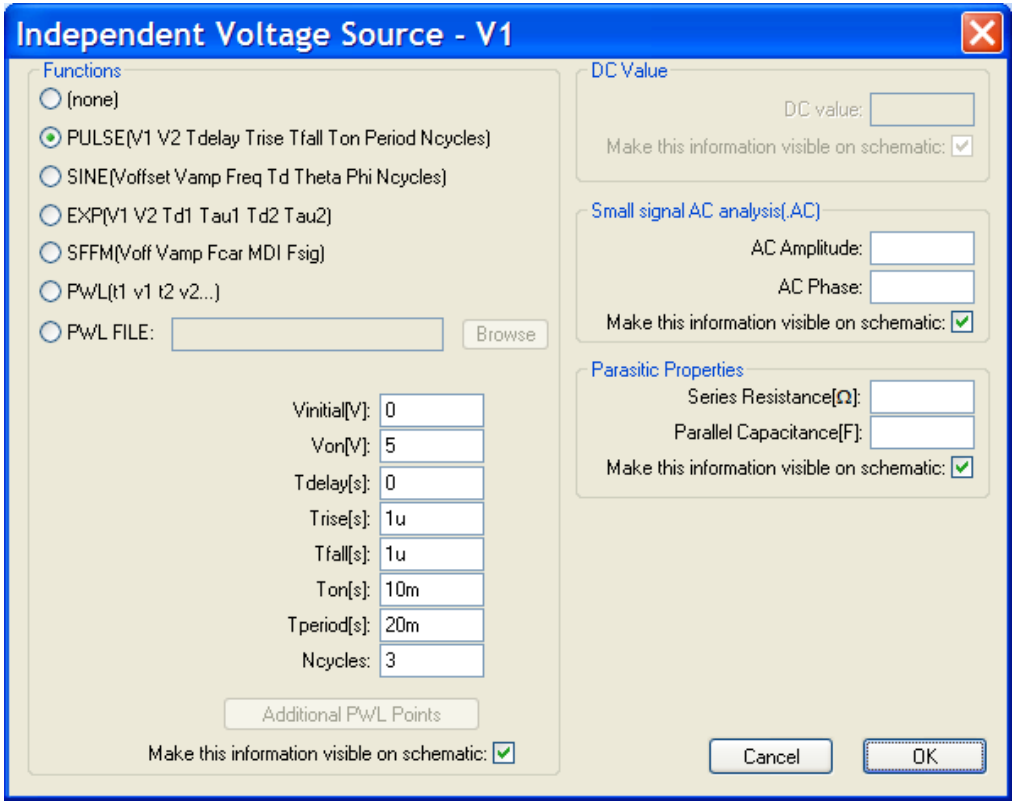


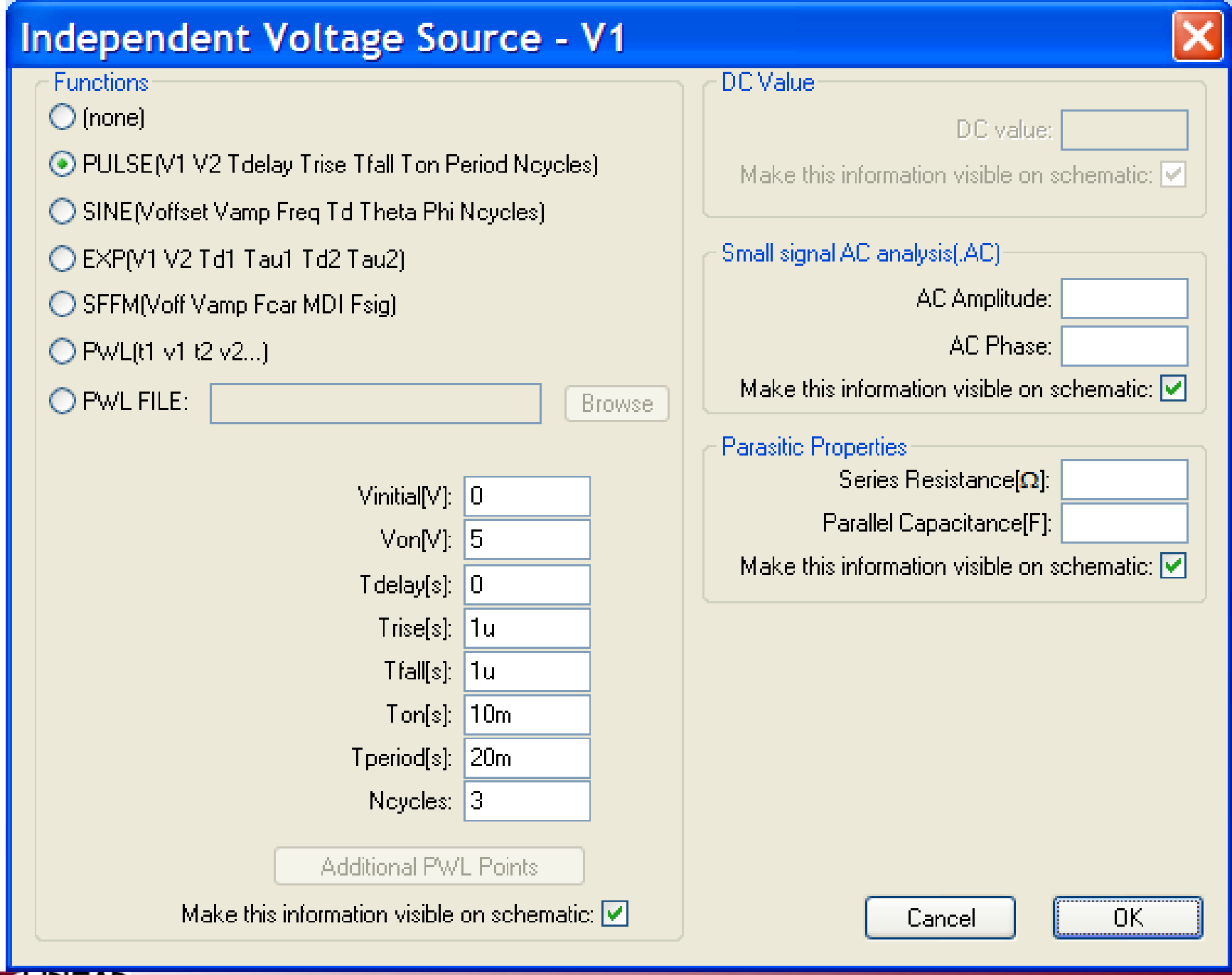


Wiring up a Simple RC Circuit

- Using the toolbar, select Wire. Wire up the RC circuit as shown below.
- Using the toolbar, select Label Net. Label the input/output nodes as shown below
- Right-Click on each component to change its value as shown below
- Right-Click on the voltage source and enter the parameters shown below under the “Advanced” tab.

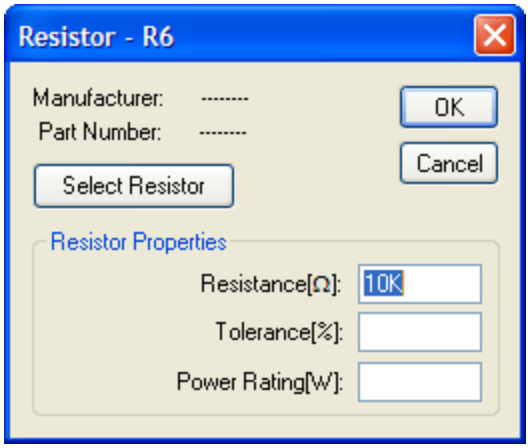


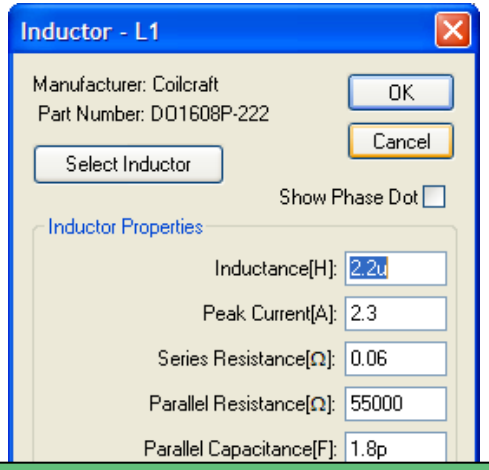


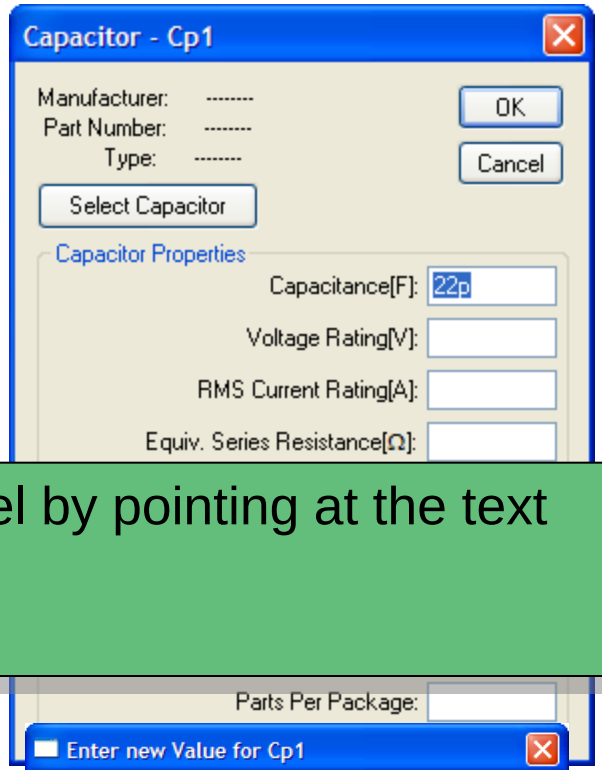

Independent Voltage Source - V1
Functions
(none)
PULSE(V1 V2 Tdelay Trise Tfall Ton Period Ncycles)
SINE(Voffset Vamp Freq Td Theta Phi Ncycles)
EXP(V1 V2 Td1 Tau1 Td2 Tau2)
SFFM(Voff Vamp Fcar MDI Fsig)
PWL(t1 v1 t2 v2...)
PWL FILE:
Browse
Vinitial[V]: 0
Von[V]: 5
Tdelay[s]: 0
Trise[s]: 1u
Tfall[s]: 1u
Ton[s]: 10m
Tperiod[s]: 20m
Ncycles: 3
Additional PWL Points
Make this information visible on schematic:
DC Value
DC value:
Make this information visible on schematic:
Small signal AC analysis(.AC)
AC Amplitude:
AC Phase:
Make this information visible on schematic:
Parasitic Properties
Series Resistance[Ω]:
Parallel Capacitance[F]:
Make this information visible on schematic:
Cancel
OK

Editing Components

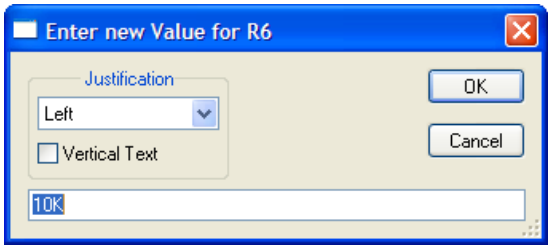
- Component attributes can be edited by pointing at the component with the mouse and Right-Clicking

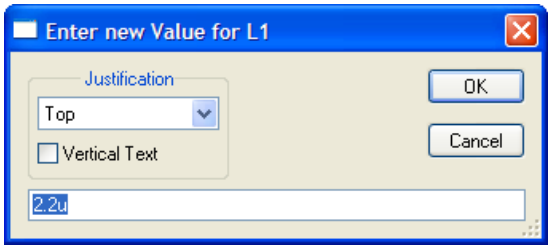


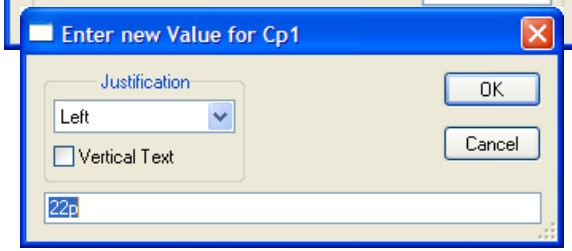




- You can also edit the visible attribute and label by pointing at the text with the mouse and then right-clicking
  - Mouse cursor will turn into a text caret

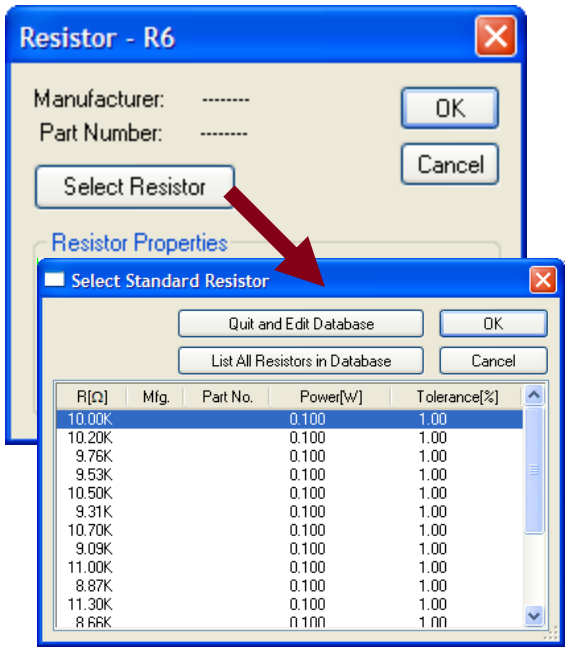


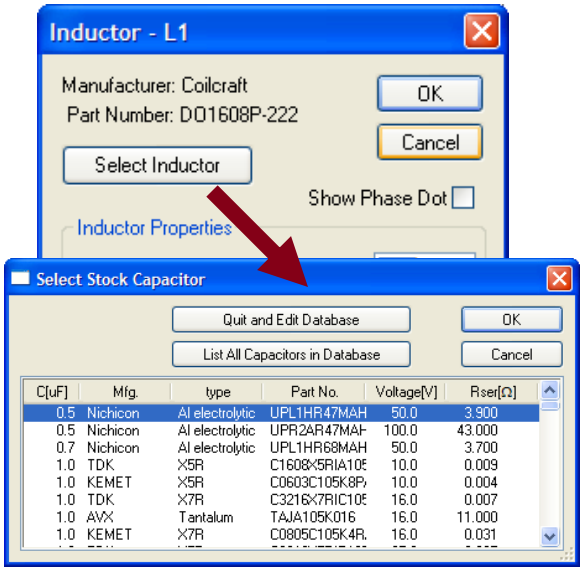


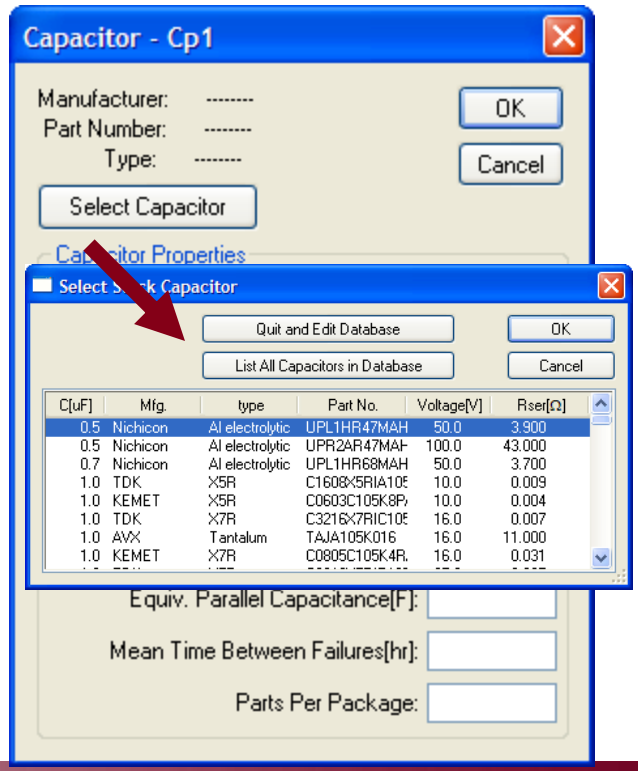


Component Database

- Components such as
  - Resistors, capacitors, inductors, diodes,
  - Bipolar transistors, MOSFET transistors, JFET transistors
  - Independent voltage and current sources
  - You can access a database of known devices







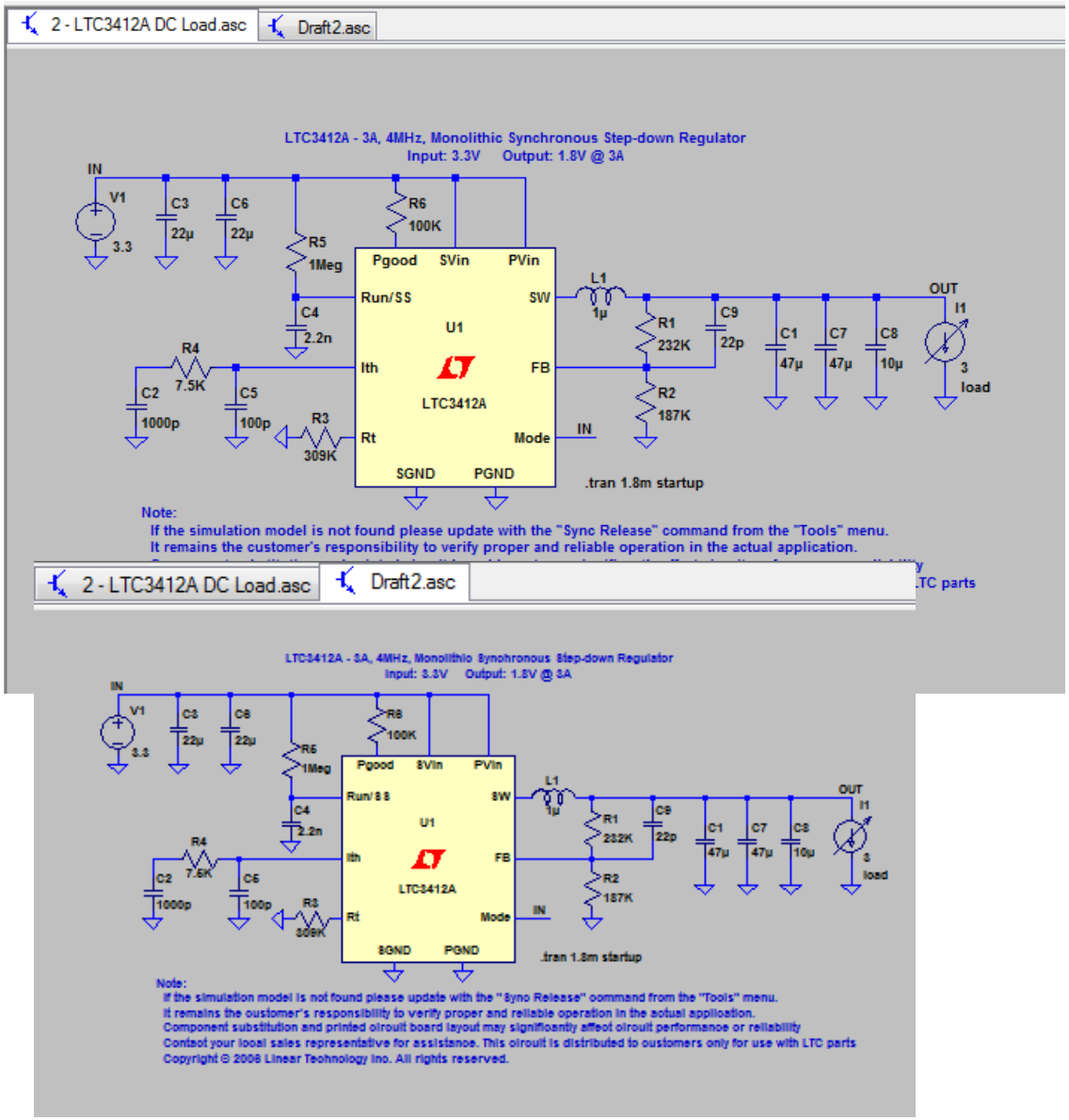
Copying from One Schematic to Another

1. **Open the schematic with the circuit you want to copy**

2. **Open the schematic where the copy will be pasted**

3. **Copy using the button** 

4. **Move the whole copied section up, out of the active window, and click on the destination schematic**

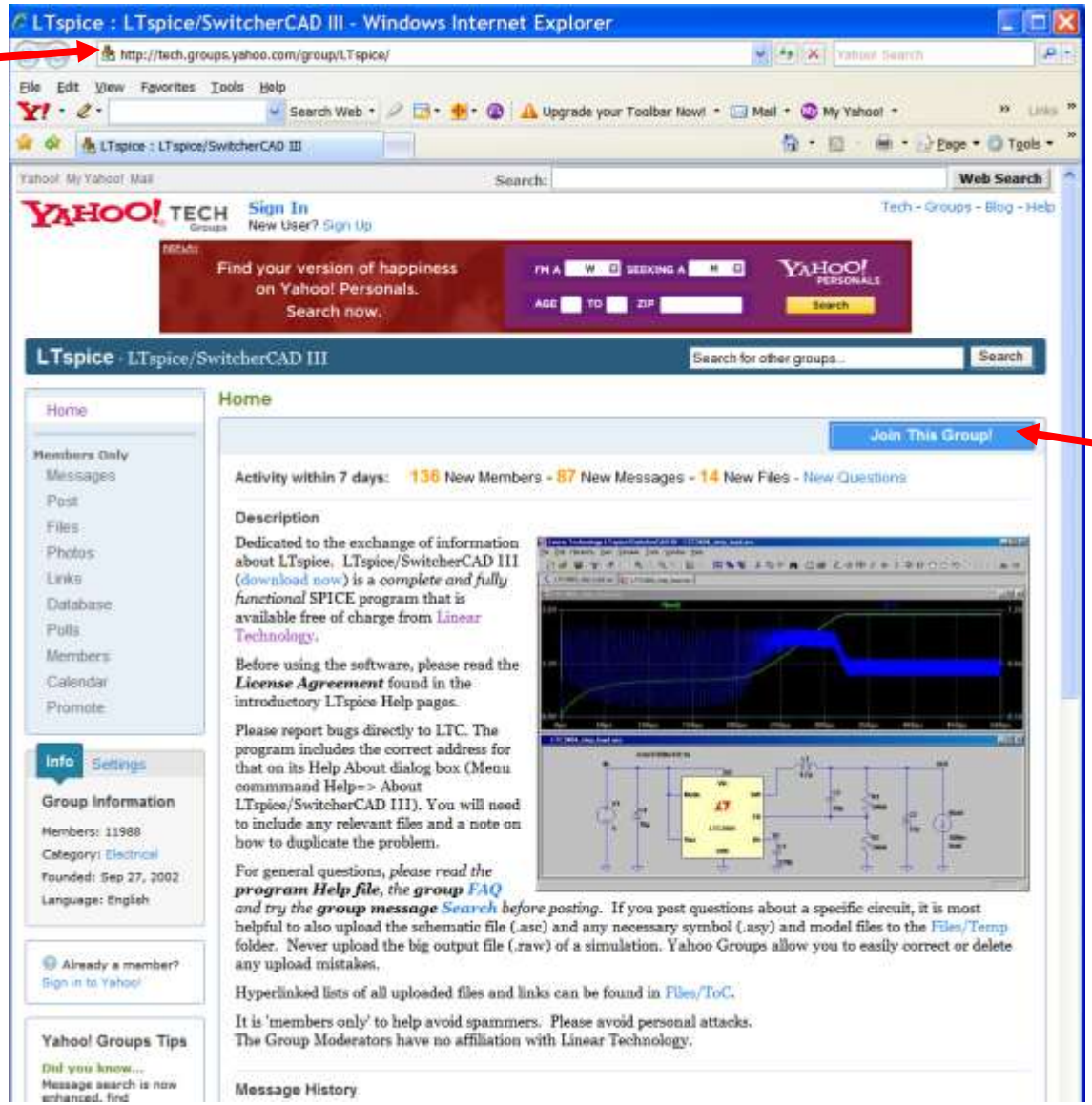


How Do I Get Started using LTspice?

- Use one of the 100's demo circuit available on linear.com
  - Designed and Reviewed by Factory Apps Group
  - Go to http://www.linear.com/software
- Use a pre-drafted test fixture (JIG)
  - Provides a good starting point, but is not production-ready
  - Used to prove out part models, and are not complete designs.
  - Components are typically "ideal" components and will need to be modified based on your operating conditions
- Use the schematic editor to create your own design
  - LTspice contains models for most LTC power devices and many more
- Use simulation circuits posted on the LTspice Yahoo! User's Group tech.groups.yahoo.com/group/LTspice
  - Also contains many very helpful discussion threads

LTspice Yahoo! User's Group Web Page

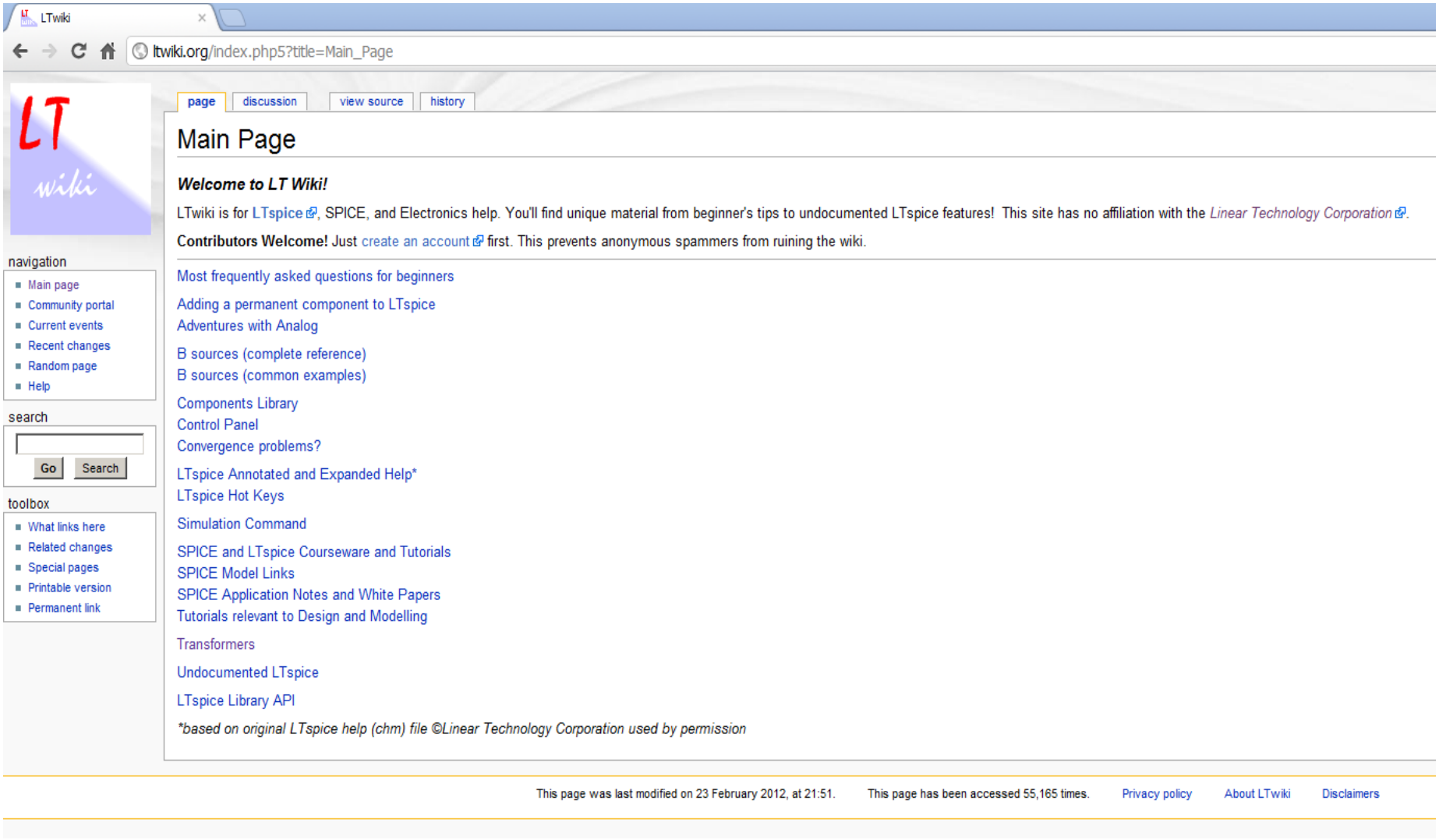
URL



Join the group here. As of Apr 2013, there are over 42300 members!

URL

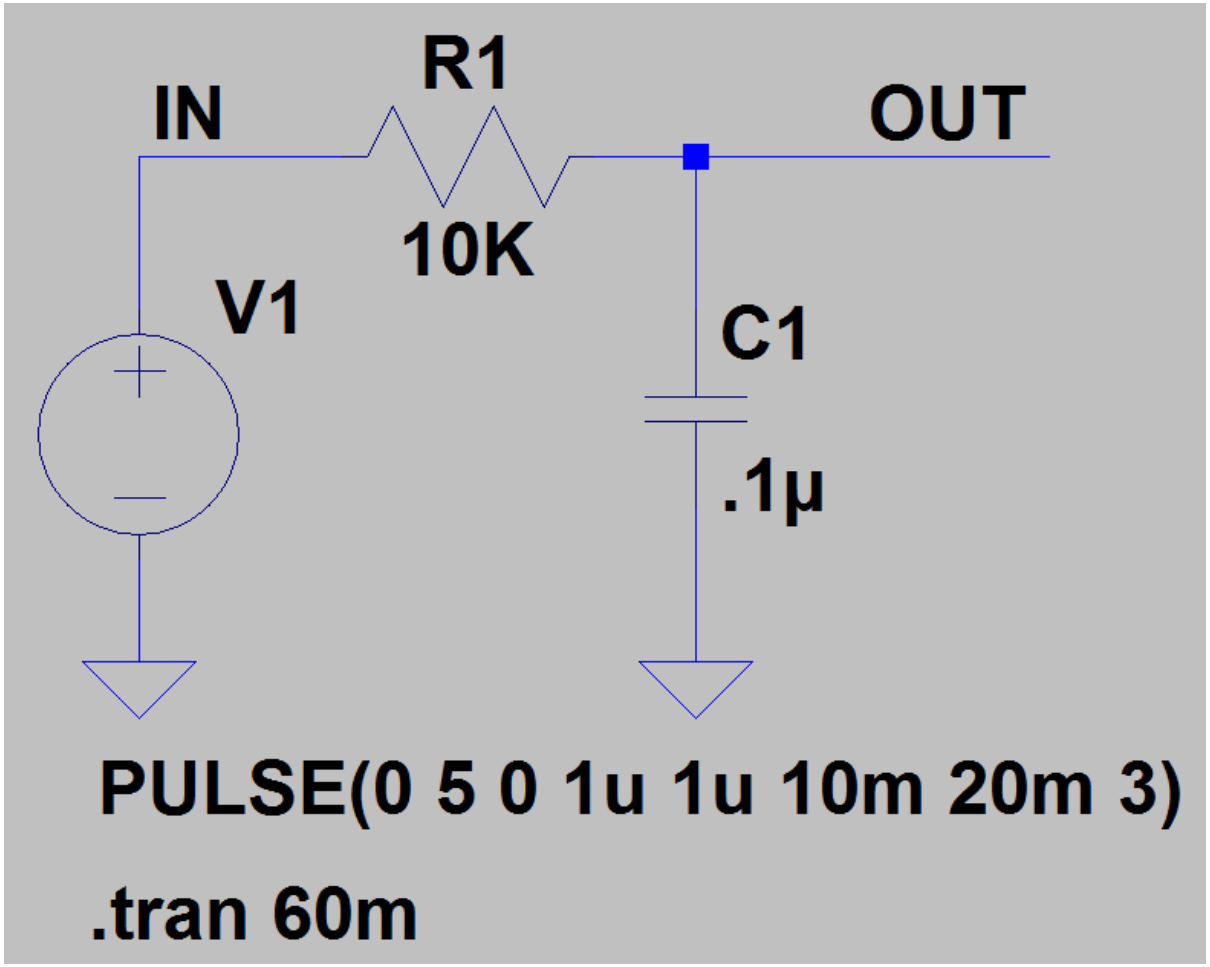
LT Wiki Web Page



How Do You Run and Probe a Circuit in LTspice?

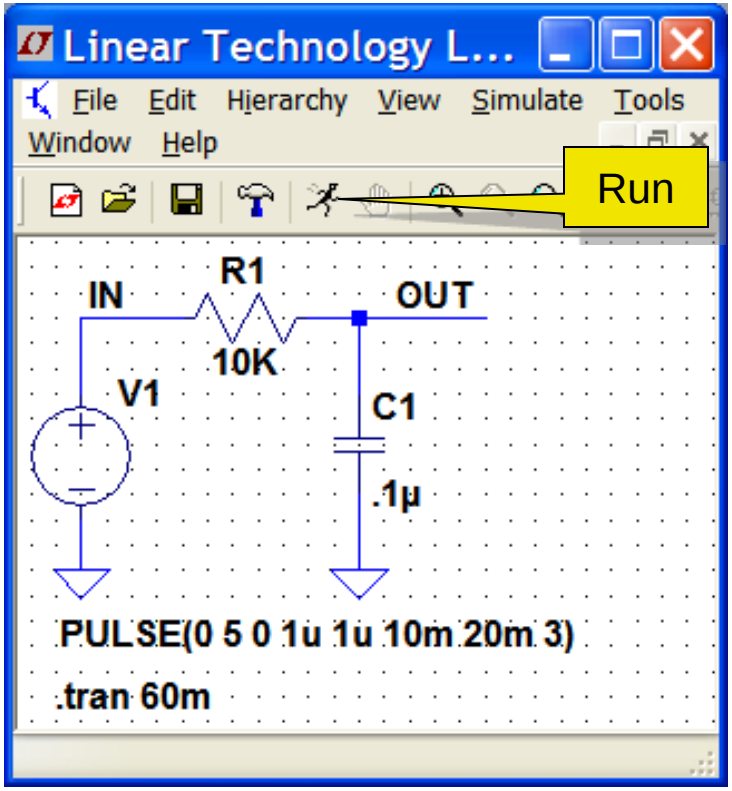
Example

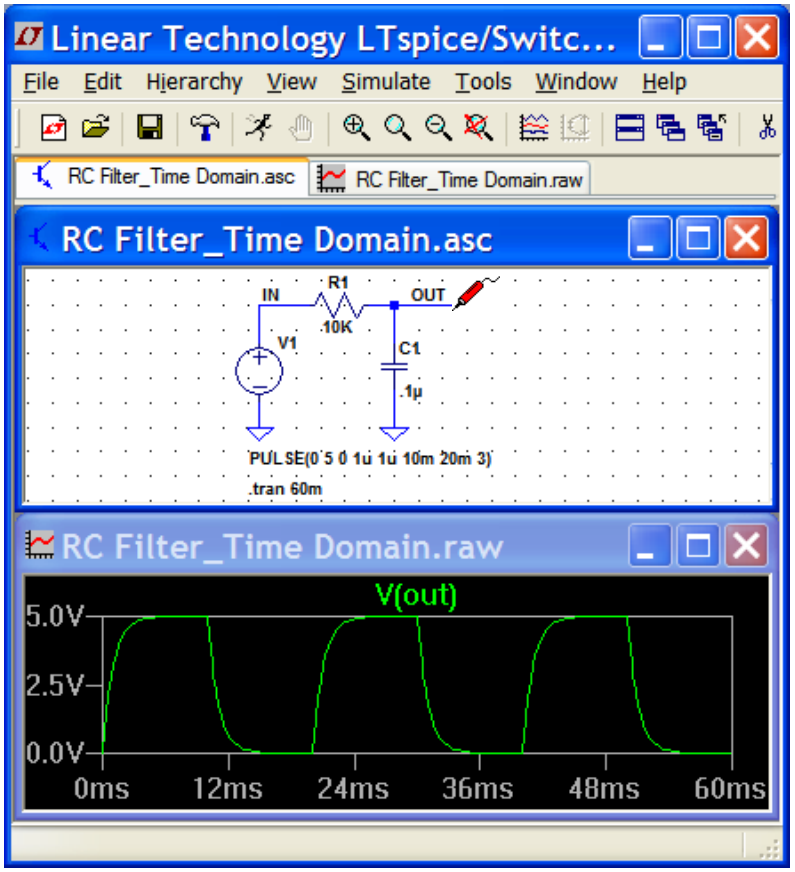
- RC Filter Time Domain.asc



Running the RC Circuit Simulation

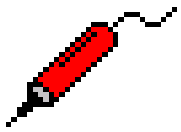
- With the RC circuit in the active window, click on the “Running Person” button on the tool bar
- The Edit Simulation Command window will appear. Set the Stop Time for 60msec, and click “OK”
- Using the mouse, click on the “OUT” node to display the output voltage waveform





Waveform Viewer

- LTspice has an integrated waveform viewer
  - Plot the voltage on any wire by simply point and click

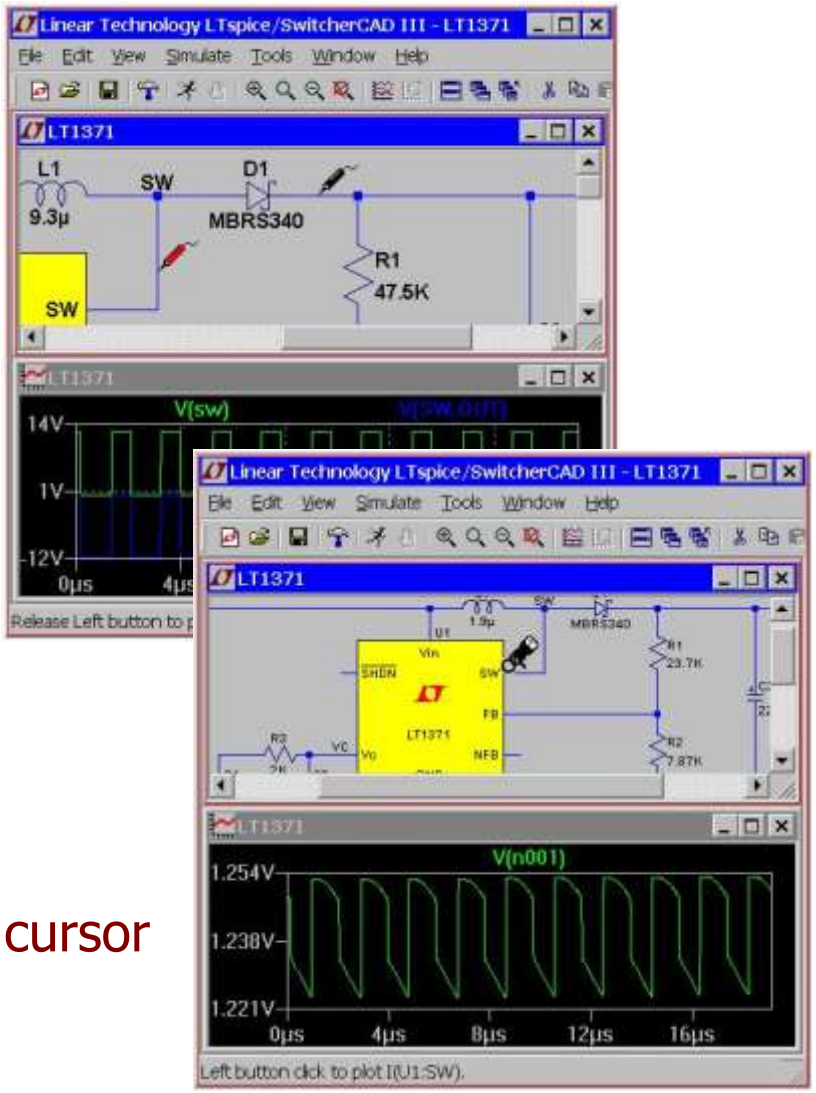


Voltage probe cursor

  - Plot the current through any component with two connections by clicking on the body of the component

Current probe cursor

    - R, C, L
    - Convention of positive current is in the direction into the pin



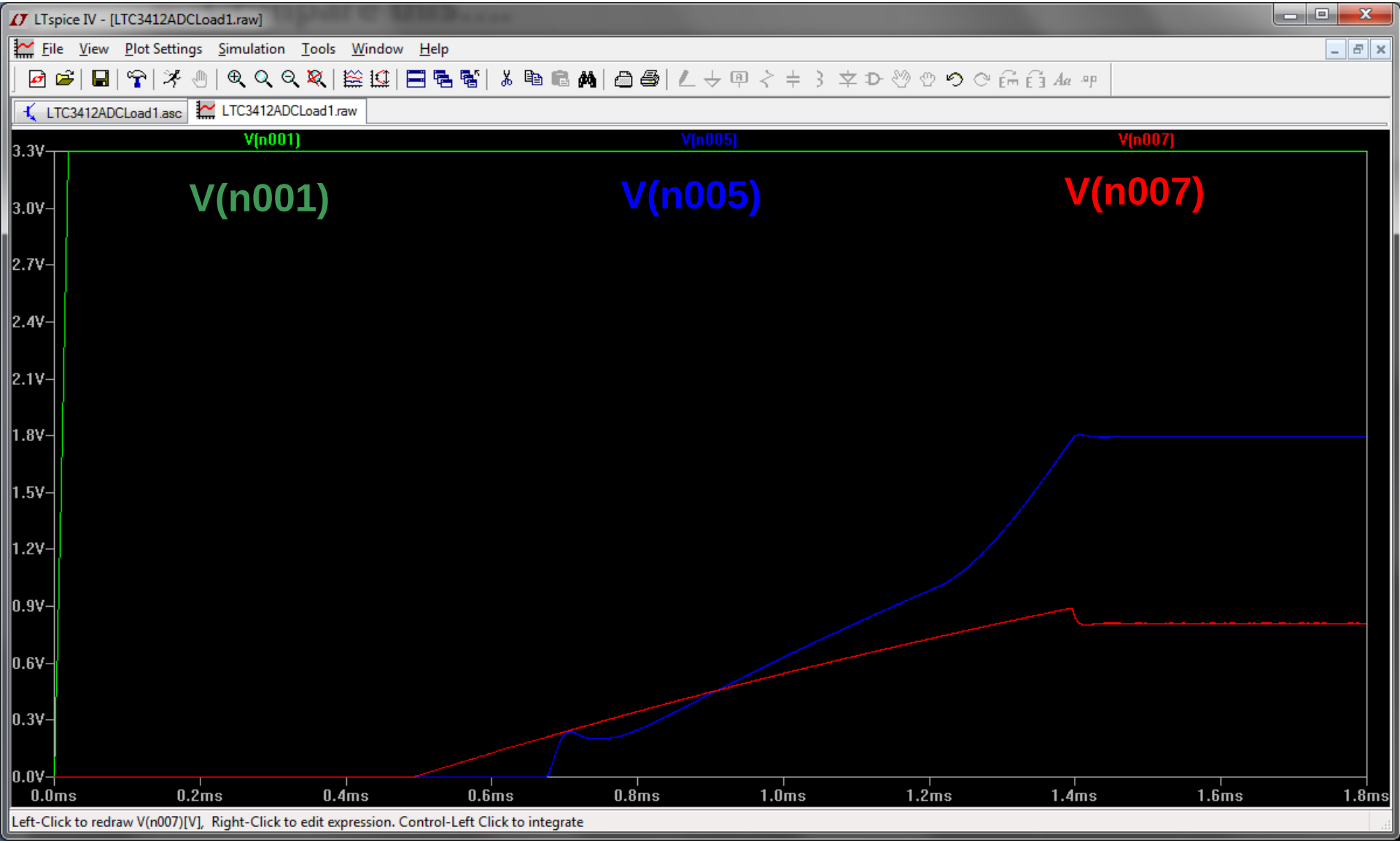
Advantages of Labeling

- Replaces arcane SPICE machine node names with easy to understand and remember human names

- Allows LTspice circuit nodes to match those on your production schematic, i.e. “TP15”

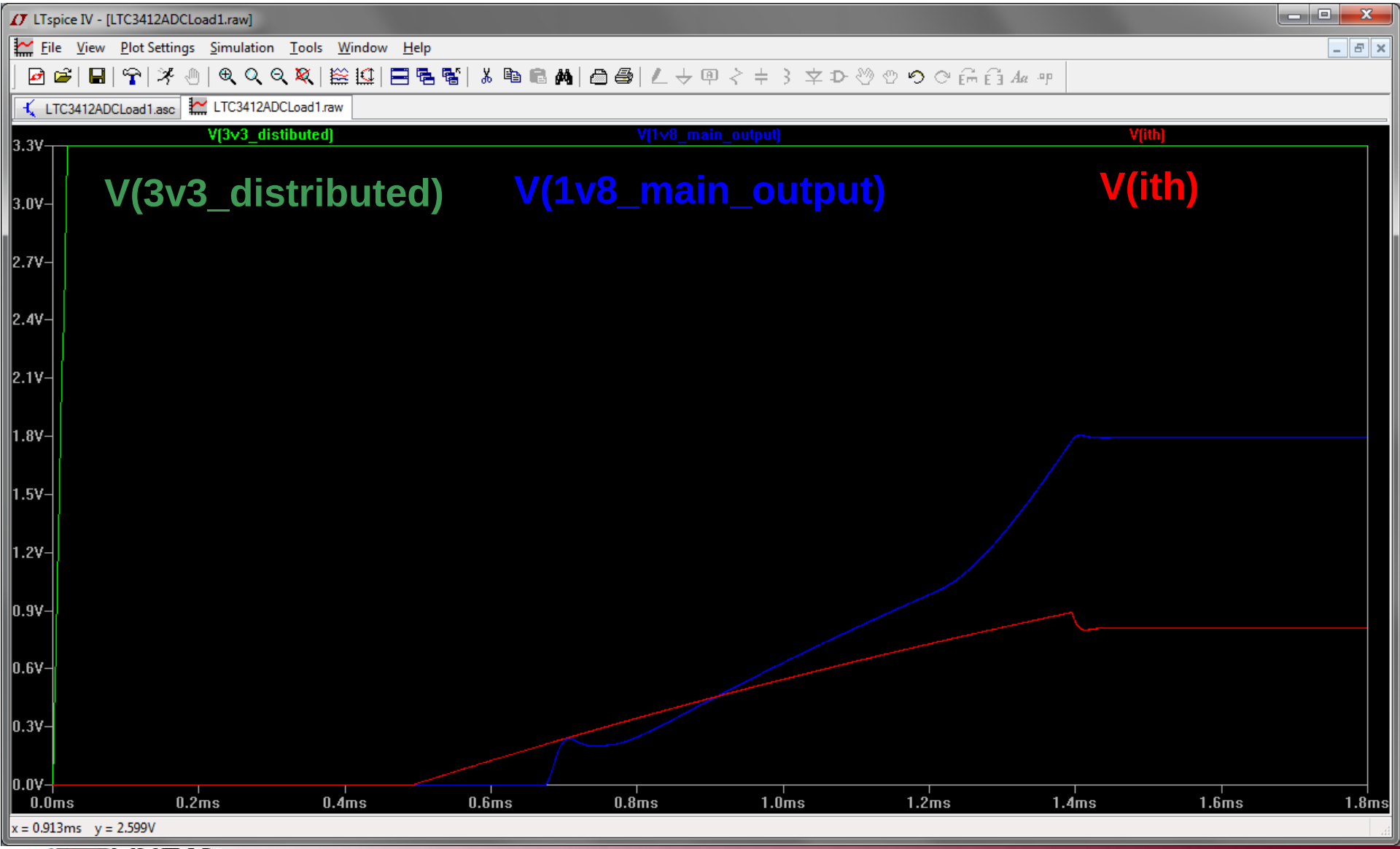
Advantages of Labeling

- **Compare this….**



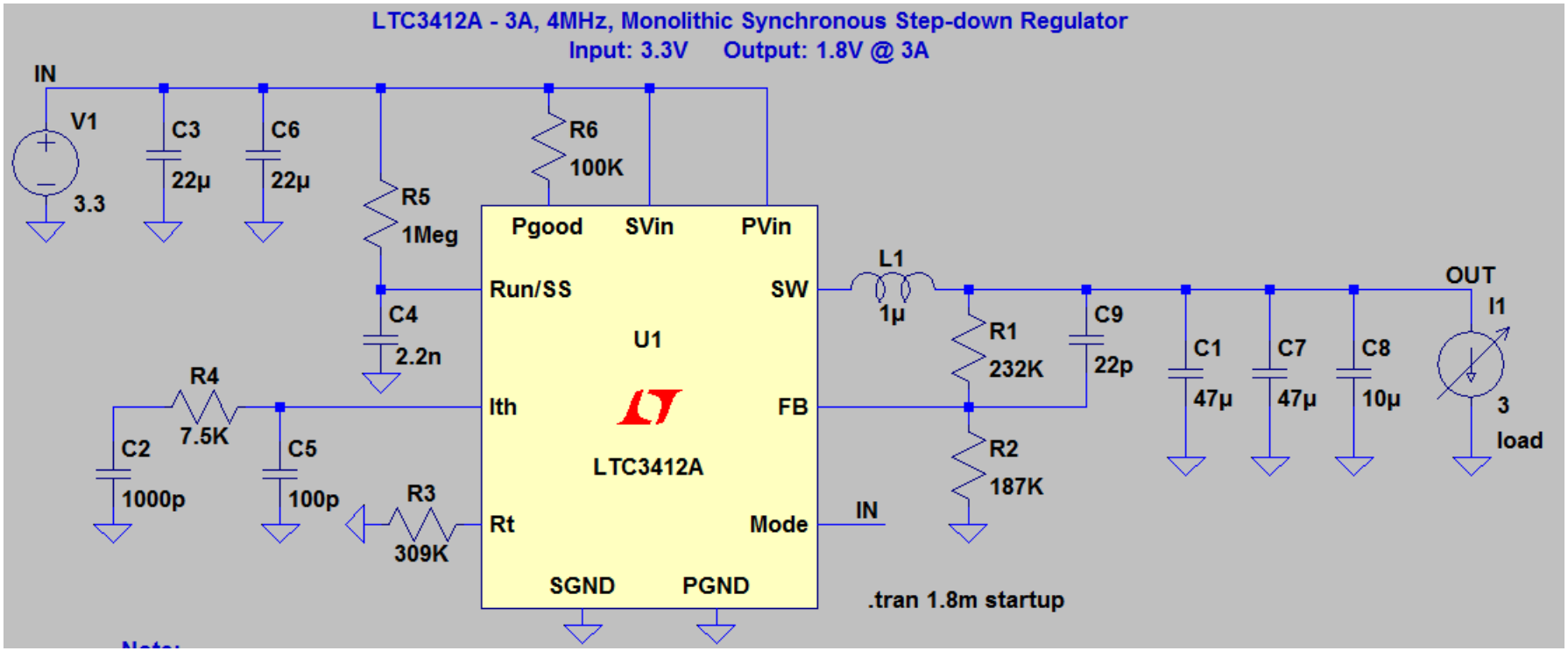
Advantages of Labeling

- To this….

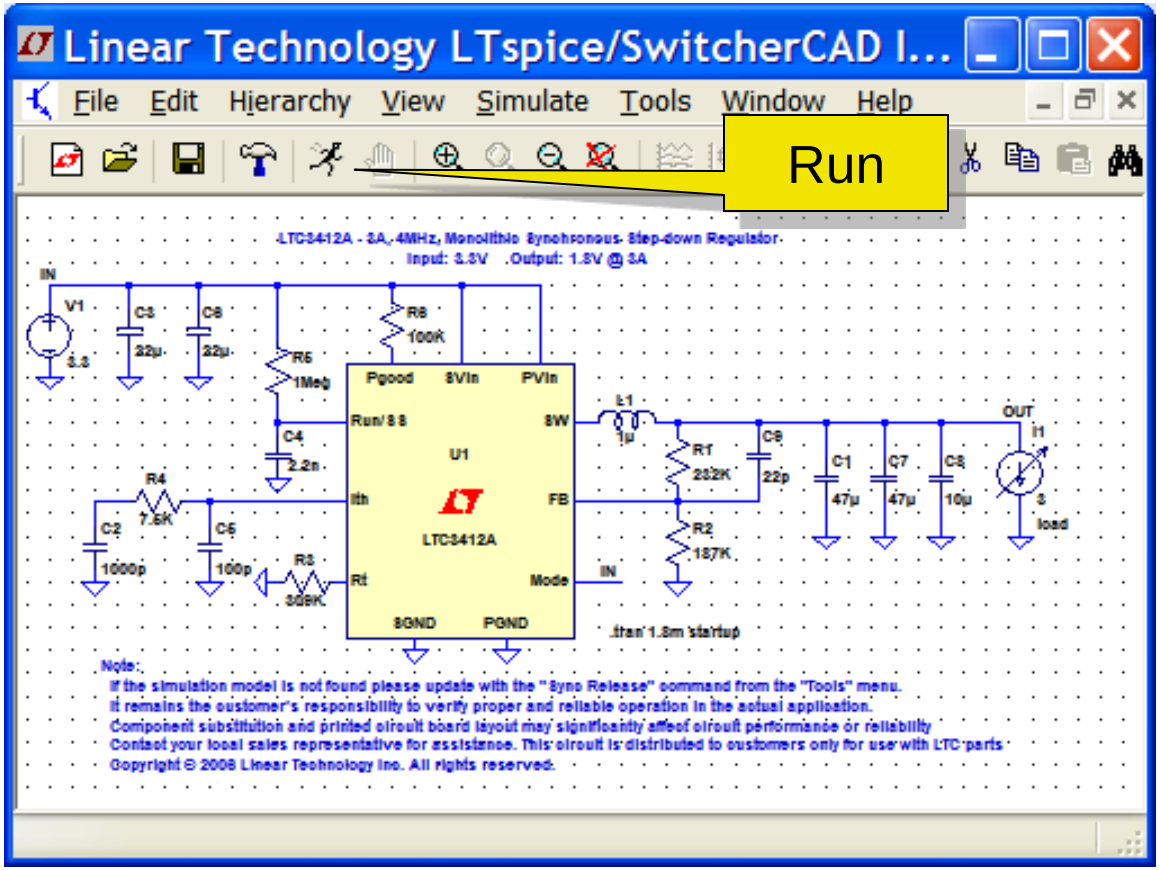


Example

- LTC3412A DC Load.asc



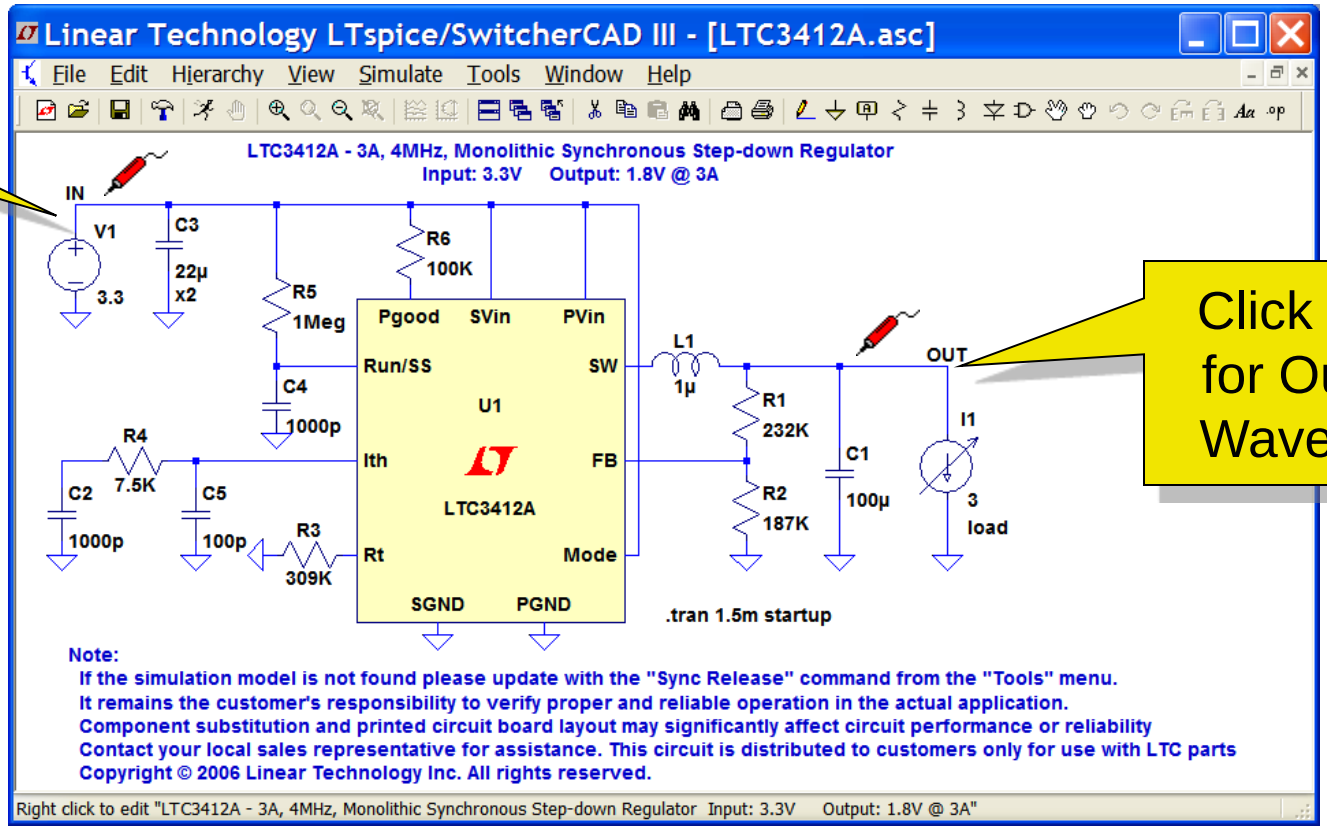
Running a Demo Circuit



- Select the “Running Man” button on the toolbar
  - The Simulation will start and waveform window will open up
  - To view waveforms, please continue to the next page….

Probing a Demo Circuit

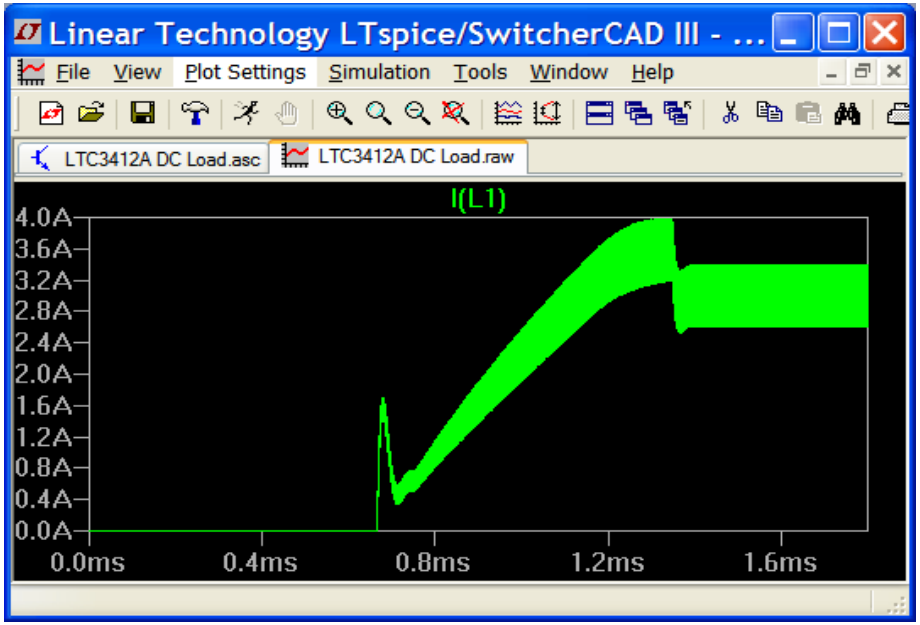
Click Here for Input Waveform



Click Here for Output Waveform

- All Demo Circuits have INs and OUTs clearly labeled to help you quickly select them
- Select the waveform of a node by clicking on IN and OUT

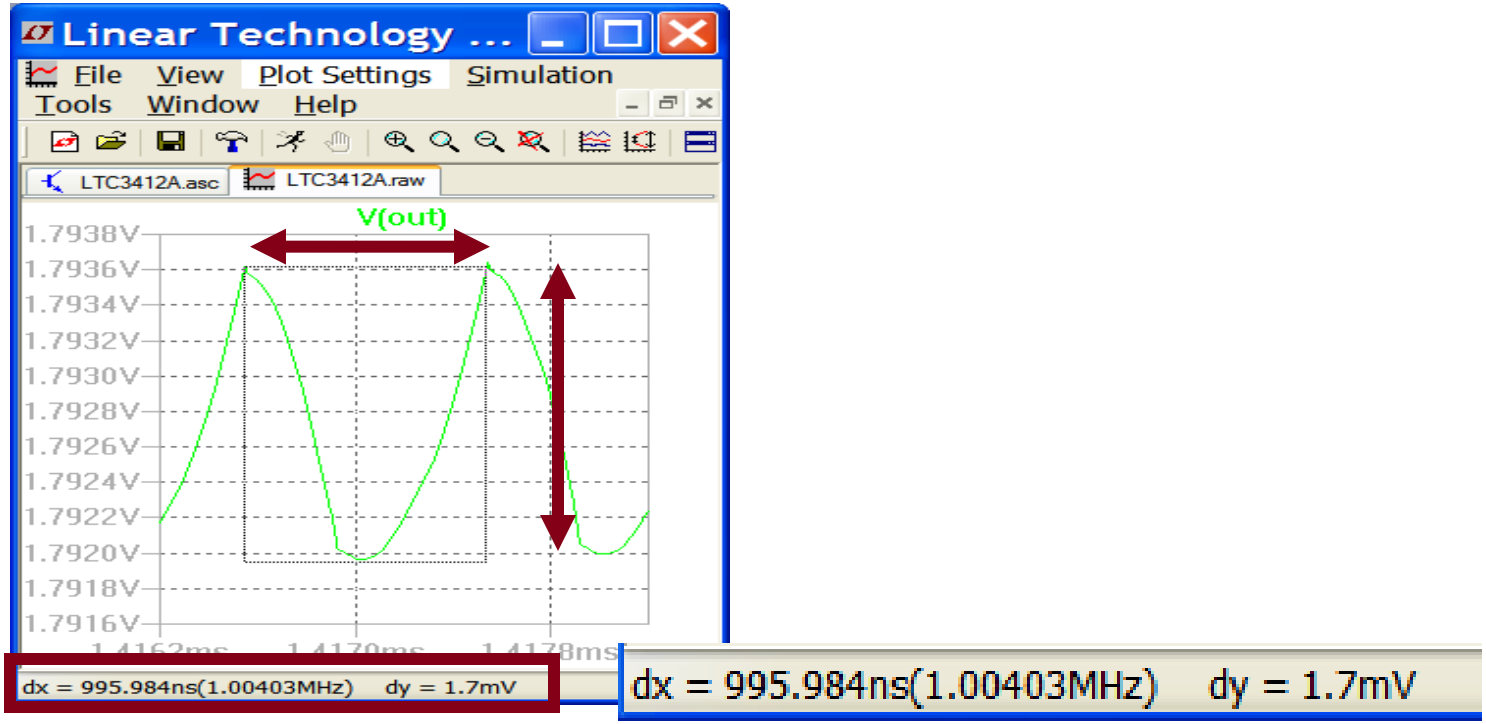
Zooming In and Out on a Waveform



- Using the mouse, click on inductor L1 to display the inductor current waveform
- In the waveform window, use the mouse to zoom in and out
  - Click and drag a box about the region you wish to see drawn larger
- Using the toolbar, click on “Zoom full extents”, to zoom back out

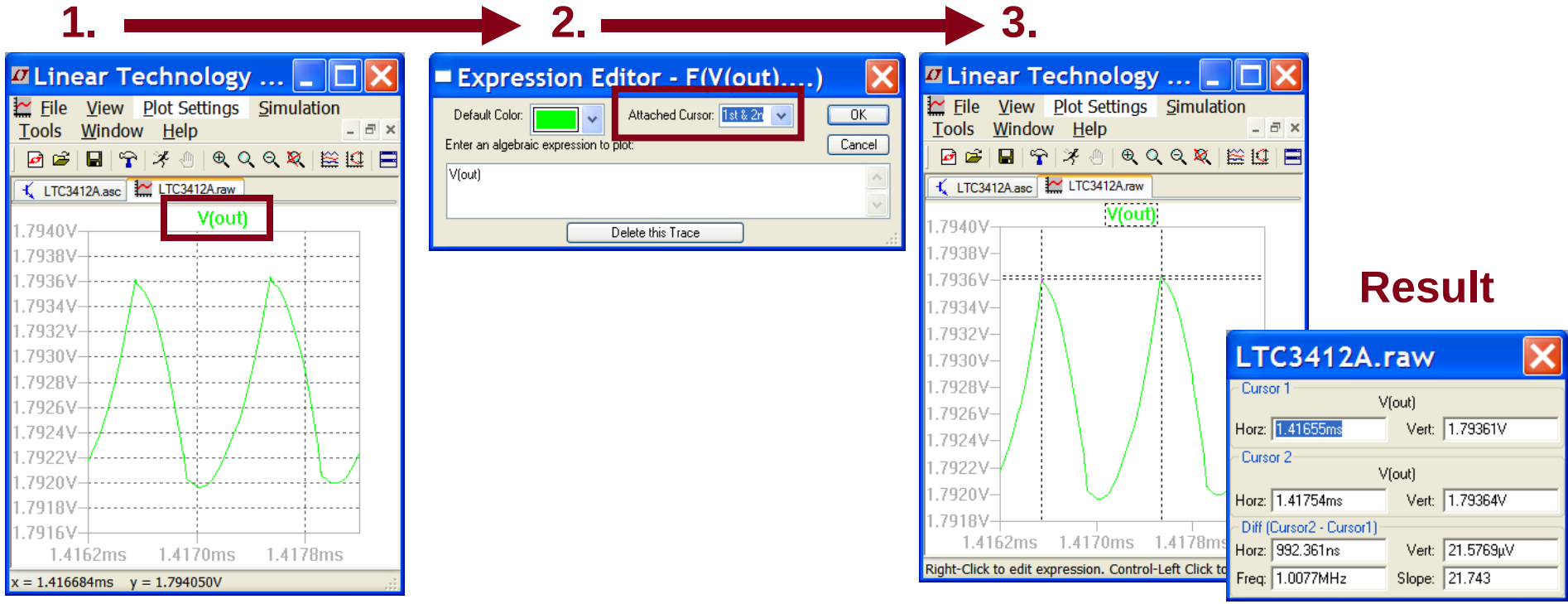
Measuring V, I and Time in the Waveform (Measurement Using Zoom)

1. Drag a box about the region you wish to measure
   - Left-Click, drag, and hold
2. View the lower left corner of the window for the status bar. The dx and dy measurement data is displayed here.
3. Use Undo from the File menu or press “F9”



Measuring V, I and Time in the Waveform (Measurement Using Cursors)

1. Right-Click on the waveform name in the waveform window
2. For “Attached Cursor”, select “1st & 2nd”
3. Position cursors to make desired measurements.

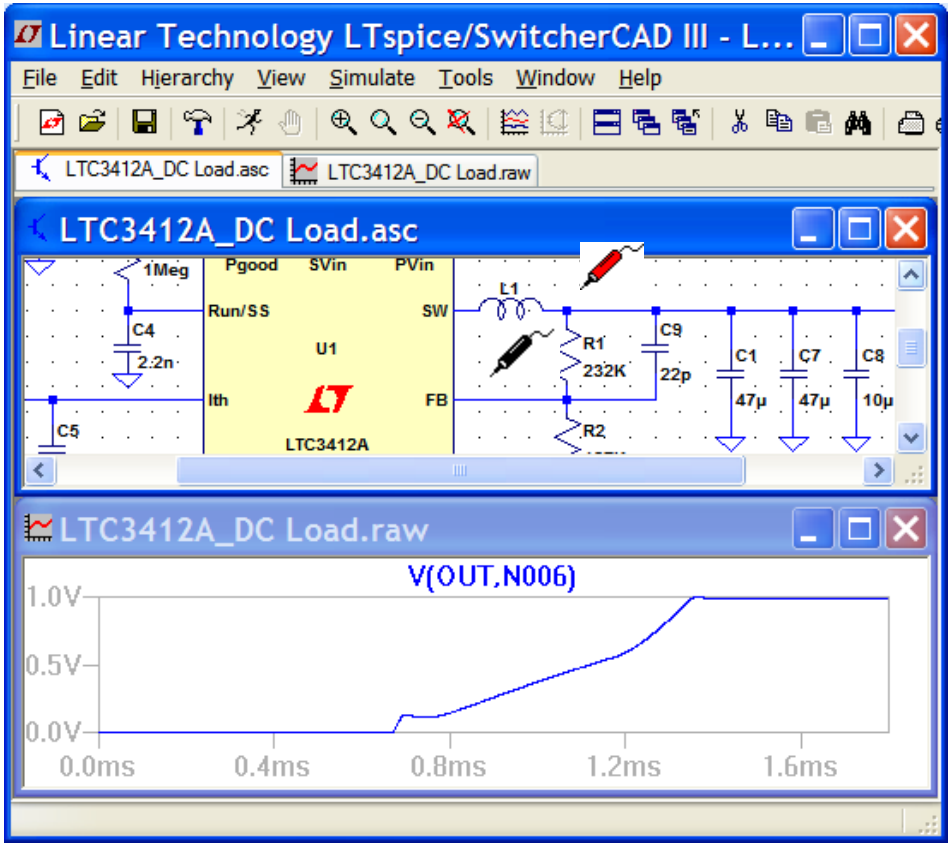


Differential Voltage Measurement

- Click on one node and drag the mouse to another node
  - Red voltage probe at the first node
  - Black probe on the second
- Will produce a differential voltage measurement

Example:

Measure across LTC3412A top resistor in feedback divider

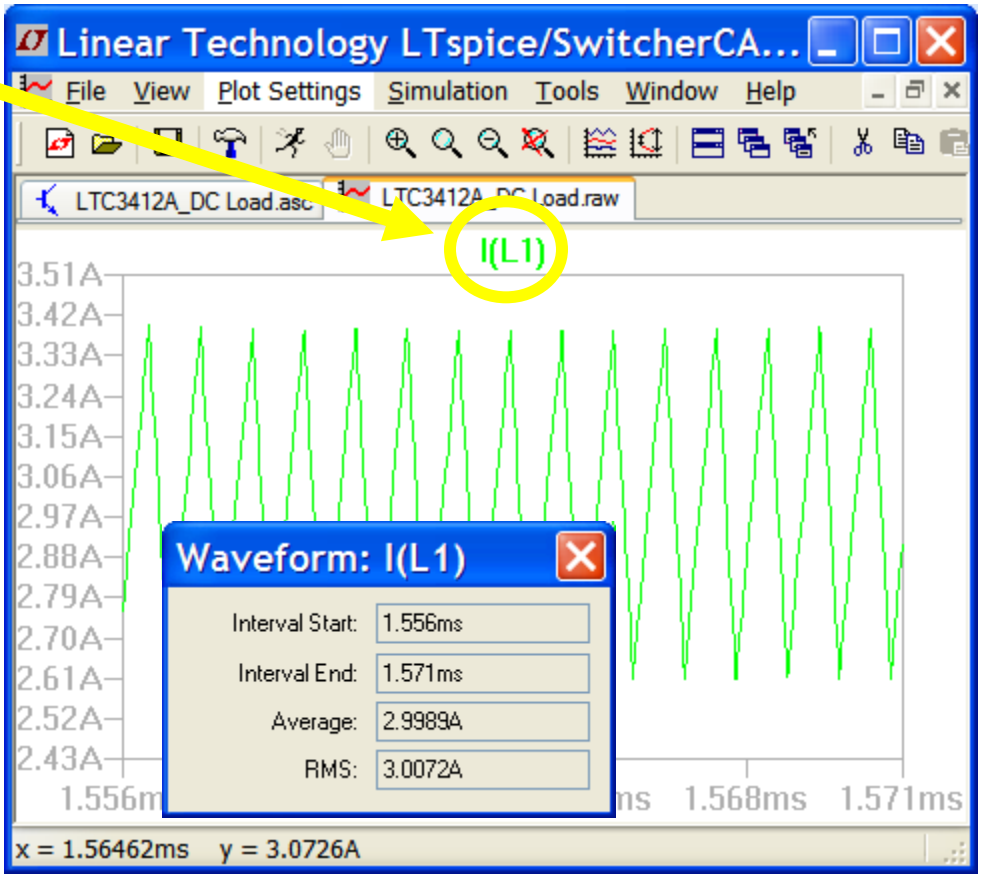


Average & RMS Calculations

- Average & RMS Current, Voltage, or Power Dissipation
- Click on inductor L1 to display the inductor current waveform
  - **Ctrl-Left-Click** the I(L1) trace label in the waveform view

Example:

Measure average and RMS current for inductor in LTC3412A circuit. Zoom in as shown for this waveform.

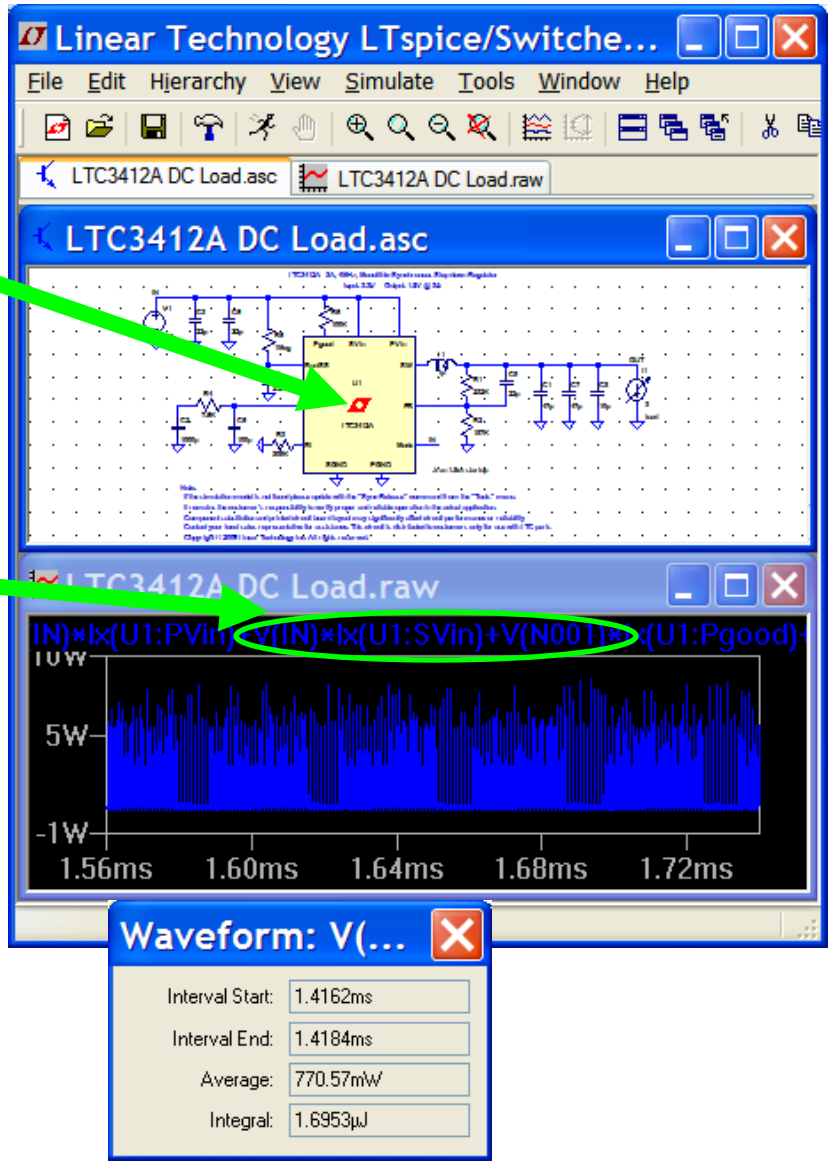


Instantaneous & Average Power Dissipation

- Instantaneous Power Dissipation
  - **Hold down the Alt key and Left-Click** on the symbol of the LTC3412A
  - Waveform is displayed in units of Watts
- Average Power Dissipation
  - **Click, hold, and drag** in the waveform window to display waveform at steady state
  - **Ctrl-Left-Click** on the Power Dissipation Trace Label in the waveform view
  - Waveform summary window will appear which shows power dissipation in the IC

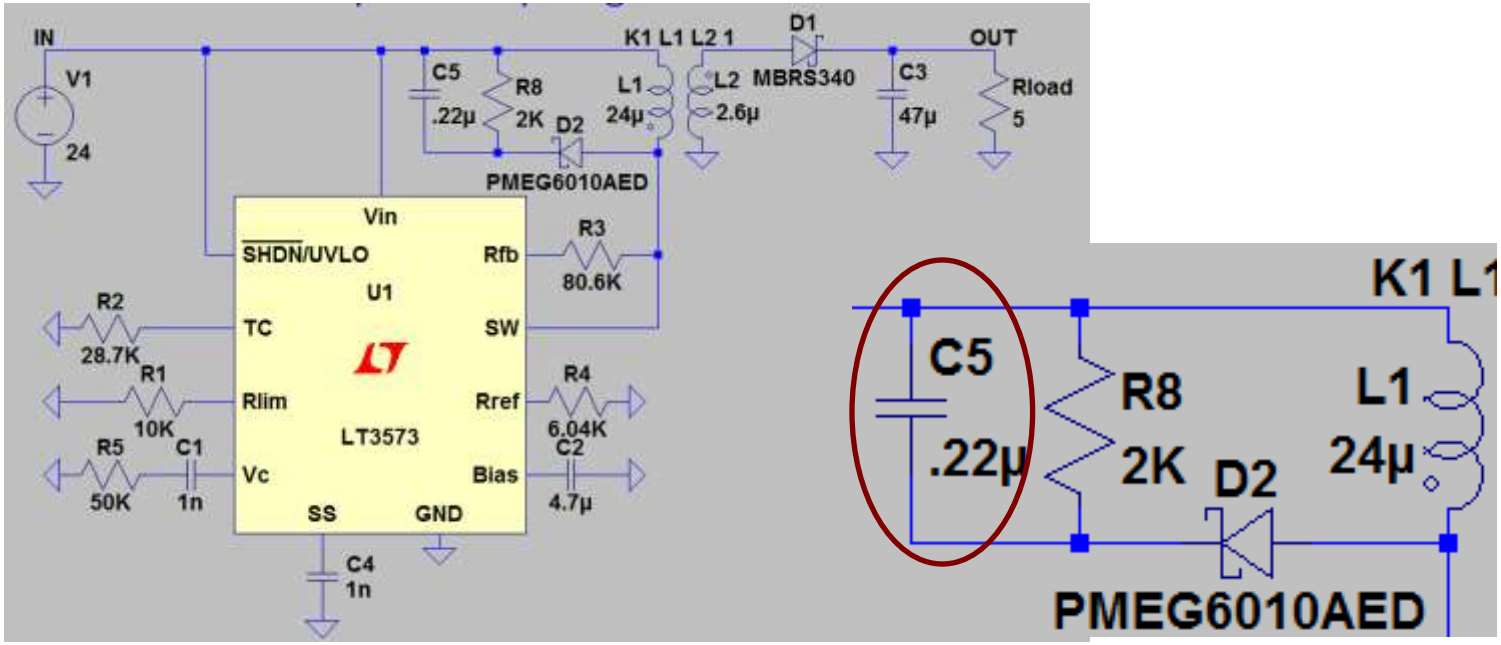
Example:

Measure the power dissipation in the LTC3412A IC



Instantaneous & Average Power Dissipation

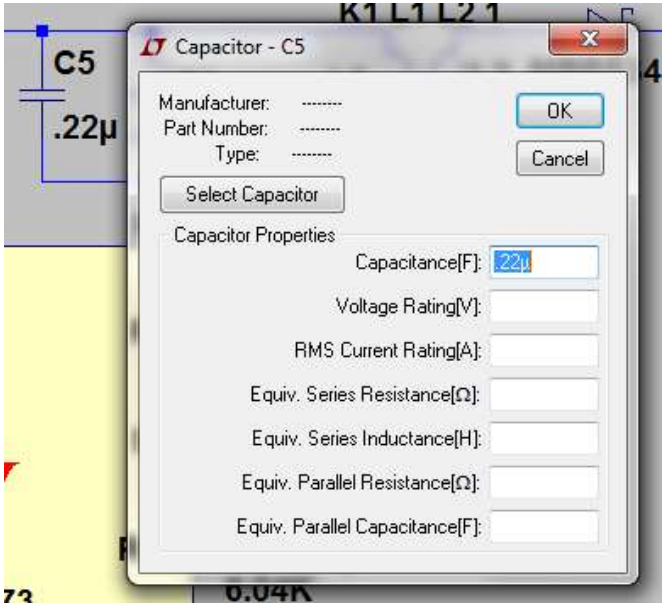
- How about an example of a component you don´t usually think about?
- Ex. Flyback snubber capacitor: barely any power….right?



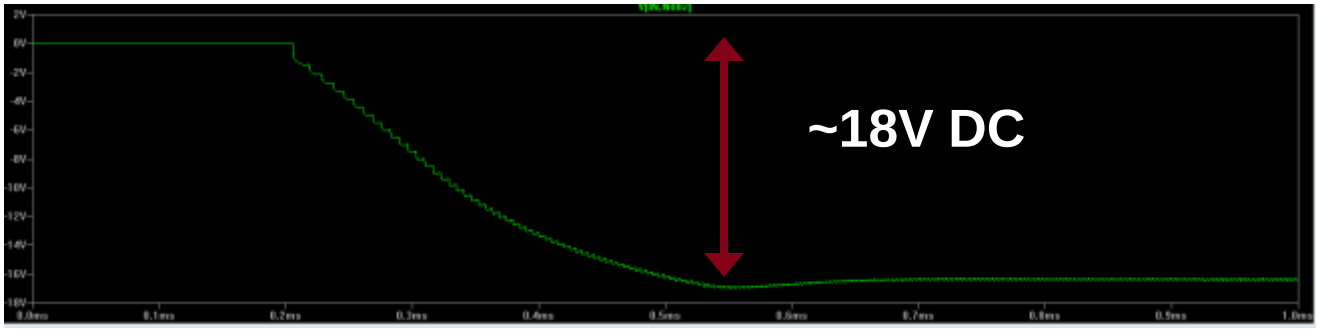
- Customer says “1206 capacitors are expensive or hard to find – can I use 0805? 0603?”

Flyback Snubber Capacitor Power

1. Check the component – an ESR value is needed, none spec´d so far:

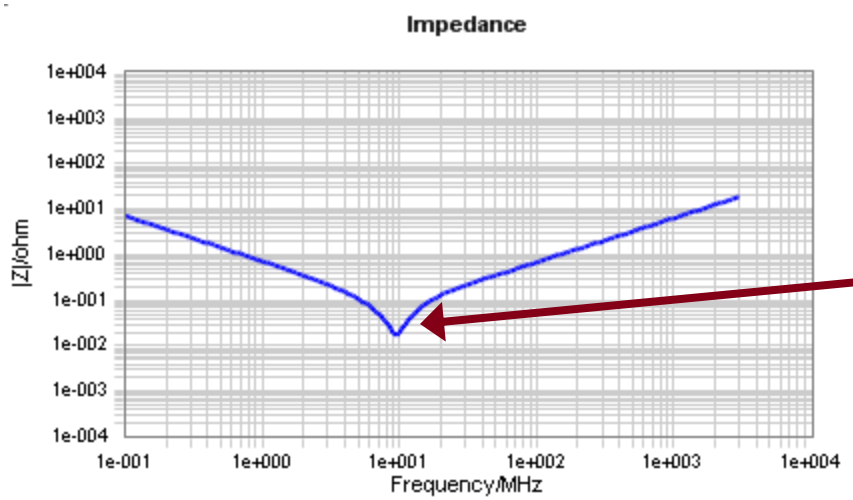


2. Check voltage, 50V rating should do:

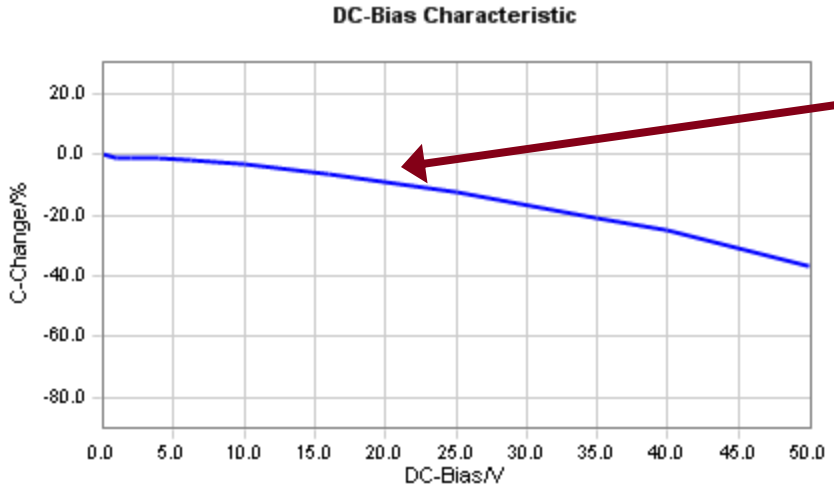


Flyback Snubber Capacitor Power

1. Look at DC bias, ESR for 220 nF, 50V, 1206 MLCC:



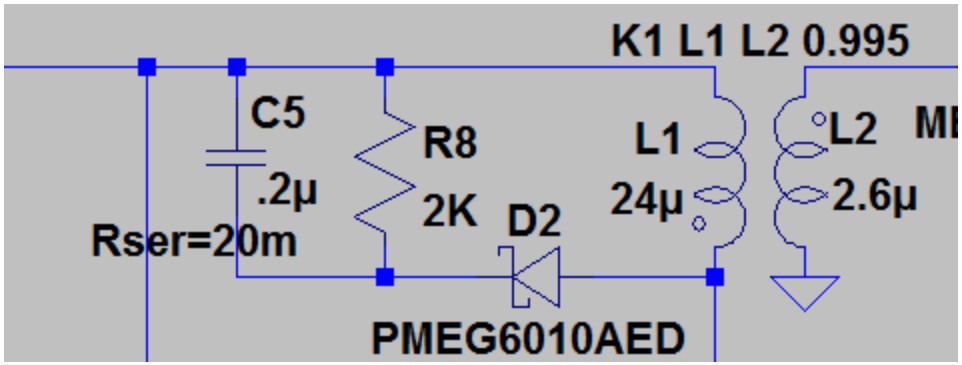
ESR is around 20 mΩ



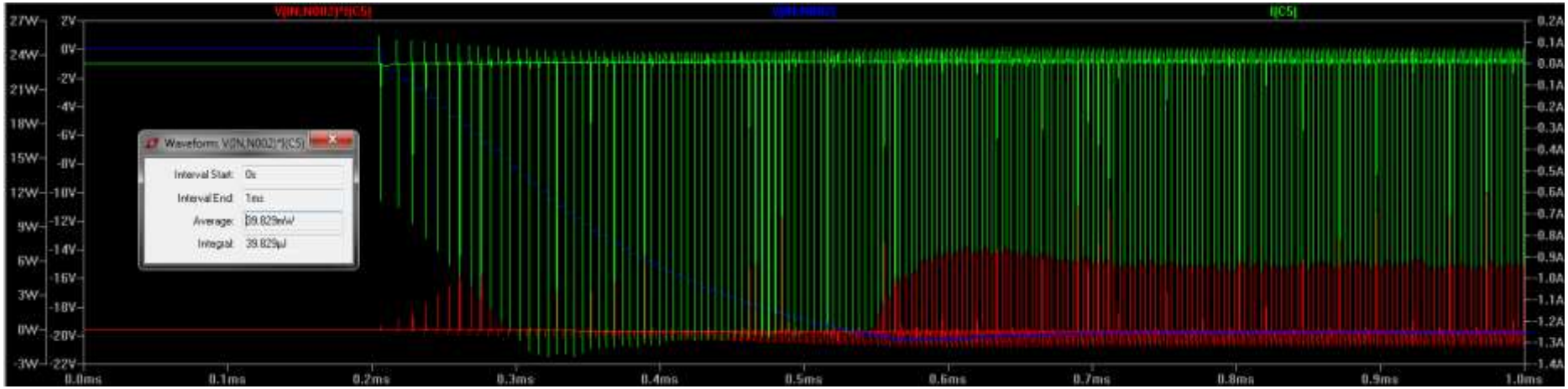
Actual capacitance is around 200 nF (- 10%)

Flyback Snubber Capacitor Power

1. A few, simple changes to the schematic:



2. And a simulation of I, V and P of C5:

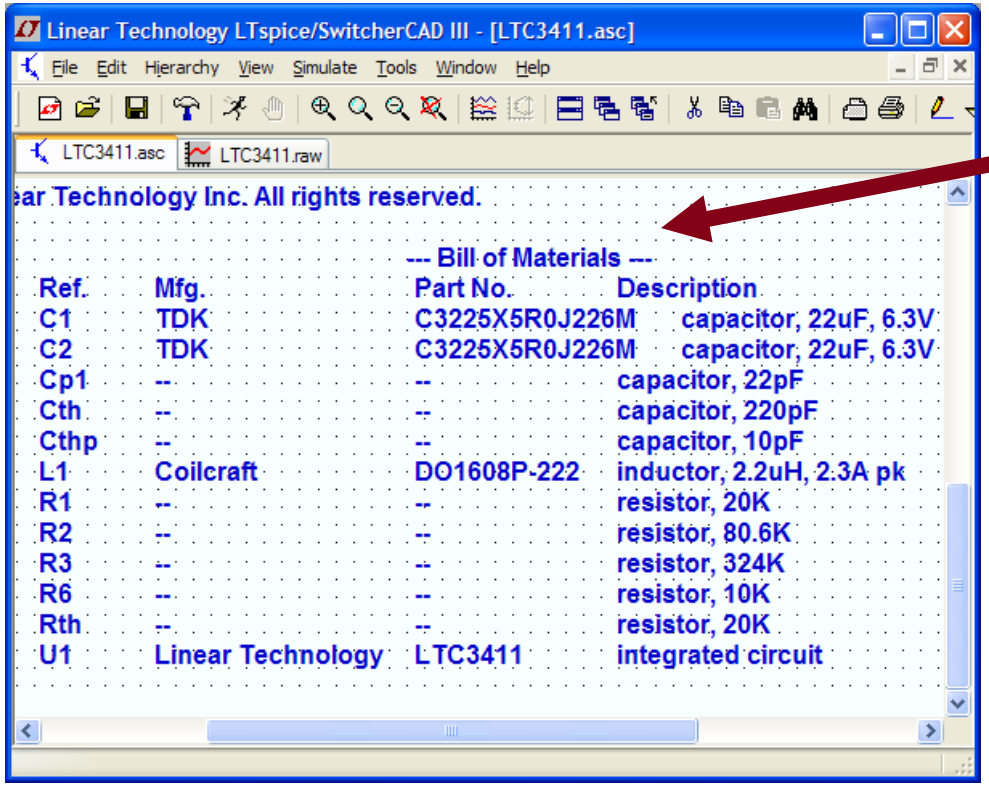


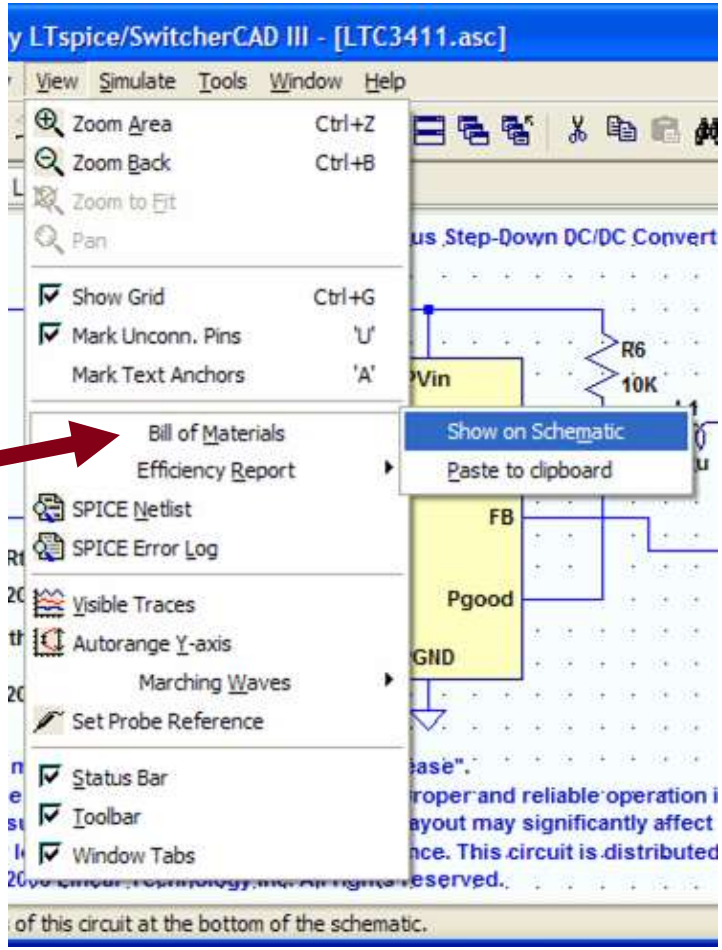
Power in C5 = 40 mW (including startup)

Generating a BOM and Efficiency Report

BOM

- Under View select Bill of Material
  - Displayed on Diagram
  - Paste to Clipboard



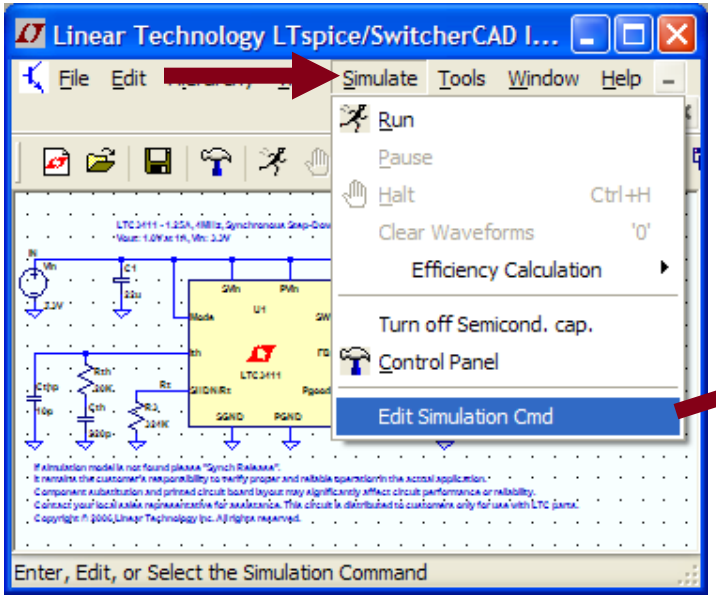


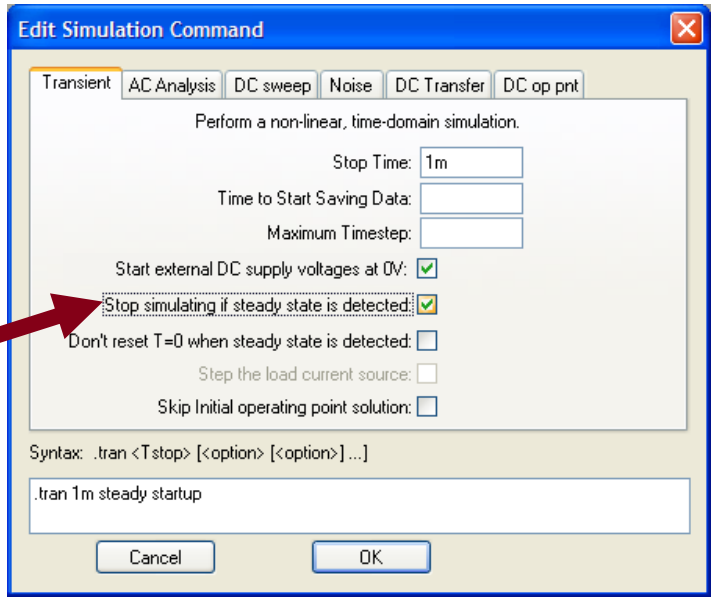
Steps to Calculate Power Supply Efficiency

1. Efficiency will only be calculated in the steady state condition
2. Right-Click the .tran statement on the schematic to bring up the Edit Simulation Command dialog box
3. Check the box "Stop simulating if steady state is detected"
4. Load must be a current source or resistor labeled "Rload"
5. Run the simulation
6. Upon completion select the View dropdown menu, then Efficiency Report, then Show on Schematic
7. Efficiency report will be pasted under the schematic

Computing Efficiency & Dissipation

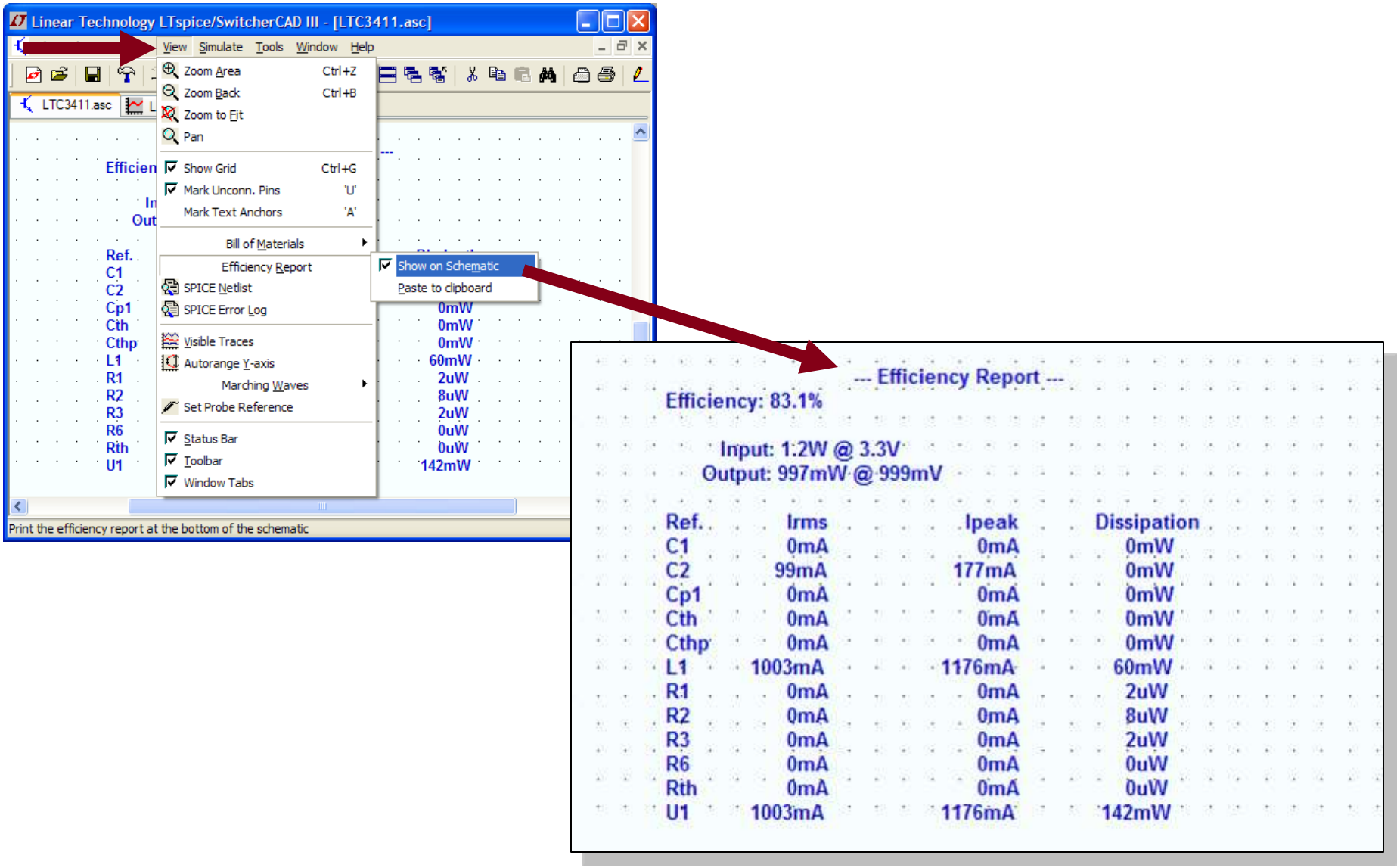
- To compute efficiency of SMPS circuits:
  - Check the "Stop simulating if steady state is detected" on the Edit Simulation Command editor
  - Rerun simulation
  - Use the menu command View=>Efficiency Report





Automatic detection of steady state may not always work – criteria for steady state detection may be too strict or too lenient

Viewing Efficiency Report



Power Supply Efficiency Caveats

- LTspice will not always be able to determine steady state, but this is rare
- Probe the OUT node and verify that it has stabilized
  - If not edit the .tran statement and increase the Stop Time parameter
  - Re-run simulation
- Efficiency must be determined partially by hand for multiple output and/or multiple input supplies
- Right-Clicking any component will report power dissipation

Manual Detection of Steady State

- You can interactively specify steady state in the following manner:
- As soon as the simulation starts, execute menu command **Simulate=>Efficiency Calculation=>Mark Start**.
- The first time you execute this command you tell LTspice you're going to manually specify the integration limits.
- After the circuit looks like it's reached steady-state, execute that command again. That will clear the history and restart the Efficiency Calculation.
- Then, after awhile, as in you see well more than 10 clock cycles, execute **Simulate=>Efficiency Calculation=>Mark End**.

Manual Detection of Steady State

- Each time you execute Simulate=>Efficiency Calculation=>Mark Start you restart the efficiency calculation and clear the waveform history.

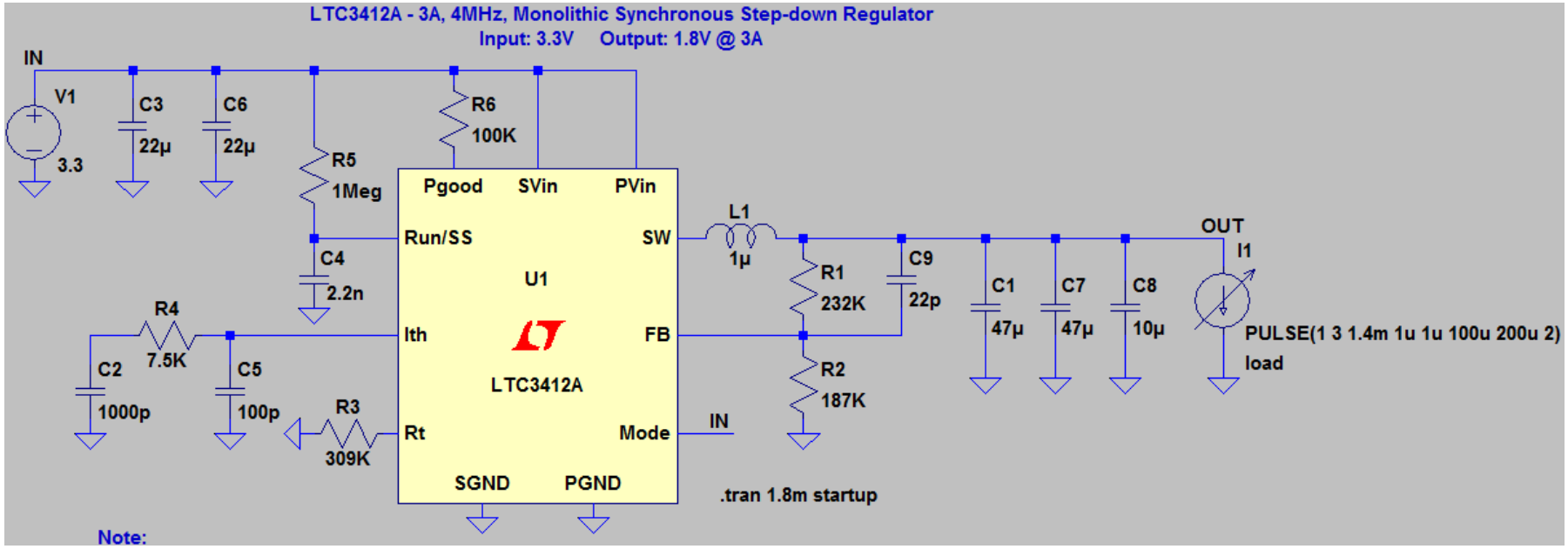
- This is a good method of preventing the data file from becoming too large and slowing down plotting, so it's recommended that you periodically execute Simulate=>Efficiency Calculation=>Mark Start whenever it is clear that you've accumulated substantial data that you don't want to be included in the integration of efficiency.

- Use the .ic directive to specify node voltages and inductor currents to reduce the length of the transient analysis required to find the steady state.

Simulating a Transient Response

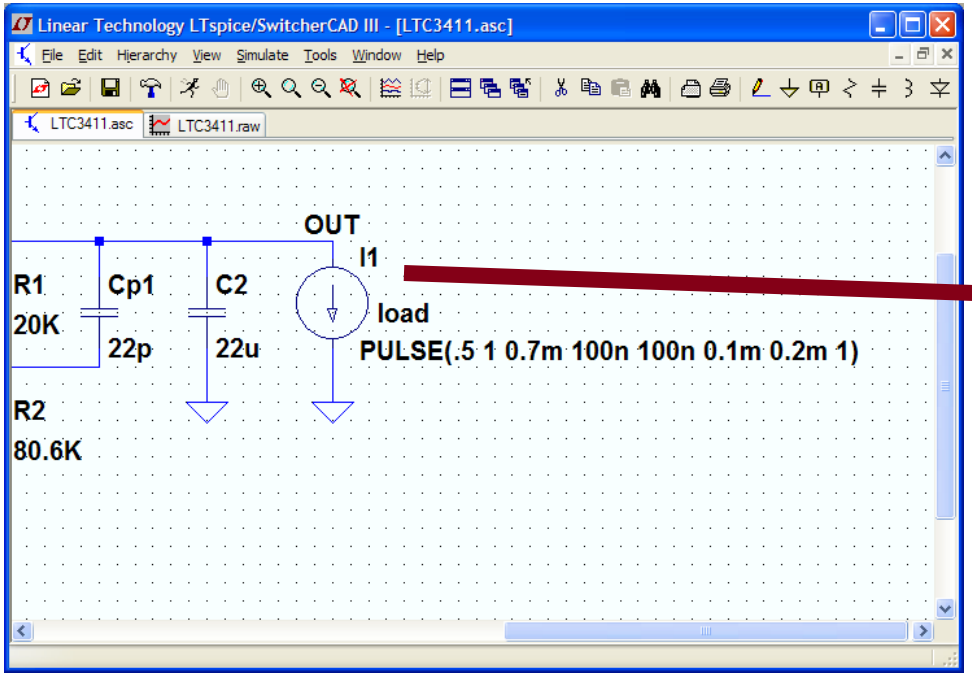
Example

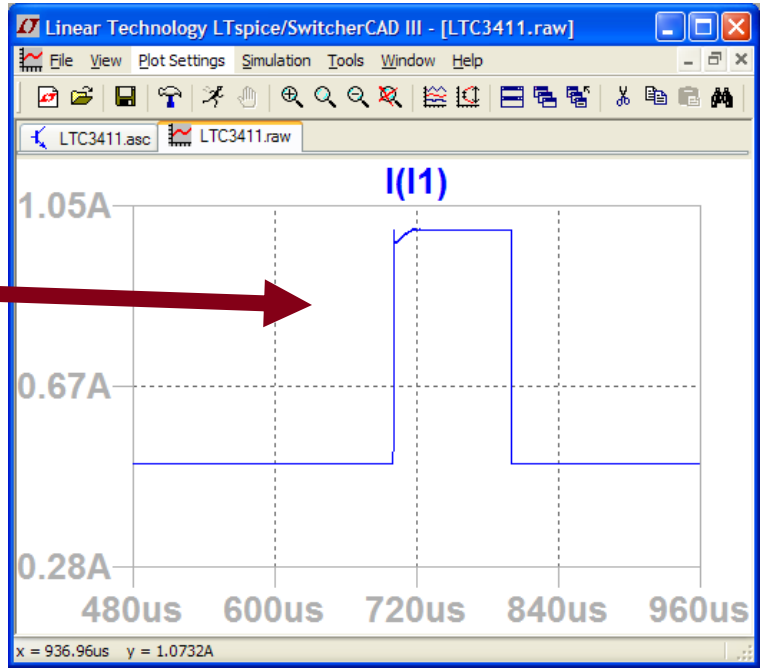
- LTC3412A Pulse Load.asc

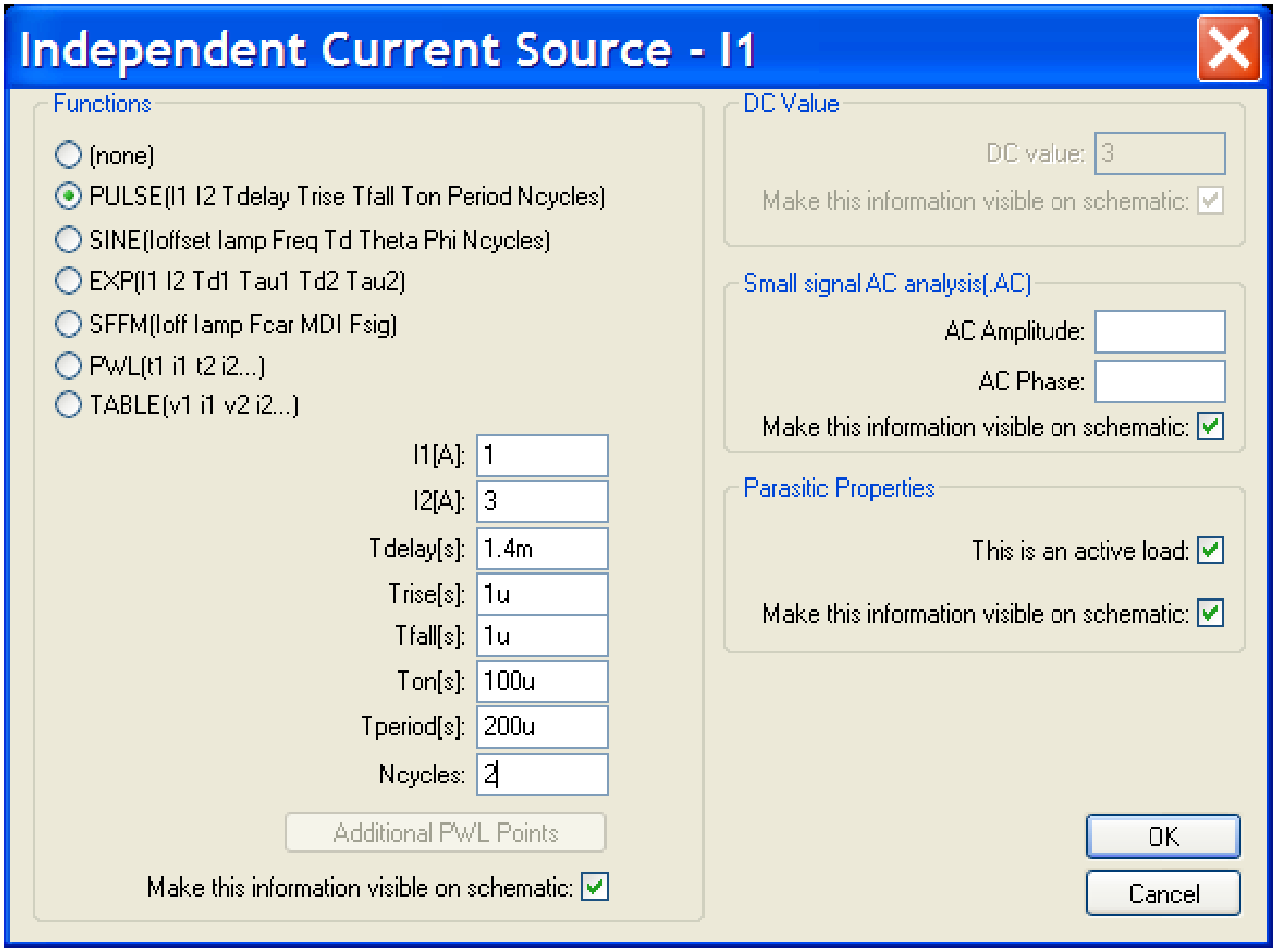


Current Load and Pulse Function

- You can simulate a load with a Resistor or Current load
- In particular the Pulse function in a current load is helpful in transient response analysis
  - Steps a current load from one value to another value

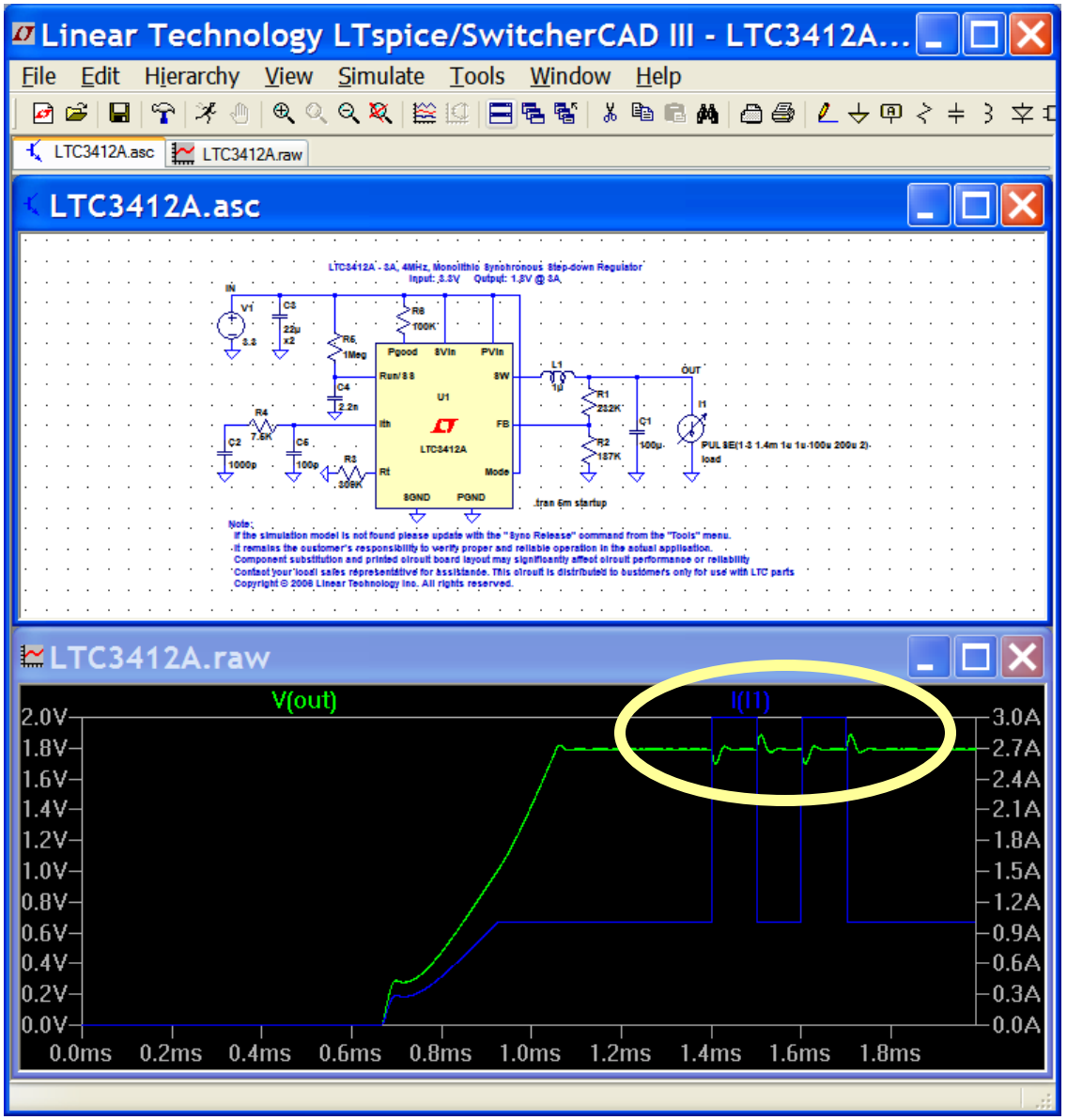





Independent Current Source - I1
Functions
(none)
PULSE(I1 I2 Tdelay Trise Tfall Ton Period Ncycles)
SINE(Ioffset Iamp Freq Td Theta Phi Ncycles)
EXP(I1 I2 Td1 Tau1 Td2 Tau2)
SFFM(Ioff Iamp Fcar MDI Fsig)
PWL(t1 i1 t2 i2...)
TABLE(v1 i1 v2 i2...)
I1[A]: 1
I2[A]: 3
Tdelay[s]: 1.4m
Trise[s]: 1u
Tfall[s]: 1u
Ton[s]: 100u
Tperiod[s]: 200u
Ncycles: 2
Additional PWL Points
Make this information visible on schematic:
DC Value
DC value: 3
Make this information visible on schematic:
Small signal AC analysis(.AC)
AC Amplitude:
AC Phase:
Make this information visible on schematic:
Parasitic Properties
This is an active load:
Make this information visible on schematic:
OK
Cancel

Run the Simulation for Transient Response

- Run the simulation
- Click on the OUT node to display Vout
- Click on the output current load to display Iout
- Notice the presence of the pulse load



.MEASURE

- .MEASURE -- Evaluate User-Defined Electrical Quantities

- There are two basic different types of .MEASURE statements:
- **refer to a point along the abscissa** (the independent variable plotted along the horizontal axis, i.e., the time axis of a .tran analysis)
- **refer to a range over the abscissa**

- The first version, those that point to one point on the abscissa, are used to **print a data value or expression thereof at a specific point or when a condition is met**.

- Syntax: .MEAS[SURE] [AC|DC|OP|TRAN|TF|NOISE] <name>
- + [<FIND|DERIV|PARAM> <expr>]
- + [WHEN <expr> | AT=<expr>]]
- + [TD=<val1>] [<RISE|FALL|CROSS>=[<count1>|LAST]]

- Note one can optionally state the **type** of analysis to which the .MEAS statement applies. This allows you to use certain .MEAS statements only for certain analysis types.

- The **name** is required to give the result a parameter name that can be used in other .MEAS statements.

.MEASURE

Example .MEAS statements that refer to a single point along the abscissa:

.MEAS TRAN res1 FIND V(out) AT=5m
Print the value of V(out) at t=5ms, this will be labeled as res1.

.MEAS TRAN res2 FIND V(out)*I(Vout) WHEN V(x)=3*V(y)
Print the value of the expression V(out)*I(Vout) the first time the condition V(x)=3*V(y) is met. This will be labeled res2.

.MEAS TRAN res3 FIND V(out) WHEN V(x)=3*V(y) cross=3
Print the value of V(out) the third time the condition V(x)=3*V(y) is met. This will be labeled res3.

.MEAS TRAN res4 FIND V(out) WHEN V(x)=3*V(y) rise=last
Print the value of V(out) the last time the condition V(x)=3*V(y) is met when approached as V(x) increasing with respect to 3*V(y). This will be labeled res4.

.MEASURE

.MEAS TRAN res5 FIND V(out) WHEN V(x)=3*V(y) cross=3 TD=1m
Print the value of V(out) the third time the condition V(x)=3*V(y) is met, but don't start counting until the time has elapsed to 1ms. This will be labeled res5.

.MEAS TRAN res6 PARAM 3*res1/res2
Print the value of 3*res1/res2. This form is useful for printing expressions of other .meas statement results. It's not intended that expressions based on direct simulation data, such as V(3), are present in the expression to be evaluated, but if they are, the data is taken from the last simulated point. The result will be labeled res6.

Note that the above examples, while referring to one point along the abscissa, the requested result is based on ordinate data(the dependent variables). If no ordinate information is requested, then the .MEAS statement prints point on the abscissa that the measurement condition occurs:

.MEAS TRAN res6 WHEN V(x)=3*V(y)
Print the first time the condition V(x)=3*V(y) is met. This will be labeled res6.

.MEASURE

The other type of .MEAS statement refers to a **range over the abscissa**.

Syntax: .MEAS [AC|DC|OP|TRAN|TF|NOISE] <name>
+ [<AVG|MAX|MIN|PP|RMS|INTEG> <expr>]
+ [TRIG <lhs1> [[VAL]=]<rhs1>] [TD=<val1>]
+ [<RISE|FALL|CROSS>=<count1>]
+ [TARG <lhs2> [[VAL]=]<rhs2>] [TD=<val2>]
+ [<RISE|FALL|CROSS>=<count2>]

The range over the abscissa is specified with the points defined by "**TRIG**" and "**TARG**".
The TRIG point defaults to the start of the simulation if omitted.
Similarly, the TARG point defaults to the end of simulation data.
If all three of the TRIG, TARG, and the previous WHEN points are omitted, then the .MEAS statement operates over the entire range of data.
The types of measurement operations that can be done over an interval are

Keyword	Operation perform over interval
AVG	Compute the average of <expr>
MAX	Find the maximum value of <expr>
MIN	Find the minimum value of <expr>
PP	Find the peak-to-peak of <expr>
RMS	Compute the root mean square of <expr>
INTEG	Integrate <expr>

.MEASURE

If no measurement operation is specified, the result of the .MEAS statement is the distance along the abscissa between the TRIG and TARG points.

Below are example interval .MEAS statements:

.MEAS TRAN res7 AVG V(NS01)
+ TRIG V(NS05) VAL=1.5 TD=1.1u FALL=1
+ TARG V(NS03) VAL=1.5 TD=1.1u FALL=1
Print the value of average value of V(NS01) from the 1st fall of V(NS05) to 1.5V after 1.1us and the 1st fall of V(NS03) to 1.5V after 1.1us. This will be labeled res7.

.MEAS AC res8 when mag(V(out))=1/sqrt(2)
The result res8 is the frequency that the magnitude of V(out) is equal to 0.7071067811865475.

.MEASURE

Also, the result of a .MEAS statement can be used in another .MEAS statement. In this example, the 3dB bandwidth is computed:

.MEAS AC tmp max mag(V(out)); find the peak response and call it "tmp"
.**MEAS AC BW trig mag(V(out))=tmp/sqrt(2) rise=1 targ mag(V(out))=tmp/sqrt(2) fall=last**
Print the difference in frequency between the two points 3dB down from peak response.

NOTE: The data from a .AC analysis is complex and so are the .measurement statements results. However, the equality refers only to the real part of the complex number, that is, "mag(V(out))=tmp/sqrt(2)" is equivalent to Re(mag(V(out)))=Re(tmp/sqrt(2)).

The AVG, RMS, and INTEG operations are different for .NOISE analysis than the analysis types since the noise is more meaningfully integrated in quadrature over frequency. Hence AVG and RMS both give the RMS noise voltage and INTEG gives the integrated total noise.

Hence, if you add the SPICE directives
.MEAS NOISE out_totn INTEG V(onoise)
.MEAS NOISE in_totn INTEG V(inoise)
to a .noise analysis, the total integrated input and output referenced rms noise will be printed in the .log file.

.MEASURE

.MEAS statements are done in post processing after the simulation is completed.
This allows you to write a script of .MEAS statements and execute them on a dataset.
To do this, make the waveform window the active window and execute menu command **File=>Execute .MEAS Script**.

Another consequence of .MEAS statements being done in post processing after the simulation is that the accuracy of the .MEAS statement output is limited by the accuracy of the waveform data after compression.
You may want to adjust the **compression settings** for more precise .MEAS statement output.

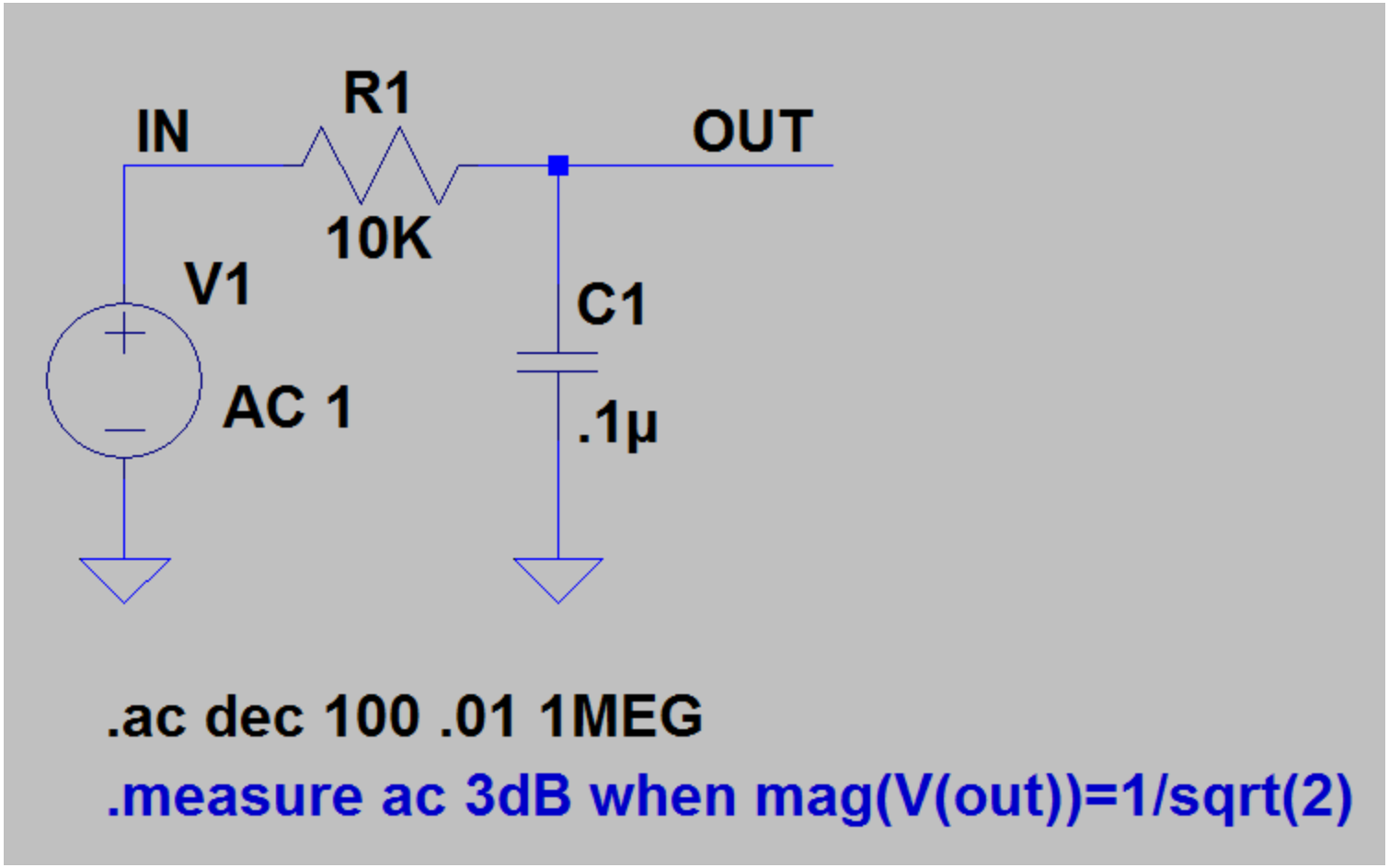
Note when testing a condition such as "when <cond1> = <cond2>" you will want the condition to go through the equality, not must meet it. This relates to the fact that floating point equality should never be required due to the finite precision used in storing numbers.

AC Analysis Overview

- Performs small signal AC analysis linearized about the DC operating point
- Useful for analysis of filters, networks, power supply stability analysis, and noise considerations

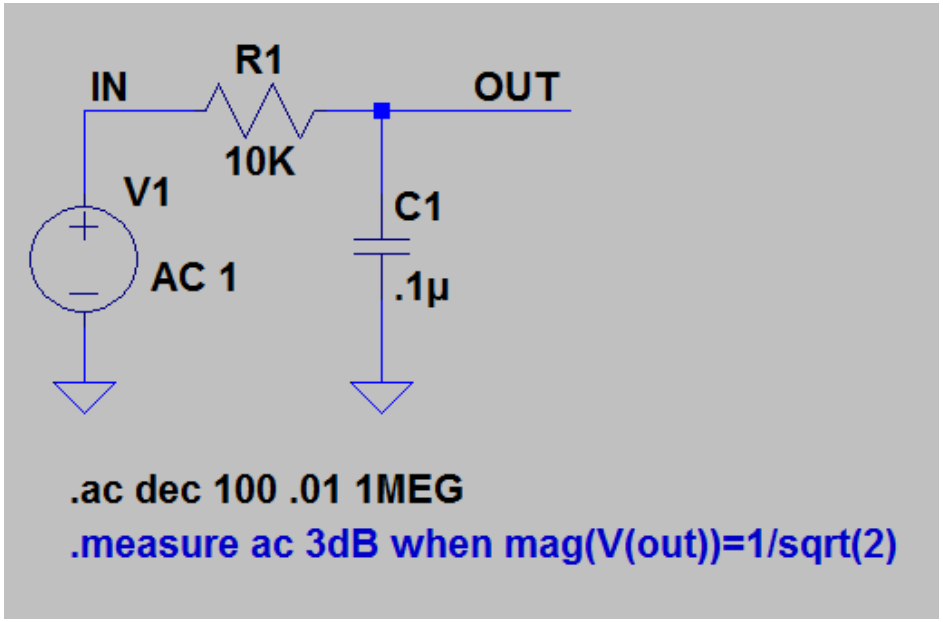
Example

- RC Filter AC Analysis.asc



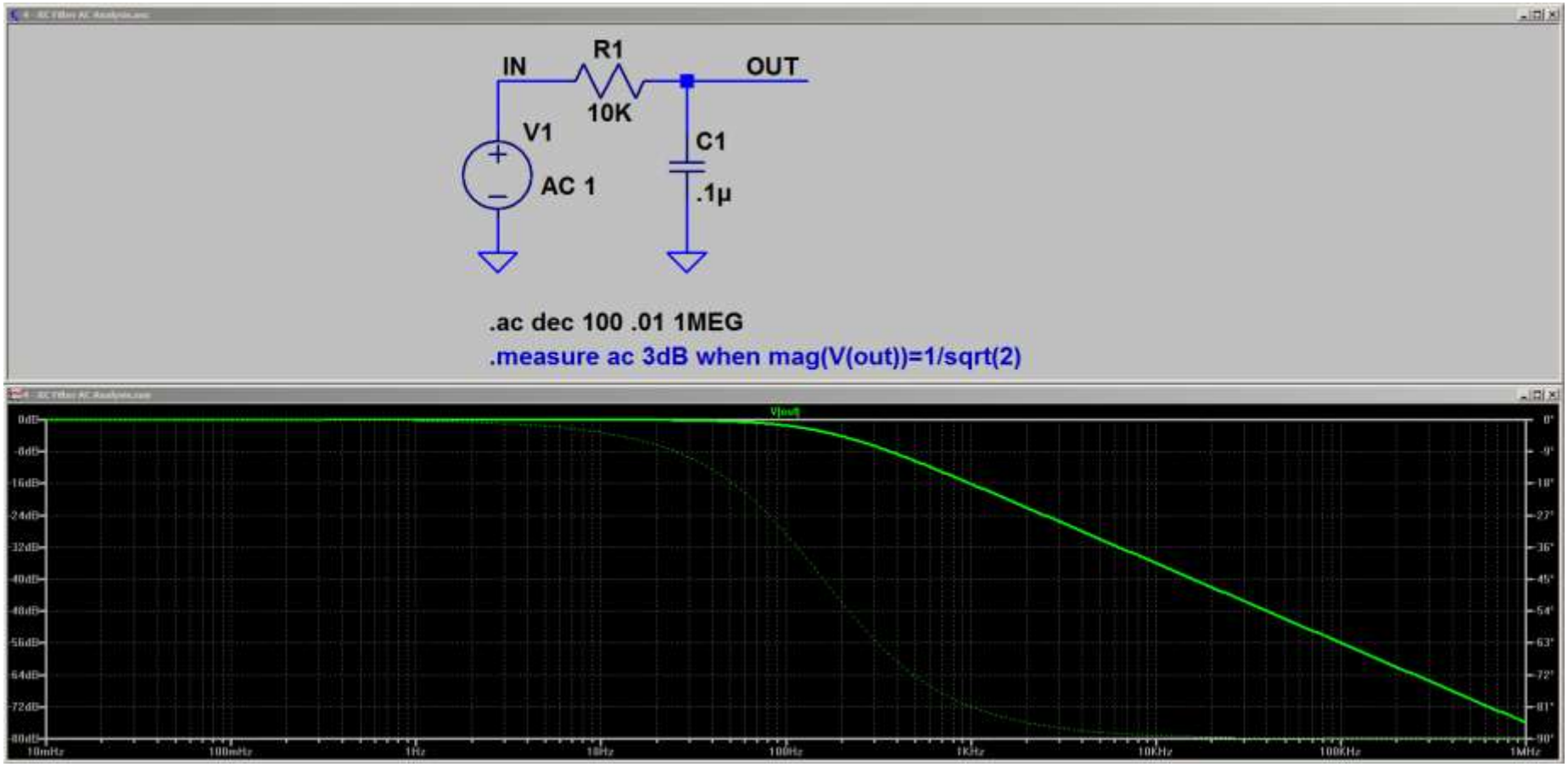
Simulating AC Analysis – RC Filter

- Single pole filter using RC network
- Syntax: .ac <oct, dec, lin> <Nsteps> <StartFreq> <EndFreq>
- Example: RC network .ac dec 100 .01 1MEG



-3dB point:
1/(2*pi*R*C) = 159Hz

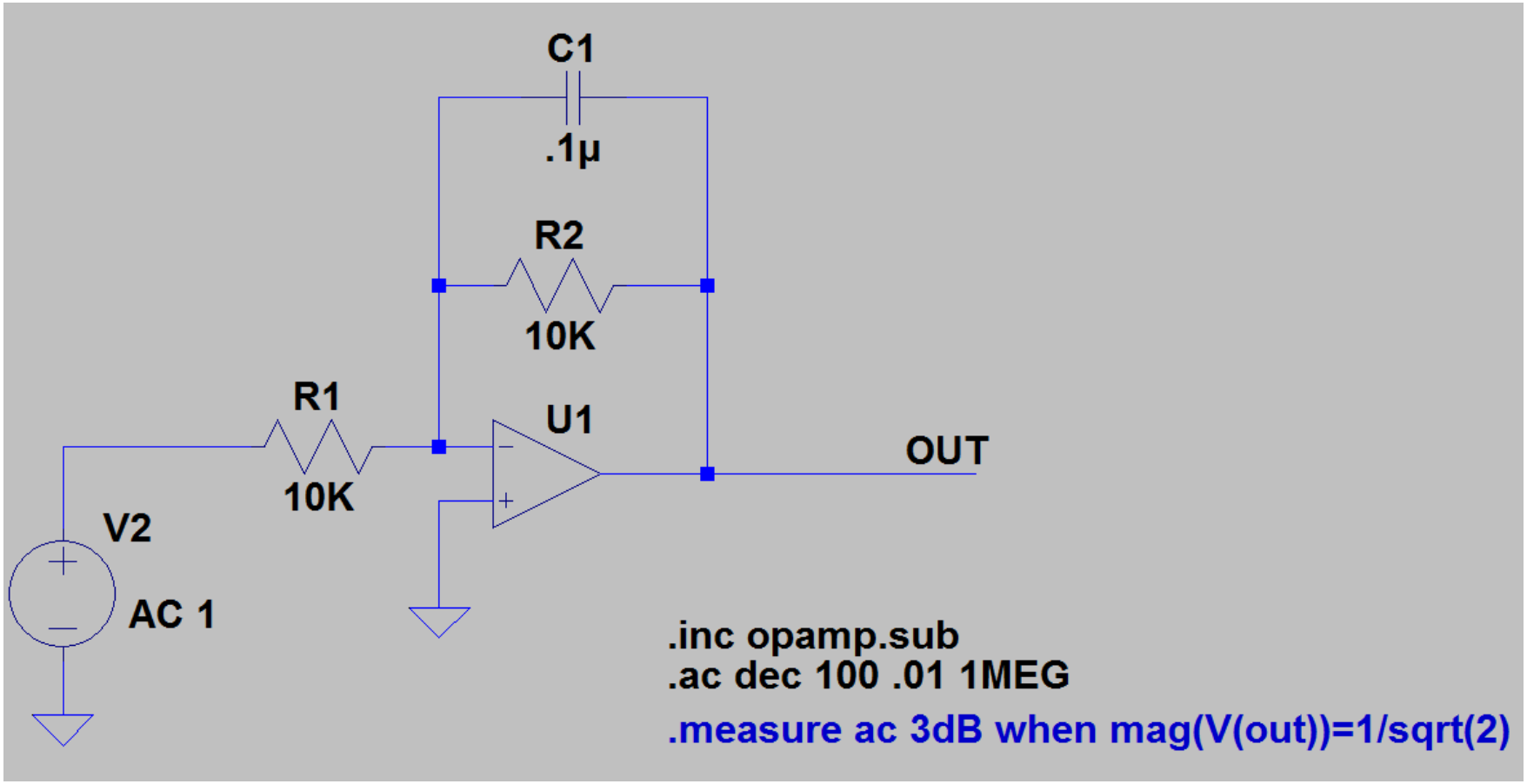
Simulating AC Analysis – RC Result



Frequency

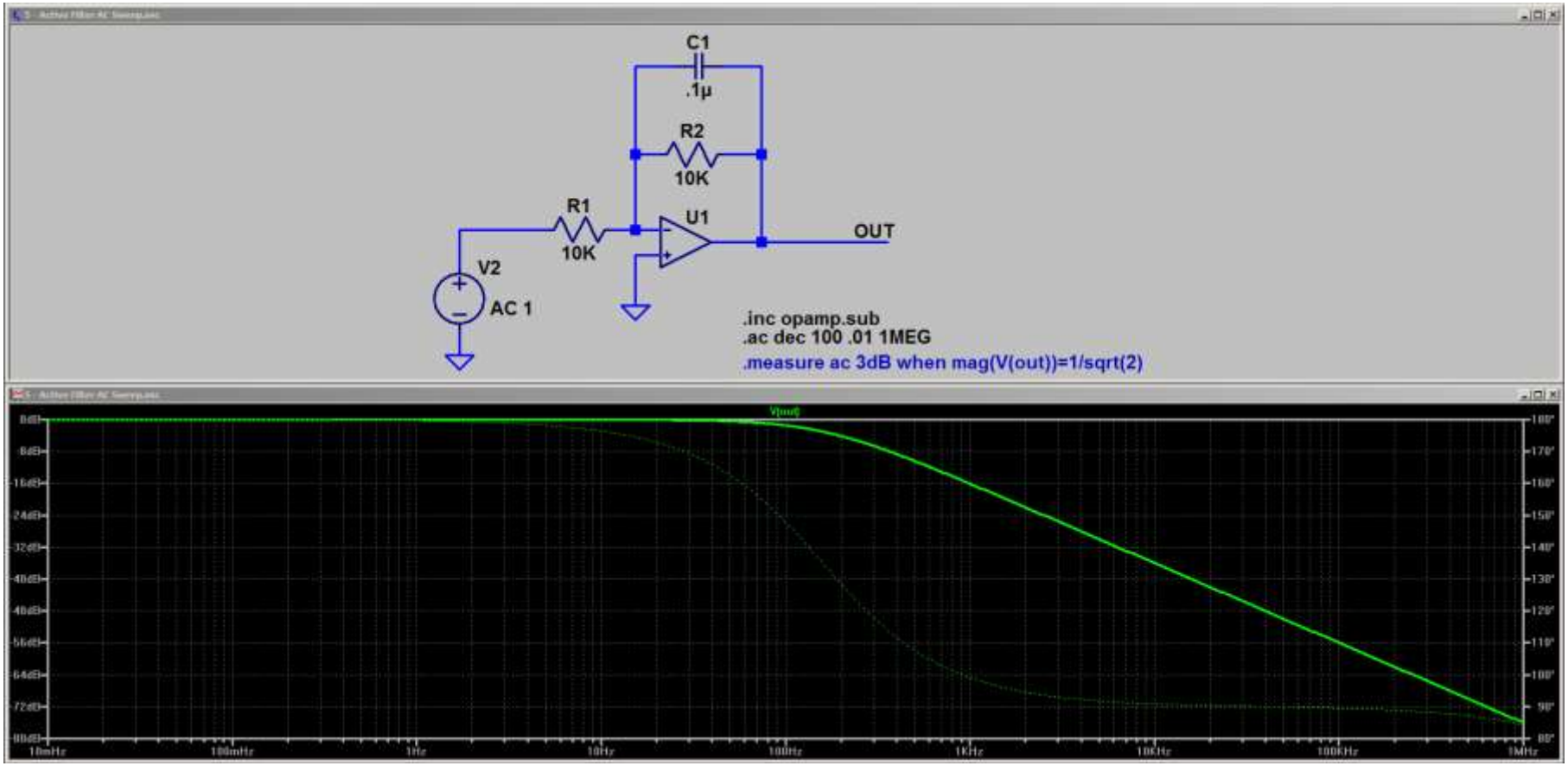
Example

- Active Filter AC Sweep.asc



Simulating AC Analysis – Active Filter

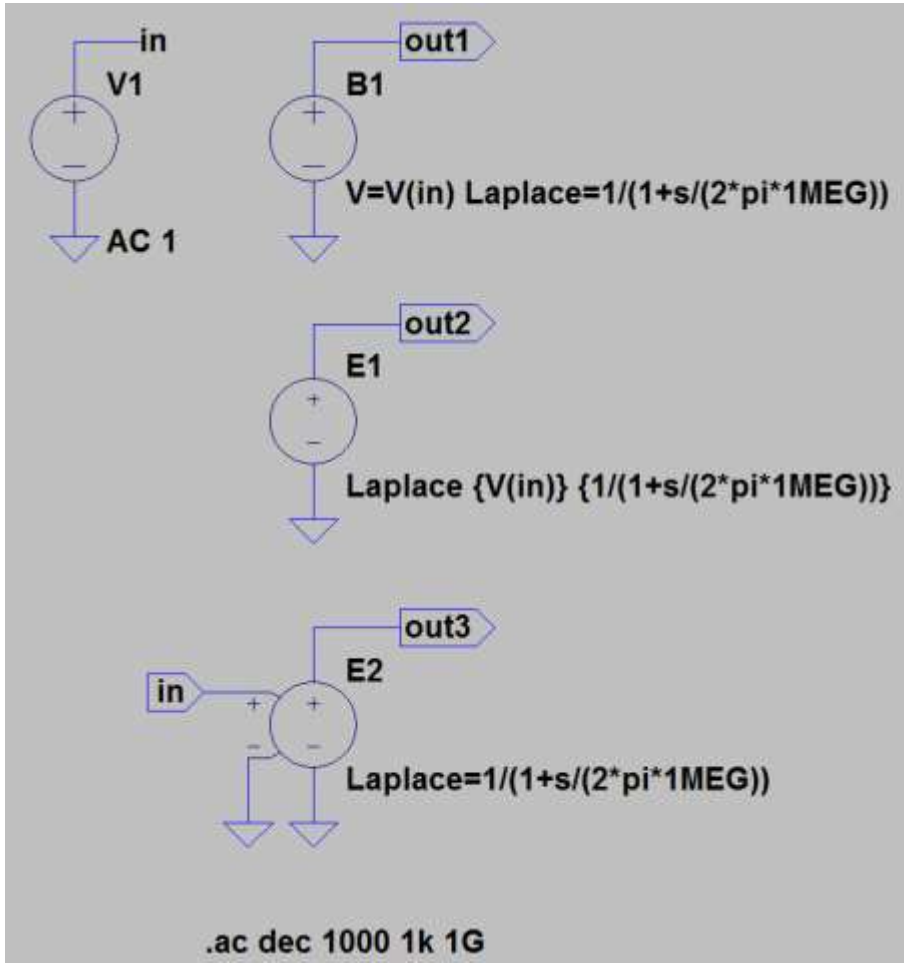
- Single pole active filter using an opamp



Frequency

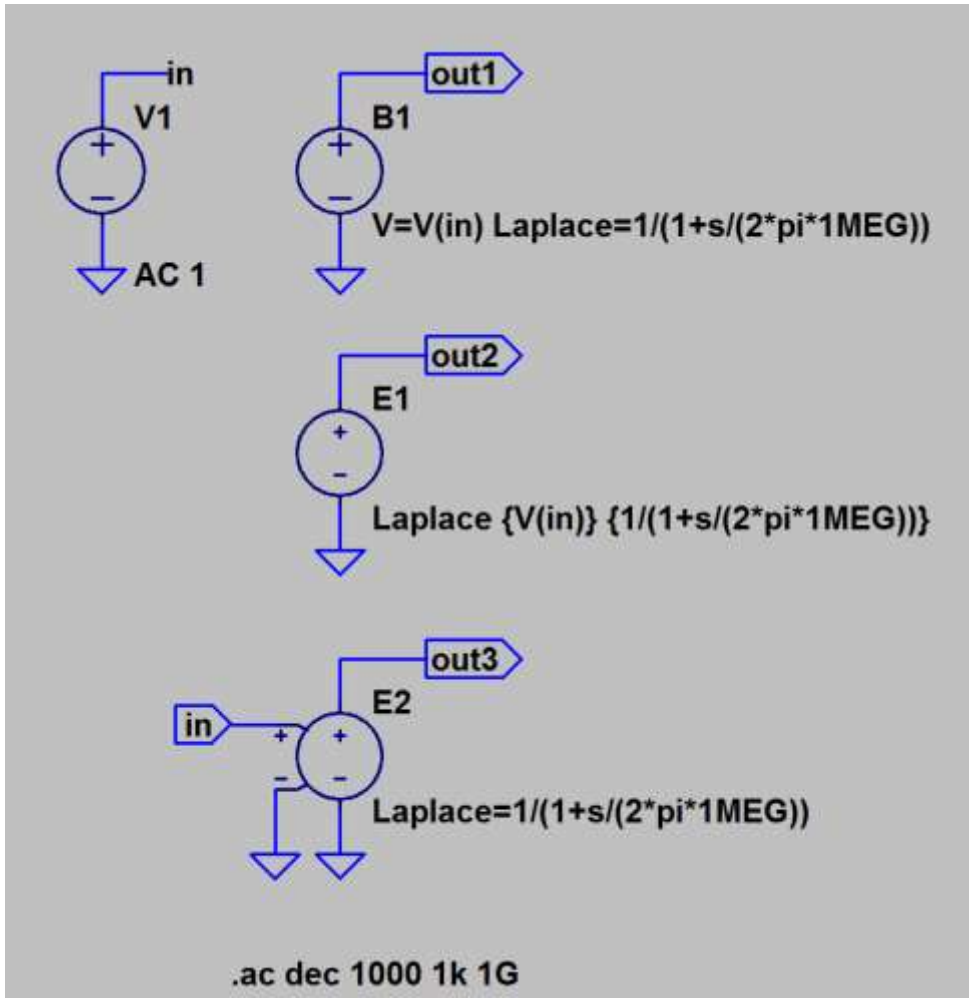
Example

- Laplace syntax.asc



Laplace Transfer Function

- Syntax for LAPLACE
- All 3 examples model the same lowpass filter with fc = 1MHz



Noise Analysis

Noise Analysis Parameters

- **.NOISE** -- Perform a Noise Analysis

- This is a **frequency domain analysis**, that computes the noise due to Johnson (thermal), shot (quantum) and flicker (1/f) noise.
- The output data is **noise spectral density**.

- Syntax: .NOISE V(<out>[,<ref>]) <src> <oct, dec, lin>+ <Nsteps> <StartFreq> <EndFreq>
- **V(<out>**[,<ref>]**)** is the node at which the total output noise is calculated.
- It can be expressed as V(n1, n2) to represent the voltage between two nodes.

- **<src>** is the name of an independent source to which input noise is referred.
- <src> is the noiseless input signal.

- The parameters **<oct, dec, lin>**, **<Nsteps>**, **<StartFreq>**, and **<EndFreq>** define the frequency range of interest and resolution in the manner used in the .ac directive.

Noise Analysis Parameters

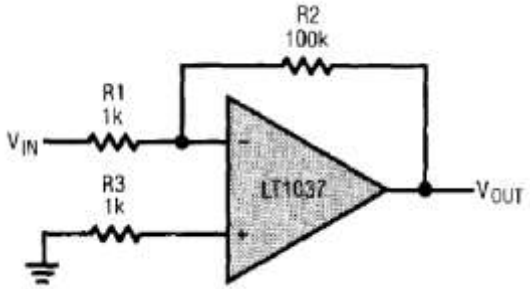
- Output data trace **V(onoise)** is the **noise spectral voltage density** referenced to the node(s) specified as the **output** in the above syntax.

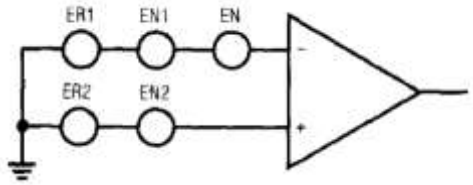
- If the **input signal** is given as a **voltage source**, then data trace **V(inoise)** is the **input-referred noise voltage density**.
- If the input is specified as a current source, then the data trace inoise is the noise referred to the input current source signal.

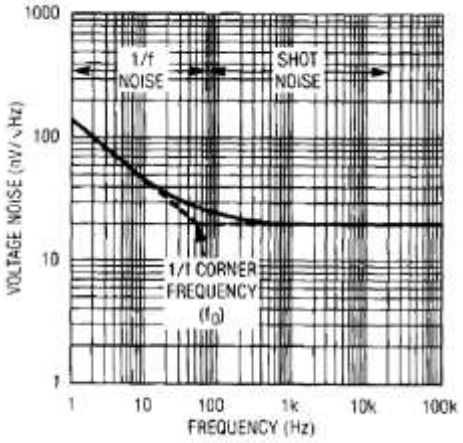
- The **noise contribution** of each component can be plotted.
- These contributions are **referenced to the output**.
- You can reference them to the input by dividing by the data trace "gain".

- The waveform viewer can **integrate noise over a bandwidth by <Ctrl-Key> + left** mouse button clicking on the corresponding data trace label.

DN15 – Noise Calculations in Op-amp Circuits





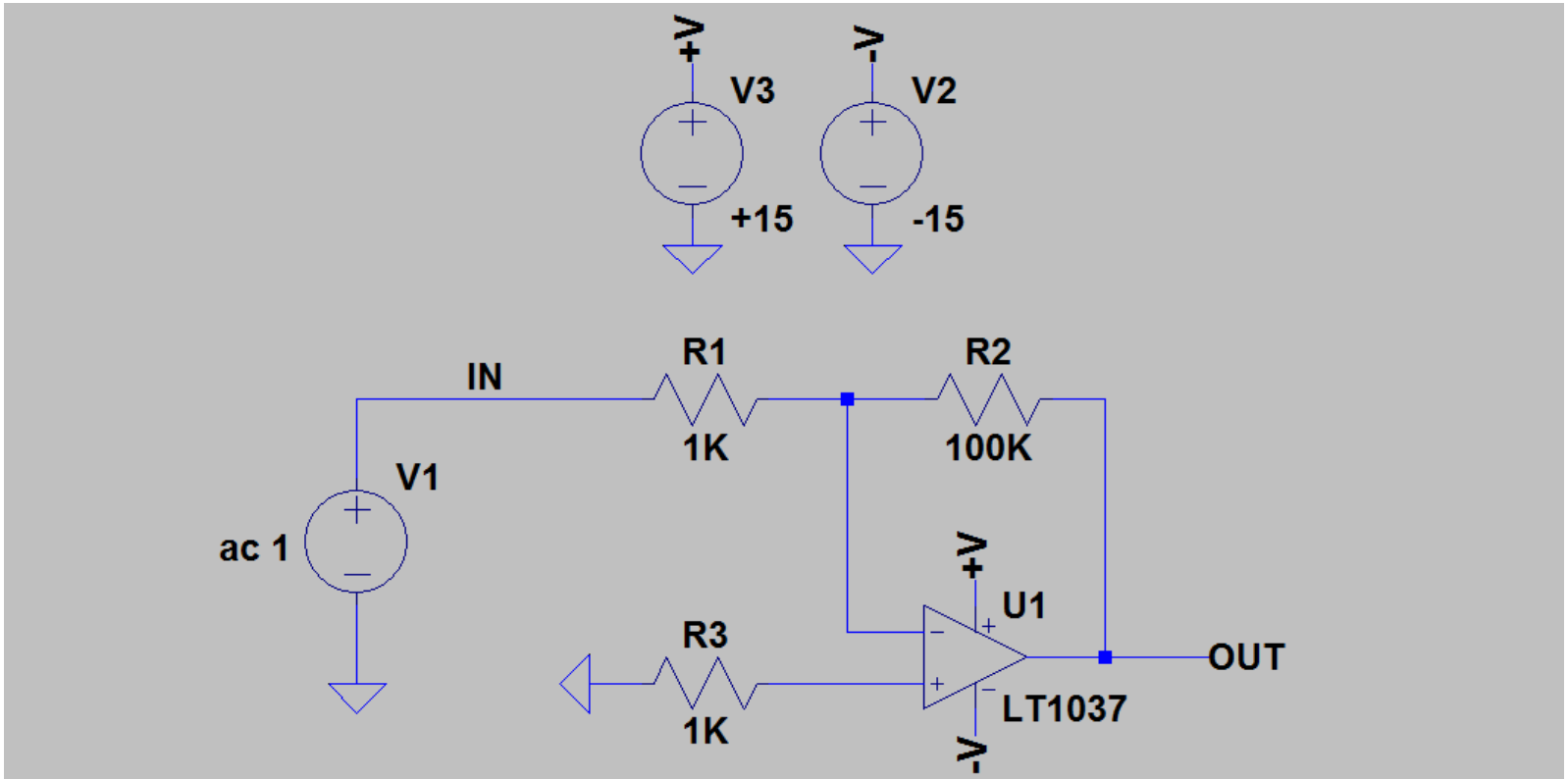


EN : voltage noise of the op-amp
EN1 : voltage noise developed by the current noise in R1 & R2
EN2 : voltage noise developed by the current noise in R3
ER1 : voltage noise of R1 & R2
ER2 : voltage noise of R3

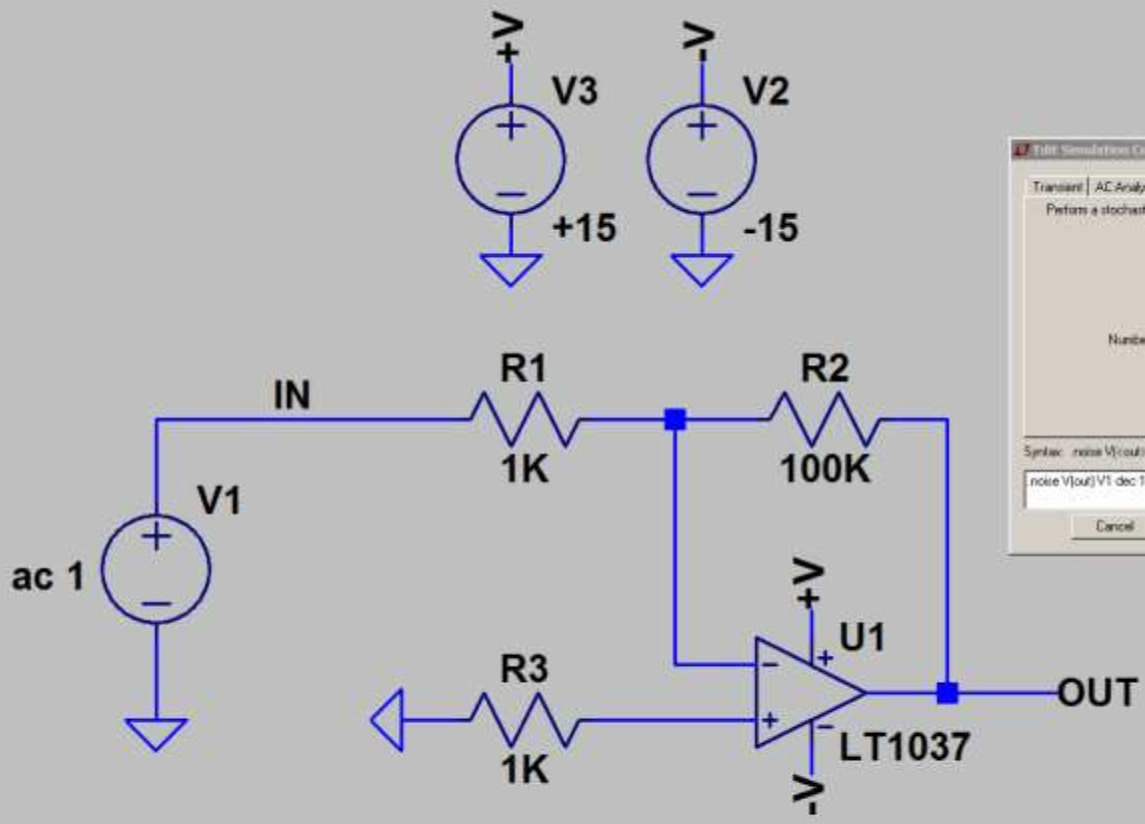
20Hz to 20kHz bandwidth
Total noise referred to the input = 880nV RMS
Output noise = 89µV RMS

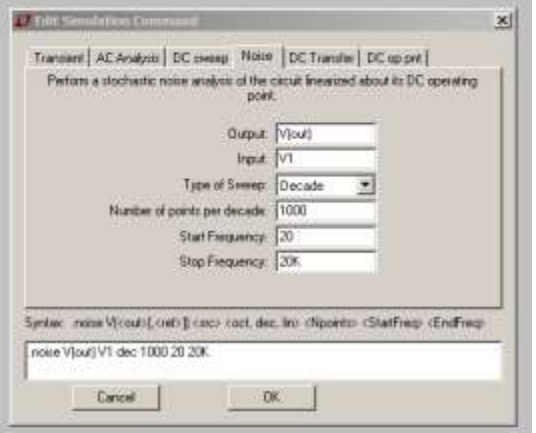
Example

- Audio Preamplifier Noise Analysis.asc



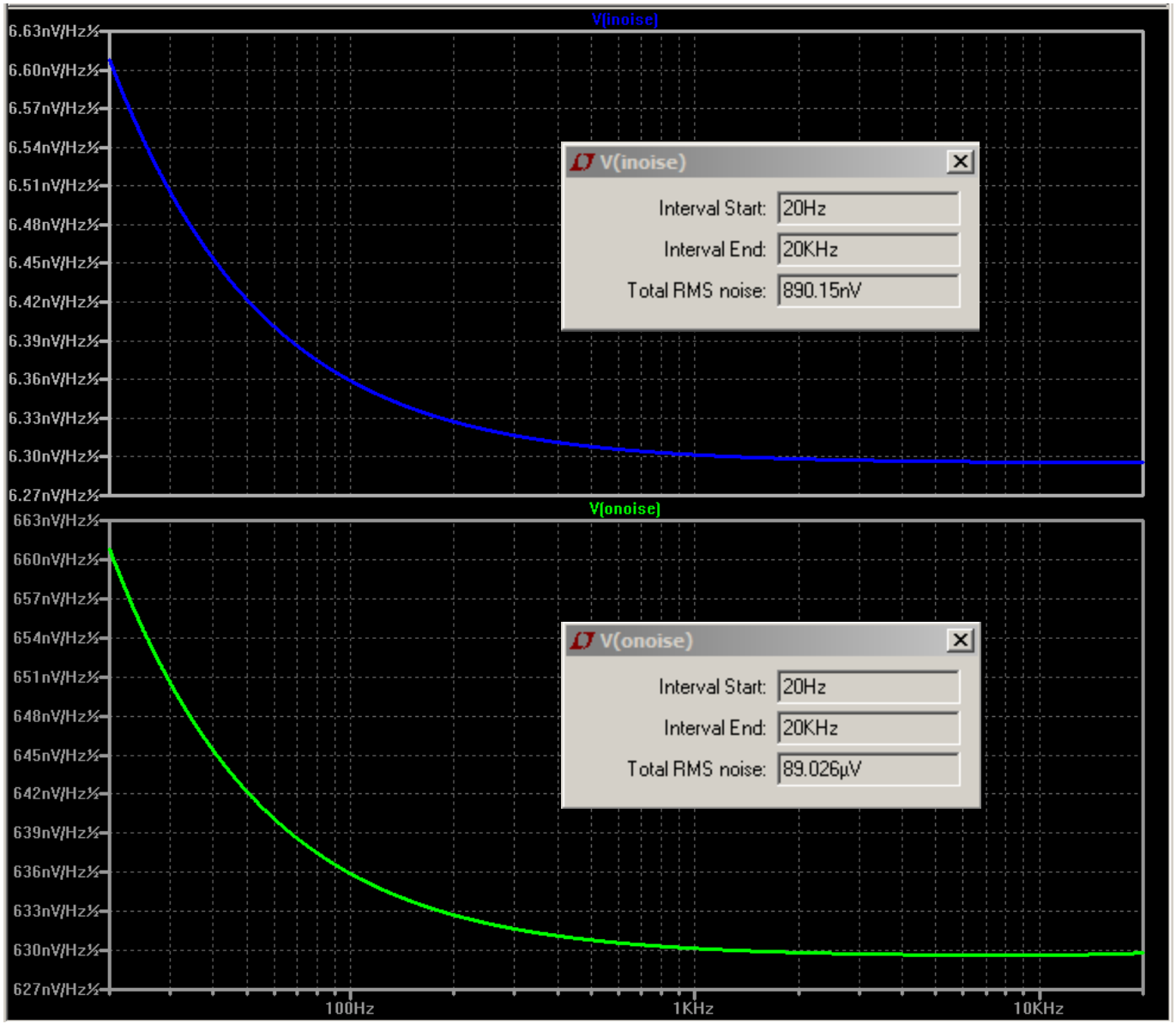
Audio Preamplifier Noise Analysis





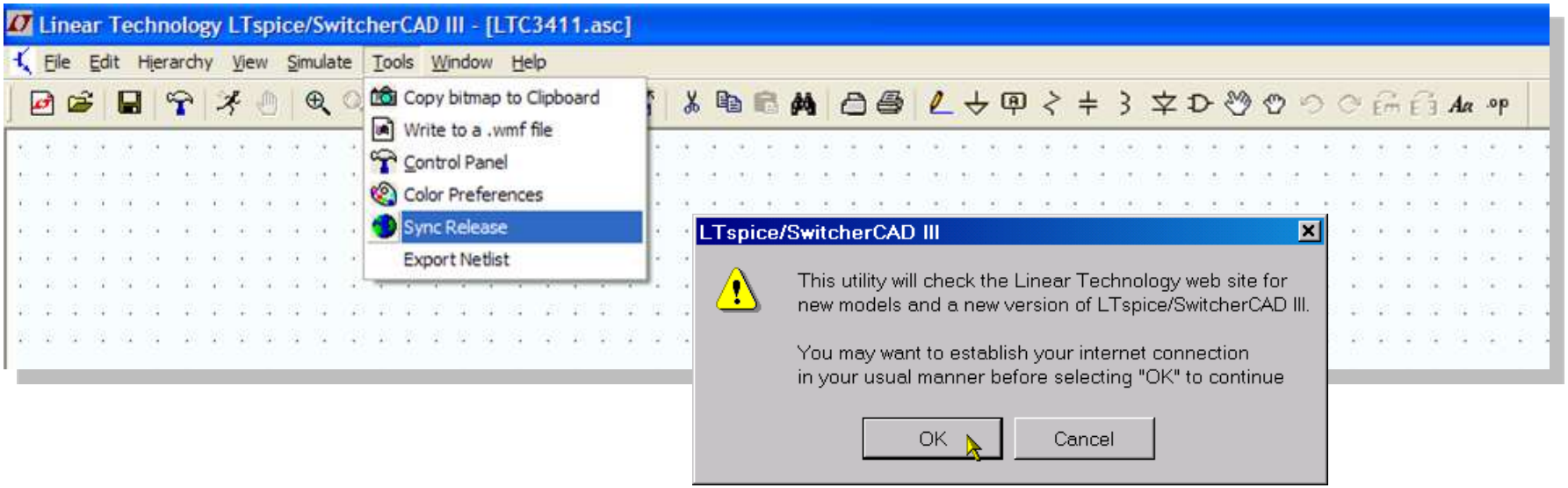
Total rms noise can be integrated by control left clicking the output data noise histogram

Input / Output Noise Density & RMS Noise



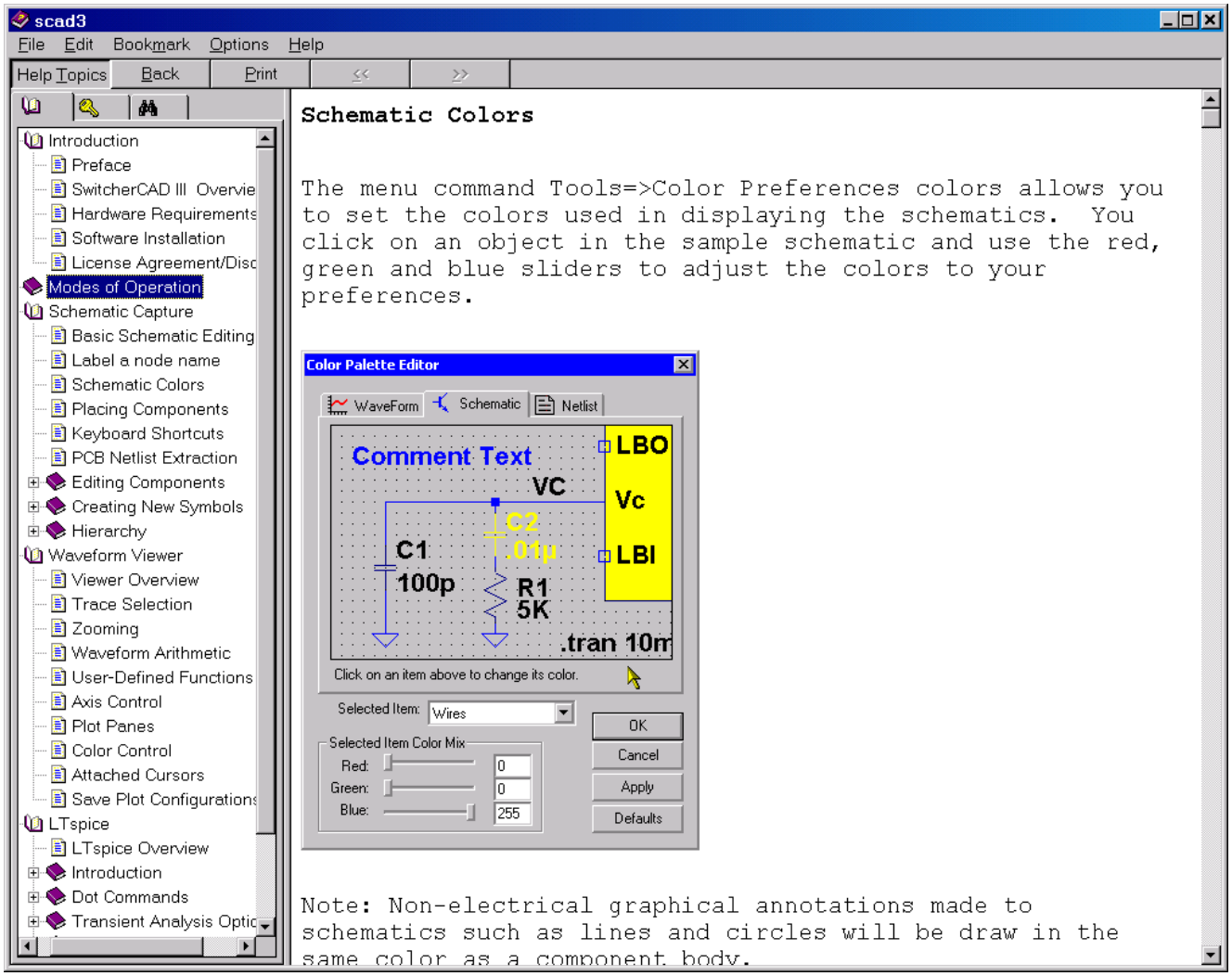
More Information and Support

Reminder to Periodically Sync Release



- It is important to sync your release of LTspice once a month to get the latest updates
  - Software update and bug fix
  - Models
  - Sample circuits and examples
- Vista users
  - You must “Run as administrator” scad.exe or its shortcut even if you are logged in as an administrator

Built-in Help System



PDF User's Guide

- Download the PDF User's Guide Manual:
  - http://LTspice.linear.com/software/scad3.pdf

Appendix A – Summary of Special Mouse and Keyboard Commands

Schematic-Based Special Commands:

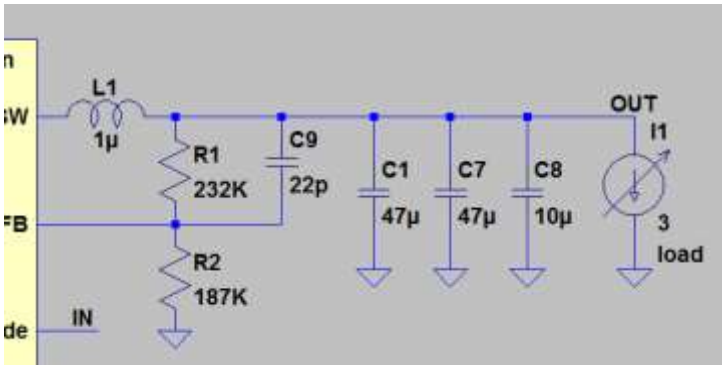
1. **Alt-Left-Click** on a **wire**
   - This will display the waveform for the **current** flowing in the wire
2. **Alt-Left-Click** on a **component** (thermometer)
   - This will display the instantaneous **power dissipation** in the component
3. **Ctrl-Right-Click** on a **component**
   - Allows you to edit embedded component **attributes**

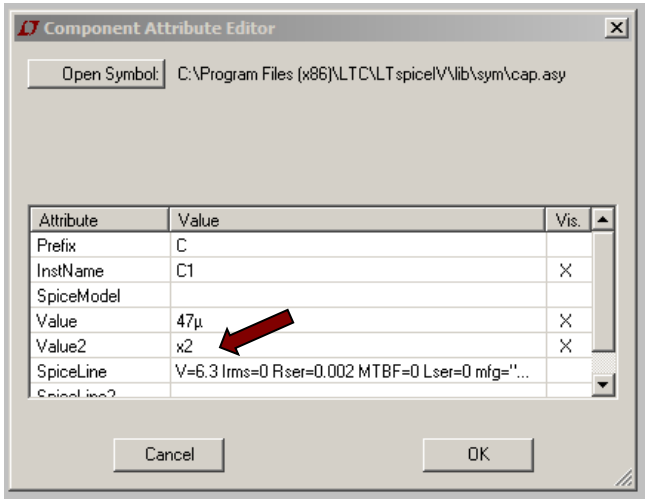
Waveform-Based Special Commands:

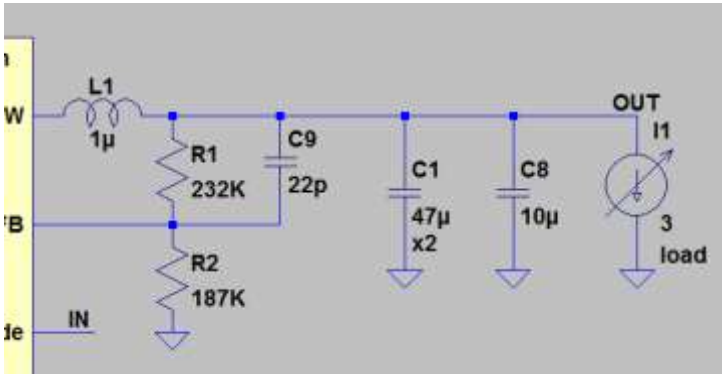
1. **Ctrl-Left-Click** on a **waveform title**
   - Displays the **average** and **RMS** values for the waveform
2. **Left-Click** on node and **drag** to another node
   - Displays **differential** voltage
3. **Alt-Left-Click** on a **waveform title**
   - This will **highlight** the corresponding **wire** on the schematic

How to Avoid Too Many Discretes on a Schematic

“x2” sets the number of parallel devices (2 in this case)

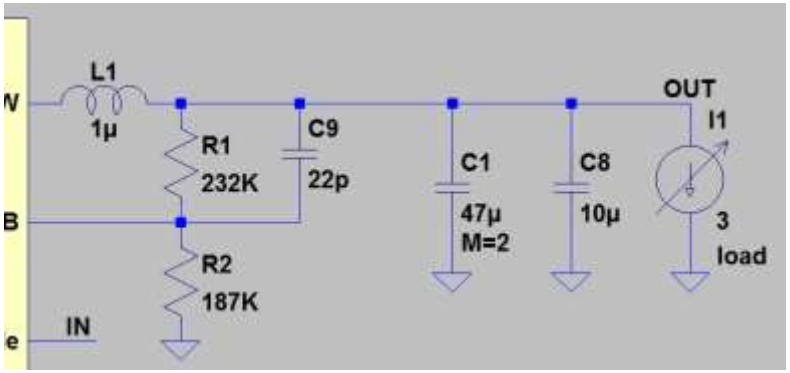


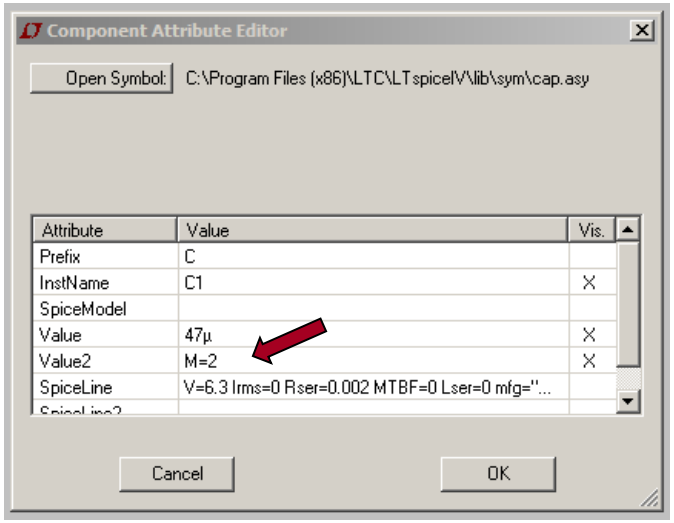




How to Avoid Too Many Discretes on a Schematic

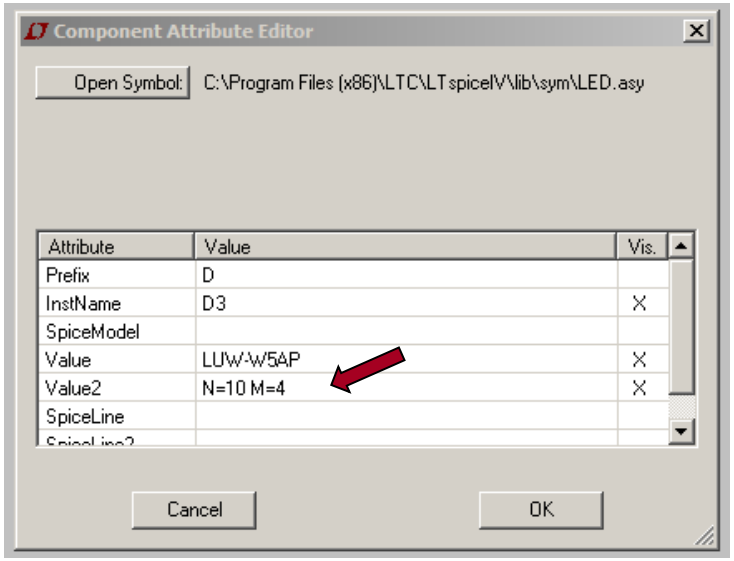
M sets the number of parallel devices.

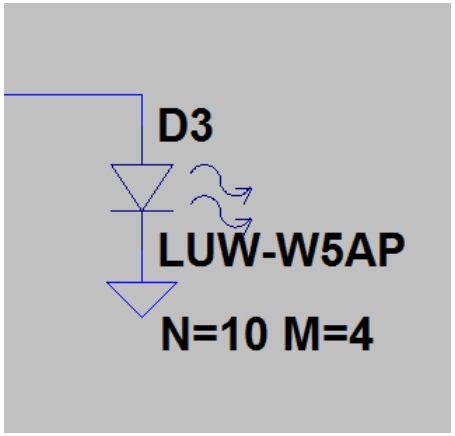




How to Avoid Too Many Discretes on a Schematic

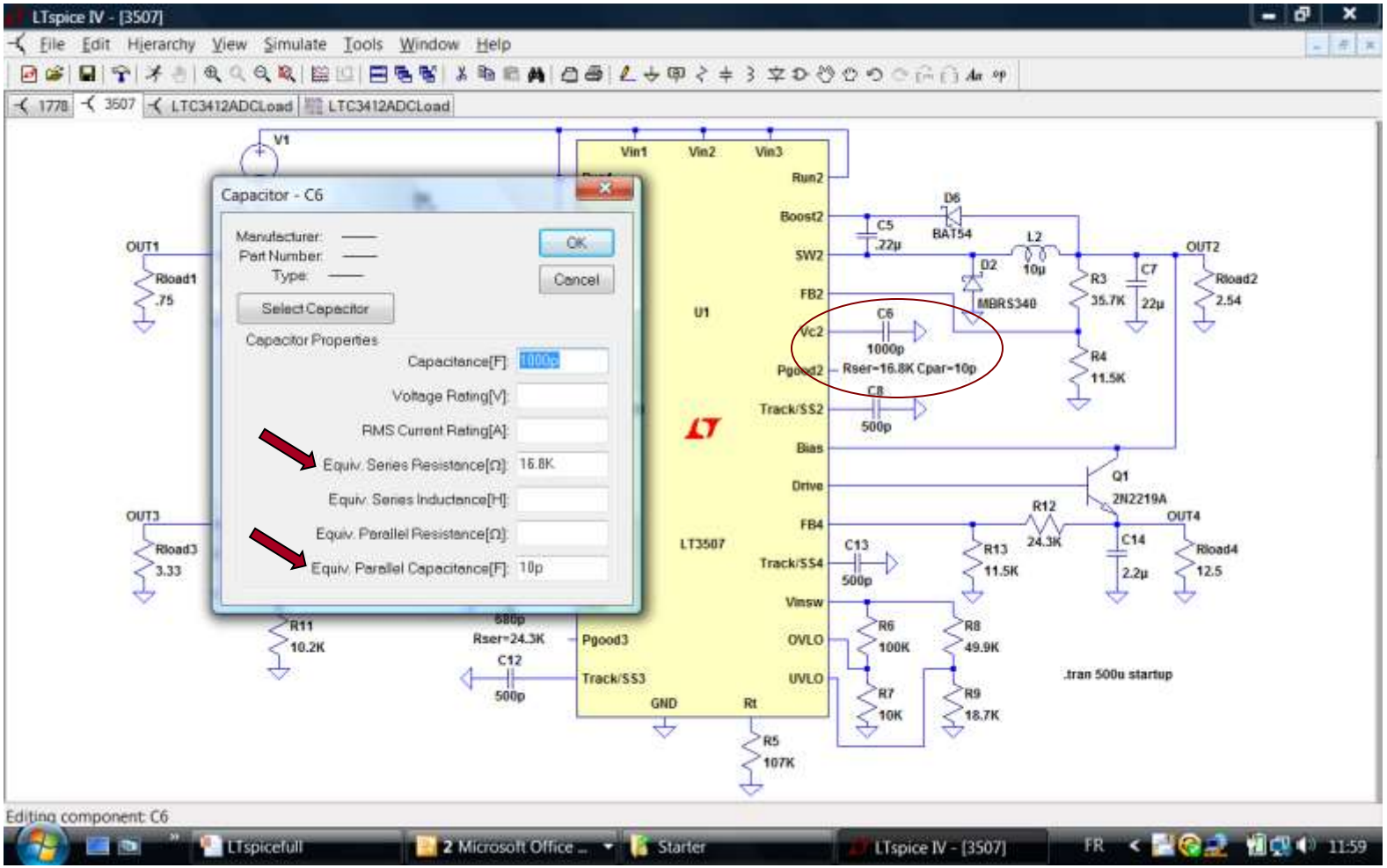
M sets the number of parallel devices (R, C, L, D)
N sets the number of series devices (D only).





LED matrix :
1 string = 10 LEDs in series (N=10)
4 strings in parallel (M=4)

How to Avoid Too Many Discretes on a Schematic



Appendix B – Summary of Additional Features

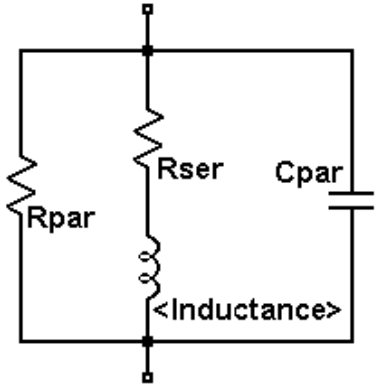
1. To **pause** a simulation:
   - "**Simulate**" pull down menu ---> **Pause**
   - There is no toolbar button for this function
2. To **zoom in/out** using the schematic editor:
   - Just use the **wheel** on your mouse

More Topics #1

1. How to simulate coils & transformers
2. Importing Third-Party Spice Models
3. Selecting a MOSFET for a DC/DC Converter
4. Managing and Customizing Model Libraries
5. Piece-wise Linear Voltage Sources

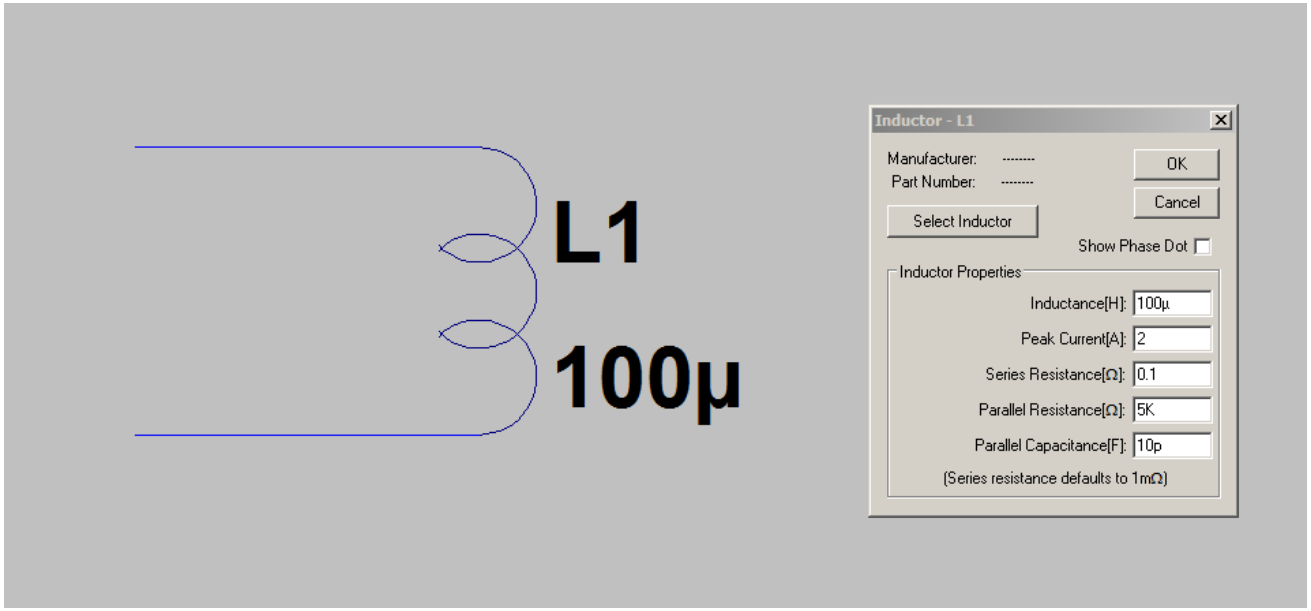
Modeling Coils

Normal Coil (no saturation)



Inductor Instance Parameters

Name	Description
Rser	Equivalent series resistance
Rpar	Equivalent parallel resistance
Cpar	Equivalent parallel capacitance
m	Number of parallel units
ic	Initial current(used only if uic flagged on the .tran card)
tc1	Linear inductance temperature coeff.
Tc1	Quadratic inductance temperature coeff.
temp	Instance temp



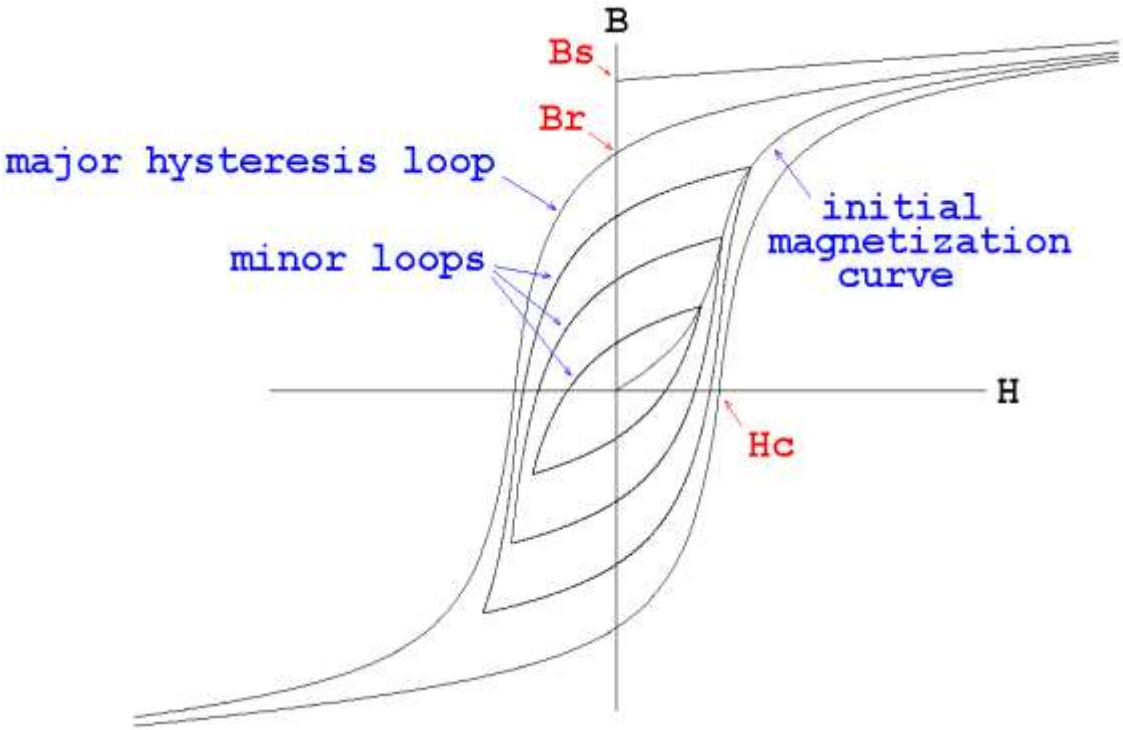
Non-linear Inductor - Chan Model

A computationally lightweight model that uses only 3 parameters to specify the core's major hysteresis loop:

Hc : Coercive force [Amp-turns/meter]

Br : Remnant Flux Density [Tesla]

Bs : Saturation Flux Density [Tesla]

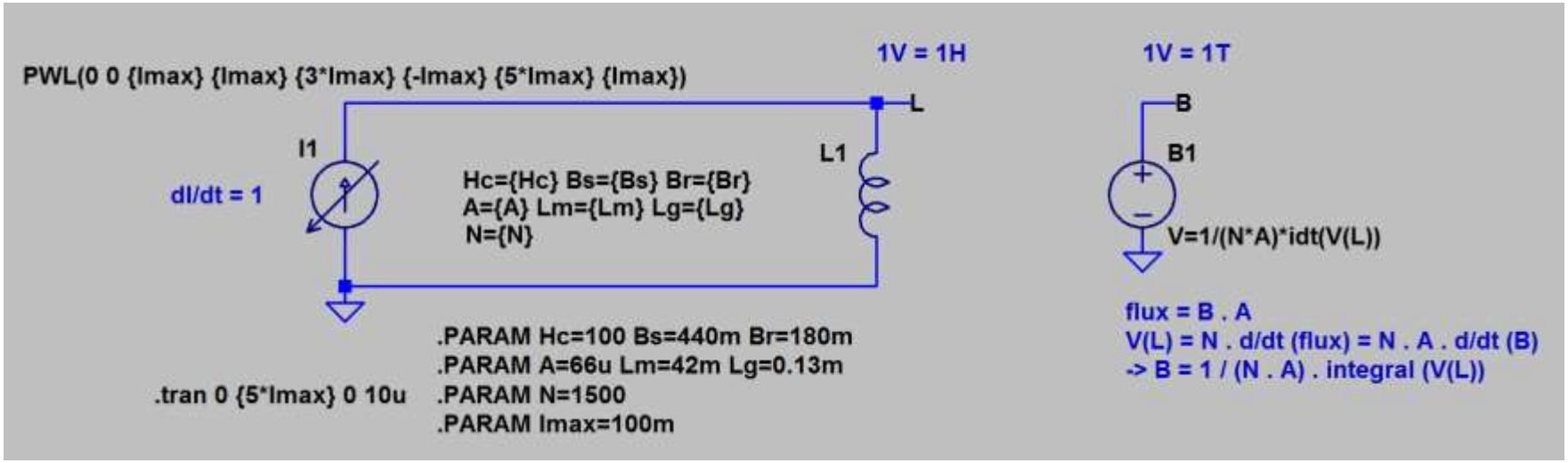


Non-linear Inductor - Chan Model

- Core physical dimensions specified with 4 parameters:
  - Lm : Magnetic Length (excluding gap) [meter]
  - Lg : Length of gap [meter]
  - A : Cross sectional area [meter^2]
  - N : Number of turns

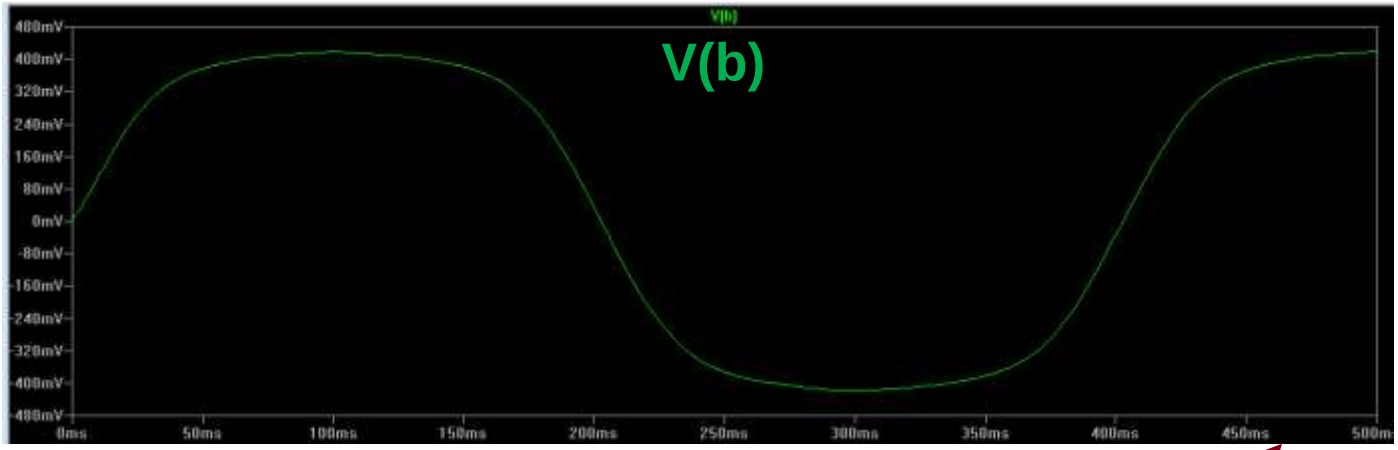
Non-linear Inductor Hysteresis Cycle B(I)

- Non-Linear Inductor Hysteresis.asc

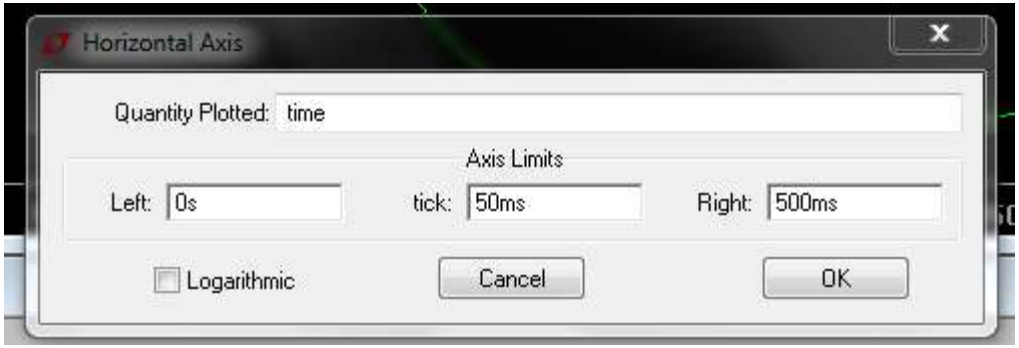


Changing the X-Axis

1. Run plot
2. X-axis is time by default



3. Mouse over the X-axis, cursor will become a ruler

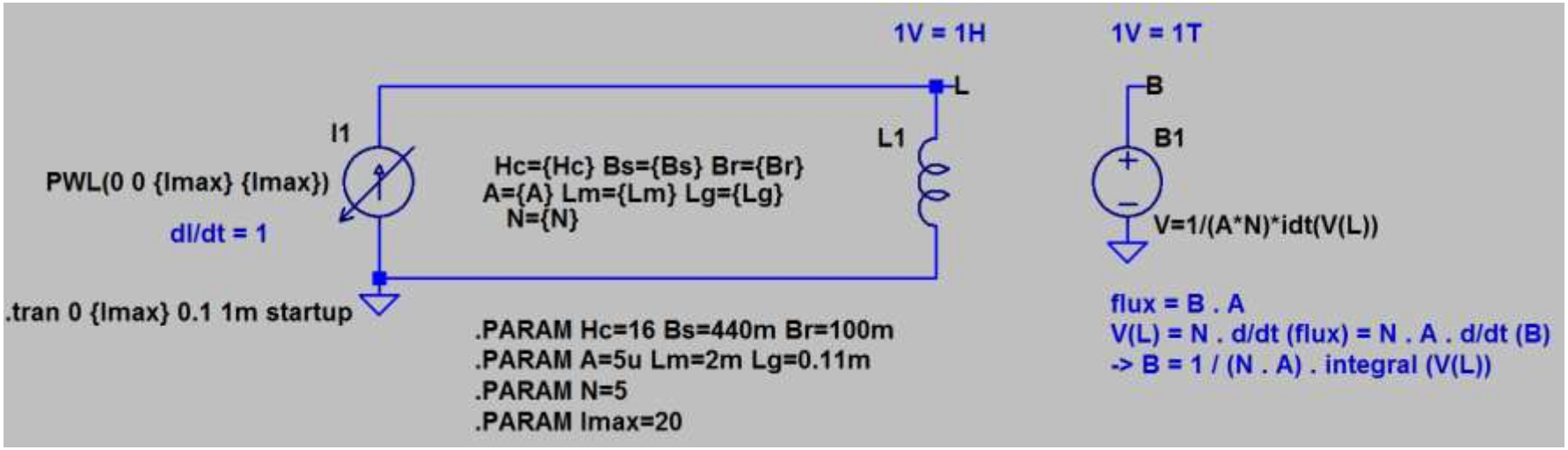


4. Put another waveform into the “Quantity Plotted” field

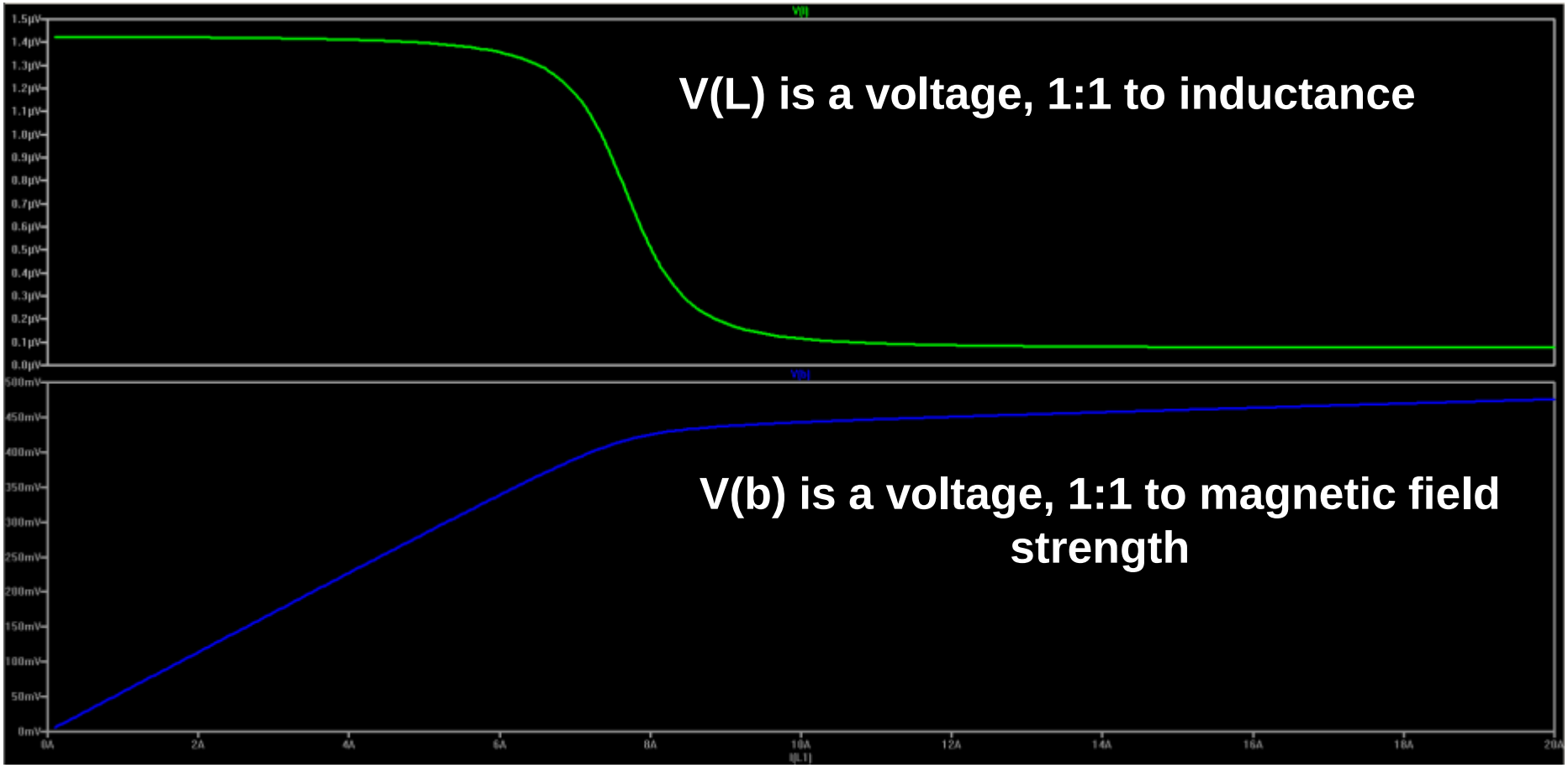
Non-linear Inductor Hysteresis Cycle B(I)

Non-linear Inductor
L(I) & B(I)

- Non-Linear Inductor LB.asc



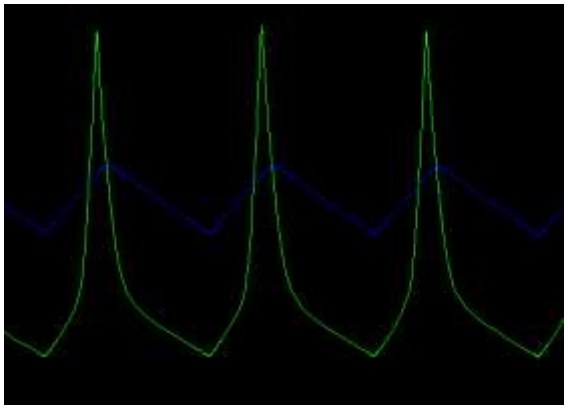
Non-linear Inductor
L(I) & B(I)

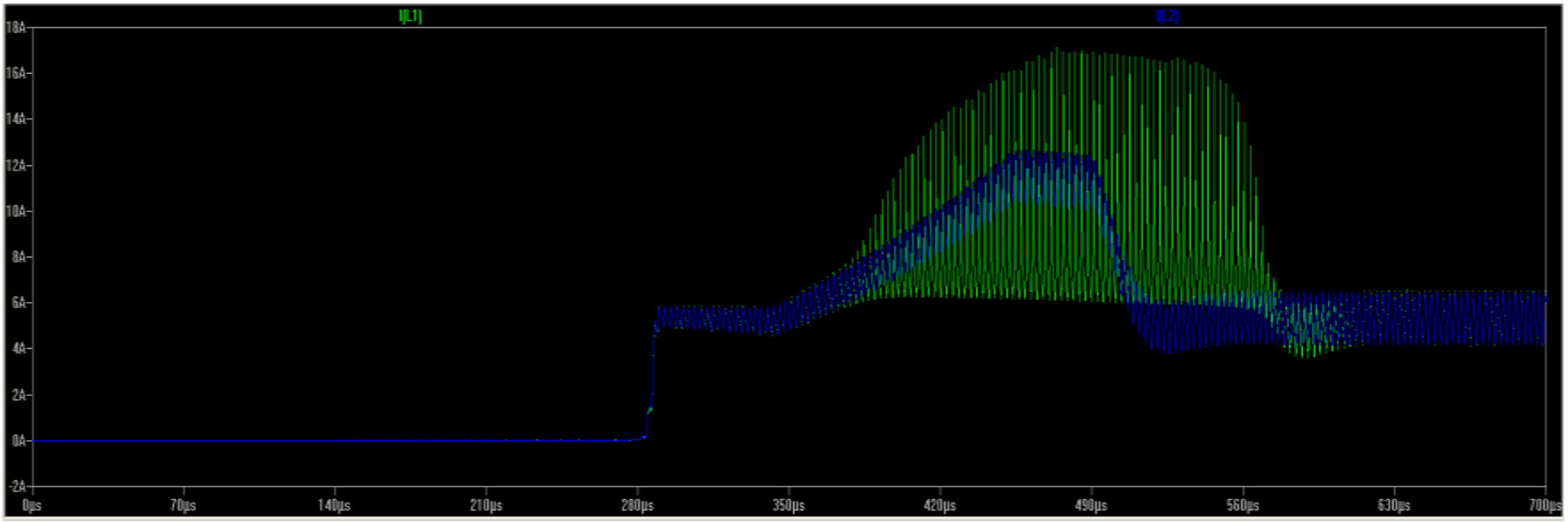


X-axis is inductor current I(L)

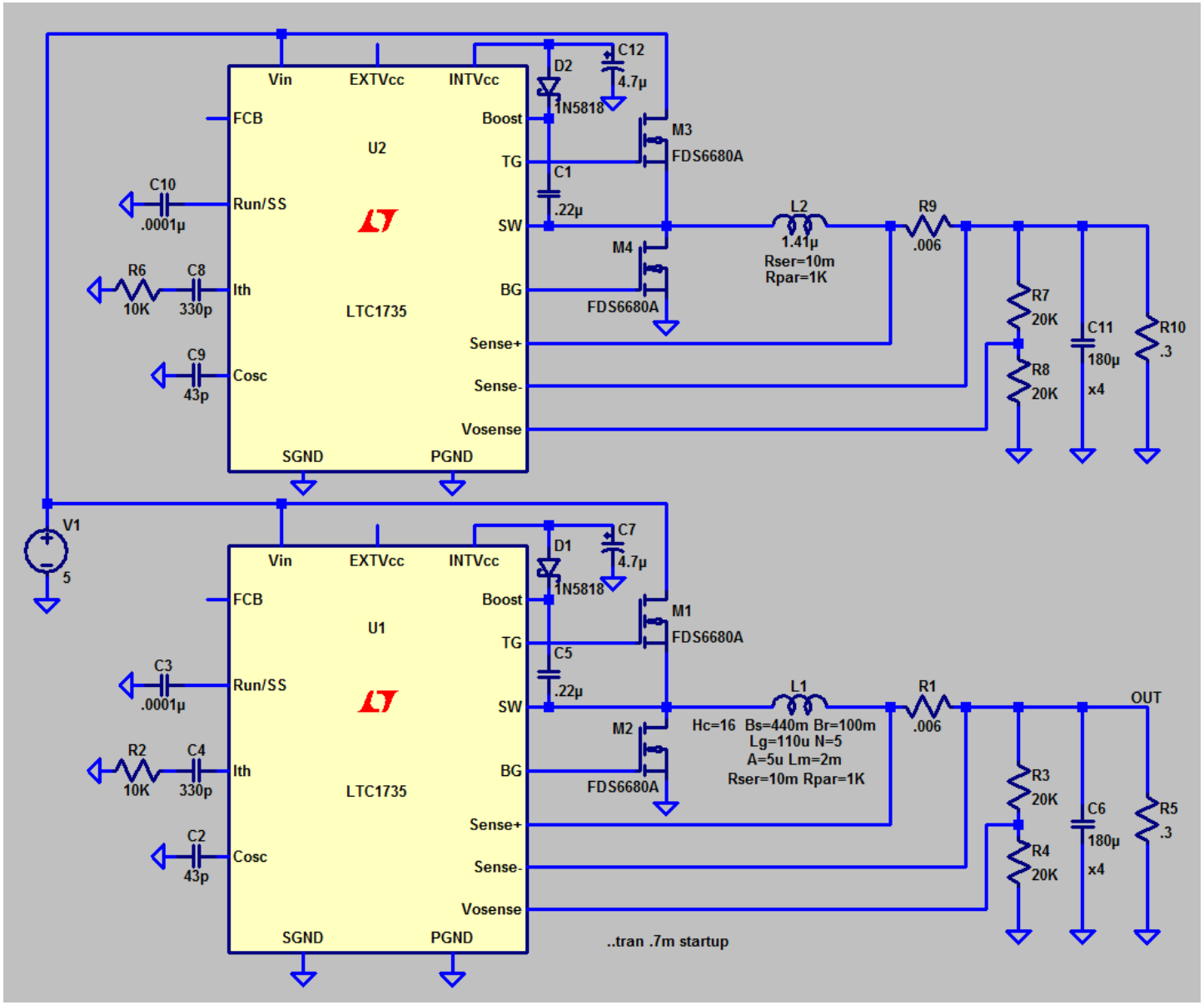
Example

- SMPS Sat Core II.asc





SMPS Inductor Saturation

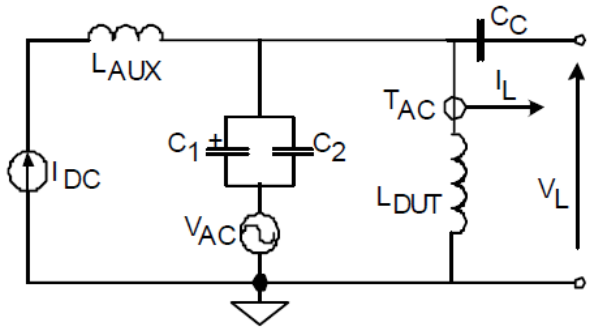


Core Saturation Considerations

- Saturation flux density (Bs) goes down monotonically with temperature
- Maximum service temperature plus self-heating
- Controller peak current production scatter
- Startup / transient / short circuit conditions

Variable Inductance Modeling

- First, characterize the saturation curve of the physical coil in the lab

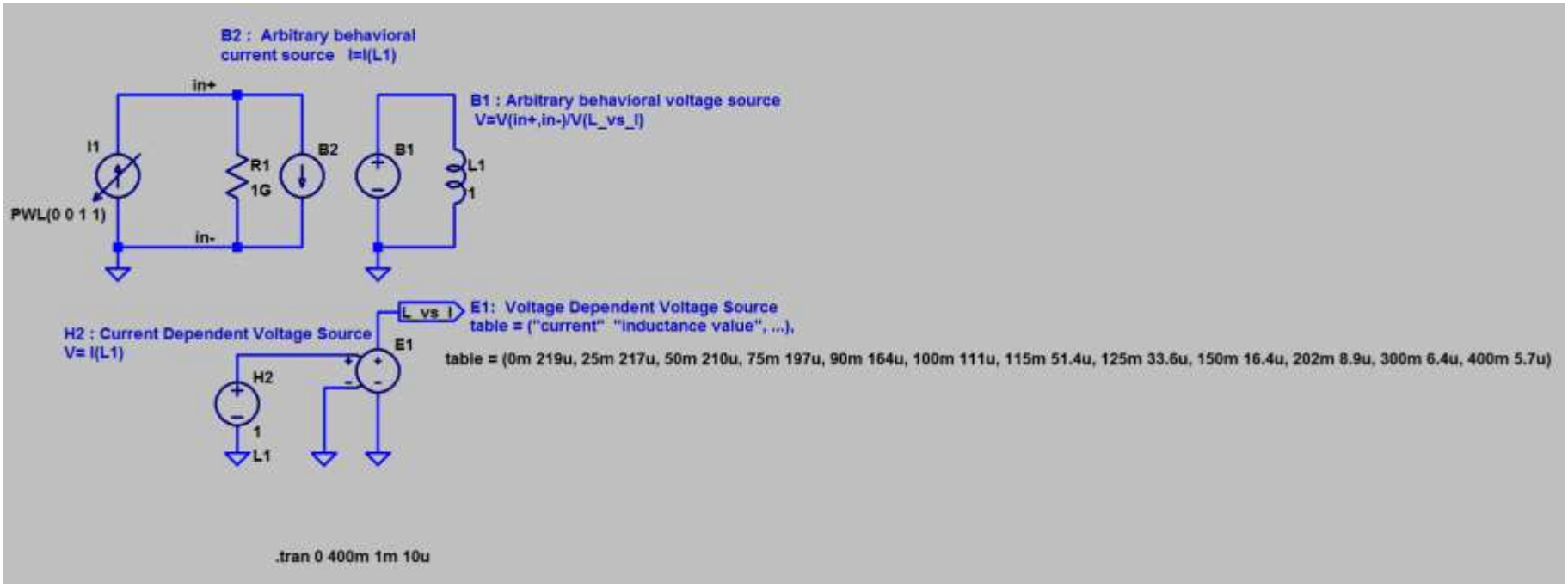


Note: you can use a RLC meter if available

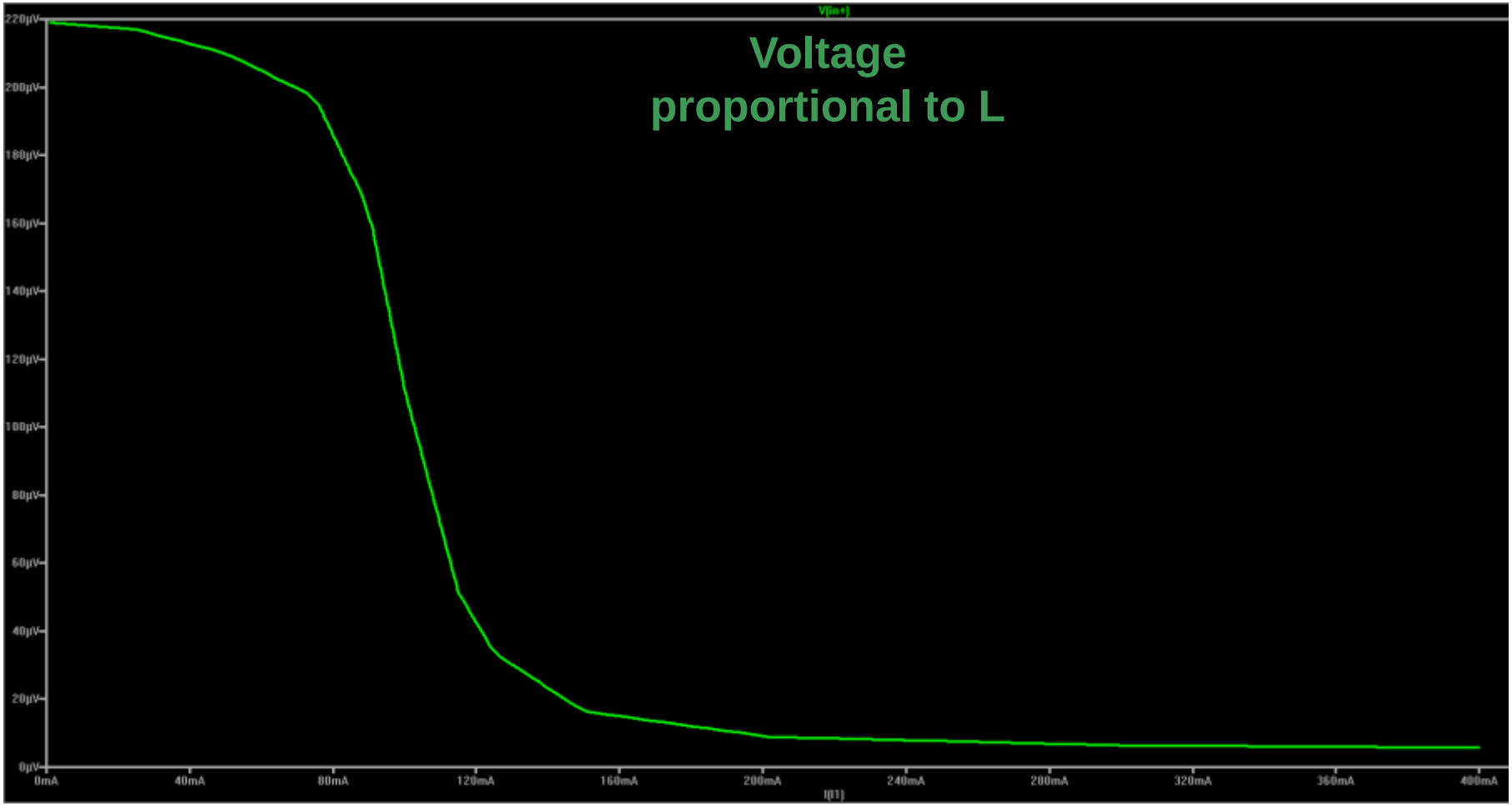
- Use “Voltage Dependent Voltage Source” or “Voltage Dependent Current Source” in order to model the coil saturation curve using the look up table capability of these behavioral sources
- A look-up table is used to specify the transfer function. The table is a list of pairs of numbers. The second value of the pair is the output voltage when the control voltage is equal to the first value of that pair. The output is linearly interpolated when the control voltage is between specified points. If the control voltage is beyond the range of the look-up table, the output voltage is extrapolated as a constant voltage of the last point of the look-up table.

Example: Variable Inductance Modeling

- Variable Inductance Modelization.asc



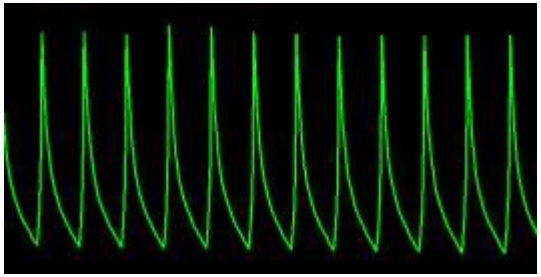
Variable Inductance Modeling

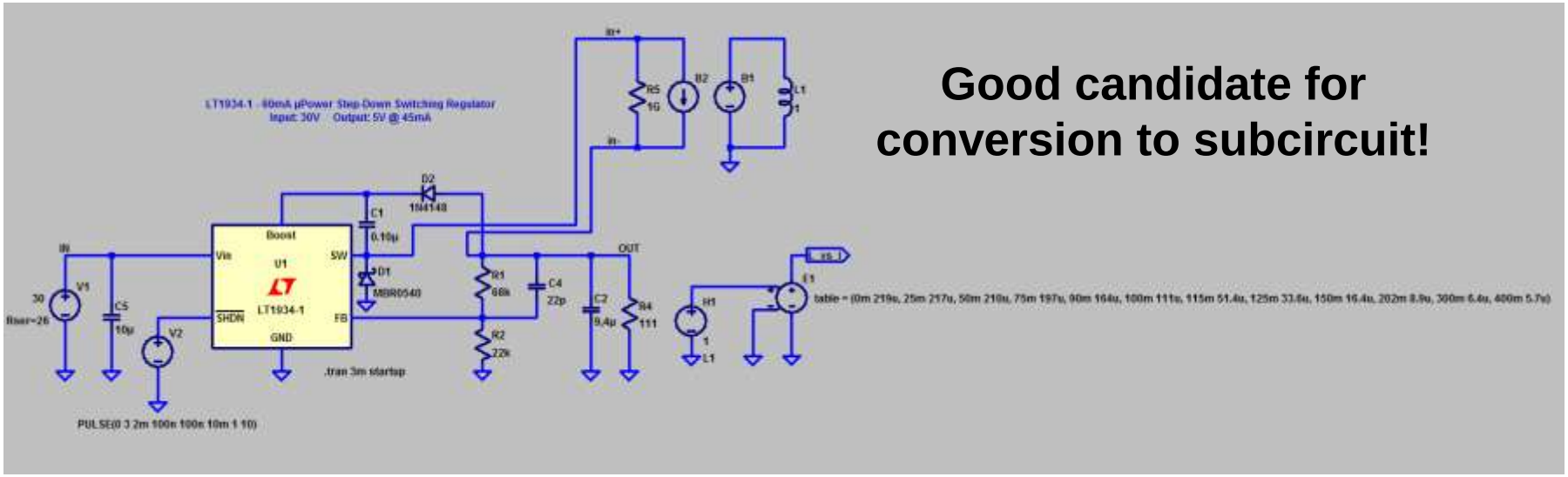


X-axis is inductor current I(L1)

Variable Inductance vs. Current

- SMPS with saturating coil.asc





Good candidate for conversion to subcircuit!

Quick´n´Dirty Saturable Inductor

- What to do if you don´t know Hc, Br and Bs!
- Typical off-the-shelf inductor datasheet:

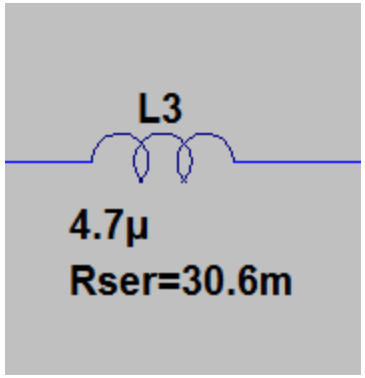
L		L measuring frequency	DC resistance	Rated current(A)*	
				max.	
(µH)	Tolerance	(kHz)	(Ω)±20%	Idc1	Idc2
4.7	±20%	100	0.0306	1.5	1.8
6.8	±20%	100	0.0442	1.3	1.5
10	±20%	100	0.0573	1	1.3

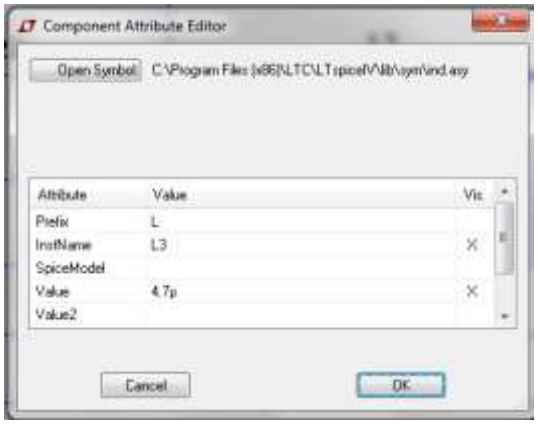
* Rated current: smaller value of either Idc1 or Idc2.
Idc1: When based on the inductance change rate (30% below the nominal value)
Idc2: When based on the temperature increase (Temperature increase of 25°C by self heating)

- Look at the note, see that Idc1 is the saturation current (because it forces L to drop)

Quick´n´Dirty Saturable Inductor

1. Place a standard inductor (or go to one already in your schematic)
2. CTRL+Right Click to enter the parameters





3. Replace “Value” with the following:

```
flux={L*Is}*tanh((x/{Is})**{a})**{1/a}
```

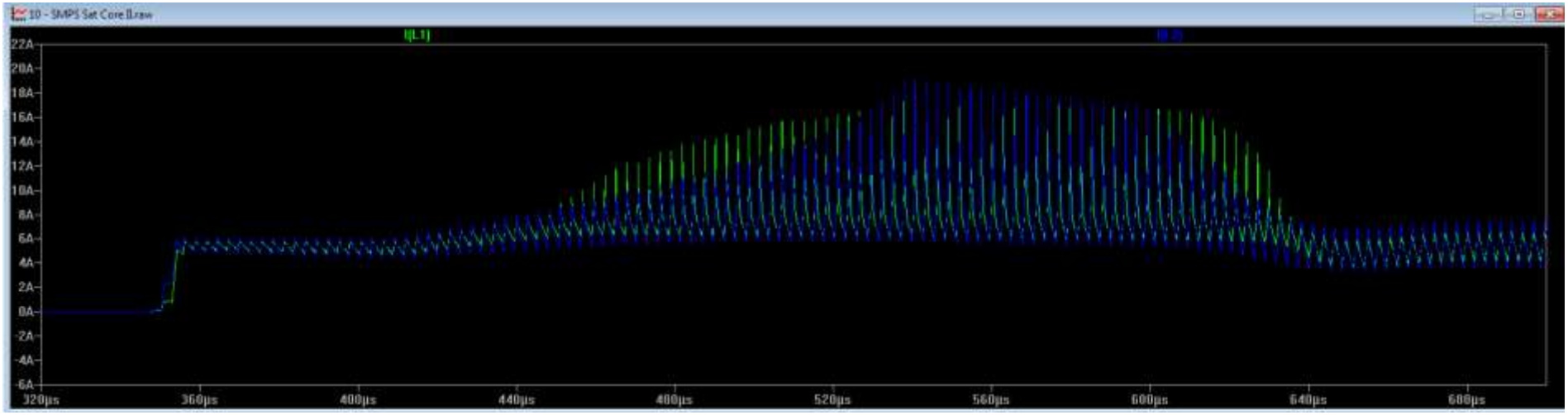
“L” is inductance, “Is” is saturation current and ”a” is the “steepness” of the saturation (exponential function)

Quick´n´Dirty Saturable Inductor

1. Place the following SPICE direction (.op) anywhere in the schematic (close to L is good for clarity)

```
.param L=4.7uH Is=1.5A a=1
```

2. For the “steepness”, a =1 is typical. Higher values lead to steeper current peaks. Use 1 to start

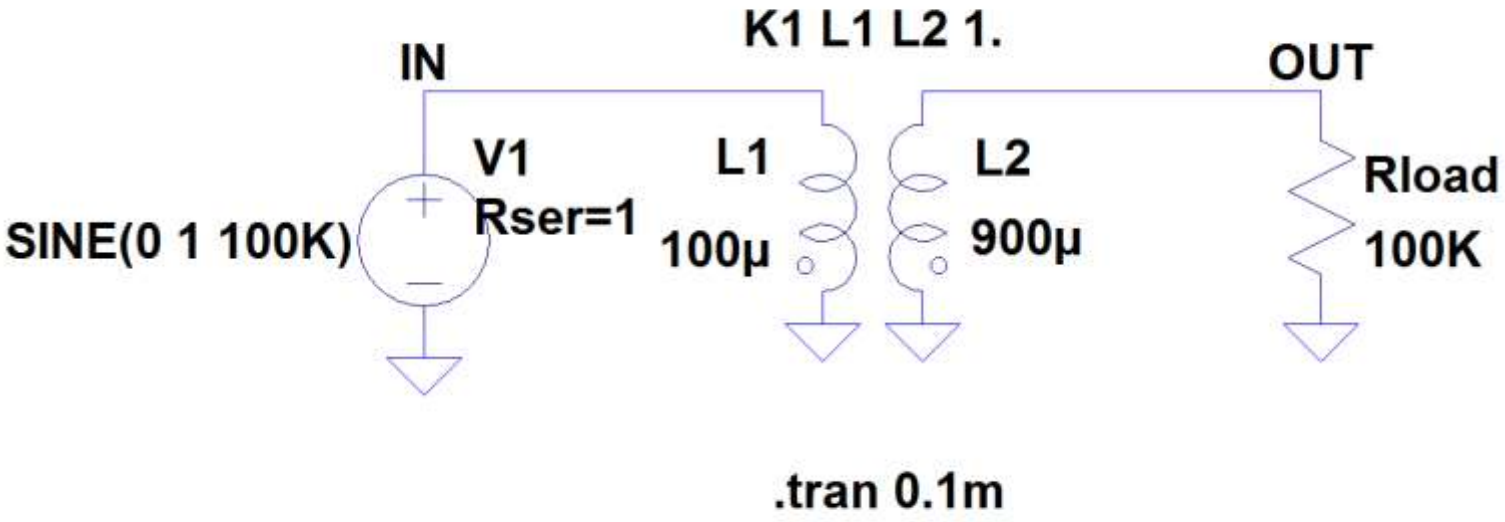


Green = model with Hc, Br, Bs. Blue = quick´n´dirty

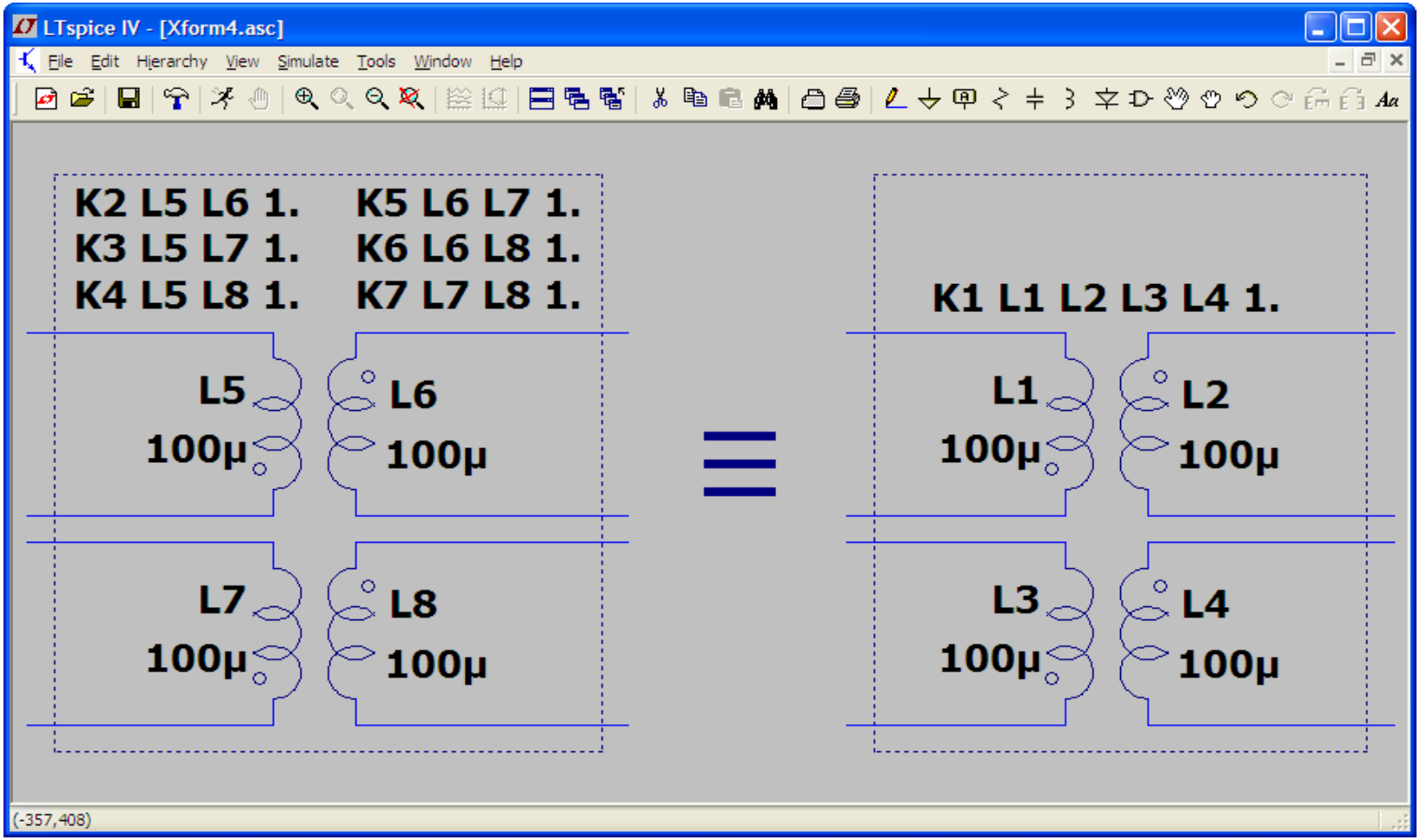
Modeling Transformers

Transformer Simulation

The following example demonstrates a transformer with 1:3 turns ratio (one to nine inductance ratio)



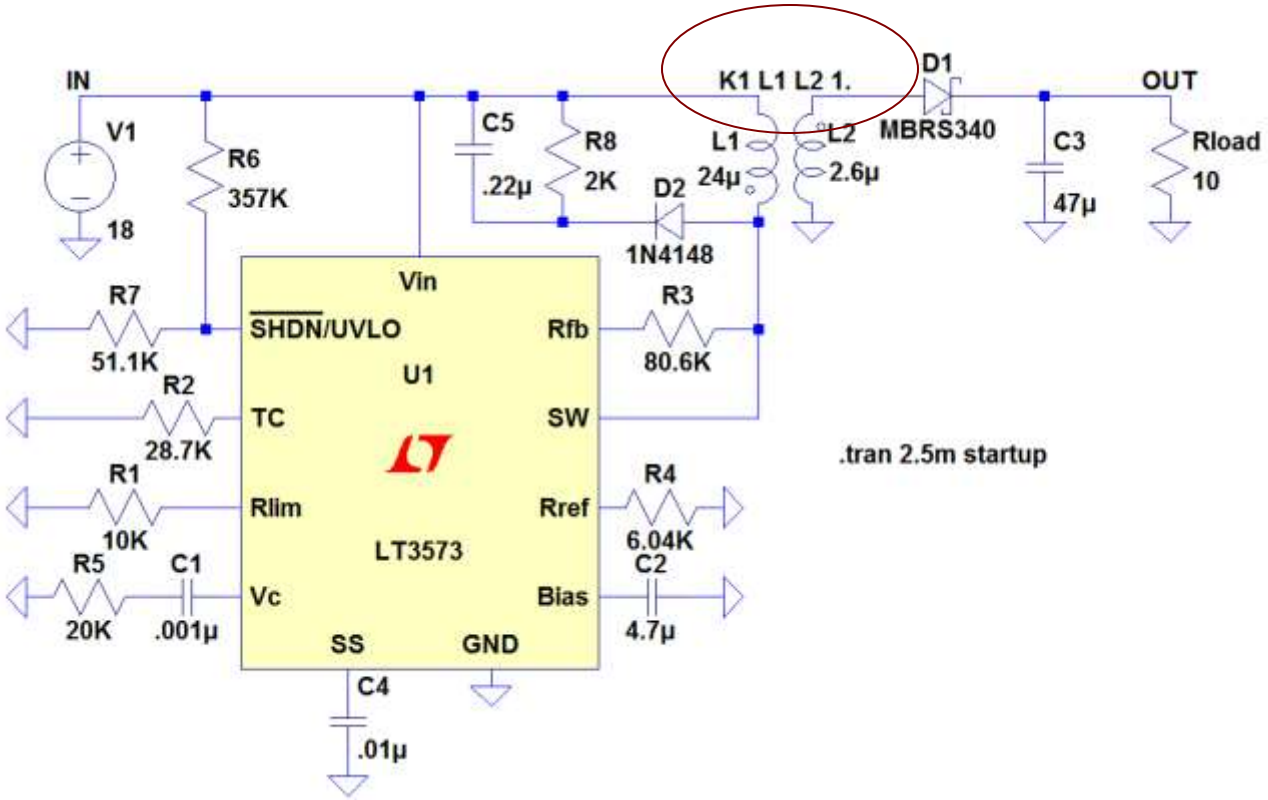
Multiple Windings



For N windings, the number of mutual couplings is $\frac{N(N-1)}{2}$

Example

- LT3573 Flyback XFMR K=1.asc

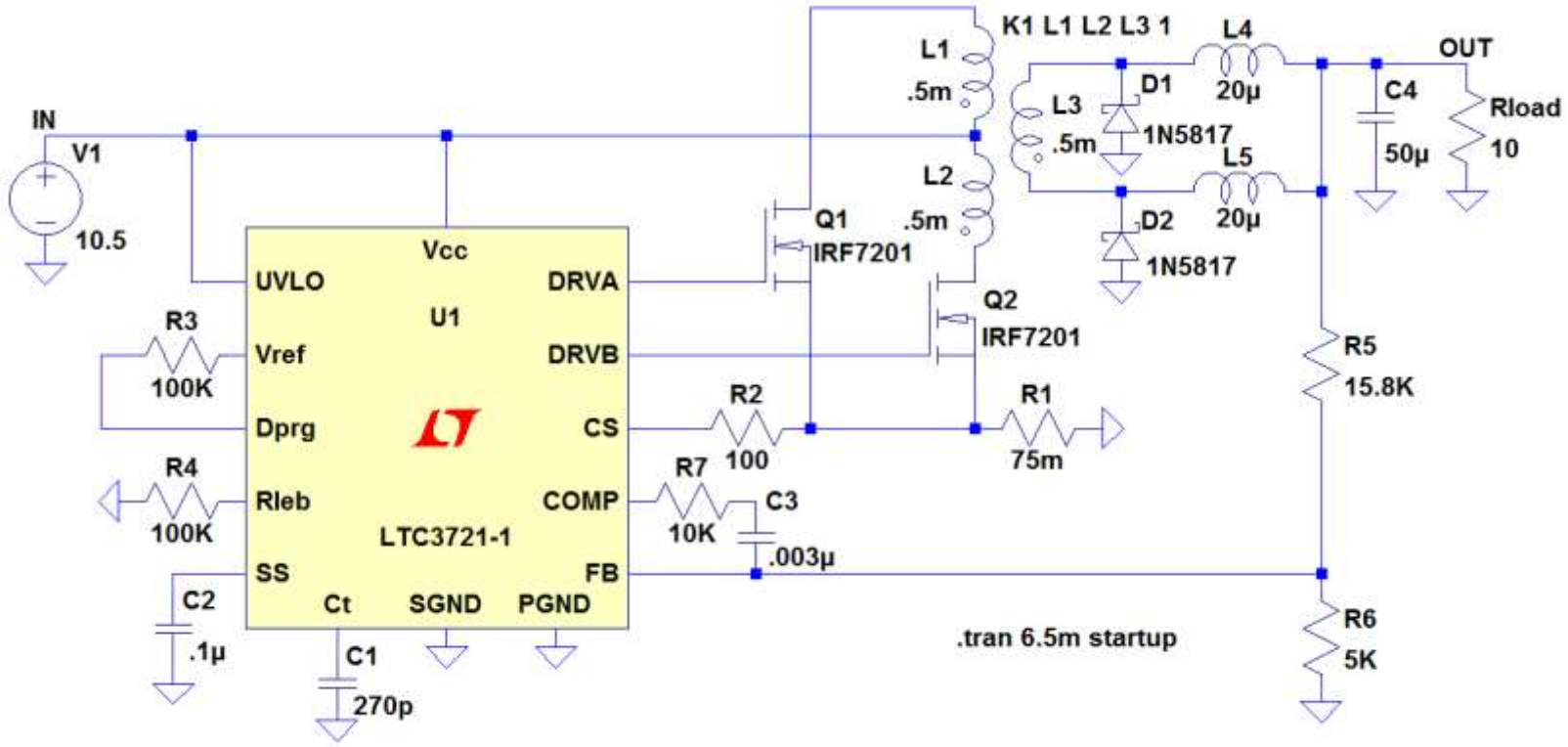


Syntax: Kxxx L1 L2 [L3 ...] <coefficient>

L1 and L2 are the names of inductors in the circuit.
The mutual coupling coefficient must be in the range of -1 to 1.

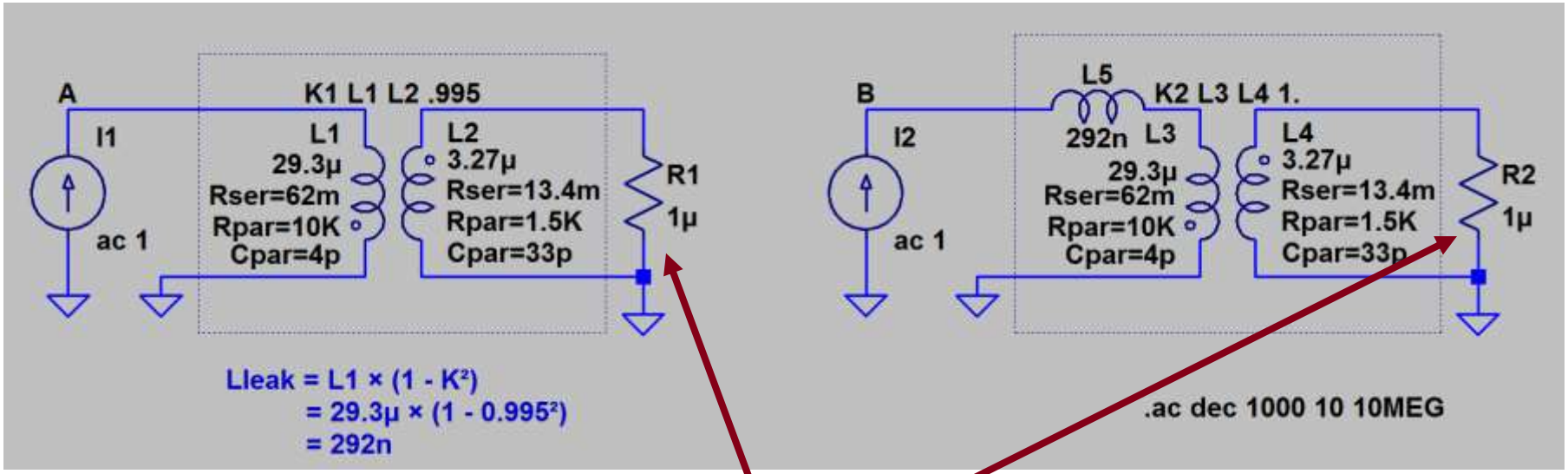
Transformer Simulation

Multiple windings can be added



Transformer with Leakage Inductance

- XFMR Leakage Inductance.asc



Short circuit the secondary(ies) to test leakage inductance

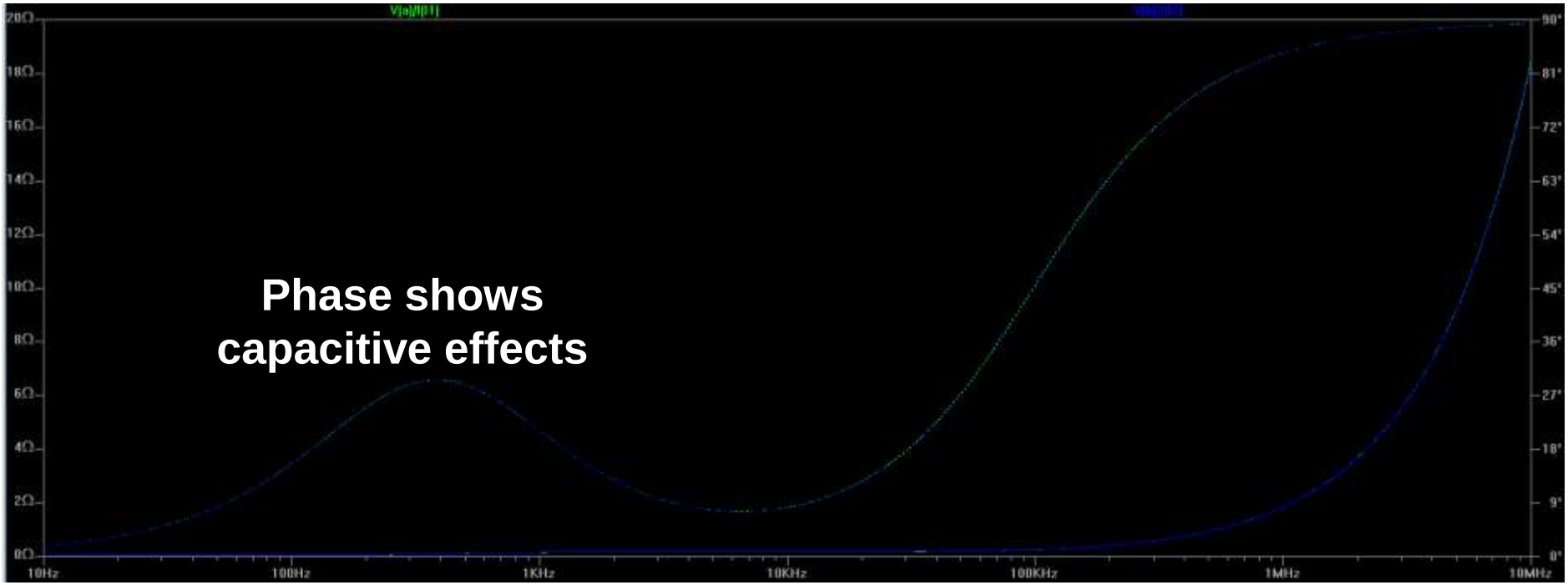
Transformer with Leakage Inductance

K = 0.995

Lleak = 292 nH

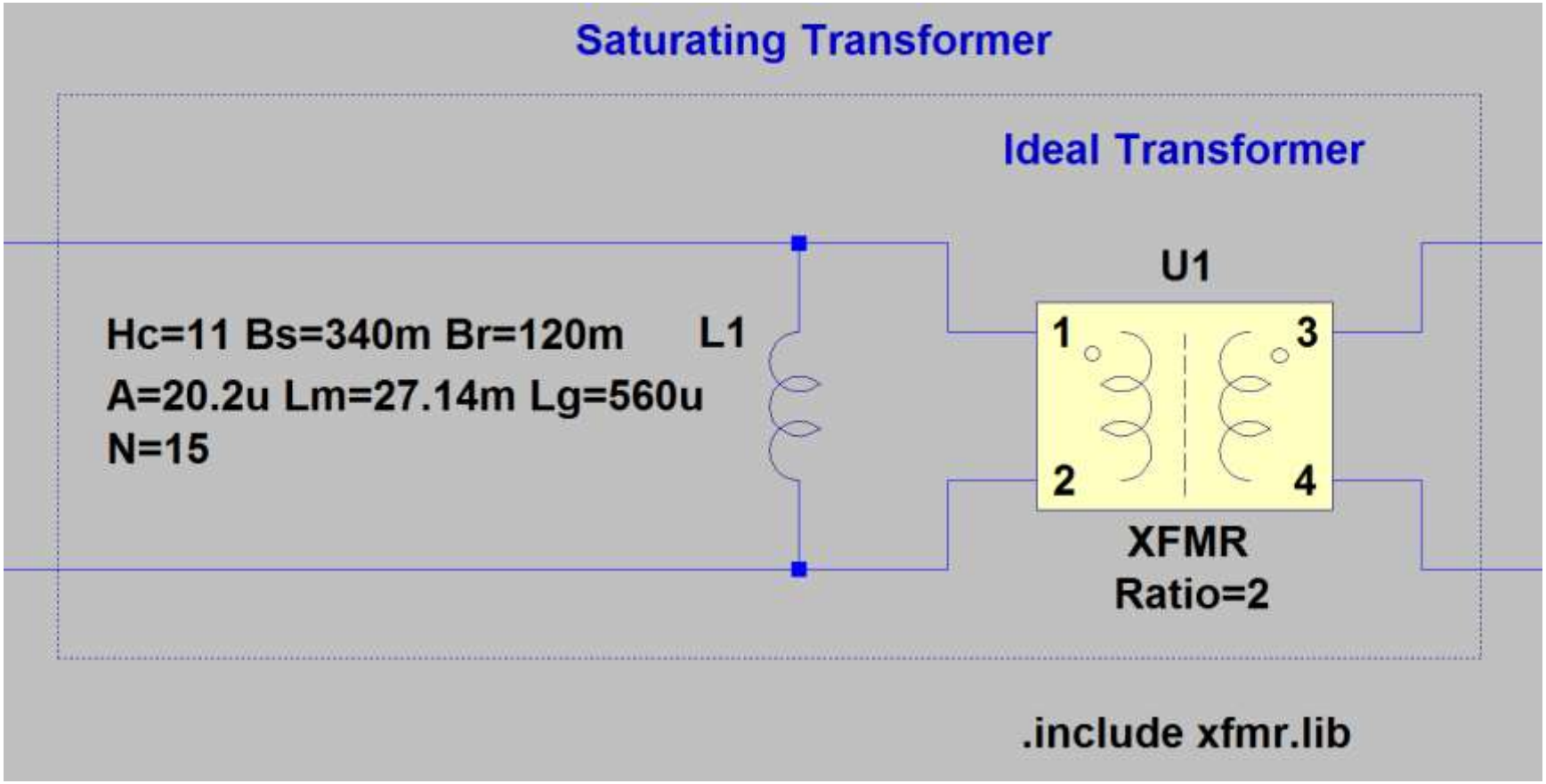
$Z_A = V_A / I1$

$Z_B = V_B / I2$



Frequency

Saturating Transformer



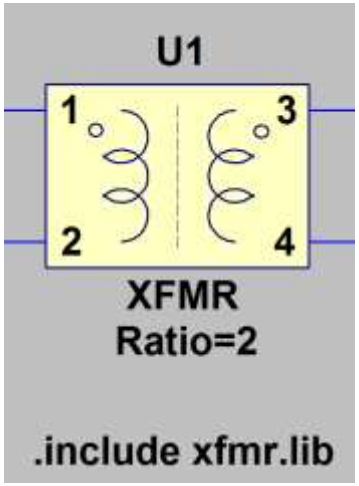
Examples

- Ideal transformer model.asc
- Ideal transformer.asc

Careful – these “transformers” will pass DC voltage and current!!

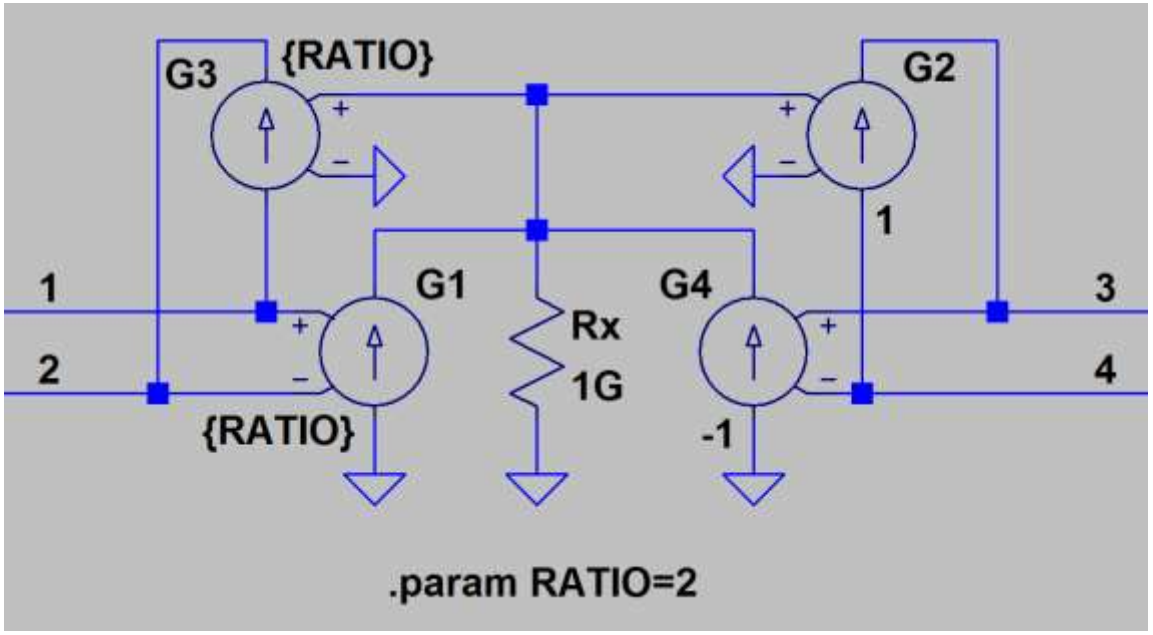
Ideal Transformer

I(1) = I(3) x RATIO

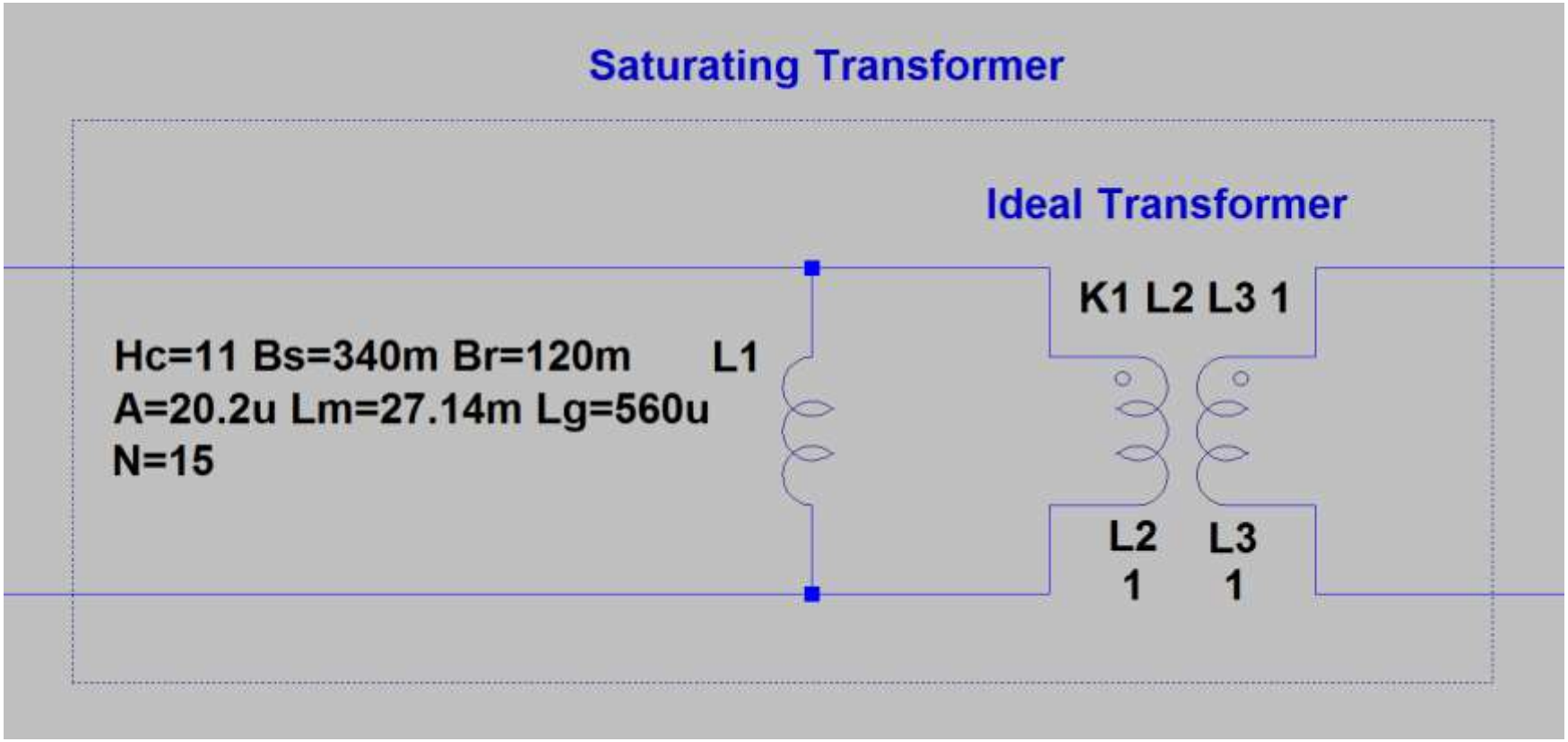


V(3,4) = V(1,2) x RATIO

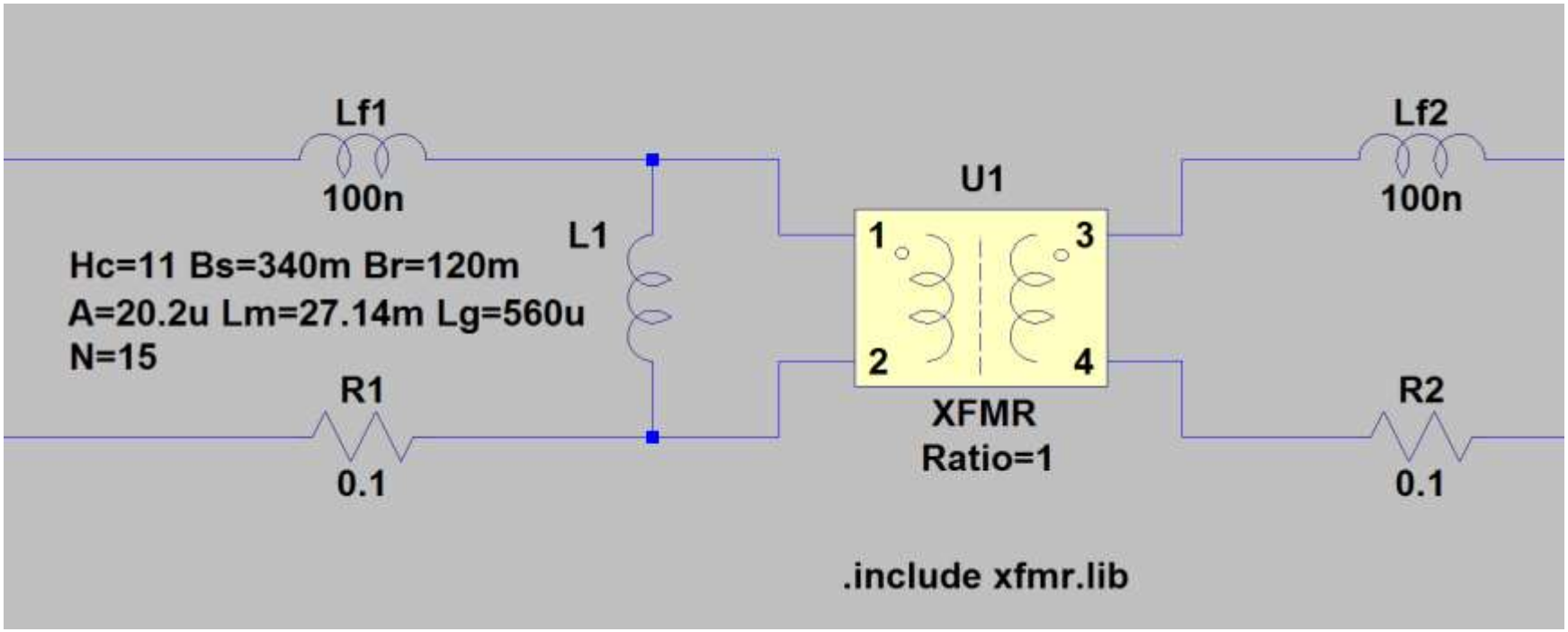
Bidirectional



Saturating Transformer



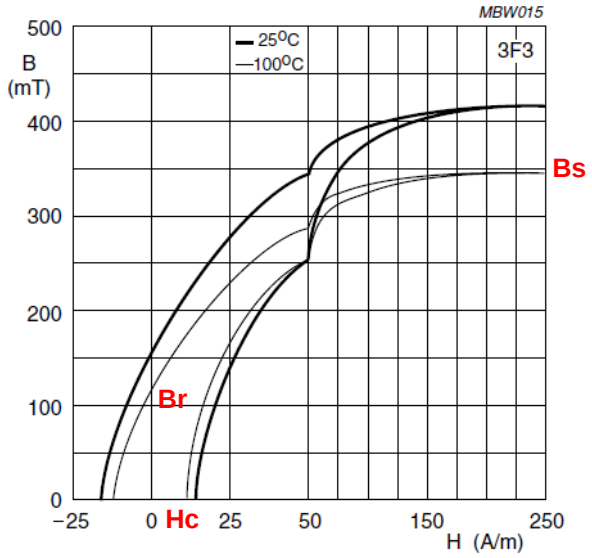
Saturating Transformer with Parasitic Elements



Transformer Parameters

Ferroxcube datasheet :

- **3F3 Material**
- **E13/6/6 core**



3F3 SPECIFICATIONS

A medium frequency power material for use in power and general purpose transformers at frequencies of 0.2 - 0.5 MHz.

SYMBOL	CONDITIONS	VALUE	UNIT
μ_i	25 °C; ≤10 kHz; 0.25 mT	2000 ±20%	
μ_a	100 °C; 25 kHz; 200 mT	≈ 4000	
B	25 °C; 10 kHz; 1200 A/m	≈ 440	mT
	100 °C; 10 kHz; 1200 A/m	≈ 370	
P_V	100 °C; 100 kHz; 100 mT	≤80	kW/m³
	100 °C; 400 kHz; 50 mT	≤150	
ρ	DC; 25 °C	≈ 2	Ωm
T_C		≥200	°C
density		≈ 4750	kg/m³

Effective core parameters

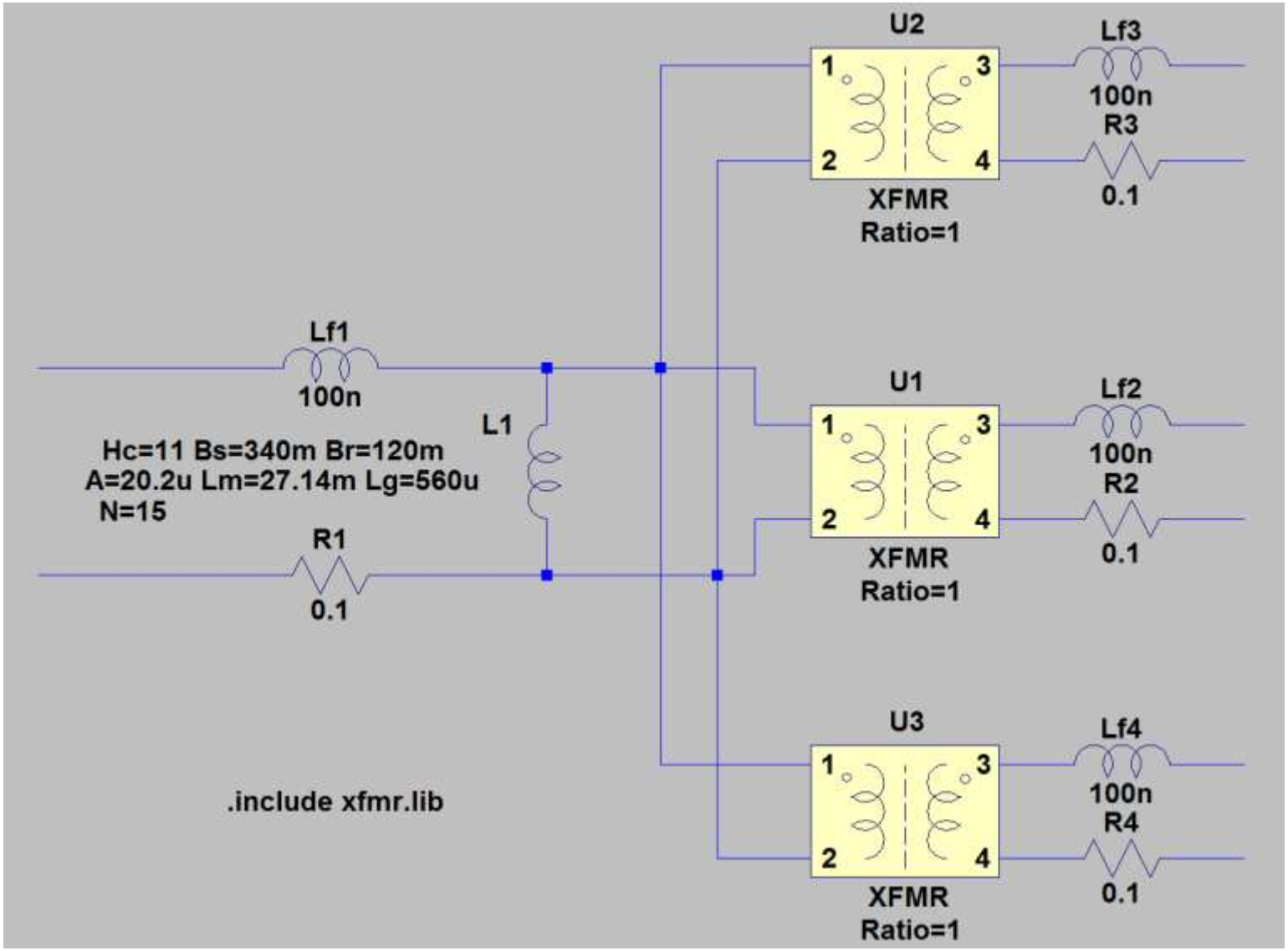
SYMBOL	PARAMETER	VALUE	UNIT
Σ(l/A)	core factor (C1)	1.37	mm^{-1}
V_e	effective volume	559	mm^3
l_e	effective length	27.7	mm
A_e	effective area	20.2	mm^2
A_{min}	minimum area	20.2	mm^2
m	mass of core half	≈ 1.4	g

Core halves

A_L measured in combination with a non-gapped core half, clamping force for A_L measurements, 15 ±5 N.

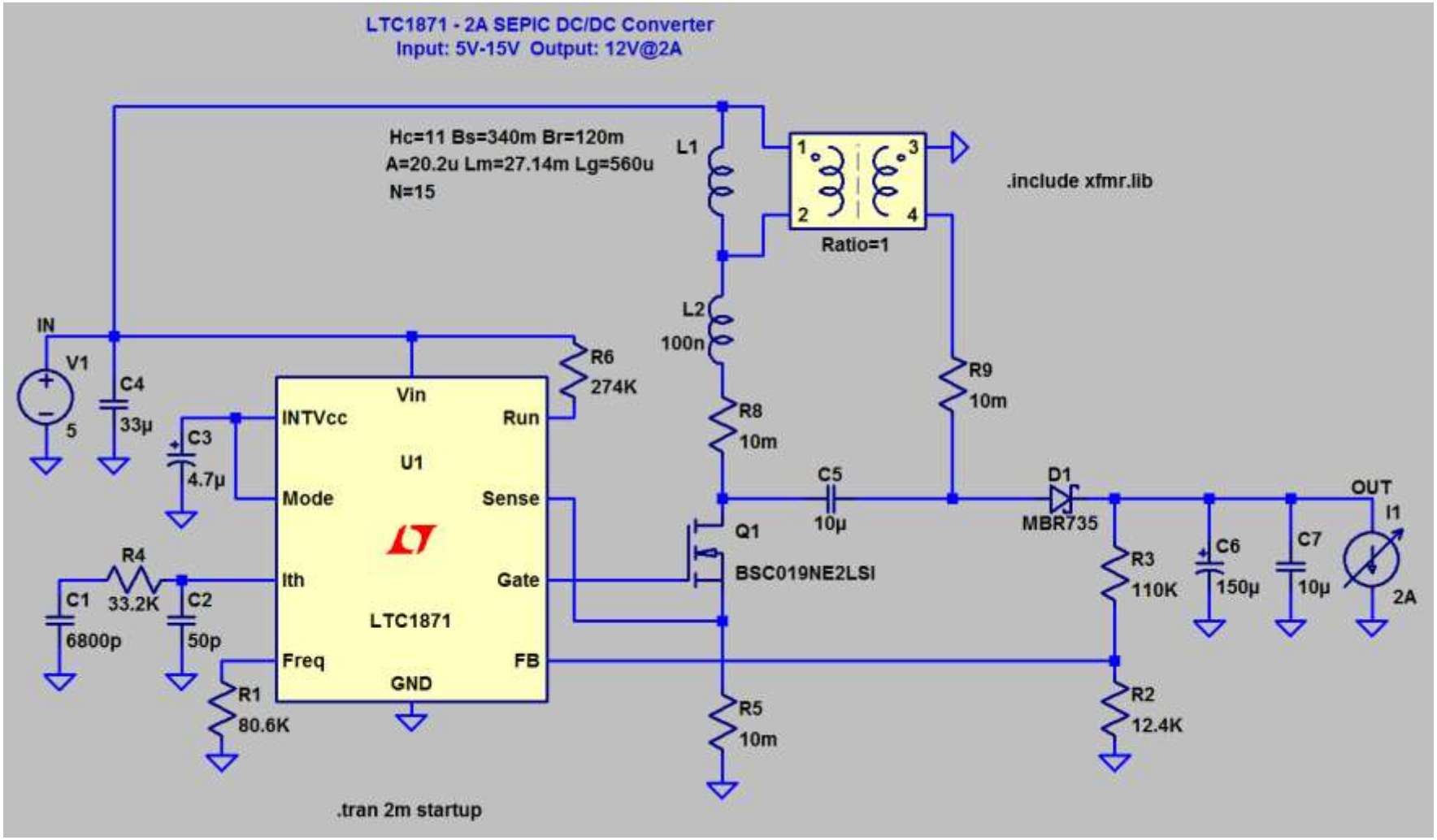
GRADE	A_L (nH)	μ_e	AIR GAP (μm)	TYPE NUMBER
3C90	63 ±5%	≈ 70	≈ 560	E13/6/6-3C90-A63
	100 ±8%	≈ 110	≈ 310	E13/6/6-3C90-A100
	160 ±8%	≈ 175	≈ 175	E13/6/6-3C90-A160
	250 ±20%	≈ 275	≈ 100	E13/6/6-3C90-A250
	315 ±20%	≈ 340	≈ 75	E13/6/6-3C90-A315
	1470 ±25%	≈ 1605	≈ 0	E13/6/6-3C90
3C92 des	1080 ±25%	≈ 1180	≈ 0	E13/6/6-3C92
3C94	1470 ±25%	≈ 1605	≈ 0	E13/6/6-3C94
3C96 des	1250 ±25%	≈ 1360	≈ 0	E13/6/6-3C96
3F3	63 ±5%	≈ 70	≈ 560	E13/6/6-3F3-A63
	100 ±8%	≈ 110	≈ 310	E13/6/6-3F3-A100
	160 ±8%	≈ 175	≈ 175	E13/6/6-3F3-A160
	250 ±20%	≈ 275	≈ 100	E13/6/6-3F3-A250
	315 ±20%	≈ 340	≈ 75	E13/6/6-3F3-A315
	1250 ±25%	≈ 1370	≈ 0	E13/6/6-3F3

Saturating Transformer with multiple windings

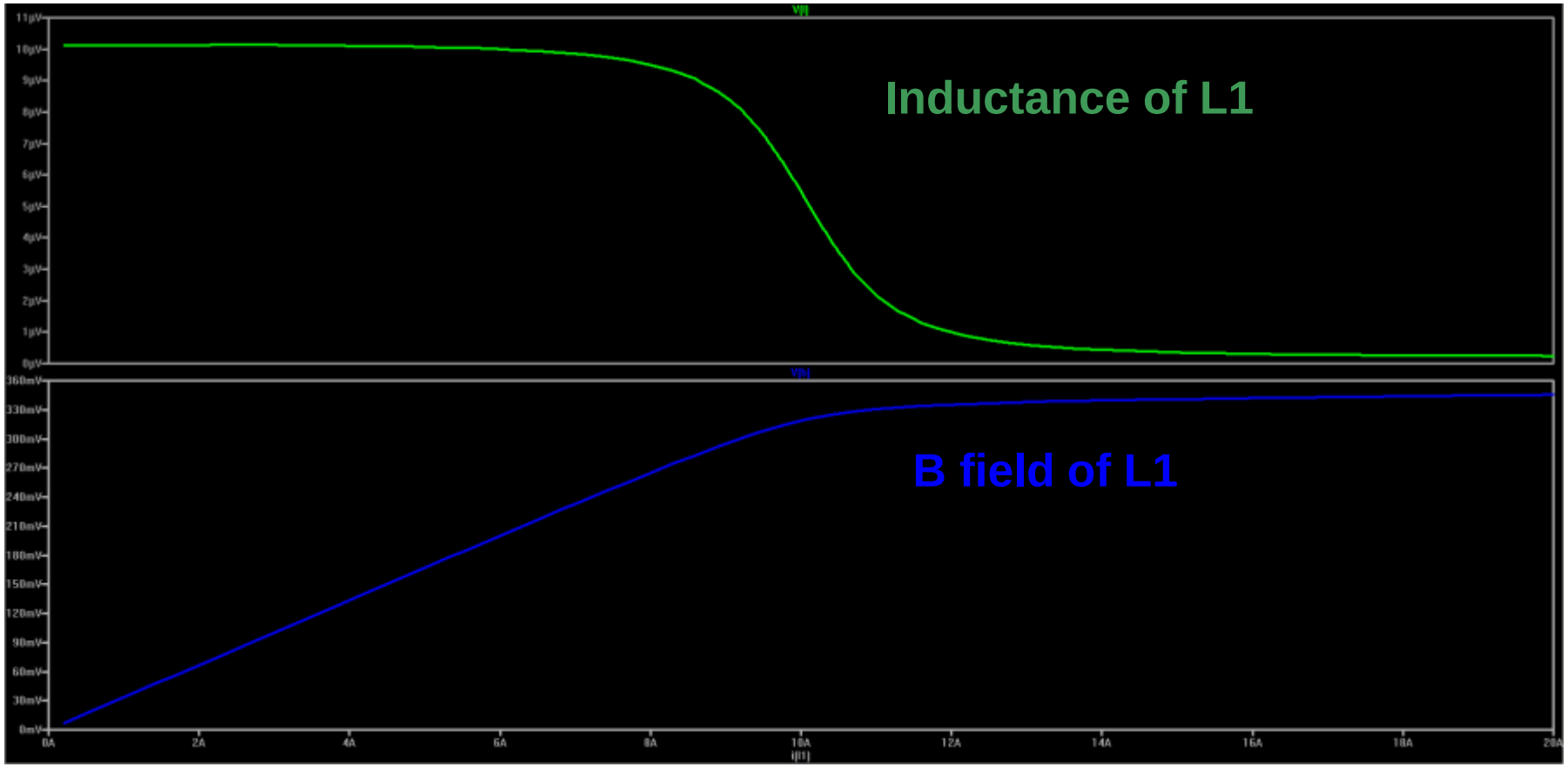


SEPIC with Saturating Transformer

- SEPIC with saturating XFMR.asc



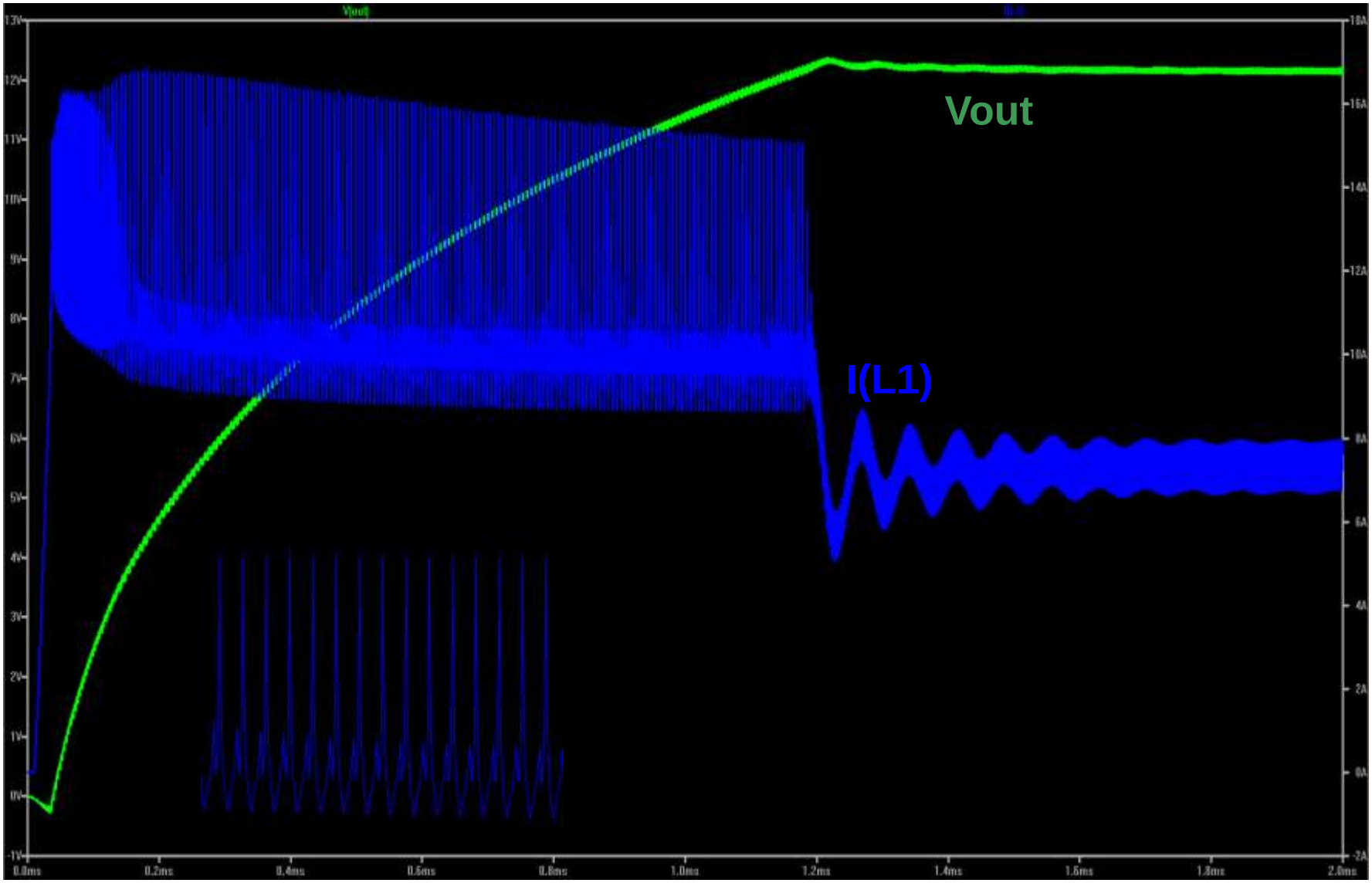
SEPIC with Saturating Transformer



Inductor Current

L = 10µH ; Isat(ΔL/L=20%) = 9.2A

SEPIC with Saturating Transformer



Importing Third-Party SPICE Models

Importing Third-Party Spice Models

To import a third party spice model:

1.) Download the spice model file from the manufacturer's website

2.) Make sure that the spice model file is located in the same directory as the LTspice simulation file

3.) Open up the spice model file and note the device name

4.) a- Place the model directly on the schematic

OR

b- Add the following spice directive to the LTspice simulation file

(Edit pull-down menu ---> SPICE Directive):
.include spice_model_file_name.abc

5.) Modify the device name in the LTspice schematic to match the device name contained in the spice model file (Right-Click on the device name, and modify the text accordingly)

Importing Third-Party Spice Models

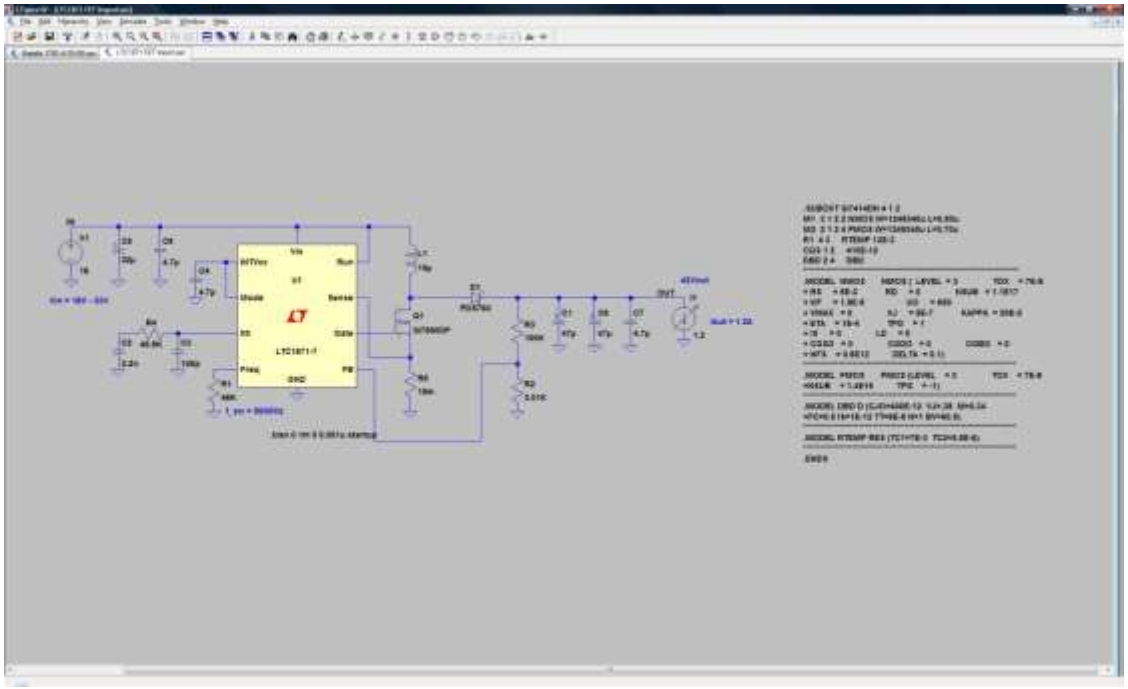
The following items are CRITICAL

1.) The file name in the .include statement must match the spice model file name identically. The file name syntax can be anything, just make sure that all of the characters match.

2.) The device name in the spice model file must match the device name in the LTspice schematic identically. The model name syntax can be anything, just make sure that all of the characters match.

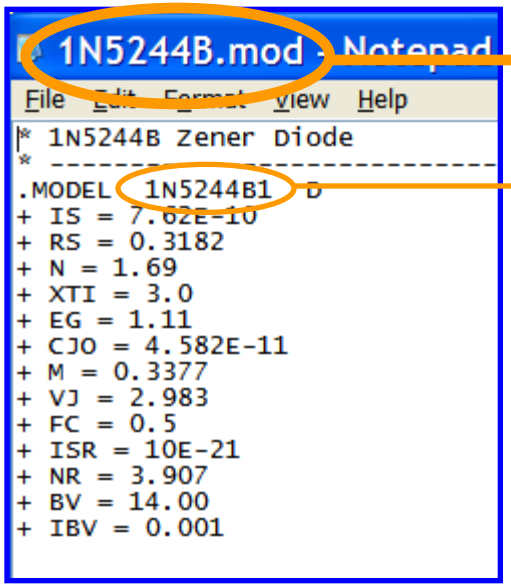
Making Circuit Files More Transportable

- Open model or subckt file and copy text to the clipboard
- Open new spice directive in LTspice
- Paste clipboard contents to directive text box
- Place on schematic
- Easier than using .include for simple models



Importing Third-Party Spice Models

Spice Model Example #1:

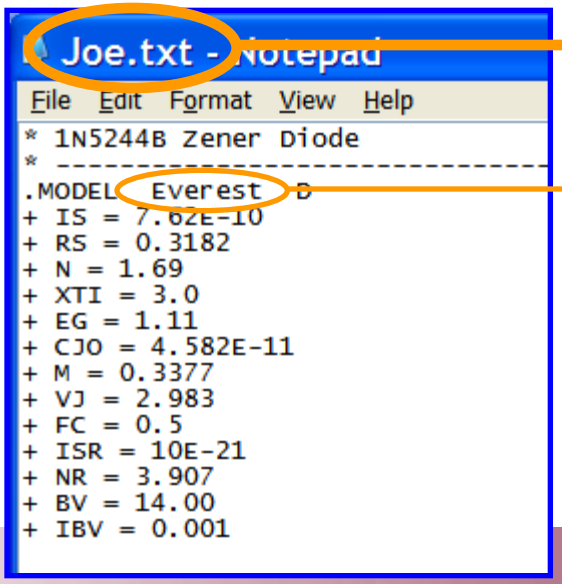


File name = 1N5244B.mod

Model name = 1N5244B1

Summary: The file and model names are irrelevant. Just make sure that the device name in the schematic and .include file_name match those of the spice model file.

Spice Model Example #2:



File name = Joe.txt

Model name = Everest

Examples

- Zener Import Example.asc
- 1N5244B Model On Schematic Example.asc
- 1N5244B Model Include Example.asc

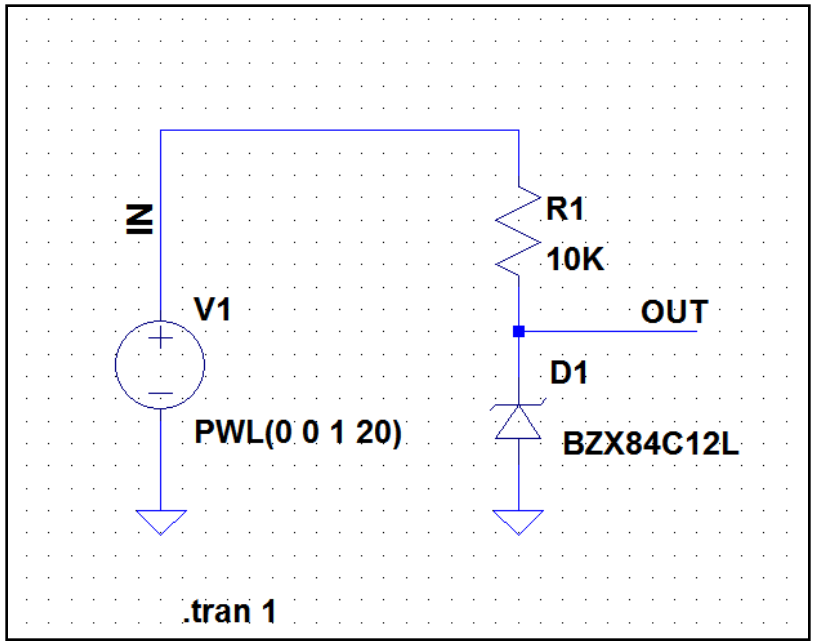
Importing Third-Party Spice Models

1) Open up the simulation file titled “Zener Import Example.asc”.

2) Open up the SPICE model file titled “1N5244B.mod” and note the device model name.

3) Modify the simulation file so that it uses the 1N5244B third-party SPICE model based on the instructions provided on the previous slides.

4) Run the simulation and probe the IN and OUT nodes.



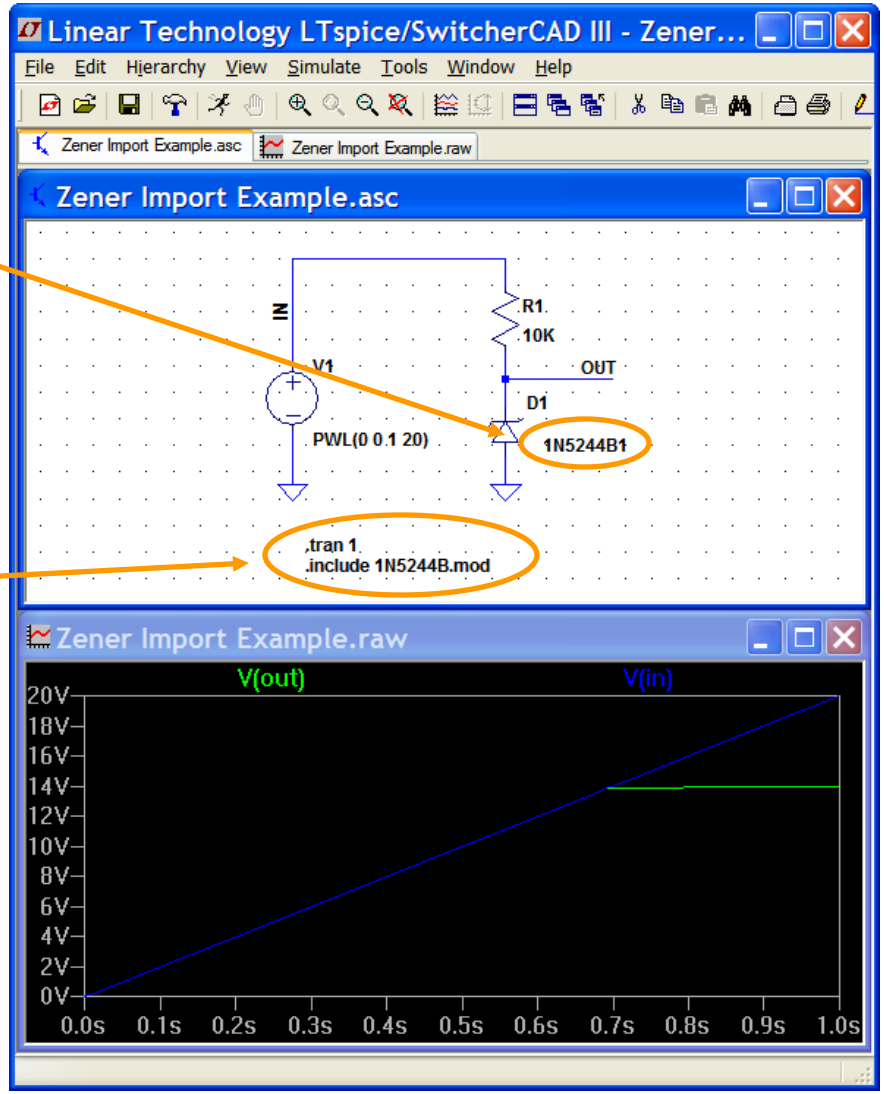
Importing Third-Party Spice Models

Solution:

1) Zener name changed to 1N5244B1 to match model name in the SPICE model file. Right-Click on the diode name text to change.

2) .include SPICE directive added to link to the SPICE model file. Use the Edit pulldown menu ---> Spice Directive to add this SPICE directive to your simulation.

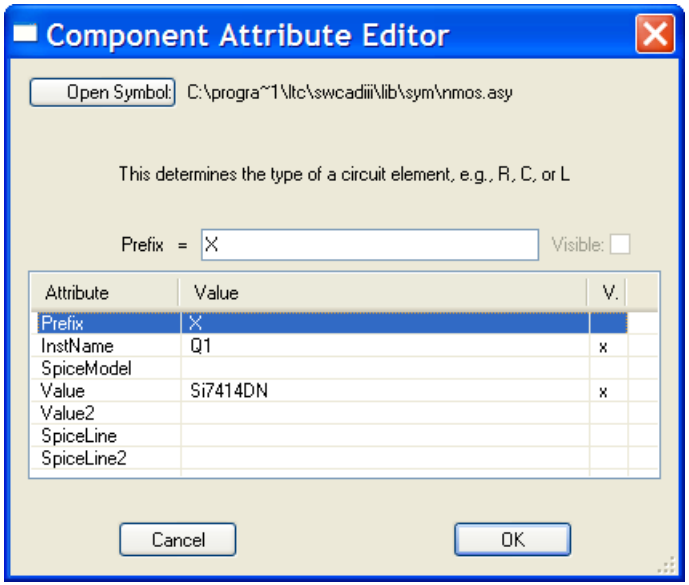
3) Result after clicking on the Running Person symbol on the toolbar and probing the IN and OUT nodes.



Importing Third-Party Spice Models

Types of SPICE Models (open up the SPICE model file to determine)

- **.MODEL definition (as covered in the previous Zener example)**
  1. **Change the device name in the simulation schematic to match the device name in the SPICE model file**
  2. **Add the SPICE directive to the schematic ".include spice_model_file.abc"**
- **.SUBCKT definition**
  1. **Same as above**
  2. **Same as above**
  3. **Must Ctrl-Right-Click on the device and change the Prefix to "X".**



Importing Third-Party Spice Models

Exercise:

- **Open up the simulation file titled “LTC1871 FET Import.asc” and follow the instructions in the simulation file.**

Selecting a MOSFET for a DC/DC Converter (LTC1871-7 Example)

Verifying an Appropriate MOSFET for a DC/DC Converter

Exercise:

- **Open up the simulation file titled “LTC1871 Boost.asc” and follow the instructions in the simulation file.**

Managing and Customizing Model Libraries

Managing and Customizing Model Libraries

LTspice Standard Library Files
(these can be opened and edited using LTspice)

- **RCL databases (easy to edit and expand)**
  - **standard.res**
  - **standard.cap**
  - **standard.ind**
  - **standard.bead**

Custom entries into the library files will not be removed by a Sync Release

- **Intrinsic Devices (more complicated to edit and expand)**
  - **standard.dio**
  - **standard.bjt**
  - **standard.mos**
  - **standard.jft**

File path for standard library files:
C:\Program Files\LTC\LTspiceIV\lib\cmp

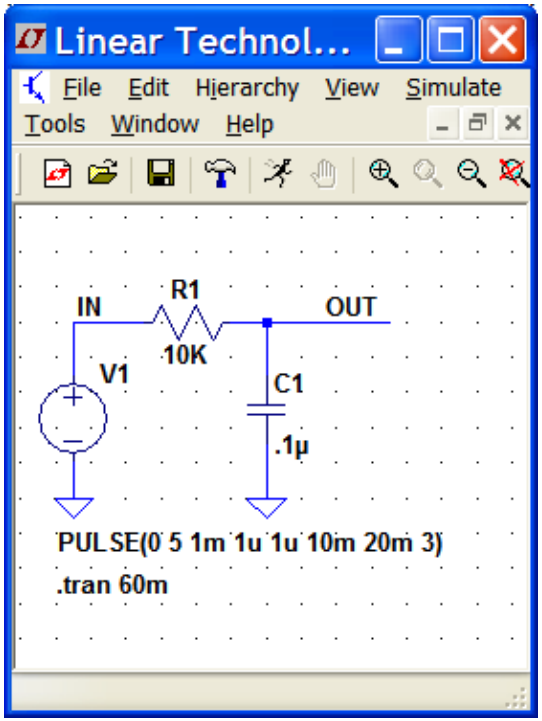
Managing and Customizing Model Libraries

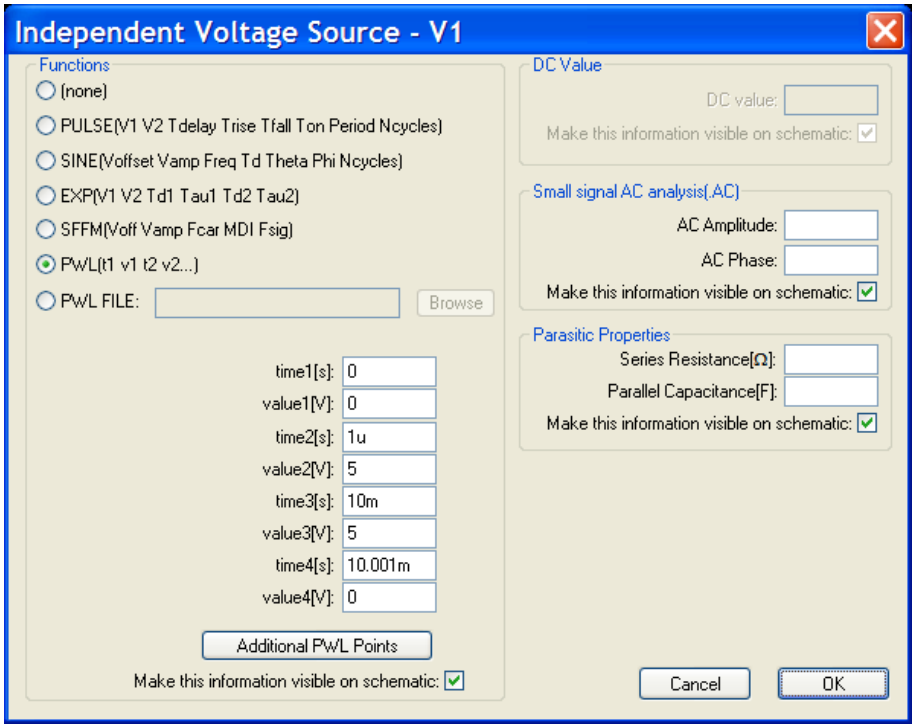
- Using LTspice, open up the RCL (resistor, capacitor, inductor) libraries and explore. File path to the library files: C:\Program Files\LTC\LTspiceIV\lib\cmp
- Using LTspice, open up intrinsic devices and explore. Modify "Zener Library Example.asc" to use a library file. Follow the instructions in the simulation file.
- Run "NPN and Library.asc". Notice that the simulation calls out the same library file as the Zener Library Example.asc. A library file can contain models for multiple devices.
- <u>If you add devices to the library files, your devices will *not* be removed when running a Sync Release.</u>

Piece-Wise Linear (PWL) Voltage Sources

Creating a PWL Voltage Source

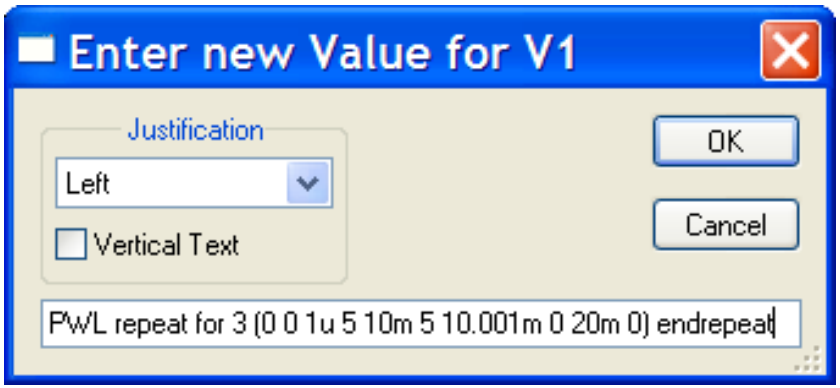
- Open up the simulation file titled “RC Filter Time Domain.asc”
- Run the simulation and probe the IN and OUT nodes
- Right-Click on the voltage source and select the PWL function
- Configure the PWL source to manually recreate the pulse waveform as shown in the voltage source window on the right
- Rerun the simulation. Notice a single pulse is now present.

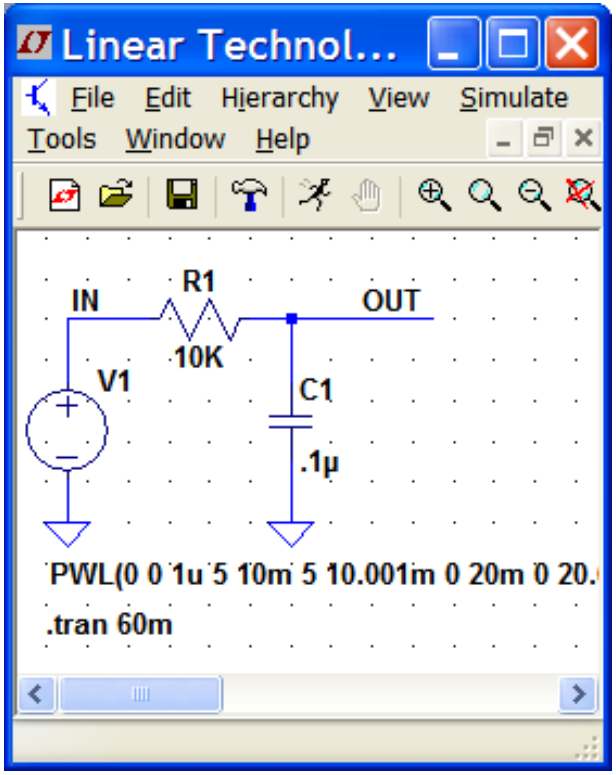




Repeating PWL Source – RC Circuit Revisited

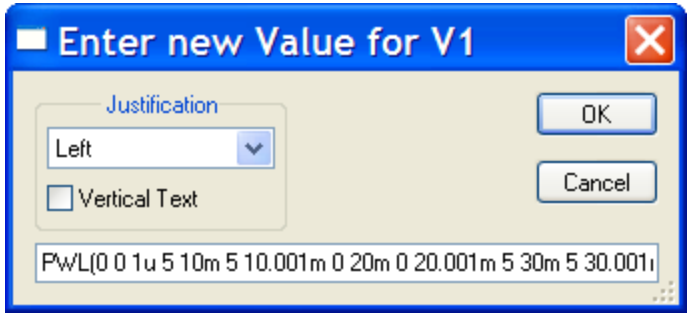
- Open up the simulation file titled "RC Filter Time Domain PWL.asc"
- Run the simulation and probe the IN and OUT nodes
- Right-Click on the PWL text string and use the repeat command to create three cycles of the input square wave.

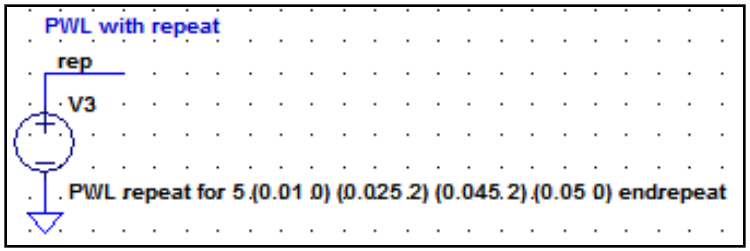


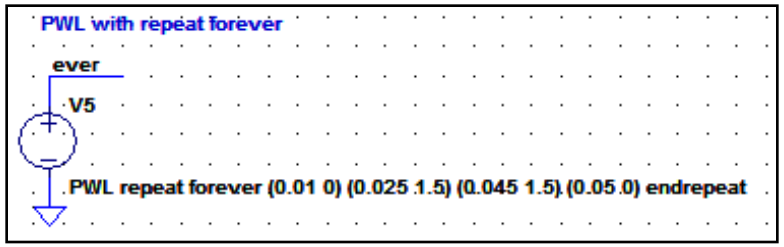


Repeating PWL Source – Additional Info

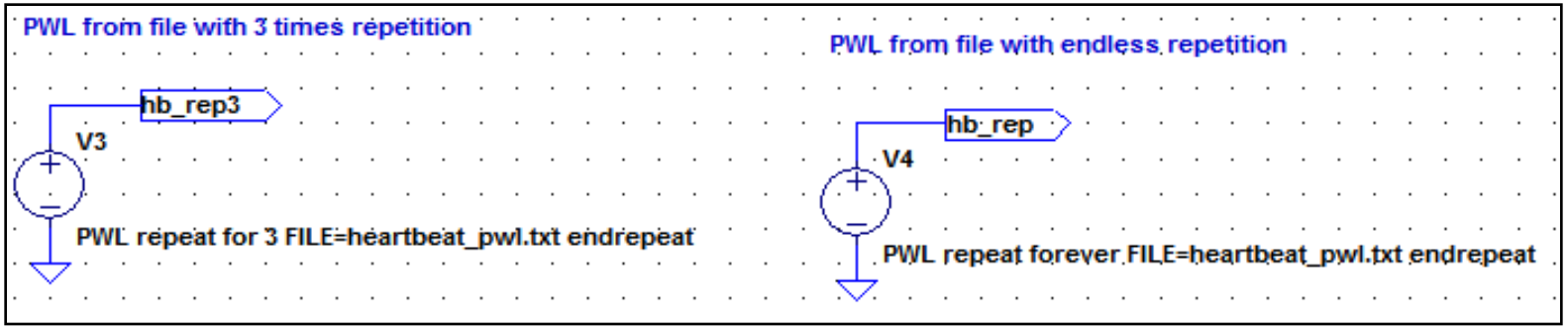
- To edit the PWL source attributes, Right-Click on the PWL text string on the schematic
- The following window will appear -->
- In the command line, modify the PWL command (examples below)







- Repeating PWL format using a PWL source file (see next page):



Importing Externally Generated PWL Sources

- To import a PWL waveform from a file, Right-Click on a voltage source, select “Advanced”, and select “PWL File”
- The file format must contain pairs of numbers separated by white space (carriage return, spaces, tabs). The first number is time (in seconds) and the second number is voltage.
- Example 1:
  - 0 0 0.1 1 0.2 0.5 0.5 0 0.7 0.3 1 0
- Example 2:
  - 0 0
  - 0.1 1
  - 0.2 0.5
  - 0.5 0
  - 0.7 0.3
  - 1 0

- Open the file “PWL Examples.asc”

More Topics #2

1. What is Usually Modeled (and What Isn't)
2. Making Circuit Files More Transportable
3. Behavioral Sources
4. Hierarchical Schematics and Automatic Creation of a Schematic Symbol
5. Parameters and Expression Evaluation
6. Thermistor Simulations: Plotting Temperature and Resistance
7. Voltage and Current Controlled Switches
8. Small signal analysis
9. Improving Simulation Speed

What Usually *is* Modeled?

- Typical performance at room temperature
- Error amp
  - Gm
  - Source/Sink Current
- Oscillator
  - Frequency
  - Duty Cycle Limits
- Switch logic
- Switch current limit
- Switch beta
- Peak current vs. error voltage
- Slope compensation
- Burst Mode
- Switch minimum on time
- Pulse skipping
- PLL capture & phase lock

What Usually is *Not* Modeled?

- **Production scatter**
- **Behavior over temperature**
- **Catastrophic failure modes**
- **Oscillator injection locked SYNC pin (unless the device has a PLL)**

What May or May Not be Modeled?

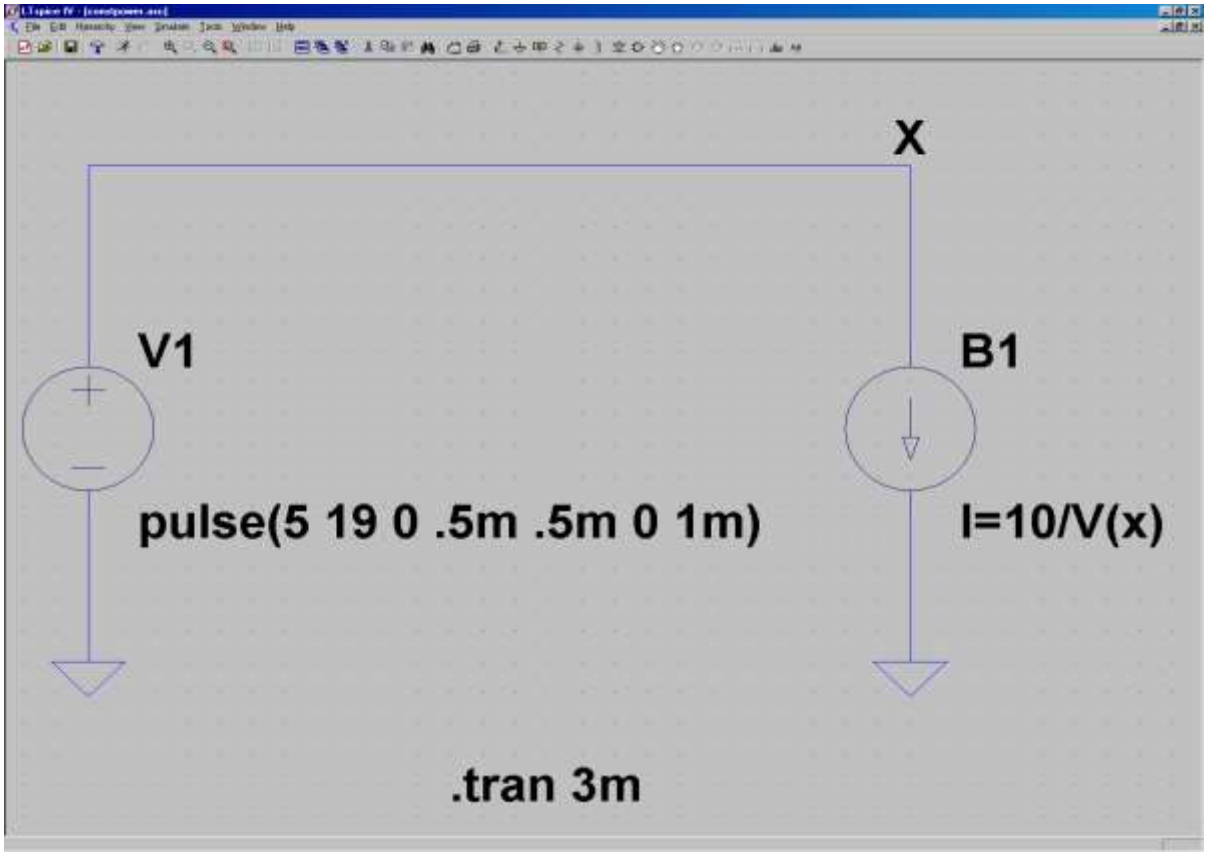
- **Iq in all modes**
- **Misc features in shutdown**

Behavioral Sources

- **Behavioral sources** are used when the user would like to define a source with an **arbitrary expression**.
- Expressions can contain the following:
  - **Node voltages**, e.g., V(n001)
  - **Node voltage differences**, e.g., V(n001, n002)
  - **Circuit element currents**; for example, I(S1), the current through switch S1 or Ib(Q1), the base current of Q1. However, it is assumed that the circuit element current is varying quasi-statically, that is, there is no instantaneous feedback between the current through the referenced device and the behavioral source output. Similarly, any ac component of such a device current is assumed to be zero in a small signal linear .AC analysis.
  - The keyword, "**time**" meaning the current time in the simulation.
  - The keyword "pi" meaning 3.14159265358979323846.
  - Various **functions** and **operations** as defined in the help file.

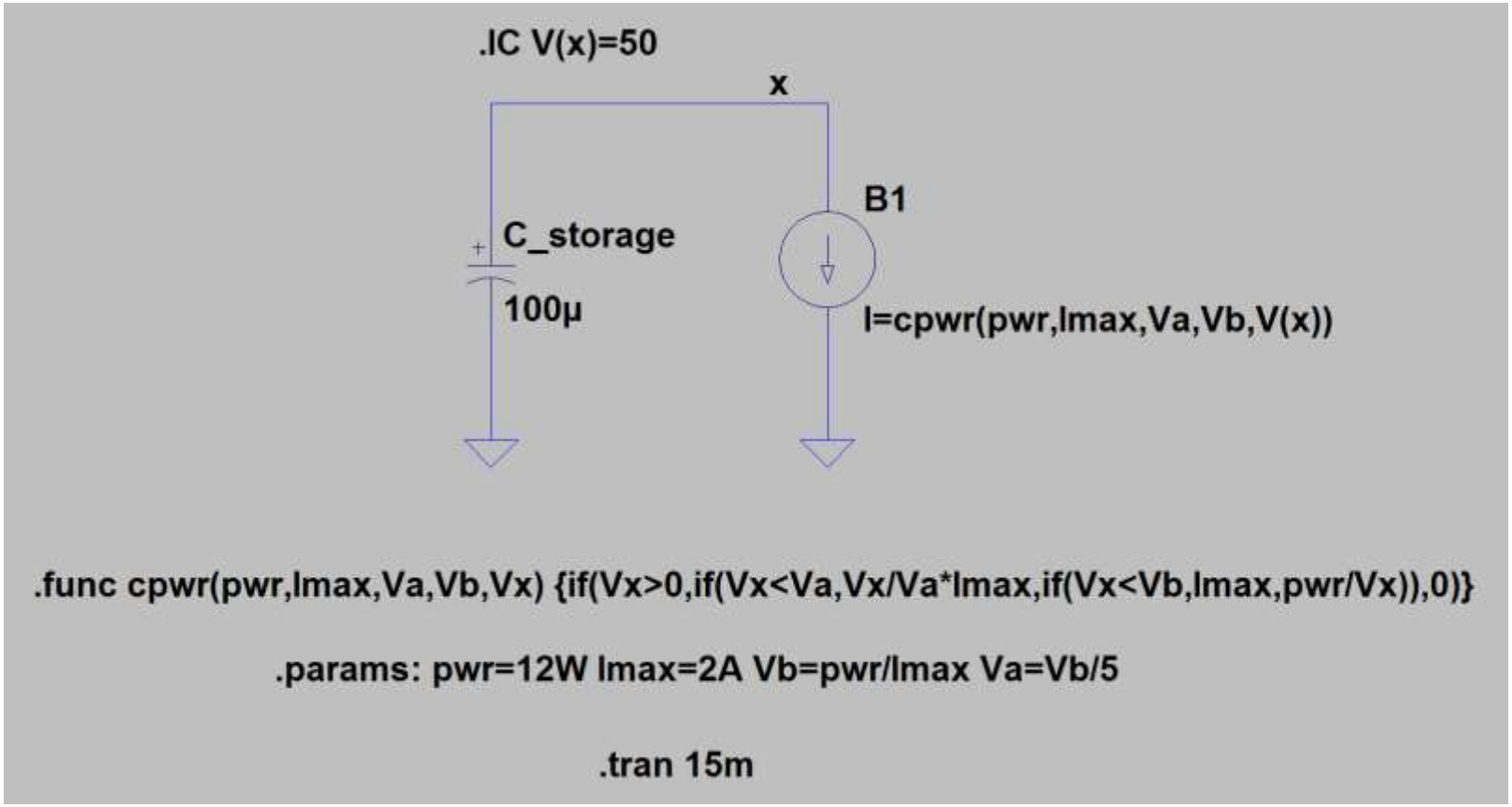
Constant Power Load

- Open up the simulation file titled “Const Power.asc” and follow the instructions in the simulation file to create a constant power load.



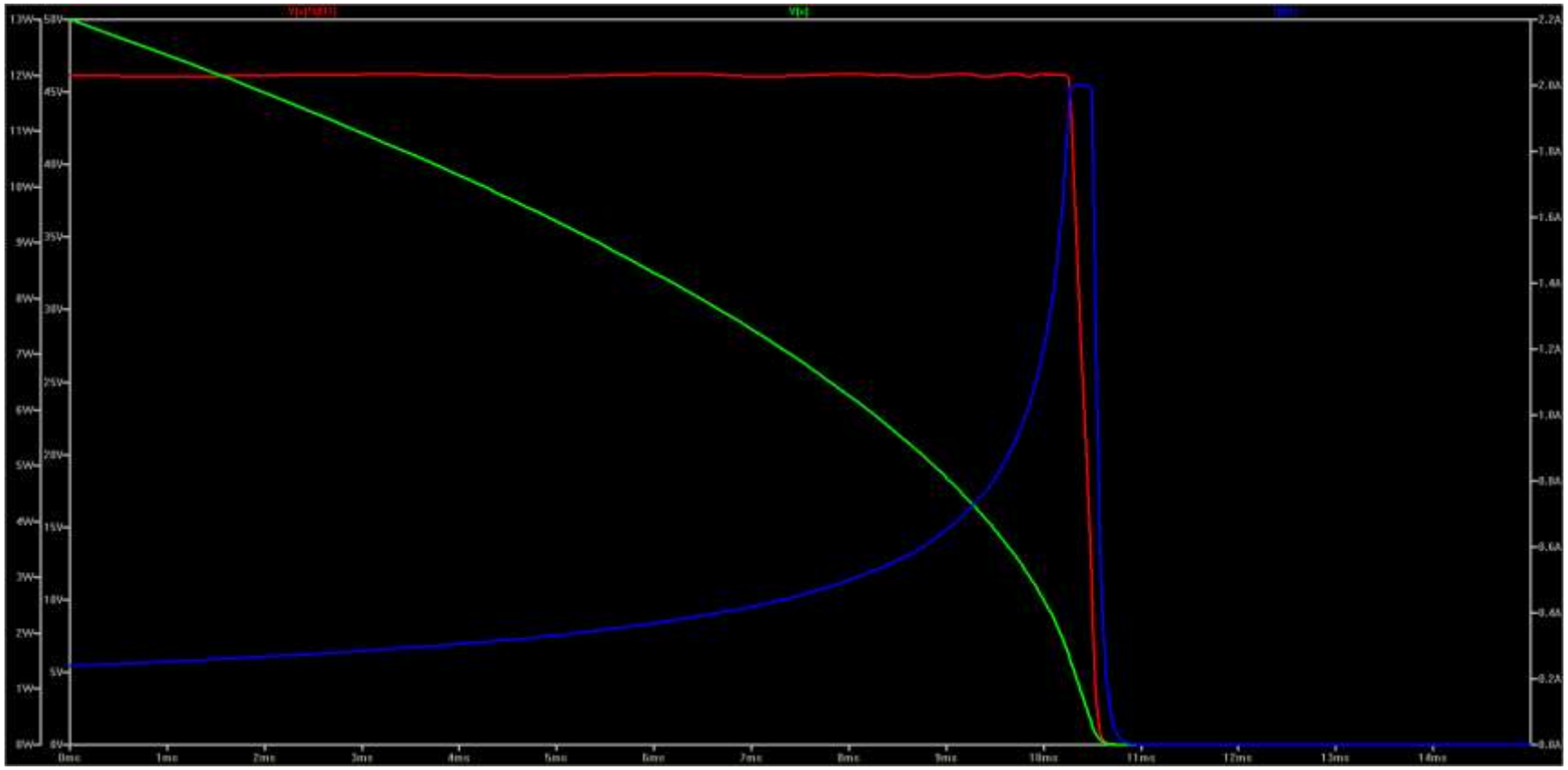
Constant Power Load with Limited Current

- **Constant power load with limited current.asc**



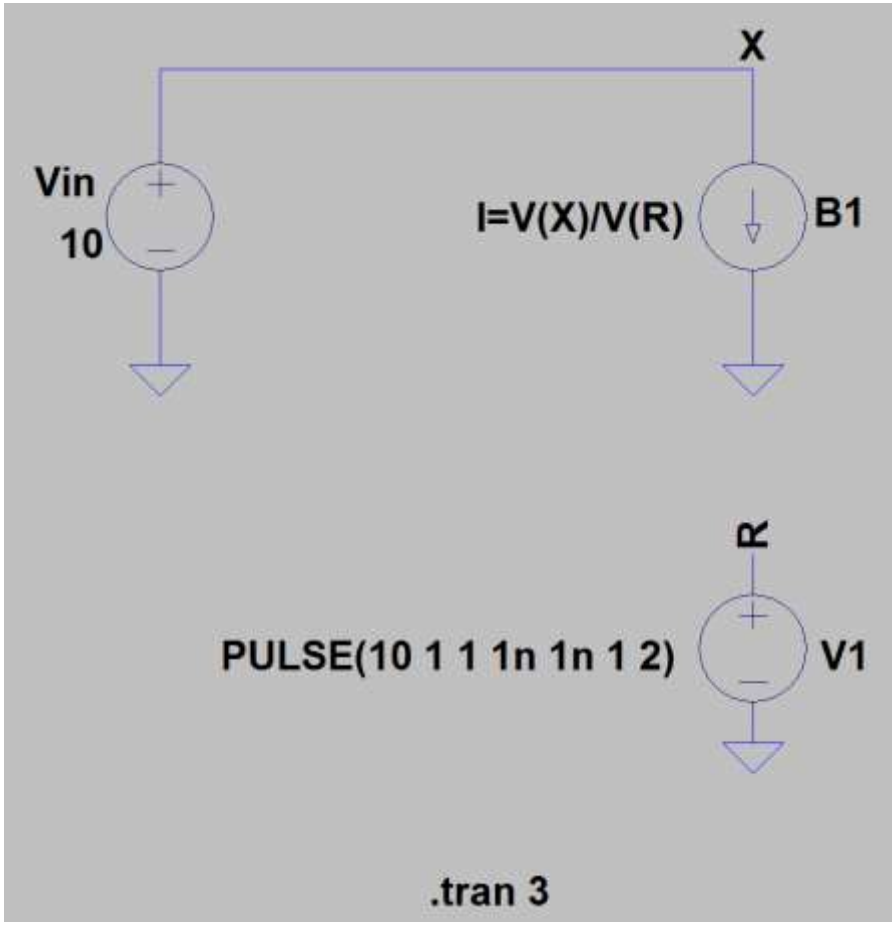
Storage Capacitor Discharge with Constant Power Load

Load Power **Load Voltage** **Load Current**



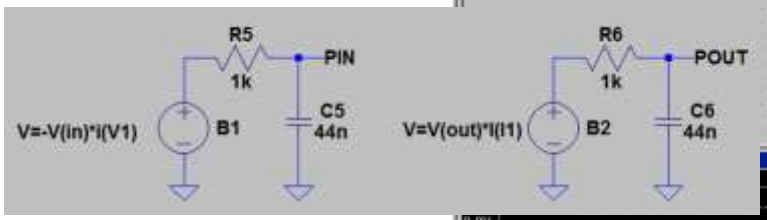
Variable R vs. Time

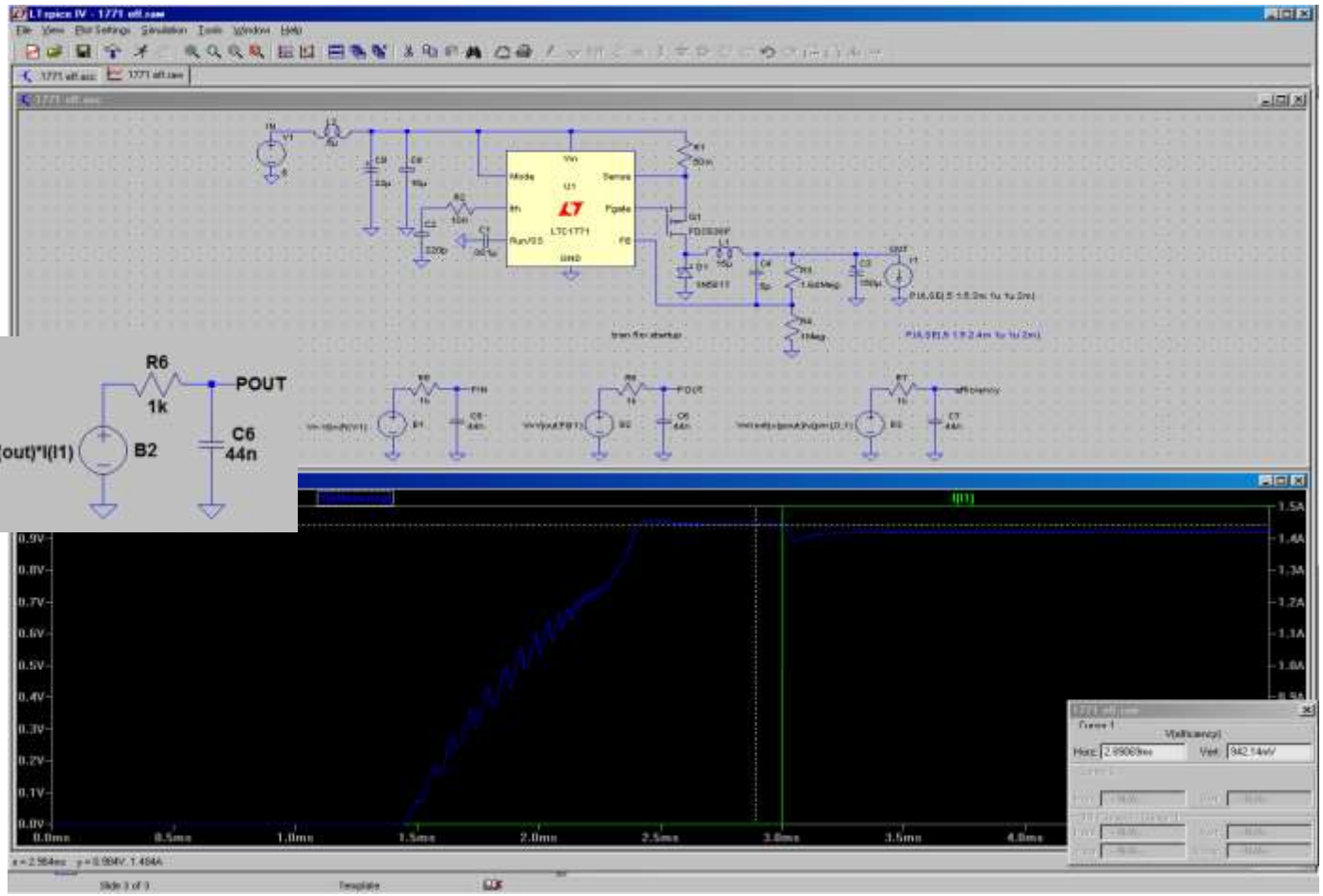
- Variable R versus time (B source).asc



Behavioral Sources

- Open up the simulation file titled “LTC1771 Efficiency.asc” and follow the instructions in the simulation file.

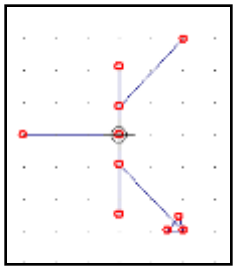




Creating a Schematic Symbol

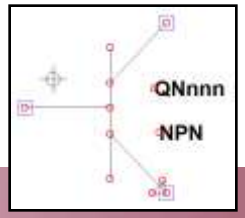
Creating a Schematic Symbol

Creating a NPN Transistor Schematic Symbol



1. **Open up LTspice**
2. **File pull down menu ---> New Symbol**
3. **Draw pull down menu ---> Line. Draft an NPN symbol.**
4. **Edit pull down menu ---> Add Pin/Port. These are the actual electrical connections for B (base), C (collector), E (emitter). Netlist order for the pins: C = 1, B = 2, E = 3.**
5. **Edit pull down menu ---> Attributes ---> Edit Attributes (to add attributes). See the screen shot here ---------->**

6. **Edit pull down menu ---> Attributes ---> Attribute Window (for attribute visibility) to make QN and NPN visible for the symbol**



Creating a Schematic Symbol

Creating a NPN Transistor Schematic Symbol (continued…..)

7. **Save the schematic symbol as “My_NPN.asy” in the same folder as the simulation file titled “NPN Schematic Symbol Import.asc”**
8. **Open up the simulation file titled “NPN Schematic Symbol Import.asc” and follow the instructions in the simulation file.**

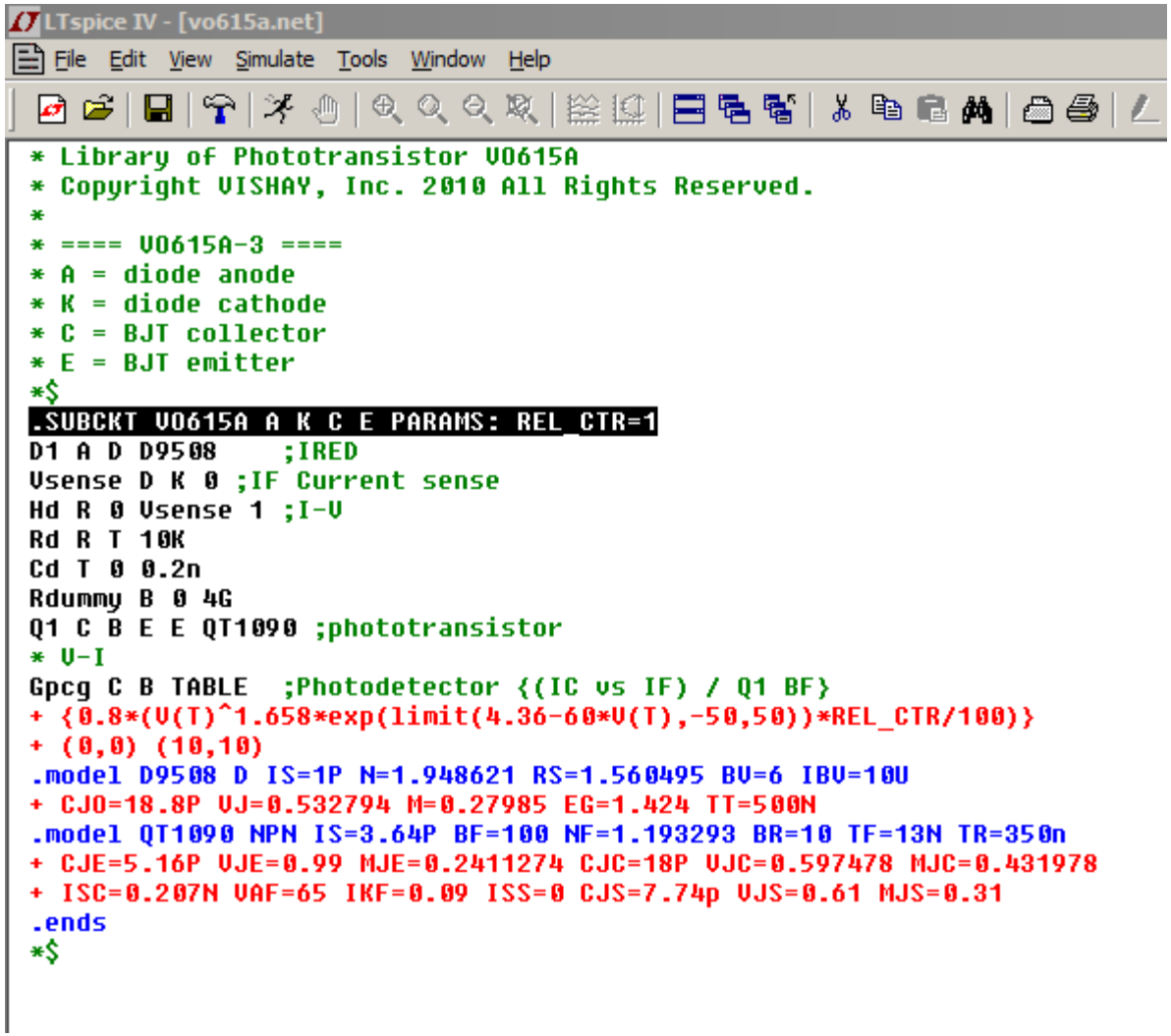
Automatic Symbol Generation from a Library File

Procedure

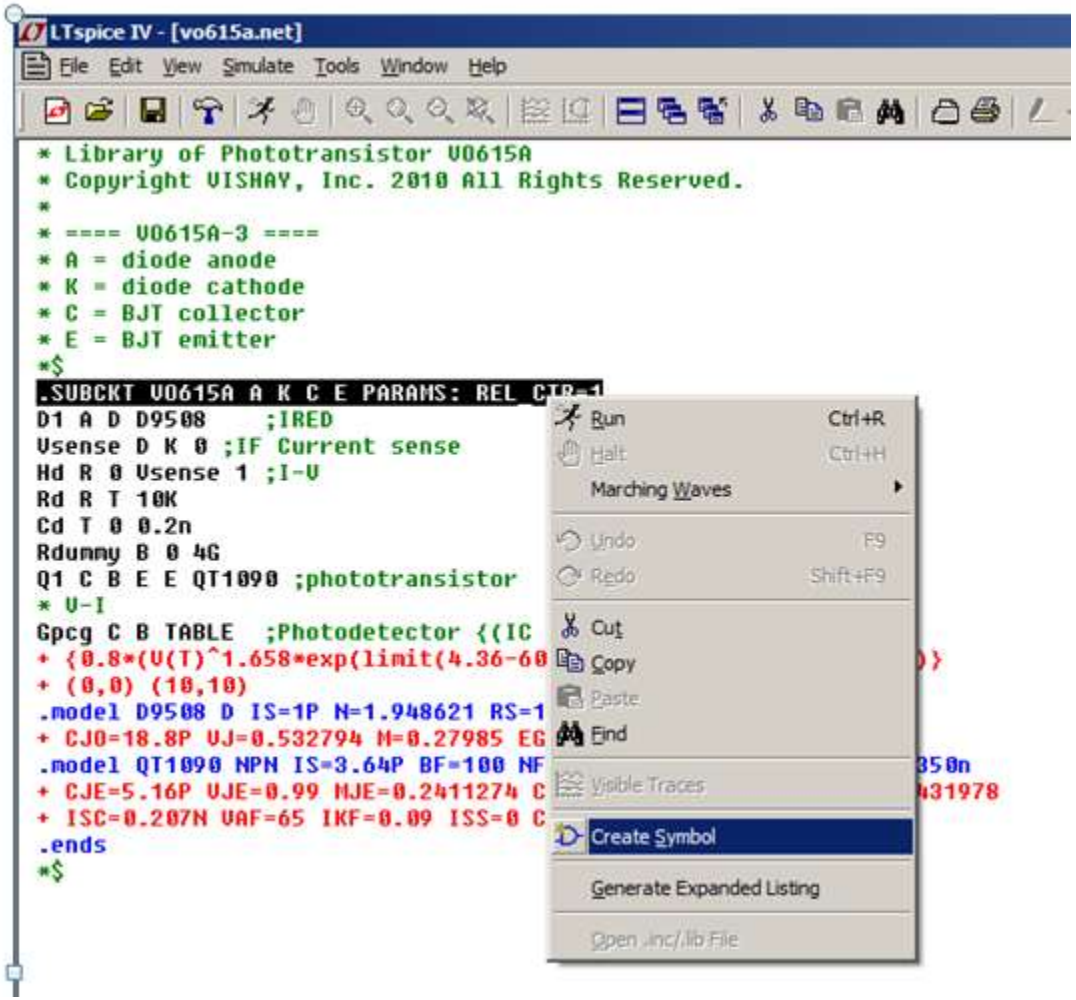
- Change the file extension to .net
- Edit the netlist file that contains subcircuit definition
- Place the cursor on the line containing the name of the subcircuit
- Right click and execute context menu item “Create Symbol”
- Click “Yes” to “Do you wish to automatically create a symbol …”
- The symbol is automatically saved in the LTspice directory : LTspiceIV \ lib \ sym \ AutoGenerated

  - Exercise : create the symbol of the optocoupler VO615A from the file vo615a.lib

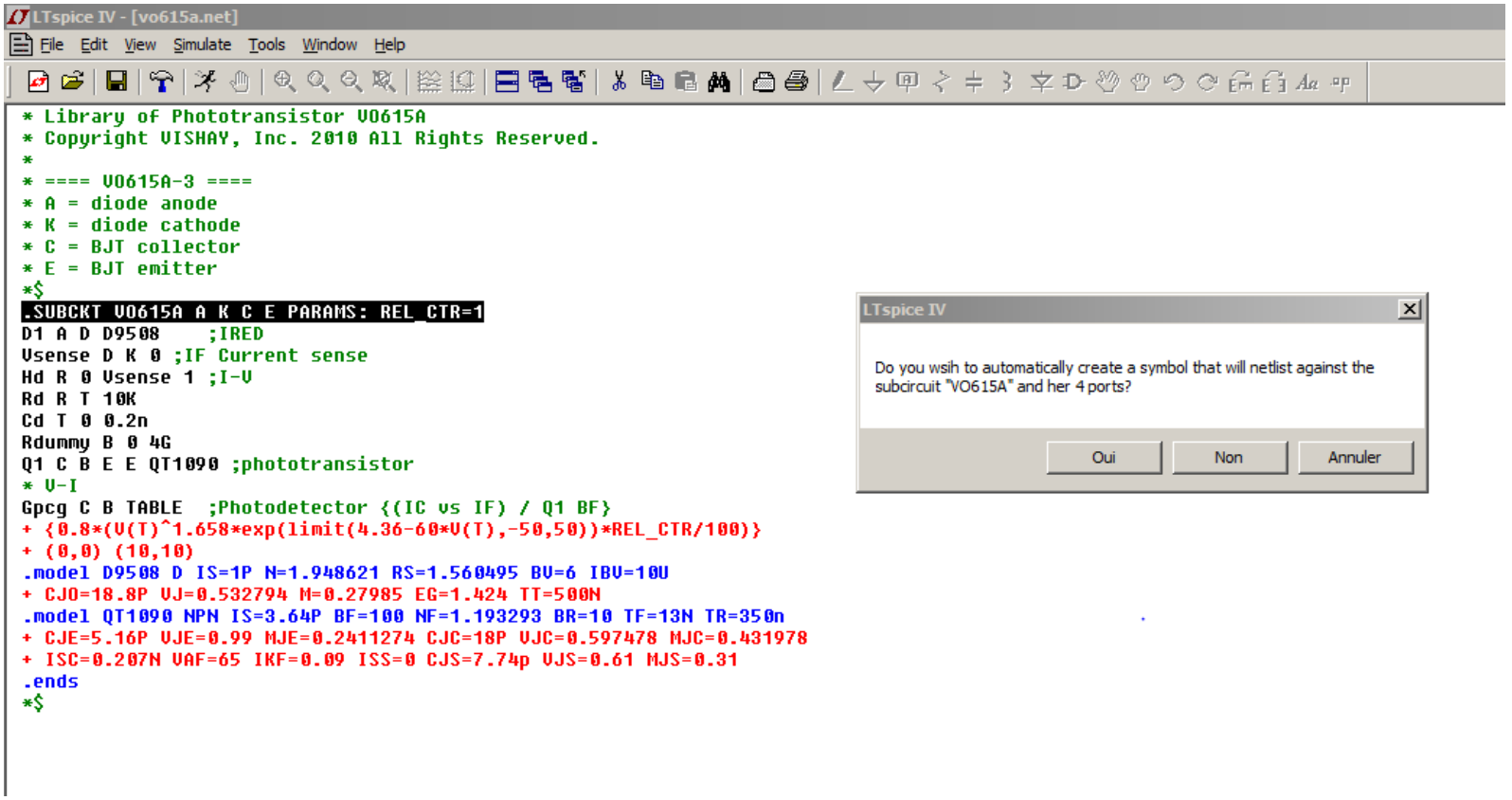
Subcircuit Definition



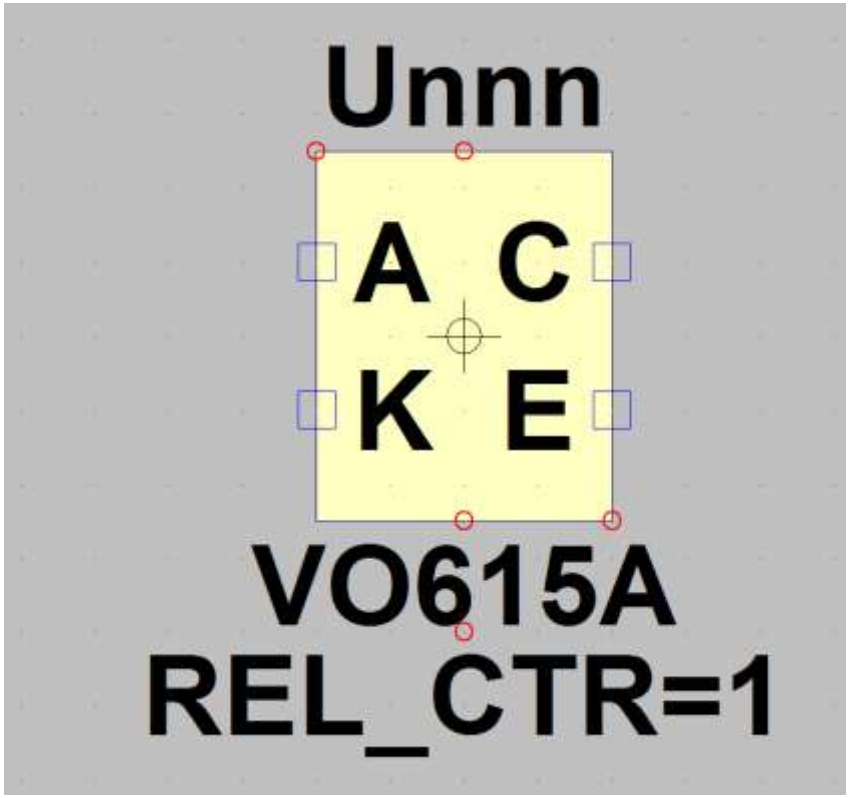
Create Symbol



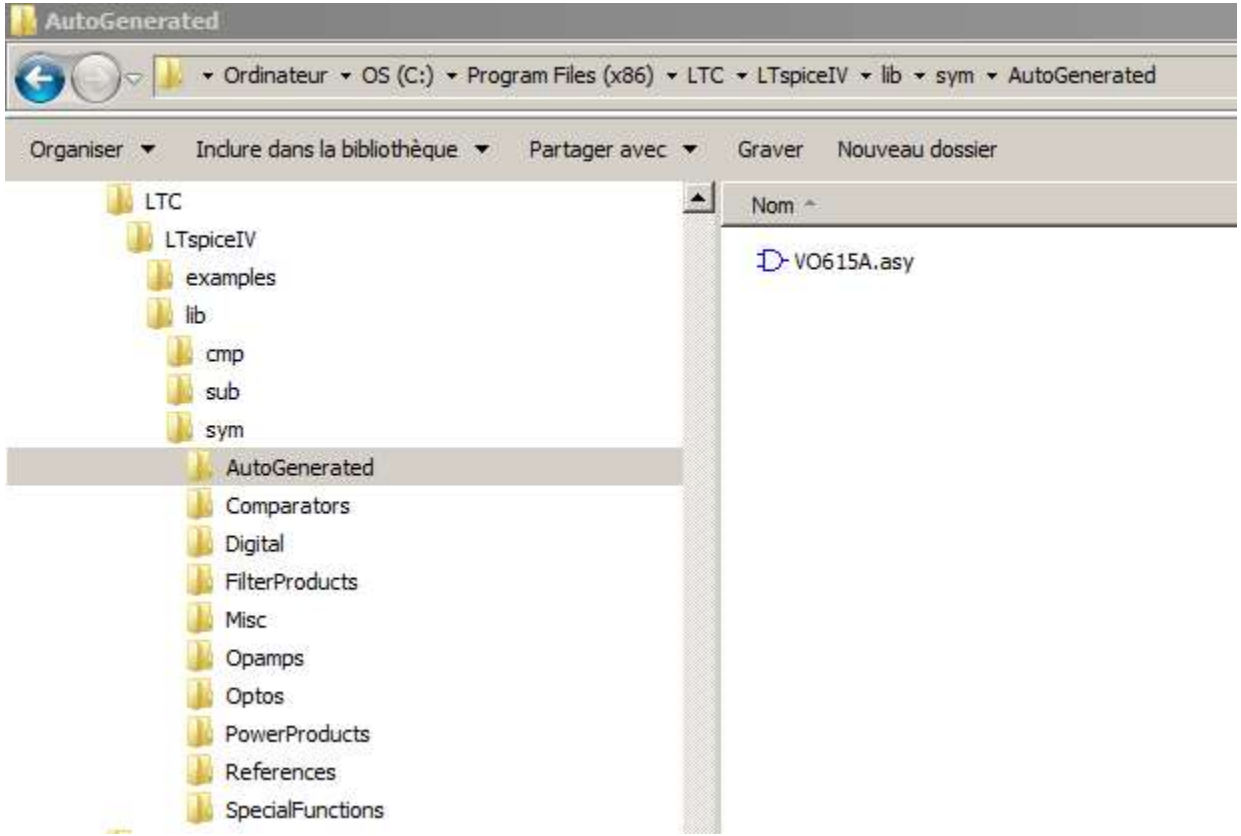
Create Symbol



Default Symbol



Saving Folder



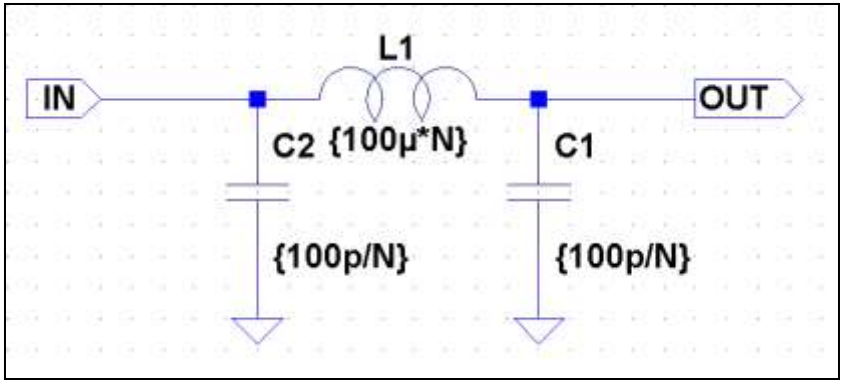
Hierarchical Schematics and Automatic Creation of a Schematic Symbol

Hierarchical Schematics / Schematic Symbols

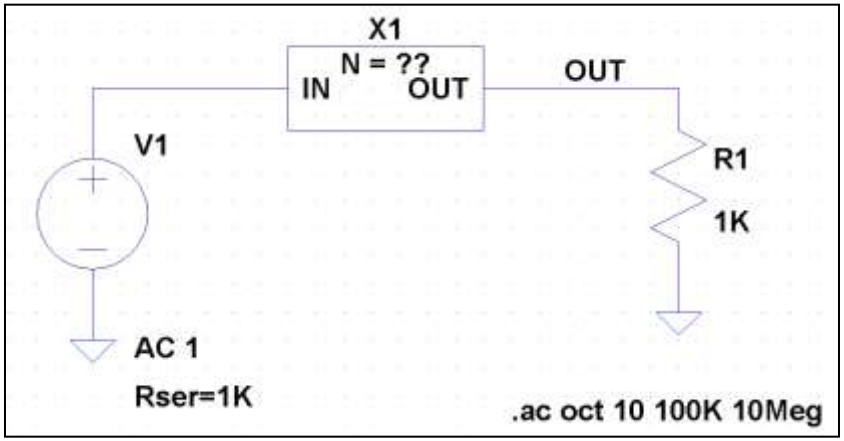
- Hierarchical schematic drafting has powerful advantages
  - Much larger circuits can be drafted than can fit onto a one sheet schematic while retaining the clarity of the smaller schematics
  - Repeated circuitry to be easily handled in an abstract manner, i.e. “black boxes” with full functionality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

Hierarchical Schematics / Schematic Symbols

The following subcircuit……..



…..can be automatically converted to a hierarchical schematic symbol (X1):

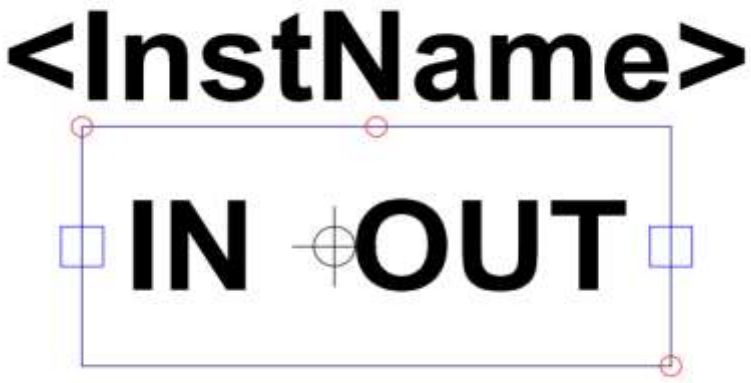


Hierarchical Schematics / Schematic Symbols

- **Exercise:**
- **Open up the simulation file titled “PISectionExample.asc” and follow the instructions in the simulation file.**

Hierarchical Schematics / Schematic Symbols

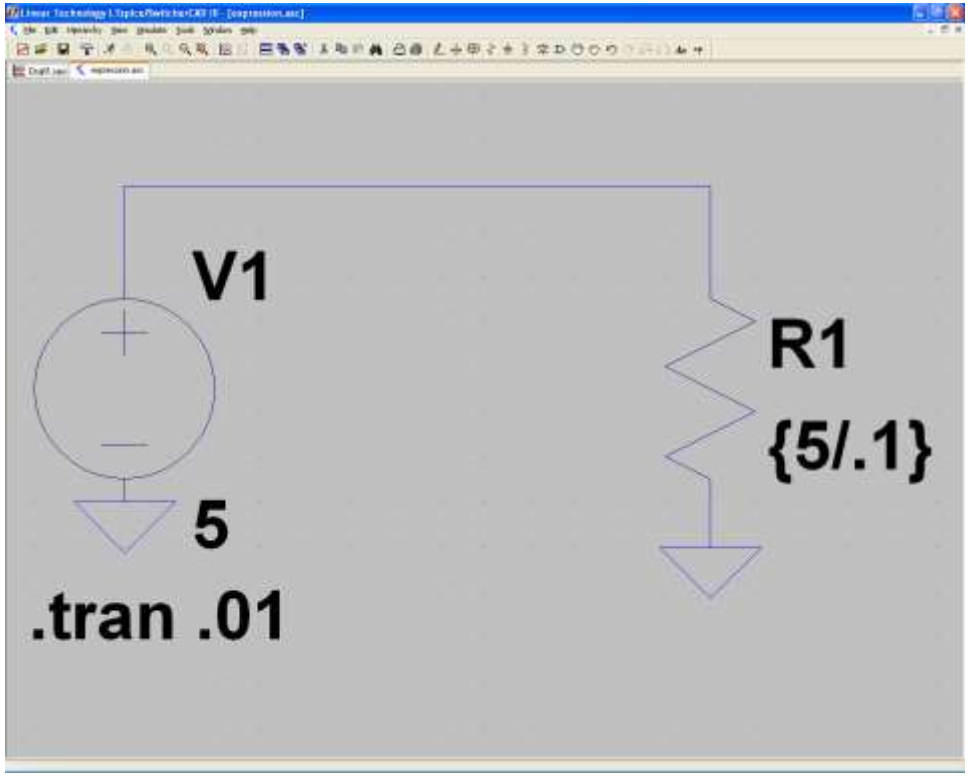
- After clicking “Yes” a new sheet titled “PISectionExample.asy” will open
- This hierarchical schematic is automatically saved to the folder – there is no need to manually save
- There is no further effort required, i.e. change <InstName>
- Both the schematic and assembly windows can be closed



Expression & Parameters

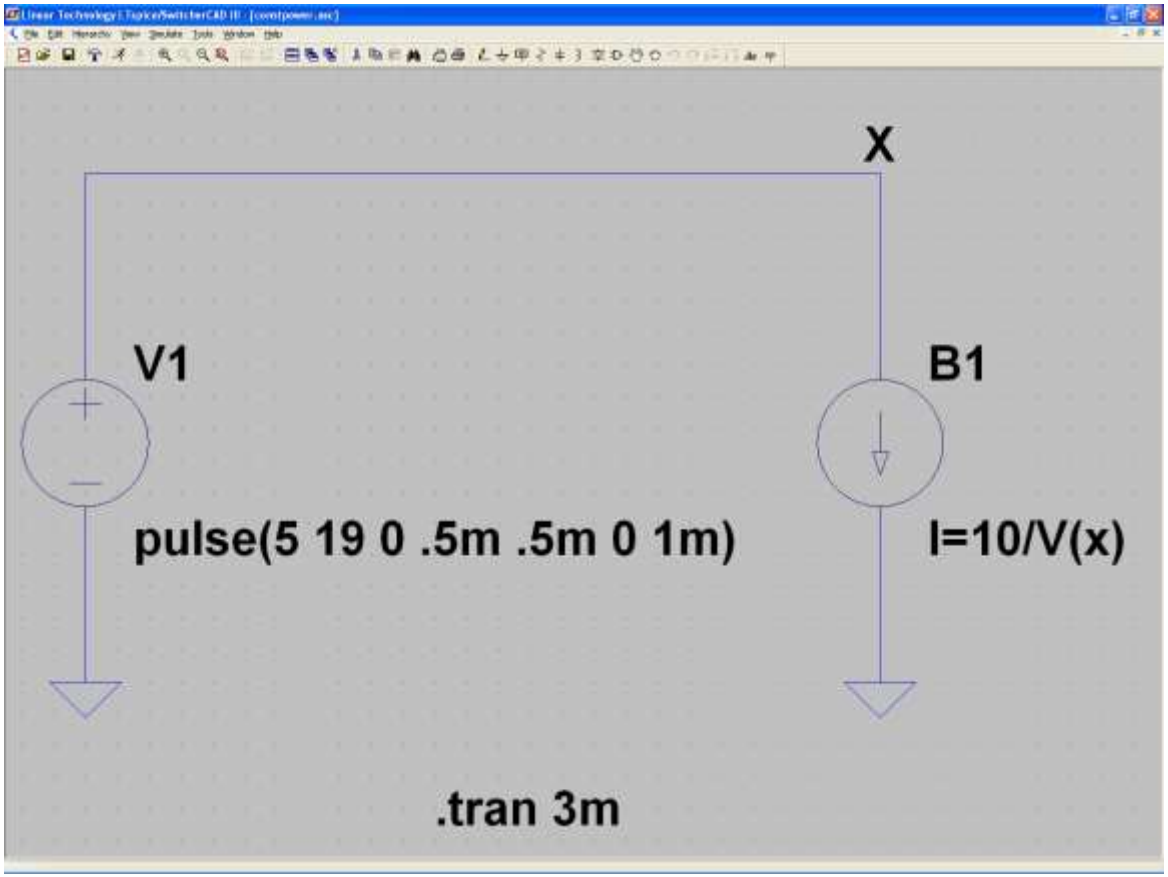
Expression Evaluation

- When curly braces **{ }** are encountered, the enclosed expression is evaluated on the basis of all relations available at the scope and reduced to a floating point value (**evaluated *before* simulation begins**).



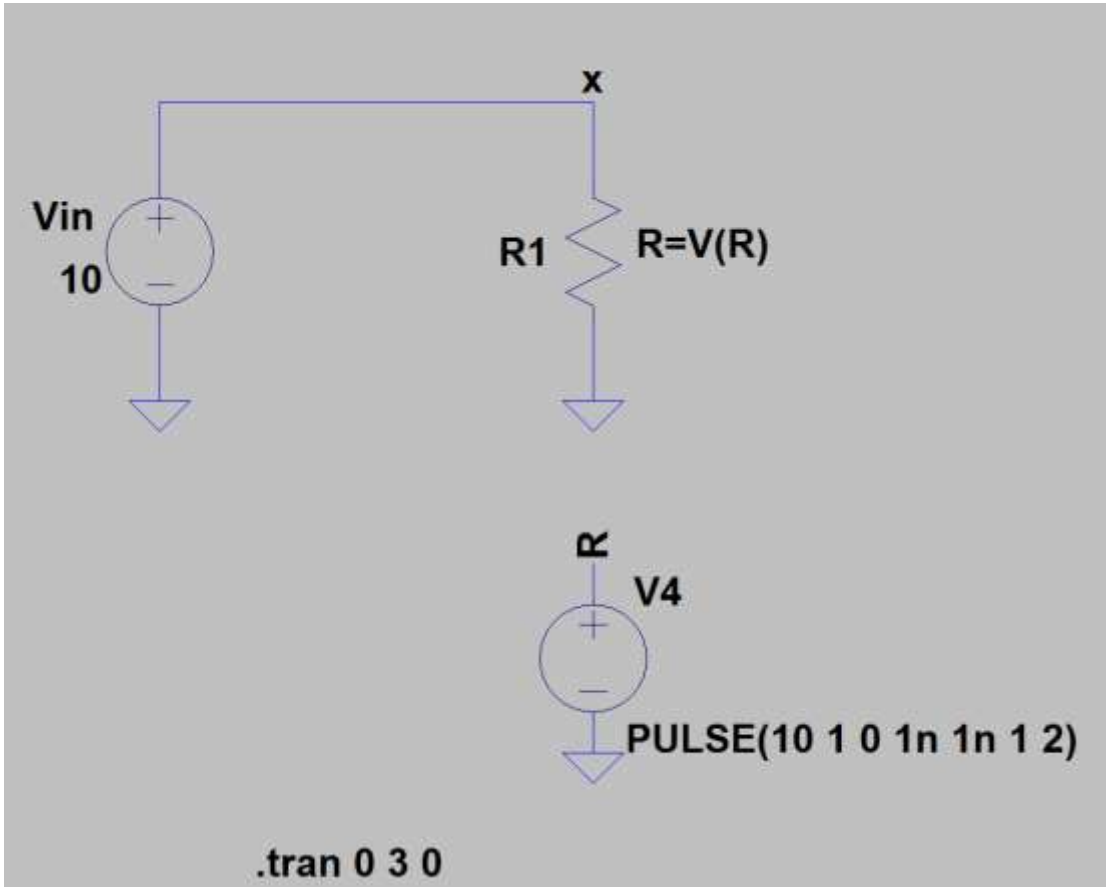
Other Places to Use Expressions

- **Without { }**, the expression is not reduced to a value before simulation, but is a **calculated *during* simulation in "real time"**. Below it is used within a behavioral source.



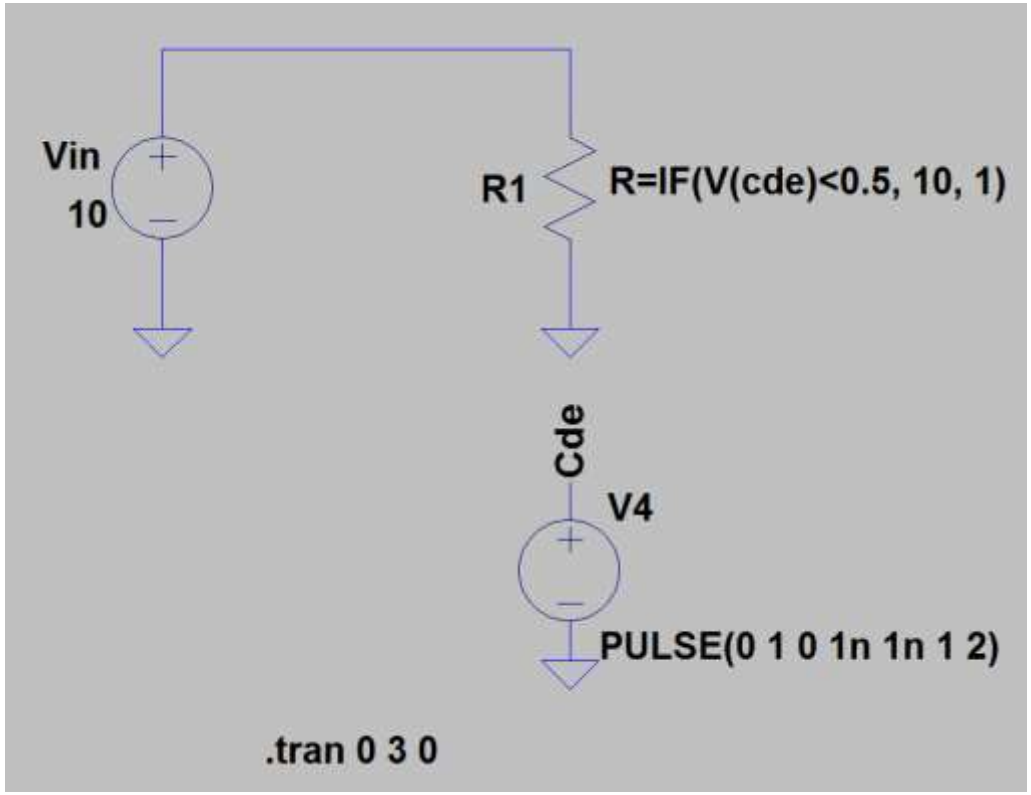
Variable R vs. Time

- Variable R versus time (expression) - R=V(R).asc



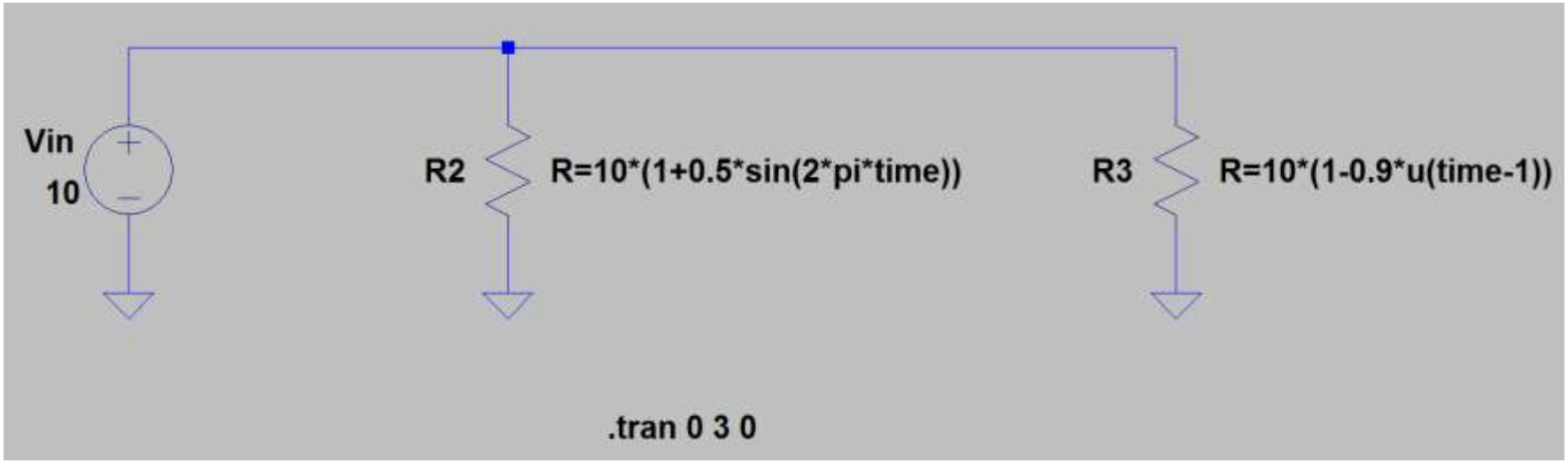
Variable R vs. Time

- Variable R versus time (expression) - IF.asc



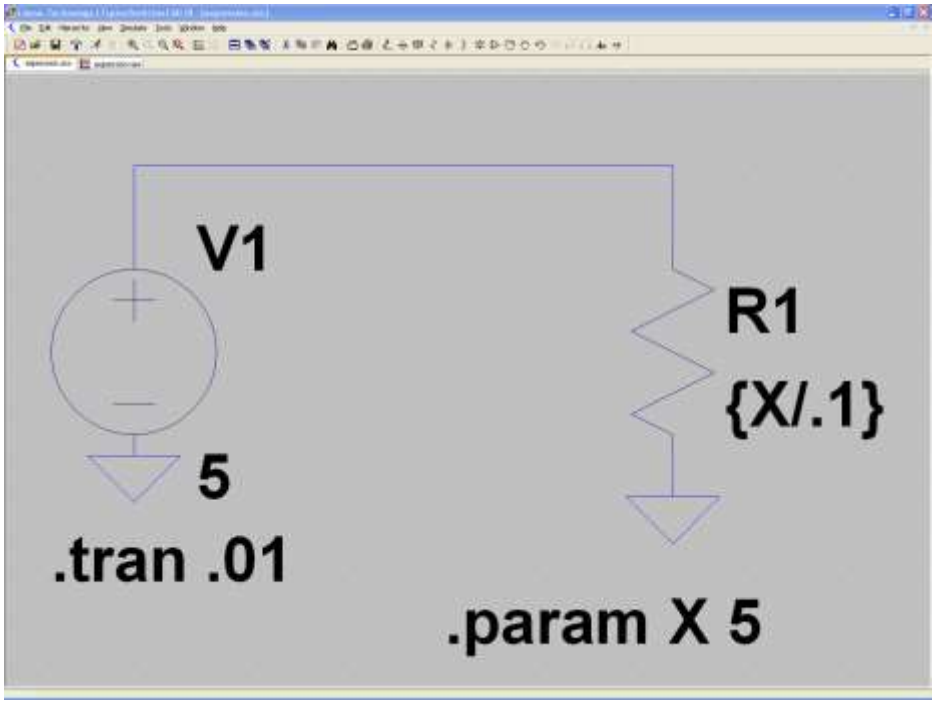
Variable R vs. Time

- Variable R versus time (expression) - R=f(time).asc



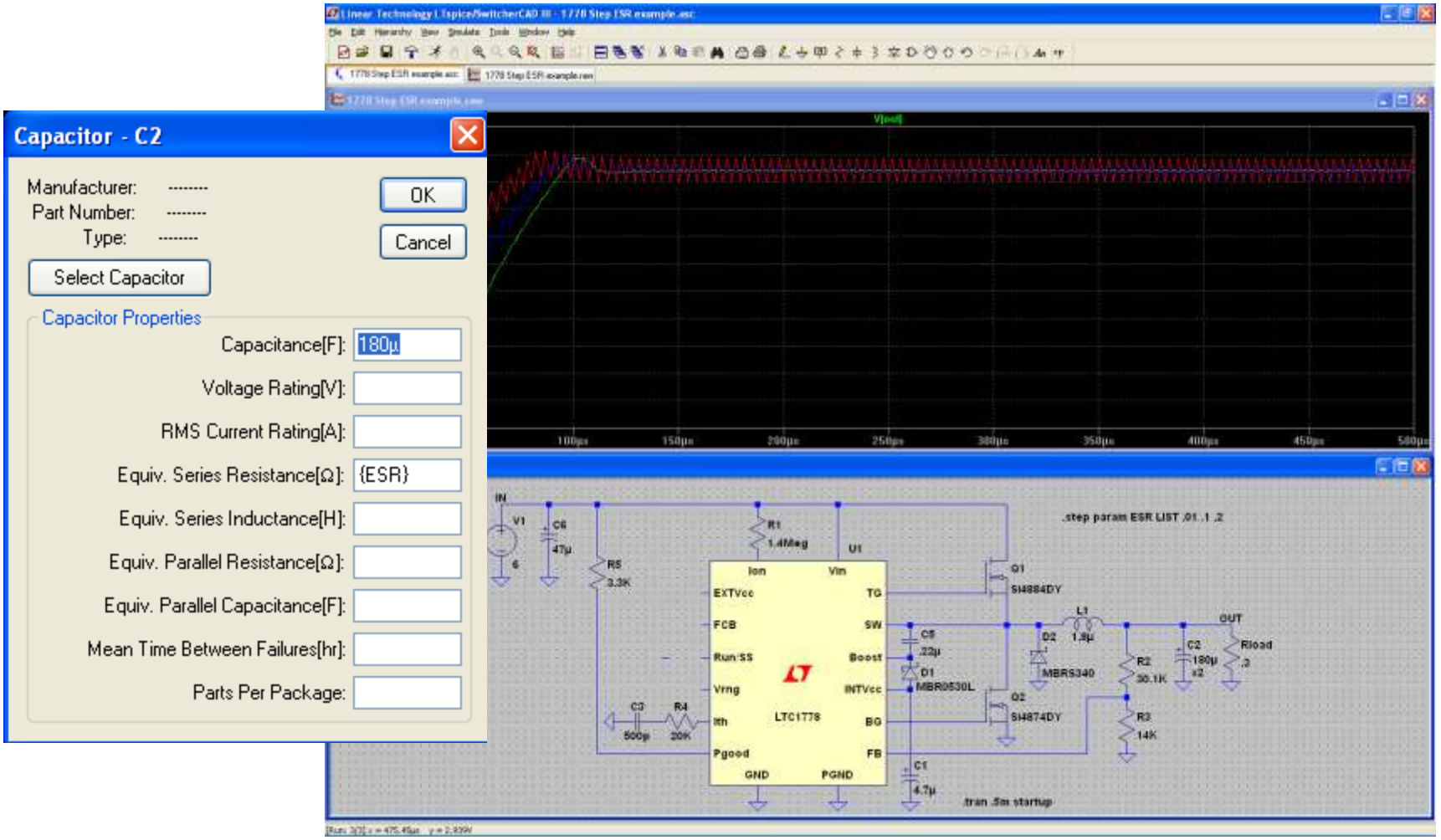
User-Defined Functions/Parameters

- The **.param** directive allows the creation of **user defined variables**
- Useful for **varying component values** without actually editing component properties



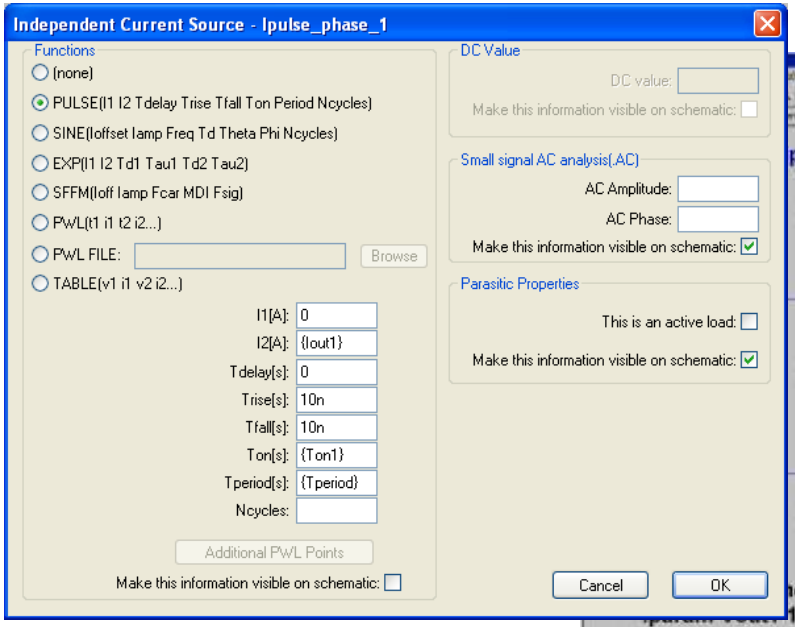
User-Defined Functions/Parameters

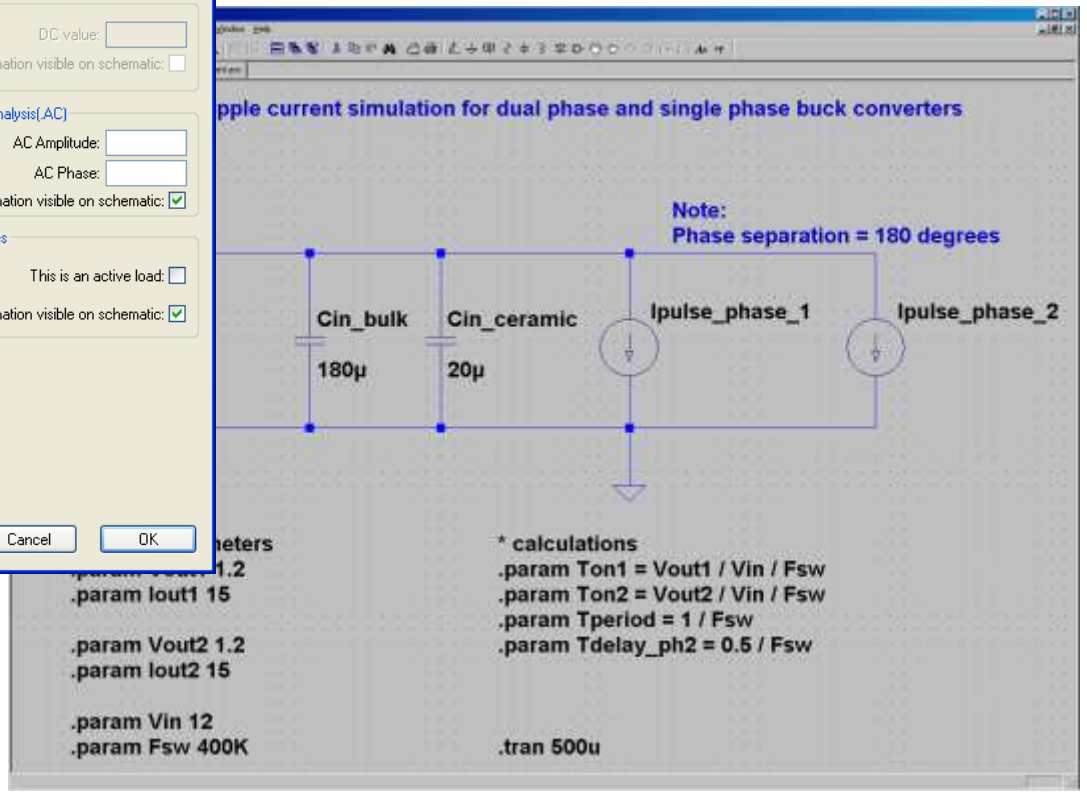
- Parameters can be used within components



User-Defined Functions/Parameters

- Parameters can also be used **within sources**.
- Parameters can be **calculation results**
- Multiple parameters can be used **simultaneously**



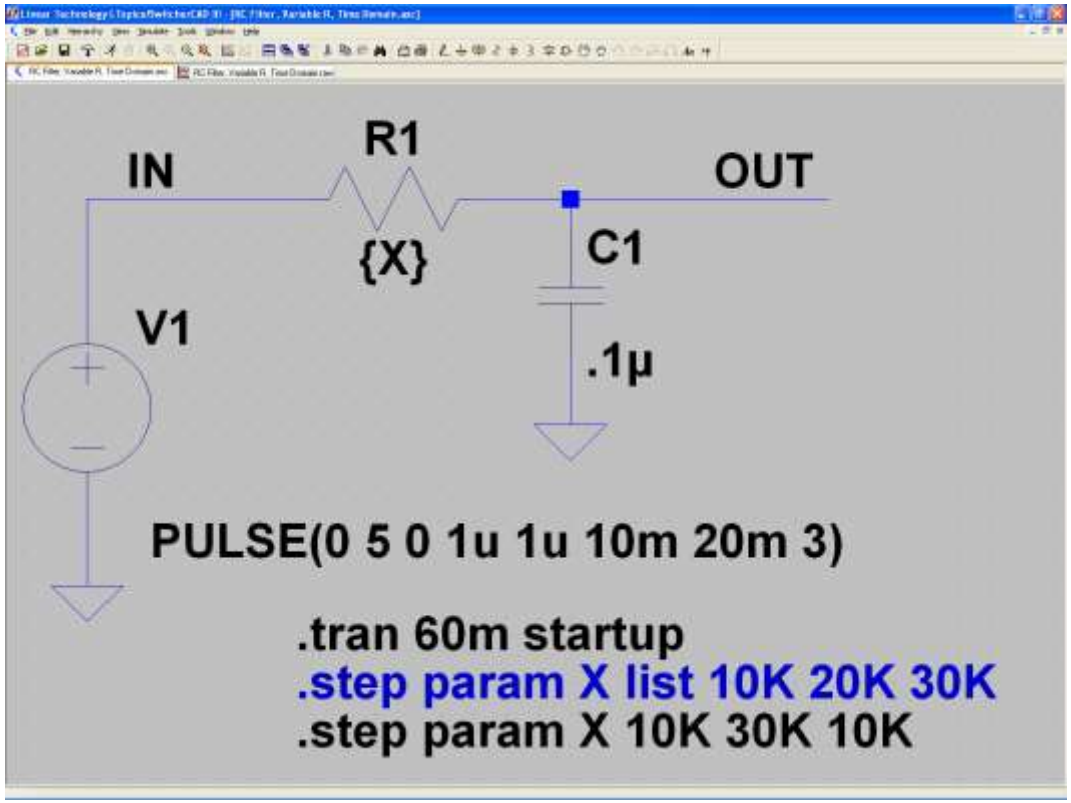


Parameter Sweeps

- The **.step** command causes the analysis to be **repeatedly performed** while stepping a model parameter
- Essentially multiple back-to-back simulation runs with the results of previous runs kept instead of being discarded
- Steps may be **linear**, **logarithmic**, or specified as a **list of values**
- Example: RC network and stepping a list of values

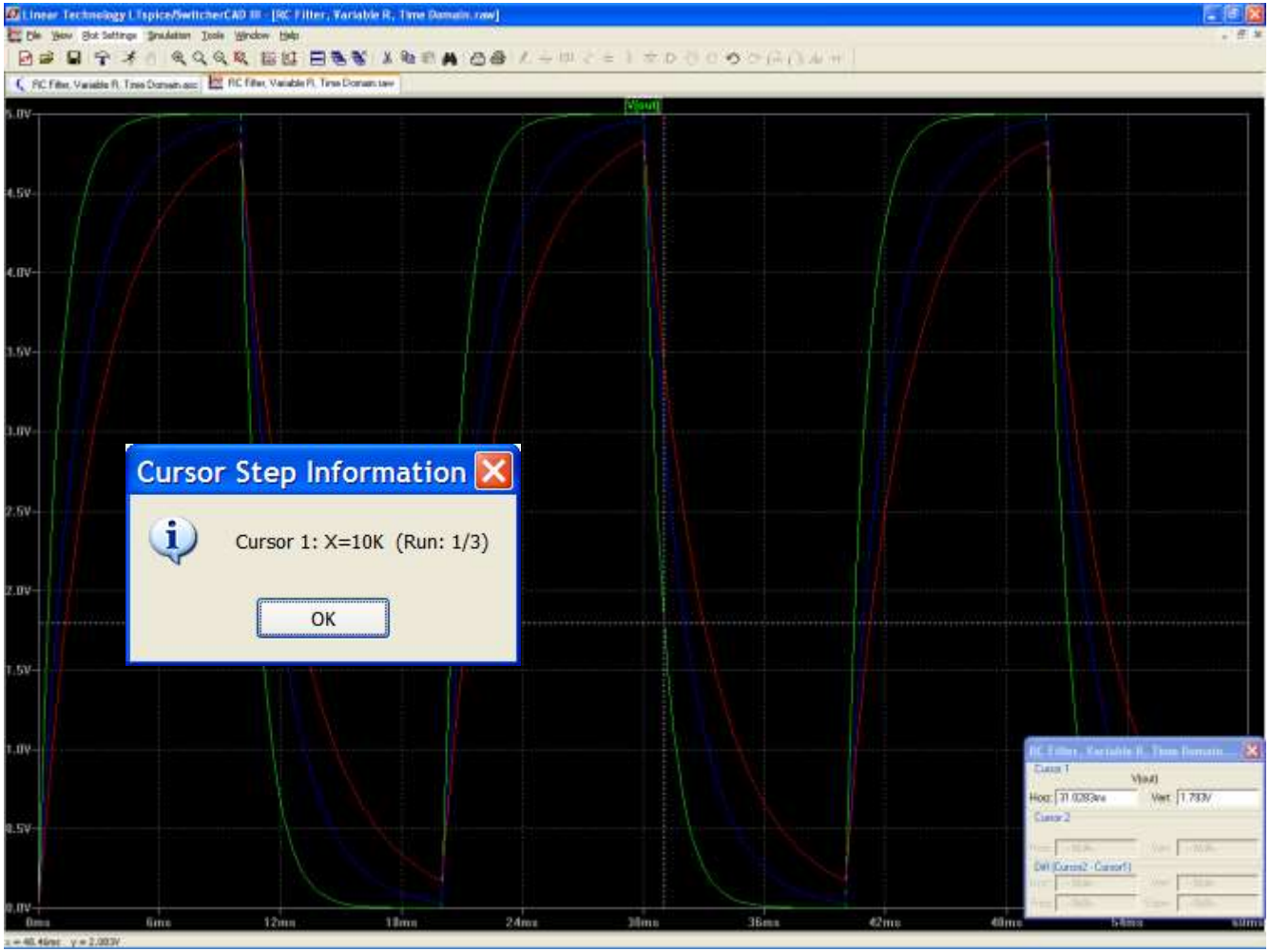
Parameter Sweeps

- Open up the simulation file titled “RC Filter Variable R Time Domain.asc” and follow the instructions in the simulation file.



Add .step command using Edit pull down menu → SPICE Directive, or by using the hotkey “s”.

Parameter Sweep – RC Filter Result

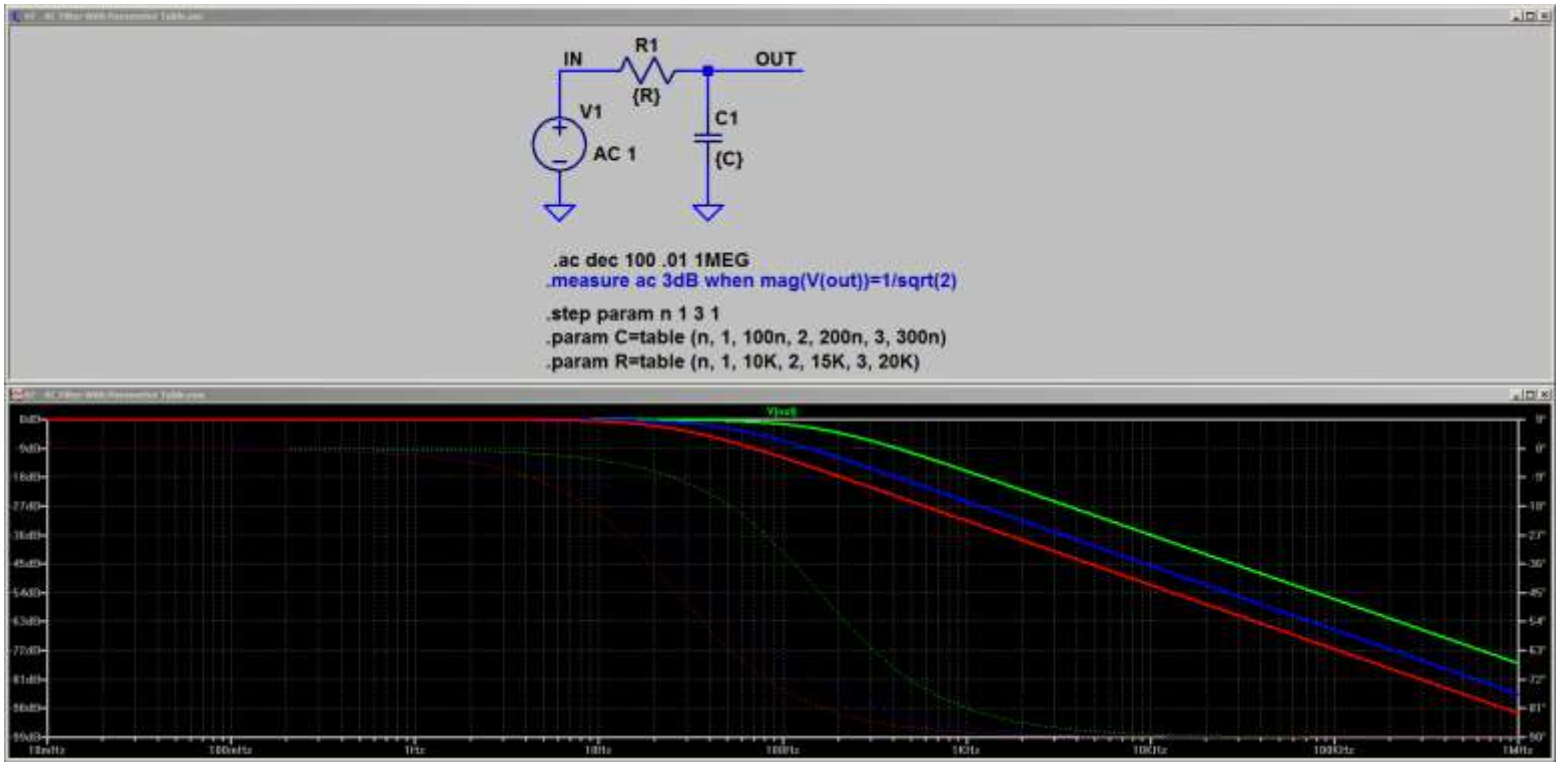


Using the **measurement cursor (left-click the *label*)**, the **up/down arrow keys** on the keyboard will **toggle** between waveforms

Right click on the cursor to display which **run** is associated with each waveform

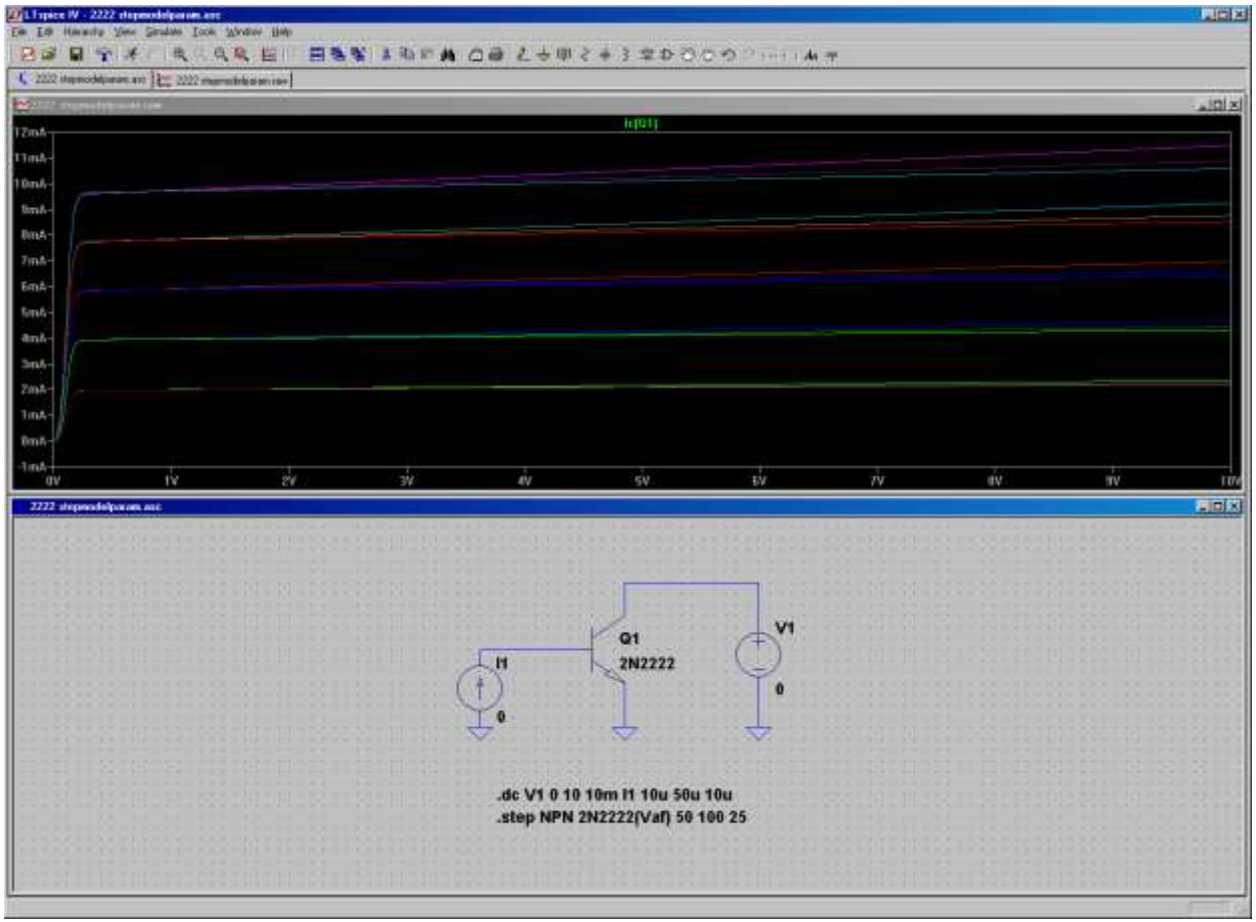
Stepping Multiple Parameters

- The **table** function can be used to **step multiple parameters simultaneously** using a table format (ex. pairs of values can be defined and simulated)
- Open up the simulation file titled “RC Filter With Parameter Table.asc” and follow the instructions in the simulation file.



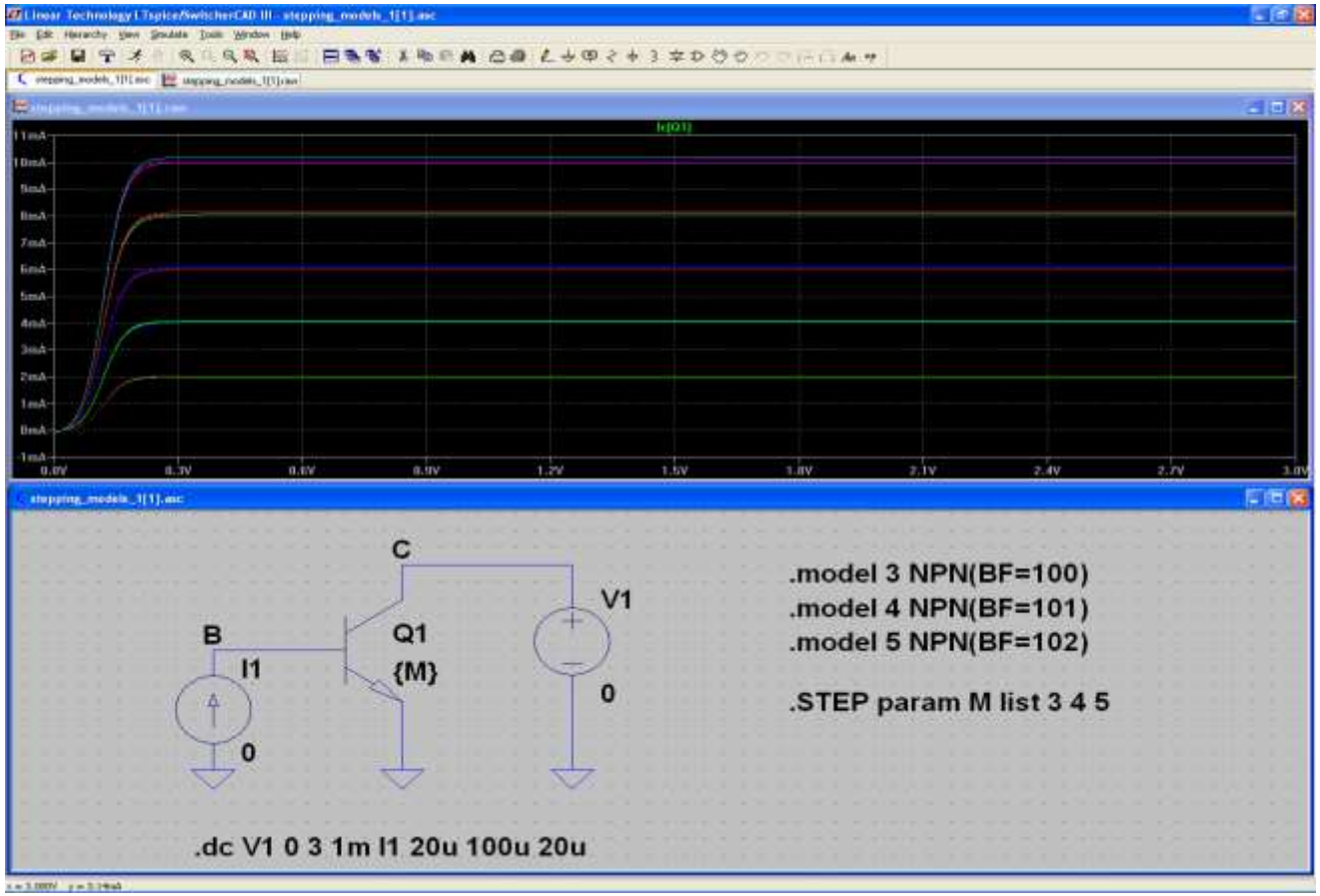
User-Defined Functions/Parameters

- **.Model parameters** can be stepped
- Open up the simulation file titled “2222 Step Model Param.asc” and follow the instructions



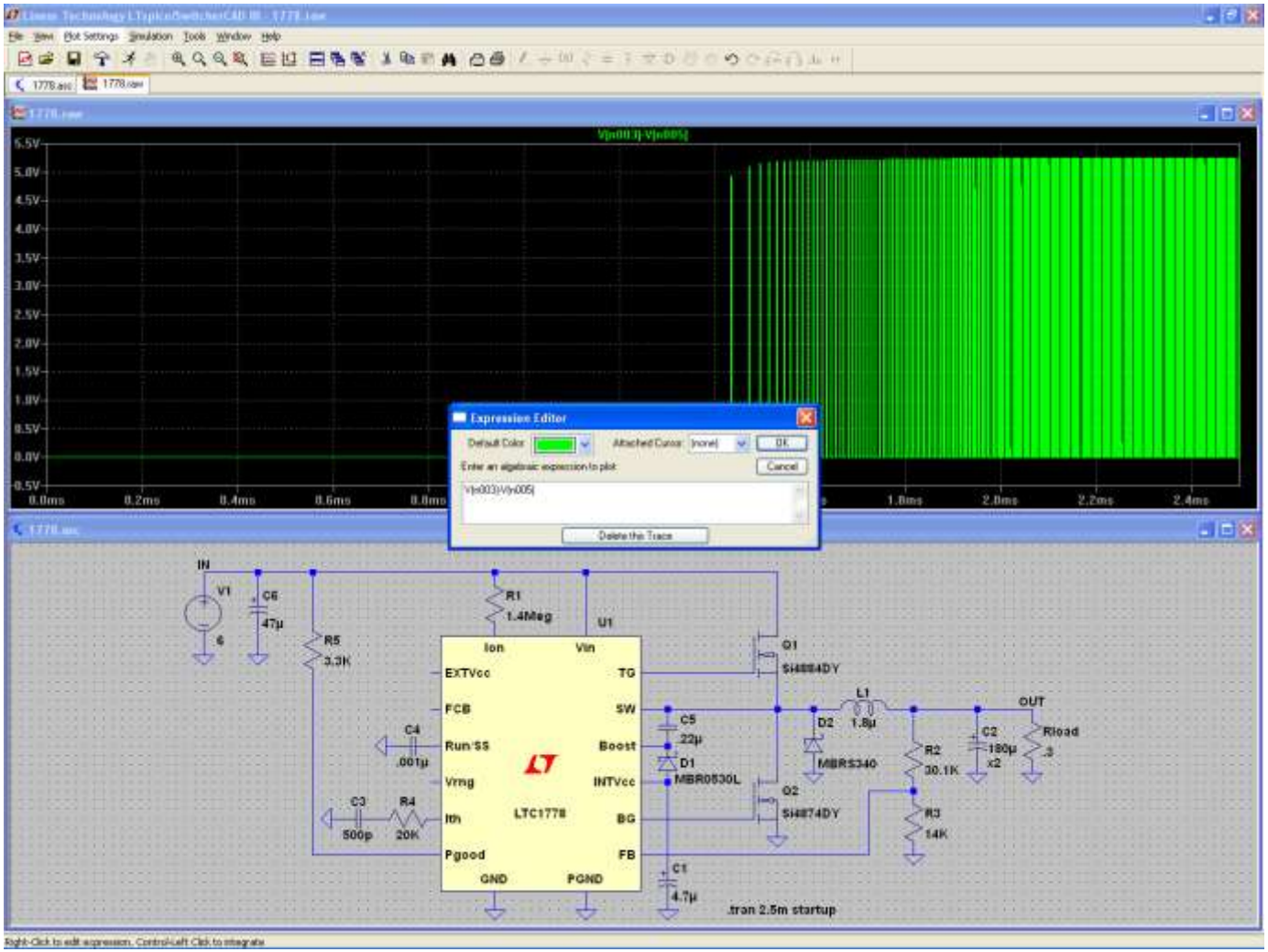
More .step Uses

- The .step command can also be used to step **which model** is being used.
- Open up the simulation file titled “Stepping Models.asc” and follow the instructions in the simulation file.



Other Places to Use Expressions

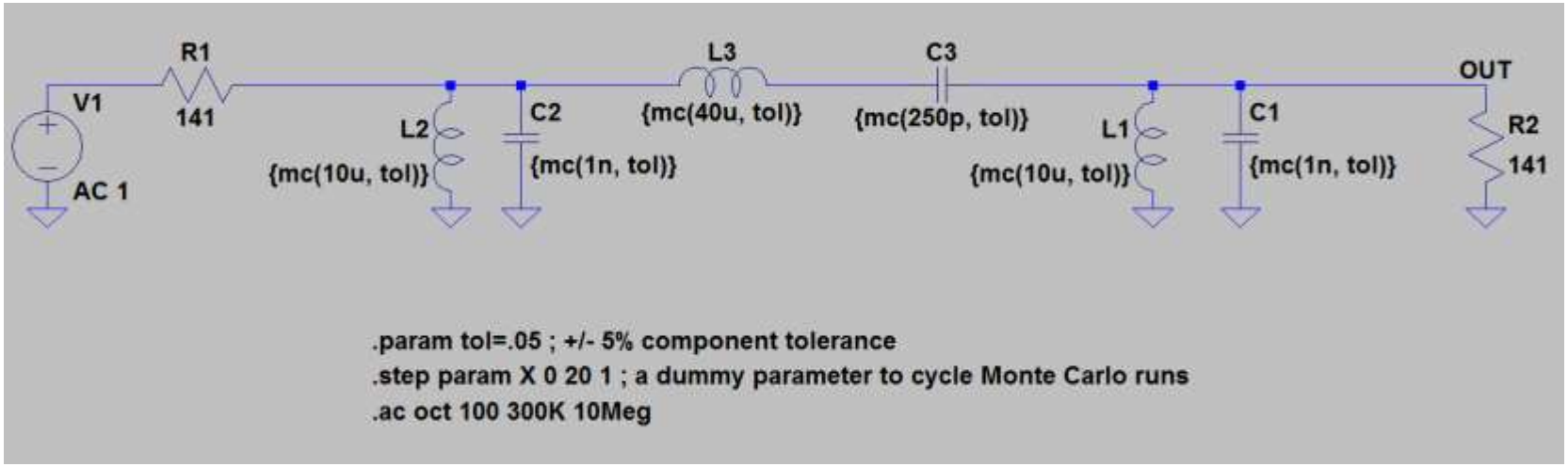
- Within the waveform editor (**right click on trace name**)



Monte Carlo Analysis

Filter Transfer Analysis

- Monte Carlo.asc

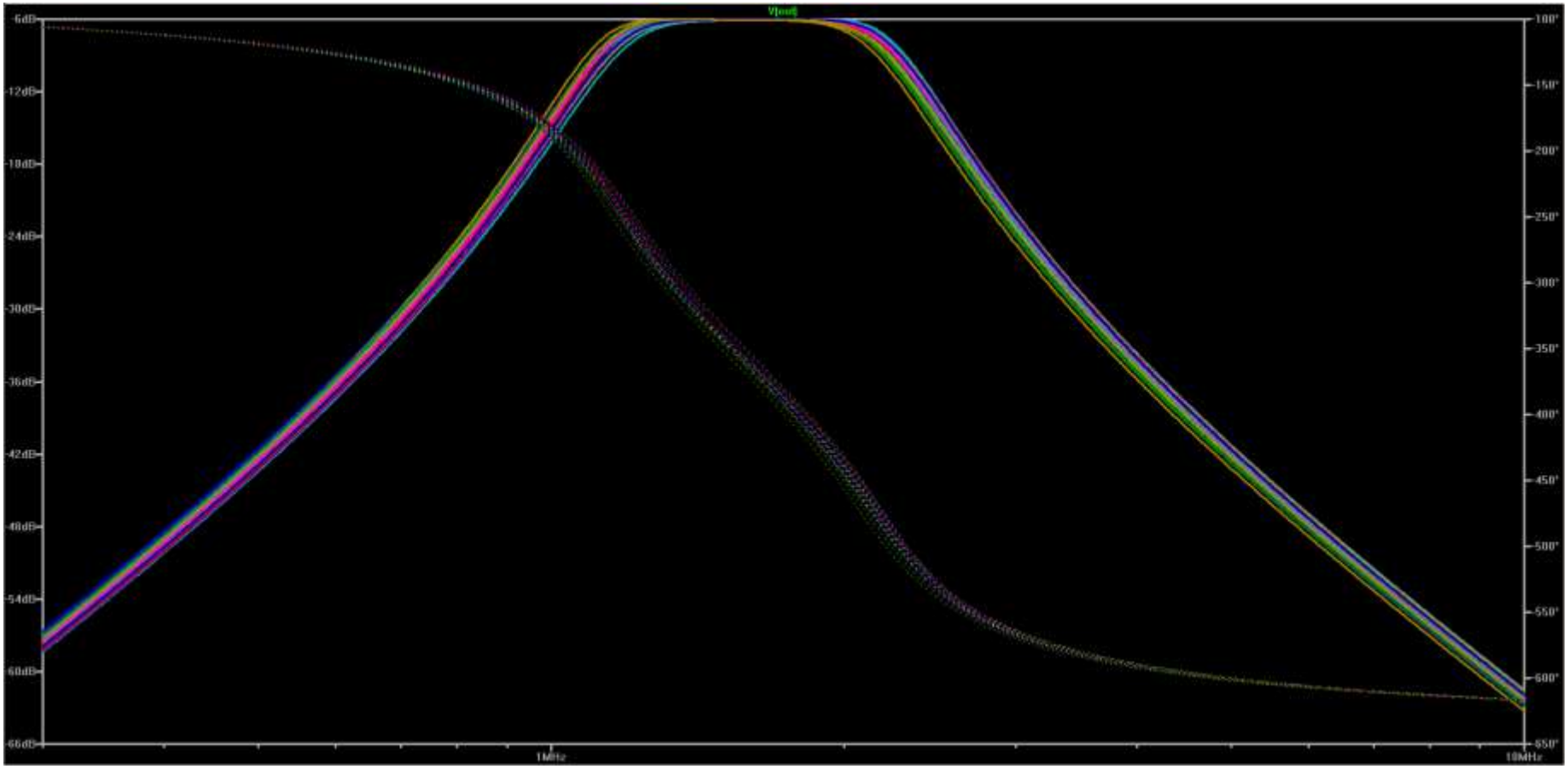


mc(val, tol) is a function that uses a random number generator with **uniform distribution** to return a value between **val*(1-tol)** and **val*(1+tol)**

Other functions of interest:
-> **flat(x)**: a function that uses a random number generator with **uniform distribution** to return a value between **-x** and **x**
-> **gauss(x)**: a function that uses a random number generator to return a value with a **centered gaussian distribution** and **sigma=x**

Filter Transfer Analysis

Vout

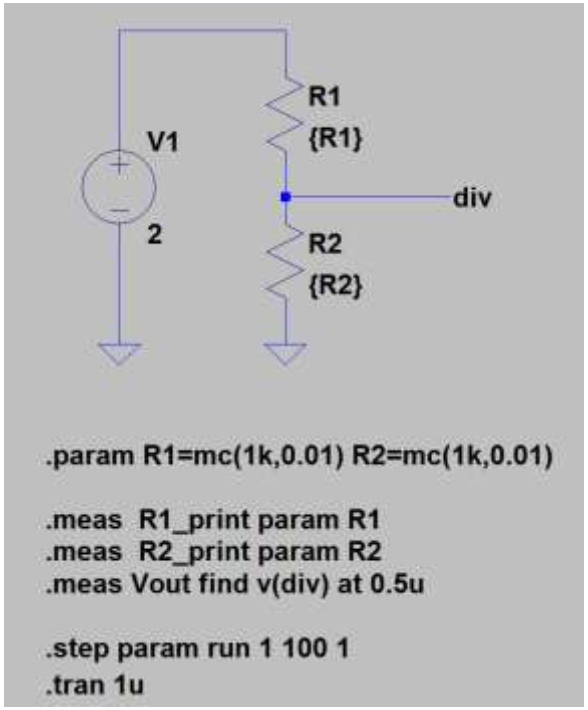


Frequency

Print Component Values

- Print component values in MC.asc

This example shows how to **print the component values** in a monte carlo run. The values are printed in the error.log file.



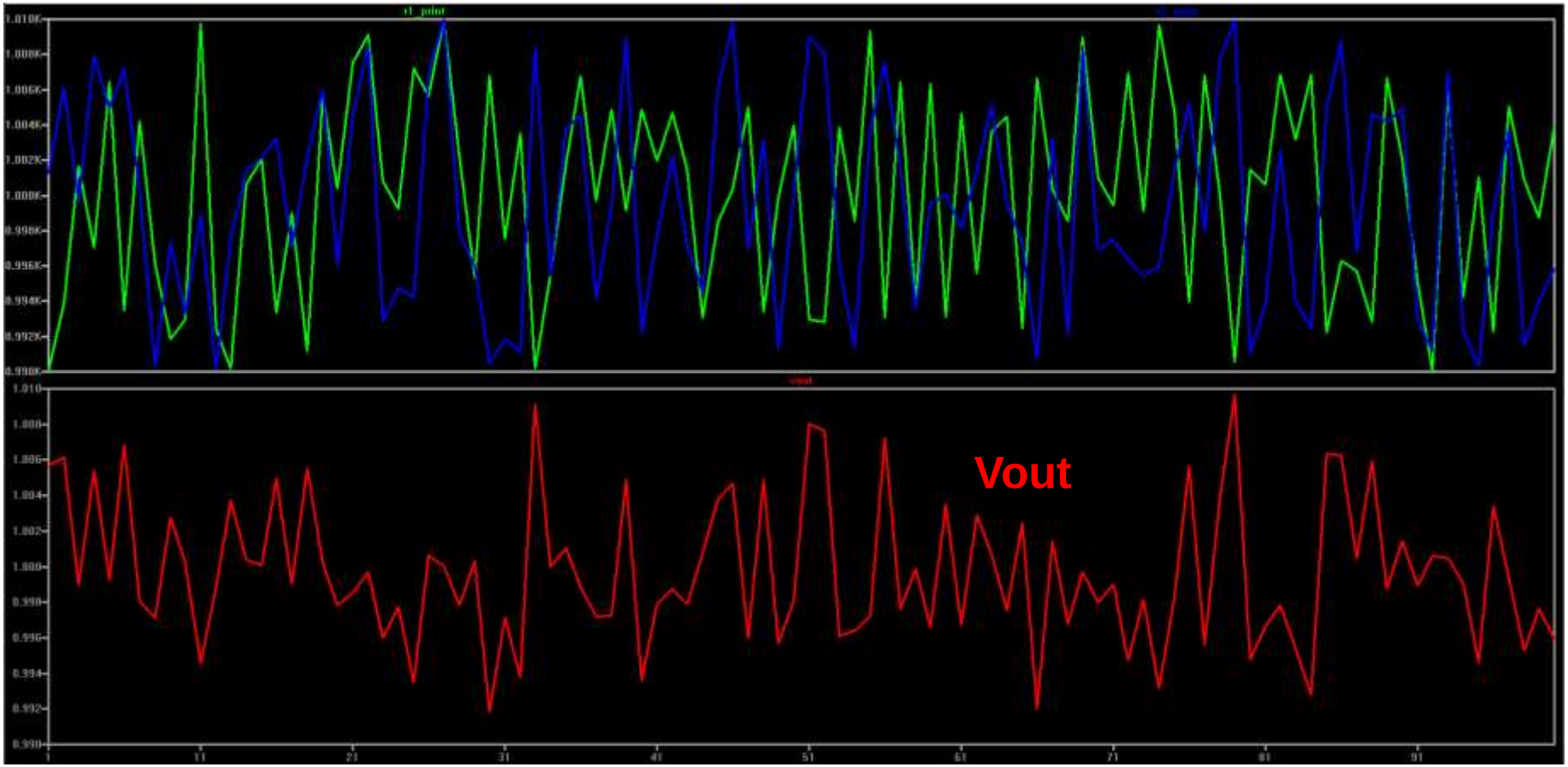
Trace the Simulation Values

- After simulation is complete, from the "**View**" menu, select '**Spice Error Log**‘
- **Right click** anywhere in the new window
- Select '**Add Trace**'
- Select '**Plot .step'ed .meas data**‘
- Select '**Yes**' in the next dialog box
- **Right click** anywhere in next window
- ‘**R1_print**' & ‘**R2_print**’ then press '**OK**’
- **Right click** and select ‘**Add new plot**’
- Select '**Add Trace**'
- Select ‘**Vout**’

Simulation Values

R1 value

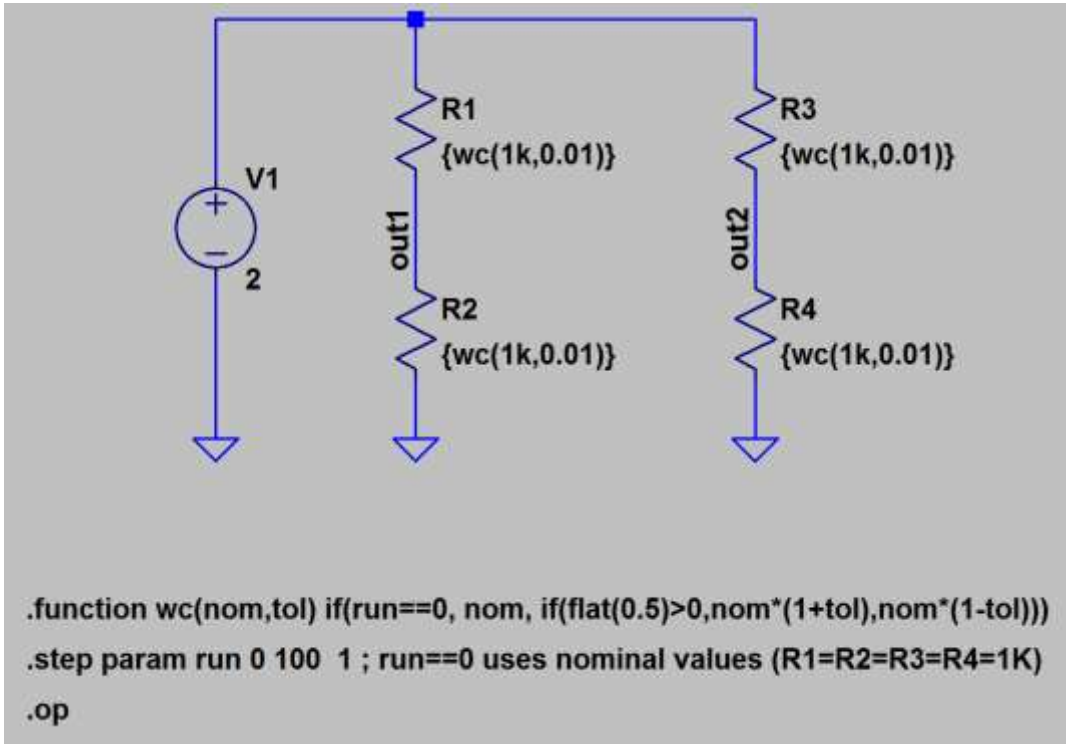
R2 value



Tolerance Stepping Min/Max Values Only

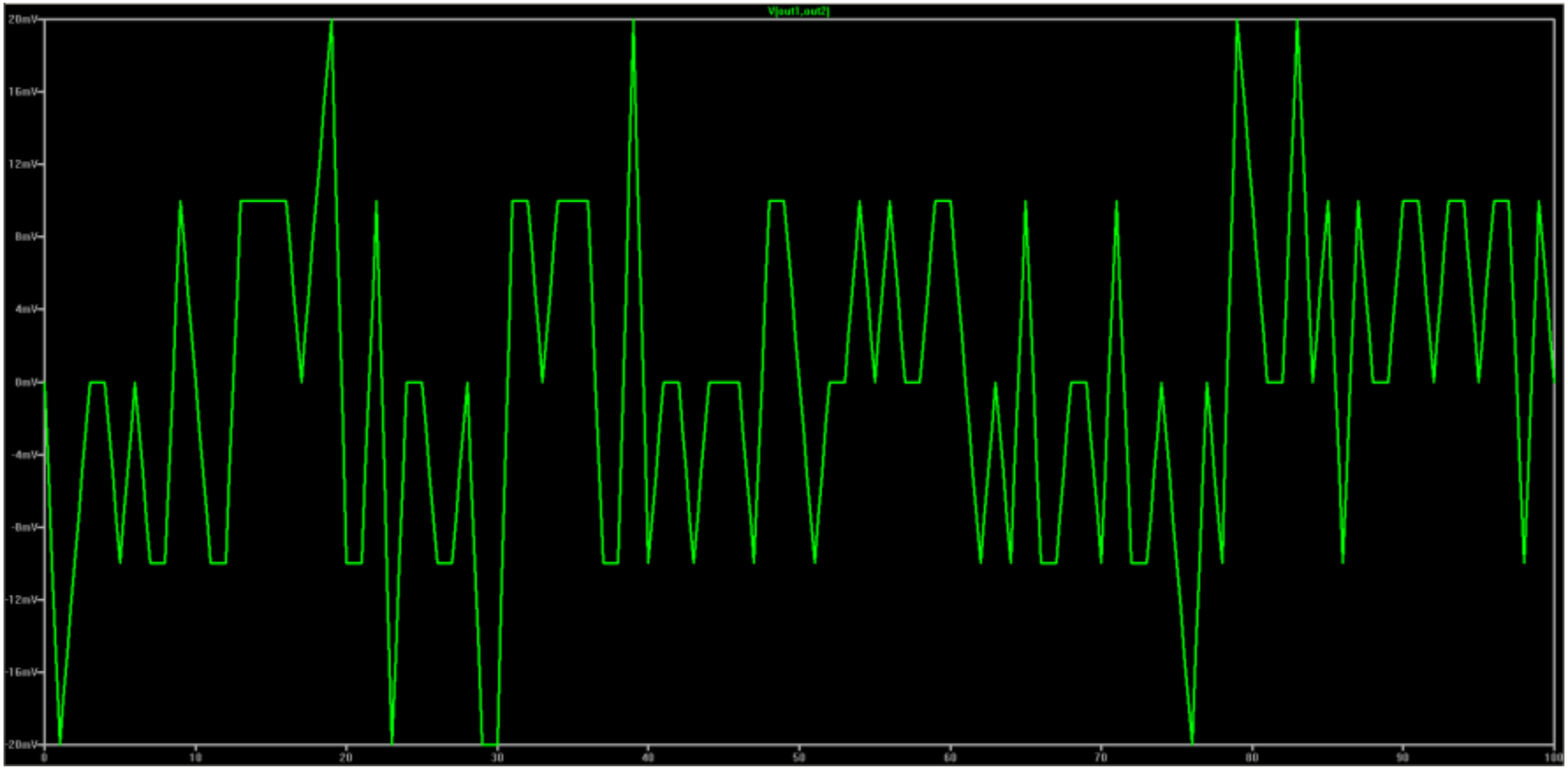
- Tolerance stepping min max only.asc

Only the extreme values **(1-Tol)**, **(1+Tol)** are used.
Run=0 is without tolerance (nominal values).
Plot **V(out1,out2)**



Tolerance Stepping Min/Max Values Only

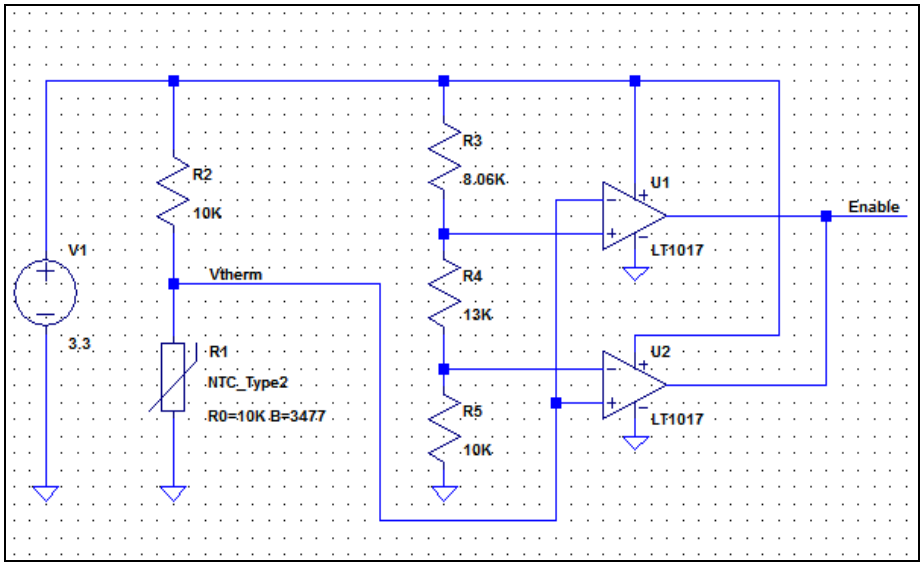
V(out1,out2)

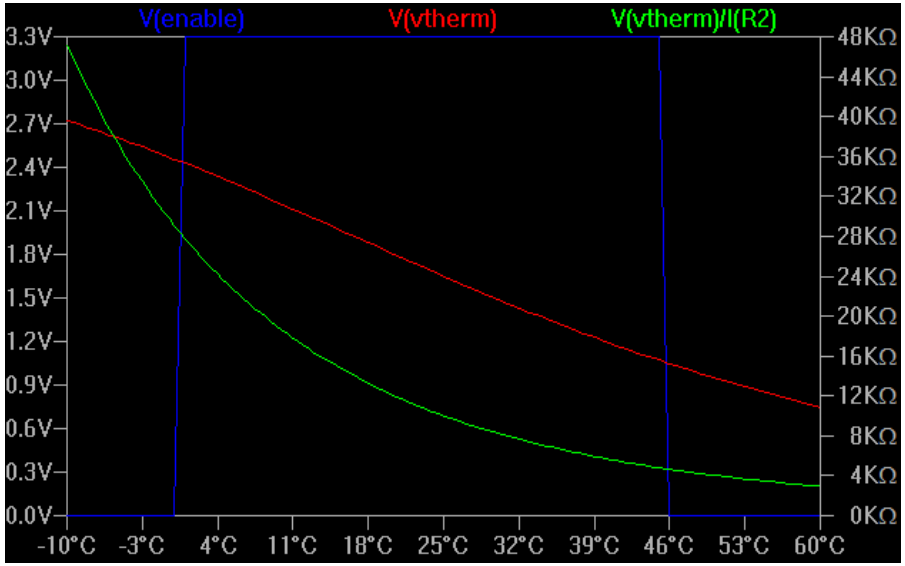


Thermistor Simulations: Plotting Temperature and Resistance

Plotting Temperature and Resistance

- Voltage and/or current are typically plotted on the vertical axis and time is typically plotted on the horizontal axis
- It is possible to plot resistance, temperature, and other parameters on the horizontal and vertical axes
- Thermistor simulation example: navigate to the NTC Circuit.asc simulation file and follow the instructions.





Plotting Temperature and Resistance

Important items to note for the NTC Circuit.asc simulation:

- **The DC operating point ".op" simulation command must be used (see LTspice help regarding DC operating point definition)**
- **The SPICE model for the thermistor is included in the simulation file**
- **A two terminal thermistor schematic symbol with the appropriate device parameters is required**
- **Additional instructions / information is included in the simulation file**

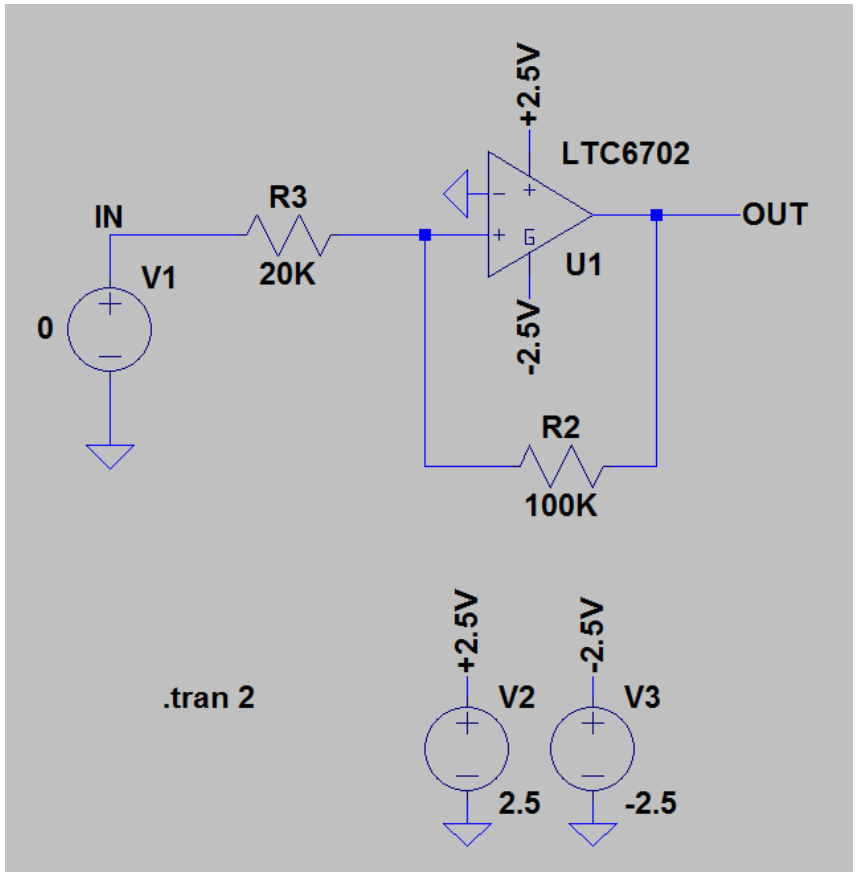
Plotting Temperature and Resistance

Important items to note for the NTC Circuit.asc simulation (cont.):

- **Voltages can be labeled and in this case the voltage across thermistor R1 is labeled Vtherm**
- **Currents cannot be labeled, thus we must determine what LTspice has called the current flowing into thermistor R1**
- **Probing the top terminal of R1 we see the current has been labeled by LTspice as "Ix(R1:A)"**
- **Plotting the expression V(vtherm)/Ix(R1:A) therefore plots resistance of R1**
- **Note that probing the bottom terminal of R1 we see that the current has been labeled Ix(R1:B) by LTspice even though in this case the current is the same as the top terminal (but reversed)**

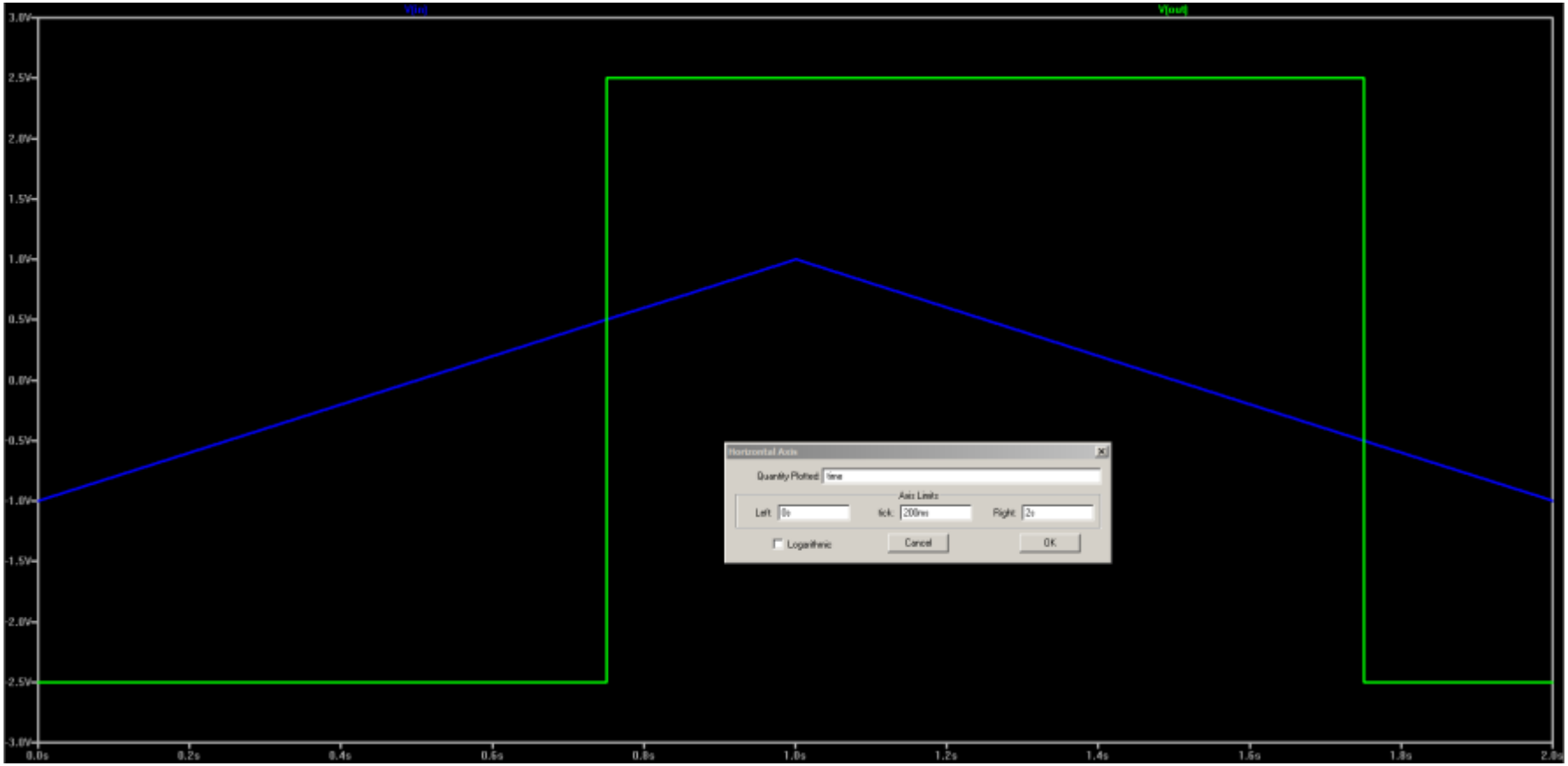
Comparator with Hysteresis

- LTC6702 comparator with hysteresis.asc



V(OUT) vs Time

VIN **Vout**

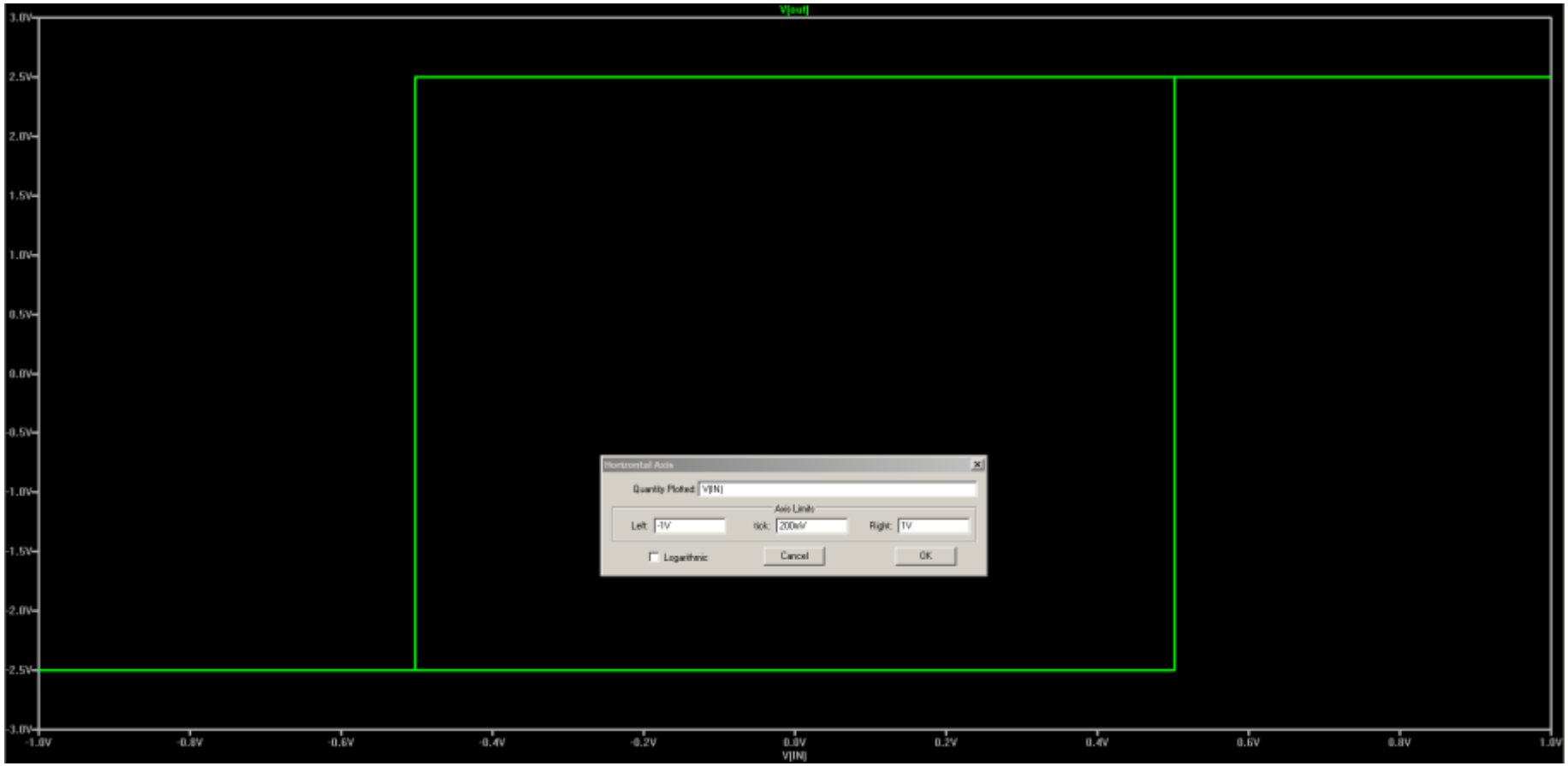


Trace Hysteresis Curve

- Move the mouse to the bottom of the screen until the cursor turns into a ruler
- Left click to display the Horizontal Axis box
- Replace the horizontal quantity plotted (Time) by V(IN)
- This lets you make parametric plots

V(OUT) vs V(IN)

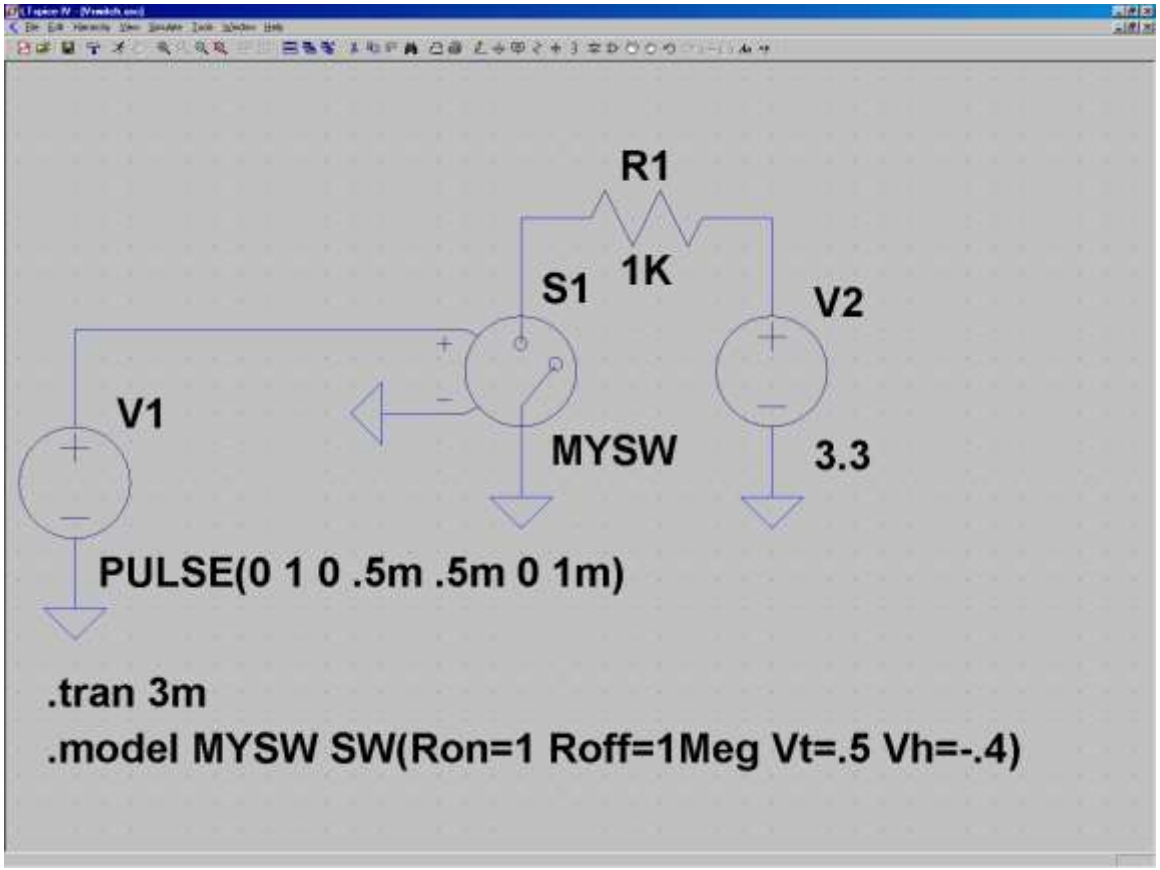
Vout



VIN (as horz. Axis)

Voltage/Current Controlled Switch

- A voltage/current controlled switch must have a model defined. This may be done as a SPICE directive directly on the schematic.
- Vswitch.asc



Switch Parameters

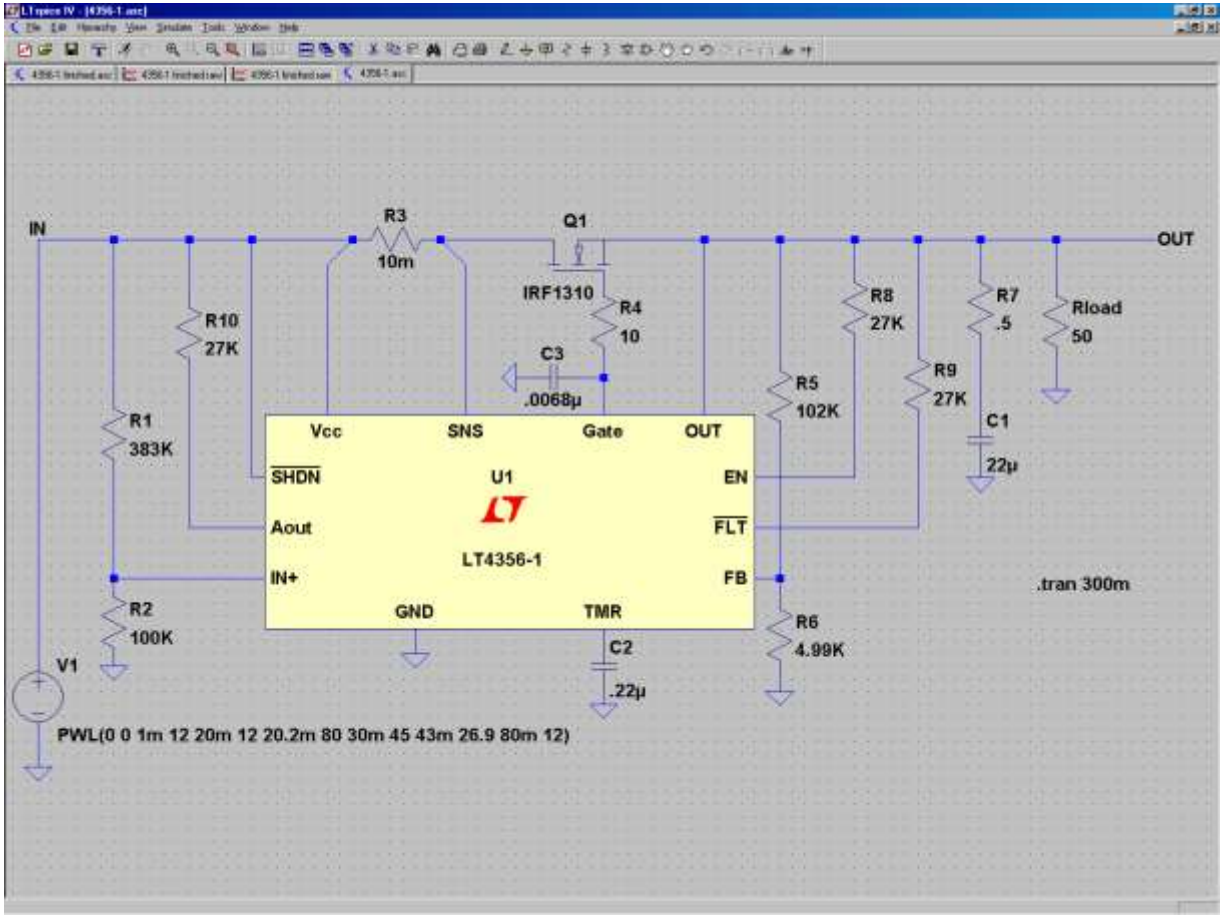
- **The parameters for the switch model are described in the help file.**

Voltage Controlled Switch Model Parameters

Name	Description	Units	Default
Vt	Threshold voltage	V	0.
Vh	Hysteresis voltage	V	0.
Ron	On resistance	Ω	1.
Roff	Off resistance	Ω	1/Gmin
Lser	Series inductance	H	0.
Vser	Series voltage	V	0.
Ilimit	Current limit	A	Infin.

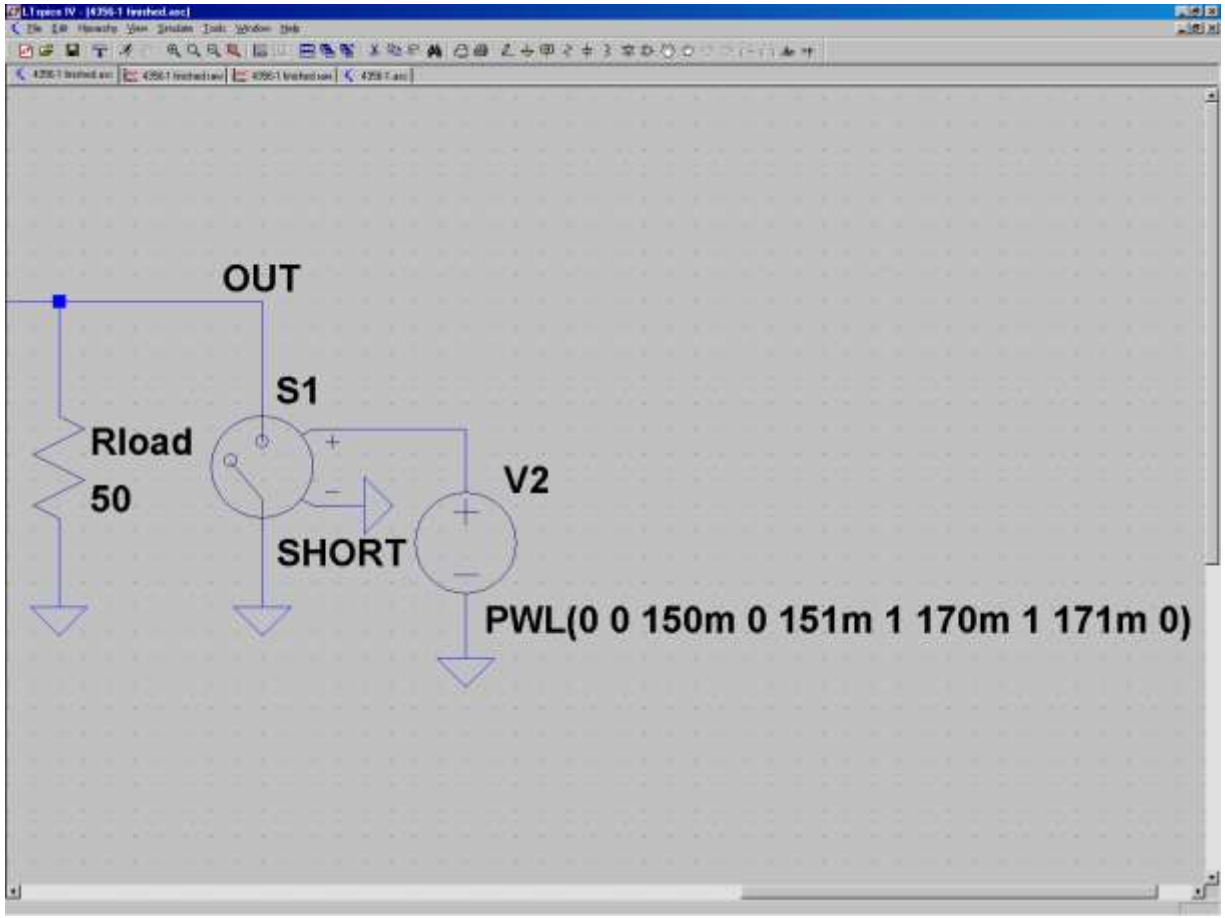
Example of Short Circuit

- Open up the simulation file titled “4356-1.asc” and follow the instructions in the simulation file.



Example of Short Circuit

- **First, set up the switch and its stimulus.**



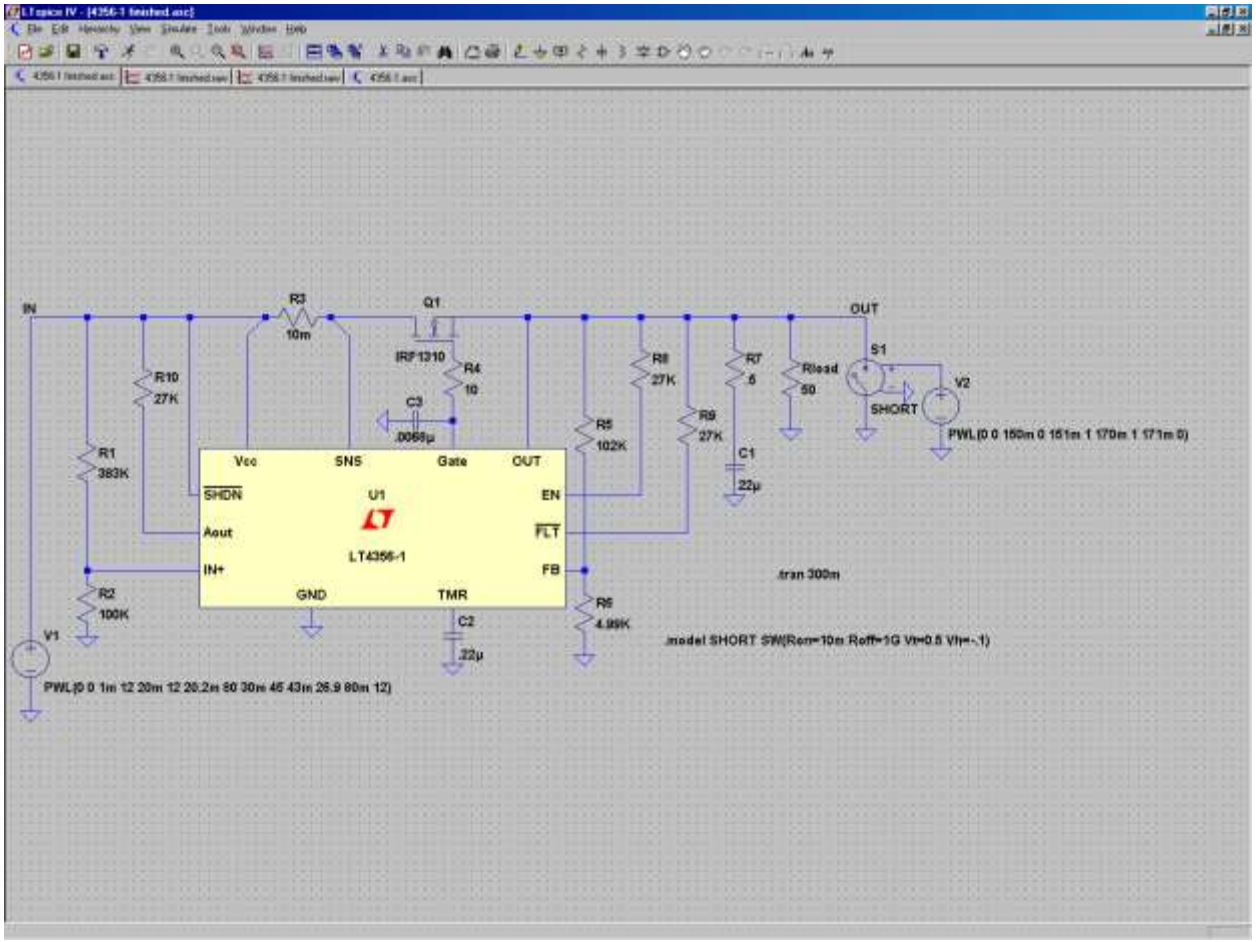
Example of Short Circuit

PWL(0 0 150m 0 151m 1 170m 1 171m 0)

.model SHORT SW(Ron=10m Roff=1G Vt=0.5 Vh=-.1)

Example of Short Circuit

- Now, define the switch model.
- ex. .model SHORT SW(Ron=10m Roff=1G Vt=0.5 Vh=-.1)



Switching Power Supply Control Loop Bode Plots

Method 1 : Use "**.step**" to size compensation network & test for load step (.TRAN)

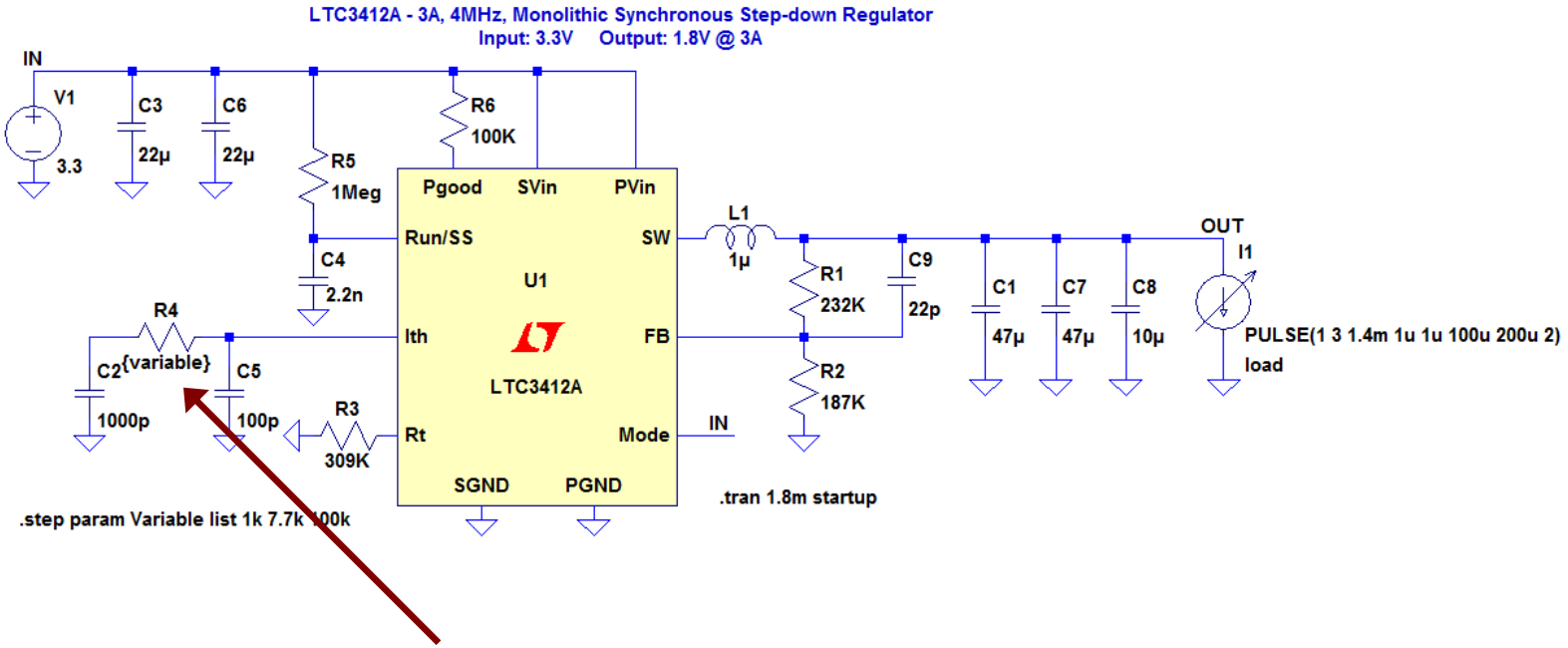
Method 2 : Use **small signal model** for Bode analysis (.AC)

Method 3 : Bode analysis using **time simulation** (.TRAN)

Method 1: .step (.TRAN)

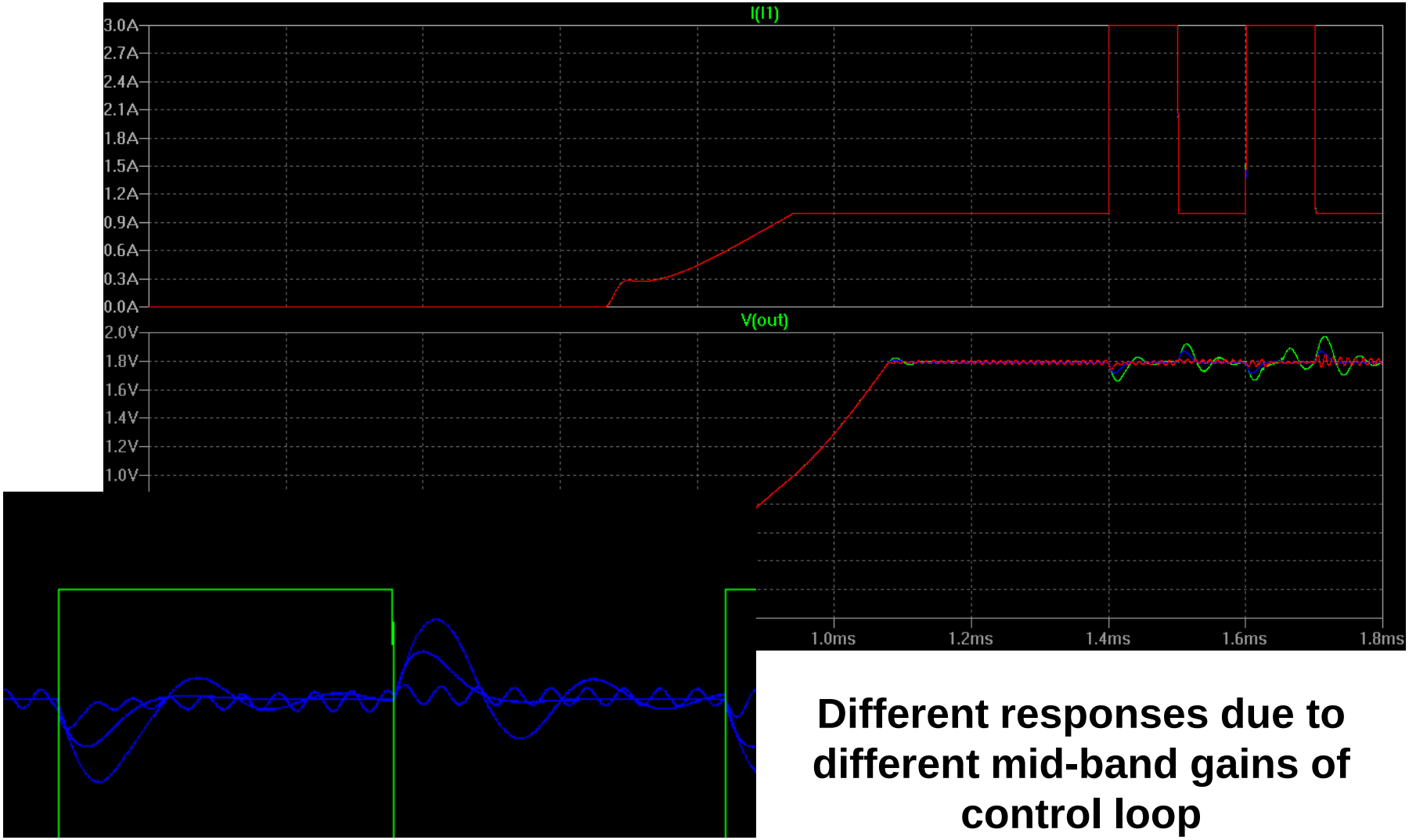
LTC3412A Example

- LTC3412A Stability.asc



Value of mid-band gain resistor is stepped from 1k to 7.7k to 100k

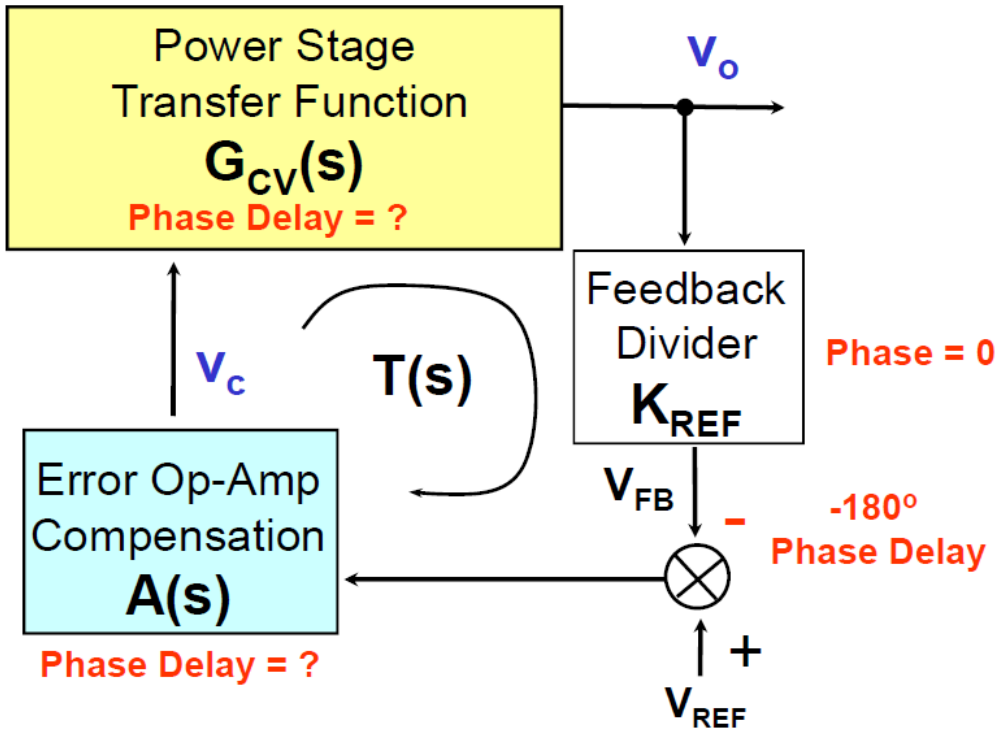
LTC3412A Example



Different responses due to different mid-band gains of control loop

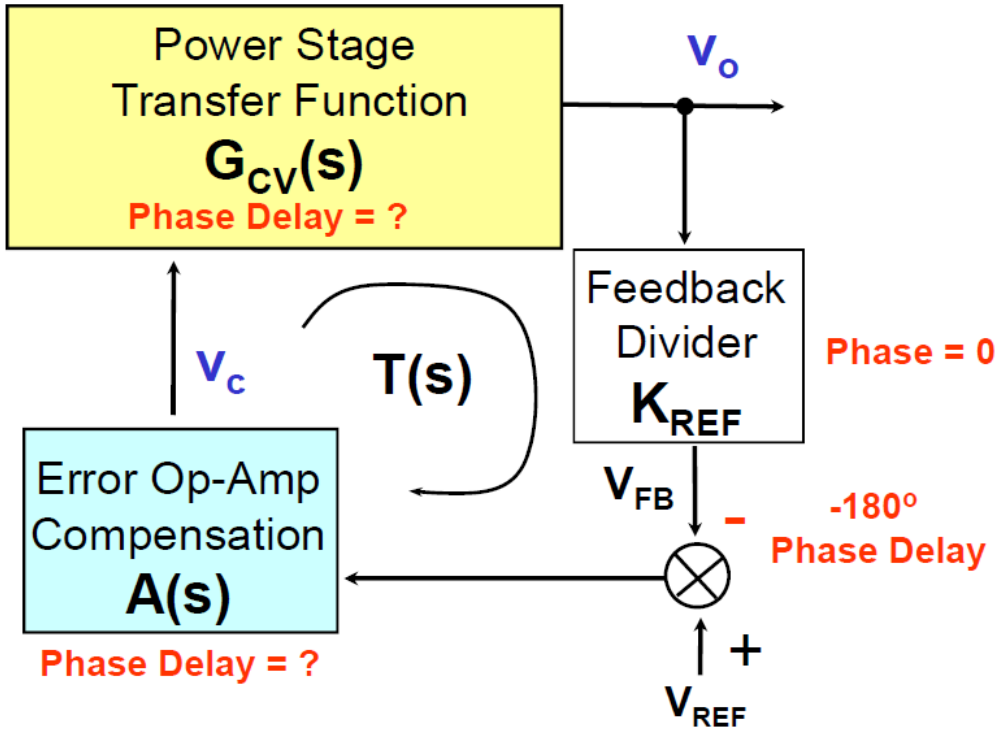
Method 2: Small Signal Model (.AC)

Control Block Diagram and Loop Gain



- - Loop Gain: $T(s) = G_{CV}(s) \cdot K_{REF} \cdot A(s)$
- - Bandwidth: crossover frequency f_c @ loop gain $|T(s)|=1$
- - Stability: Total phase > -360° at crossover frequency
- Total Phase = $(-180° + \varphi[A(s)] + \varphi[G_{CV}(s)])_{f=fc} > -360°$

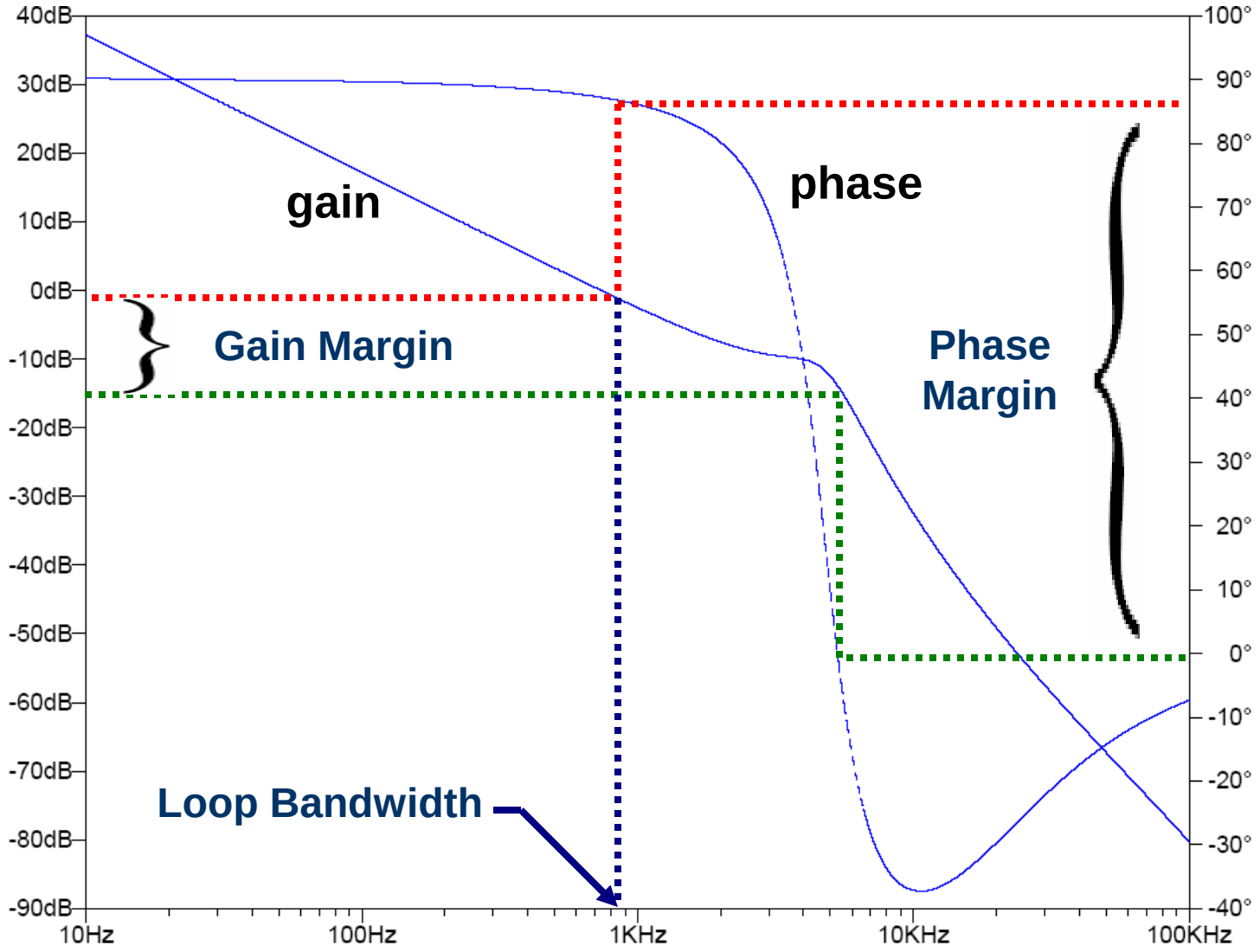
Control Block Diagram and Loop Gain



Design Targets:

- Regulation accuracy, line rejection, low Z_{OUT}: $|T(s)| >> 1$ for $f < f_c$
- - Bandwidth: high bandwidth for fast transient response
- - Stability: phase margin > 45°

Open Loop Stability Criteria



LINEAR TECHNOLOGY

LT1976/LT1976B

High Voltage 1.5A, 200kHz Step-Down Switching Regulator with 100μA Quiescent Current

FEATURES

- **Wide Input Range: 3.3V to 60V**
- **1.5A Peak Switch Current (LT1976)**
- **100μA Quiescent Current (LT1976)****
- **1.6mA Quiescent Current (LT1976B)**
- **Low Shutdown Current: I_Q < 1μA**
- **Power Good Flag with Programmable Threshold**
- **Load Dump Protection to 60V**
- 200kHz Switching Frequency
- Saturating Switch Design: 0.2Ω On-Resistance
- Peak Switch Current Maintained Over Full Duty Cycle Range*
- 1.25V Feedback Reference Voltage
- Easily Synchronizable
- Soft-Start Capability
- Small 16-Pin Thermally Enhanced TSSOP Package

APPLICATIONS

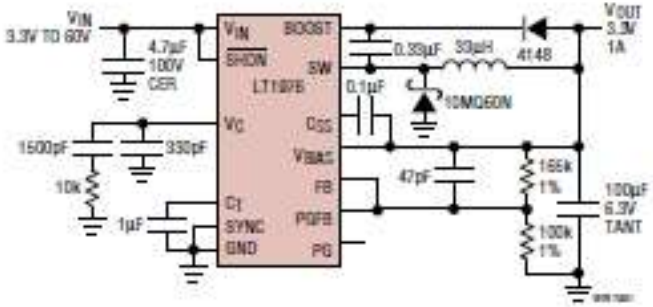
- High Voltage Power Conversion
- 14V and 42V Automotive Systems
- Industrial Power Systems
- Distributed Power Systems

DESCRIPTION

The LT®1976/LT1976B are 200kHz monolithic step-down switching regulators that accept input voltages up to 60V. A high efficiency 1.5A, 0.2Ω switch is included on the die along with all the necessary oscillator, control and logic circuitry. Current mode topology is used for fast transient response and good loop stability.

Innovative design techniques along with a new high voltage process achieve high efficiency over a wide input range. Efficiency is maintained over a wide output current range by employing Burst Mode operation at low currents, utilizing the output to bias the internal circuitry, and by using a supply boost capacitor to fully saturate the power switch. The LT1976B does not shift into Burst Mode operation at low currents, eliminating low frequency output ripple at the expense of efficiency. Patented circuitry maintains peak switch current over the full duty cycle range.* Shutdown reduces input supply current to less than 1μA. External synchronization can be implemented by driving the SYNC pin with logic-level inputs. A single capacitor from the C_{SS} pin to the output provides a controlled output voltage ramp (soft-start). The devices

14V to 3.3V Step-Down Converter with 100μA No Load Quiescent Current



Small Signal Model

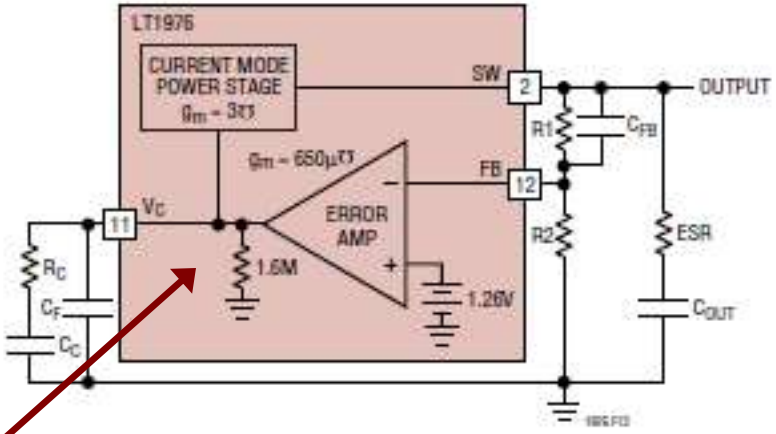


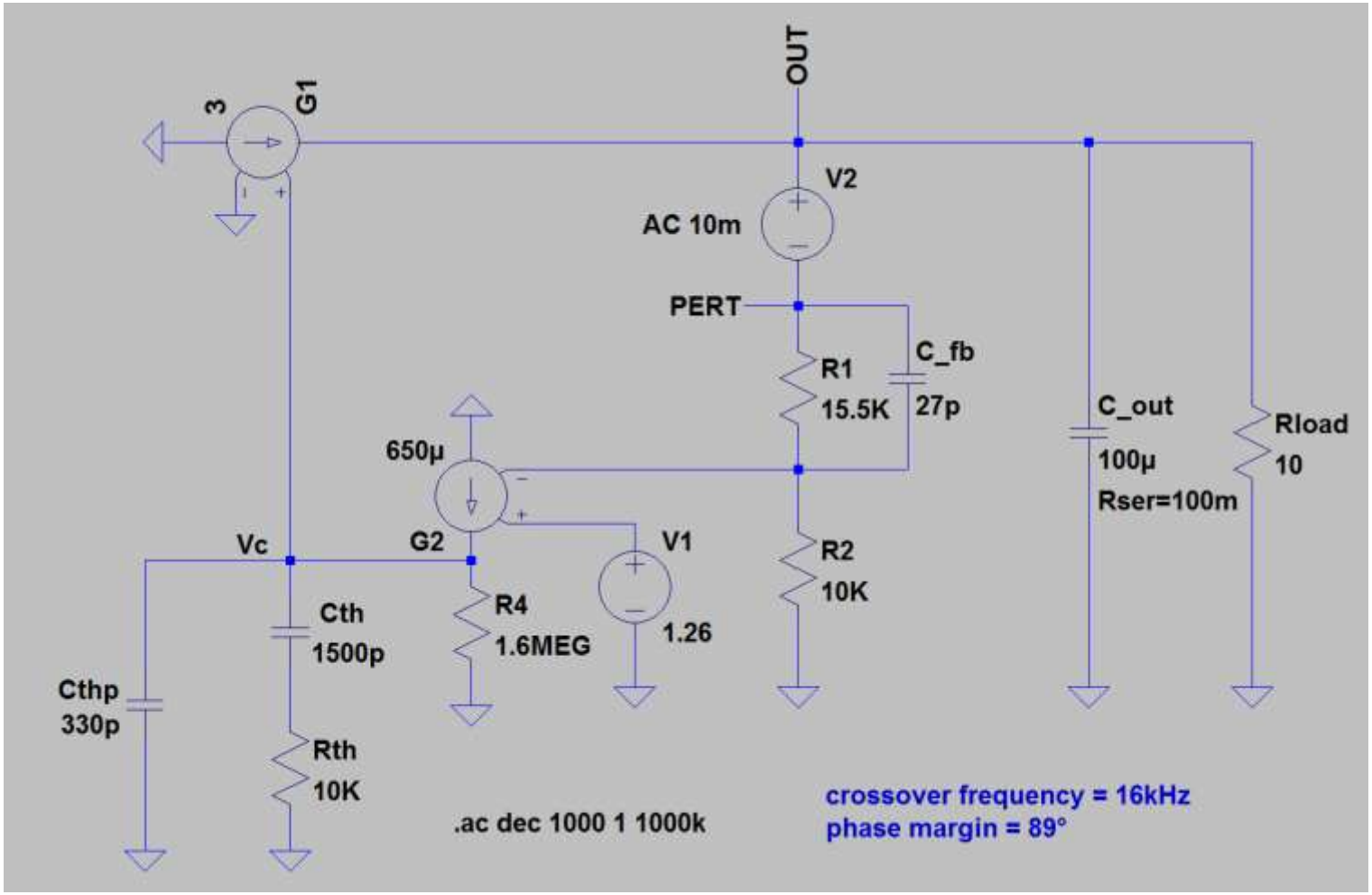
Figure 13. Model for Loop Response

Rout = Voltage Gain / Voltage g_M

	EA Voltage Gain (Note 8)				900		V/V
	EA Voltage g_m	dI(V_C)= ±10μA		400	650	800	μMho
	EA Source Current	FB = 1.15V		20	40	55	μA
	EA Sink Current	FB = 1.35V		15	30	40	μA
	V_C to SW g_m				3		A/V

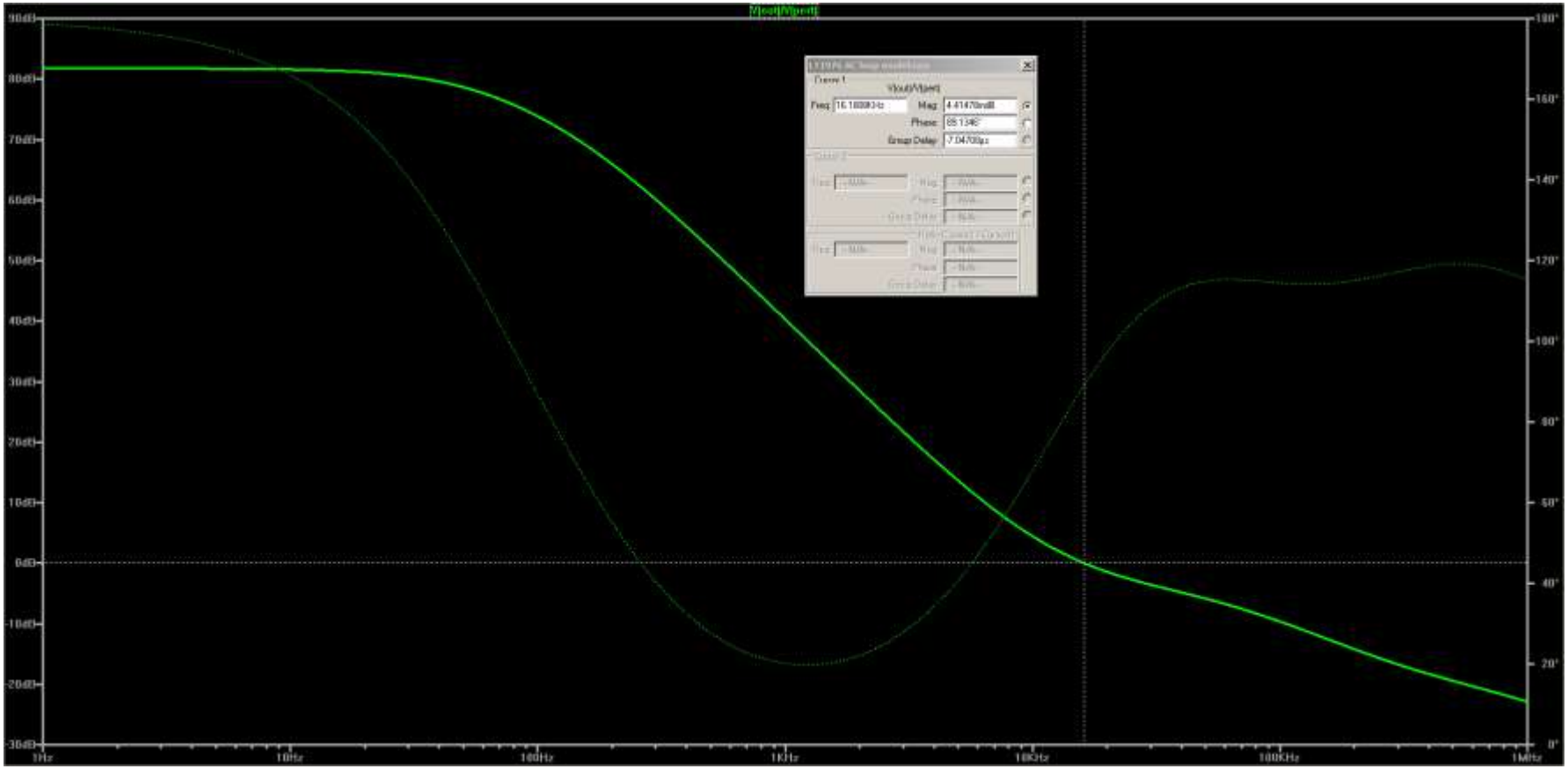
LTspice Model

- LT1976 AC loop model.asc



LTspice Model

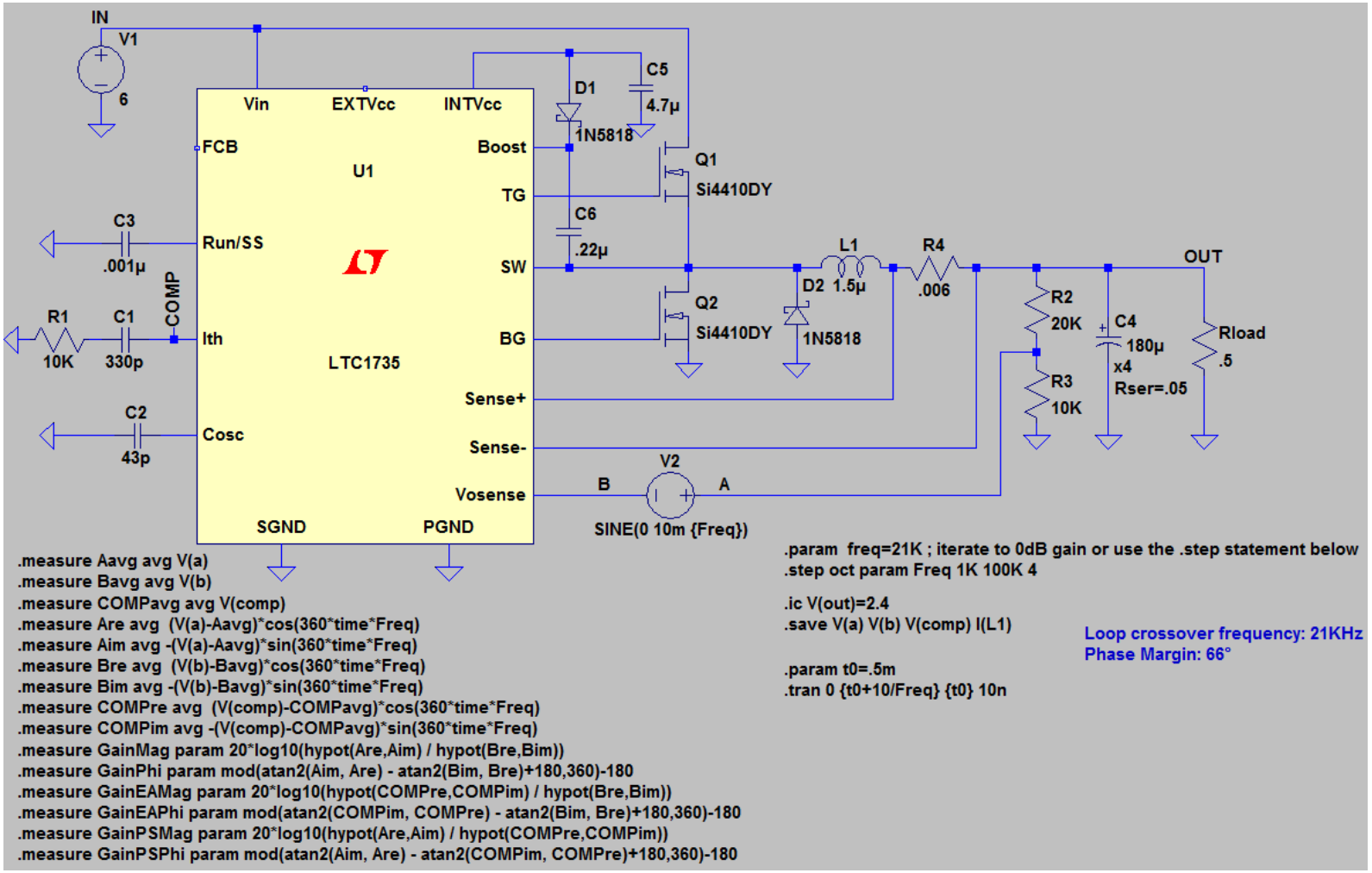
Gain = Vout / Vpert



Method 3: Bode analysis (.TRAN) + (.step)

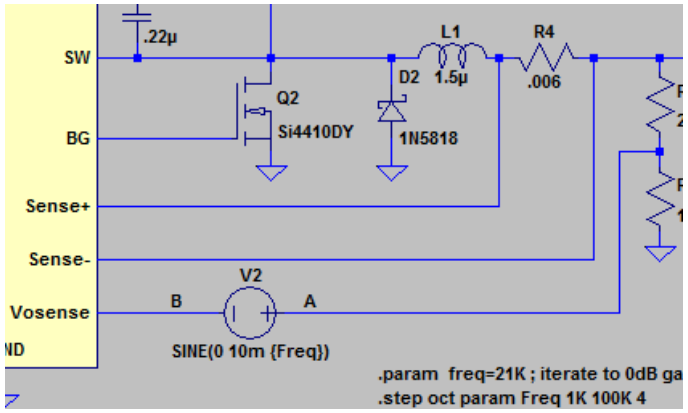
LTC1735 Example

- LTC1735 Bode.asc



Switcher Control Loop Bode

1. **This AC source is the "injector" of a network analyzer:**



Point "A" goes to a low-Z node (Vout) and Point "B" to a high-Z node (Vfb)

2. **Theory and background in-depth:**
   http://www.linear.com/solutions/4672

   Simulation must be run for each point along freq. axis

   `.step oct param Freq 1K 100K 4` **4x per octave = 28 runs of sim**

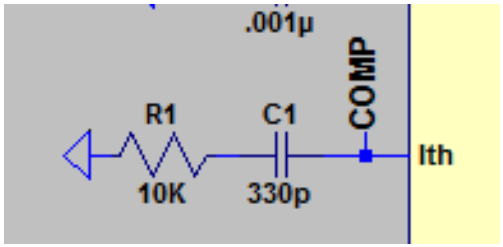
3. **Speed up sim by adding initial conditions:** `.ic V(out)=2.4`

4. **Run simulation, go for coffee and donut (takes ~ 13 minutes)**

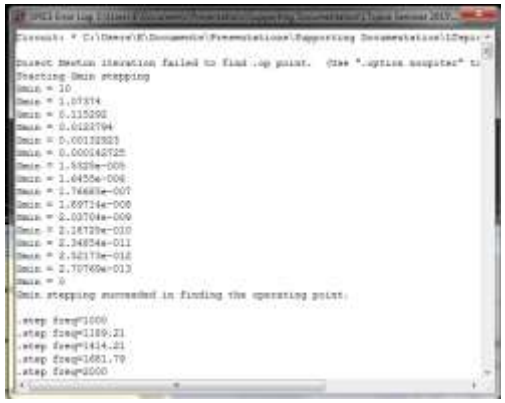
Switcher Control Loop Bode

1. **While you eat donuts, drink coffee:**

   LTspice runs a series of simulation and measures gain and phase at Point A, Point B, and also Point "COMP" at the output of the g_M error amplfier

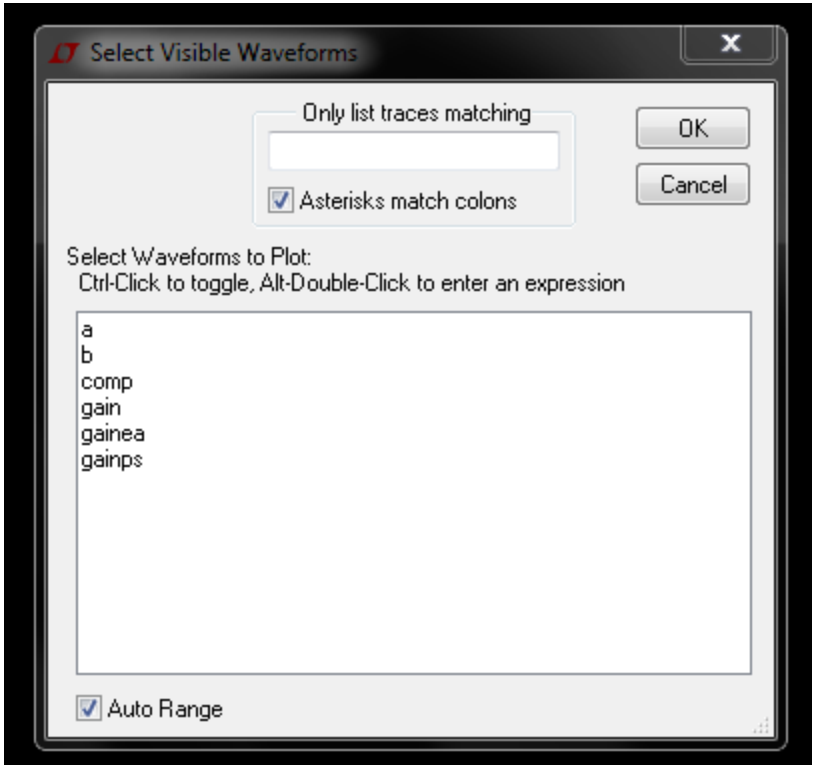


2. **When all sims are done….nothing happens**
3. **Go to View->SPICE Error Log (or CTRL-L)**
4. **Right-click in the Error Log text box**
5. **"Plot step´ed .meas data"**
6. **A blank window pops up**



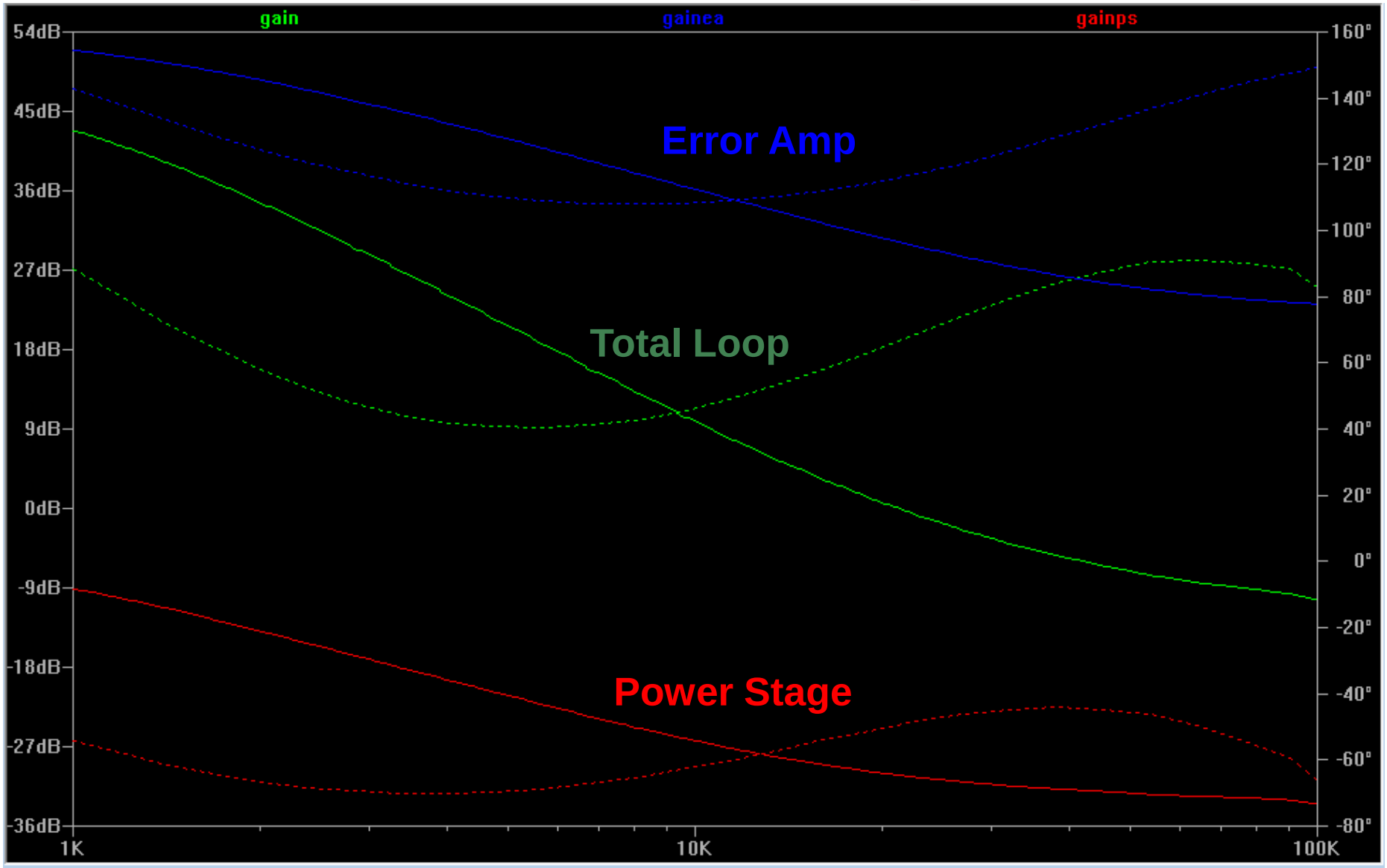
Switcher Control Loop Bode

1. **A blank trace window pops up. Right click again:**



2. **“gain” is the complete control loop gain and phase**
3. **“gainea” is only the Error Amplifier**
4. **“gainps” is only the Power Stage**

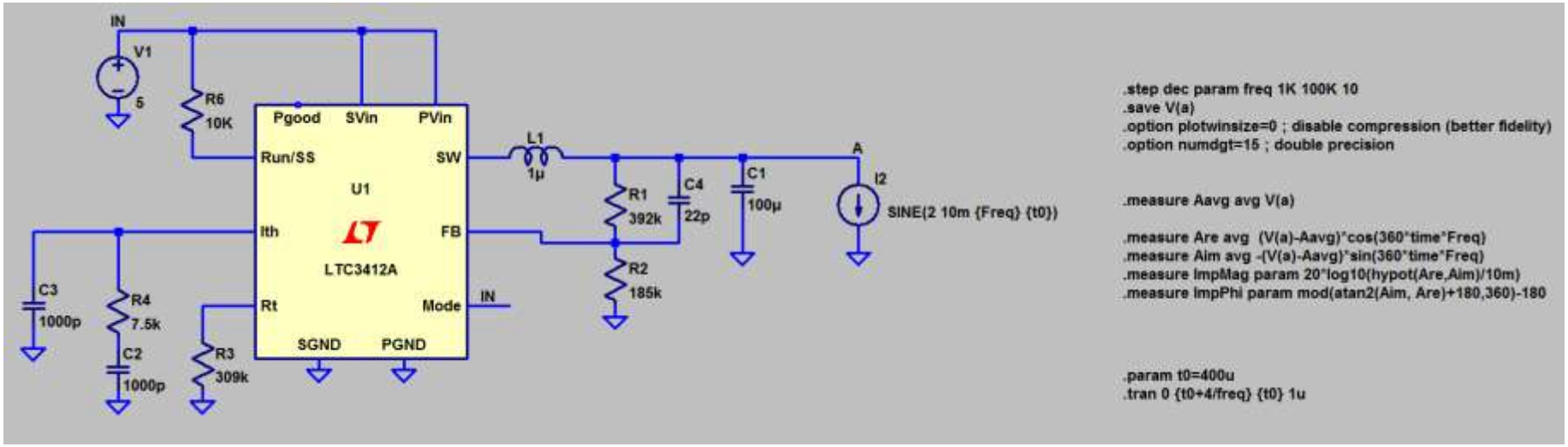
Switcher Control Loop Bode



Output Impedance

- LTC3412A Zout.asc

The same methodology can be applied to measure output impedance



Start sim, go for an organic smoothie and a power bar…

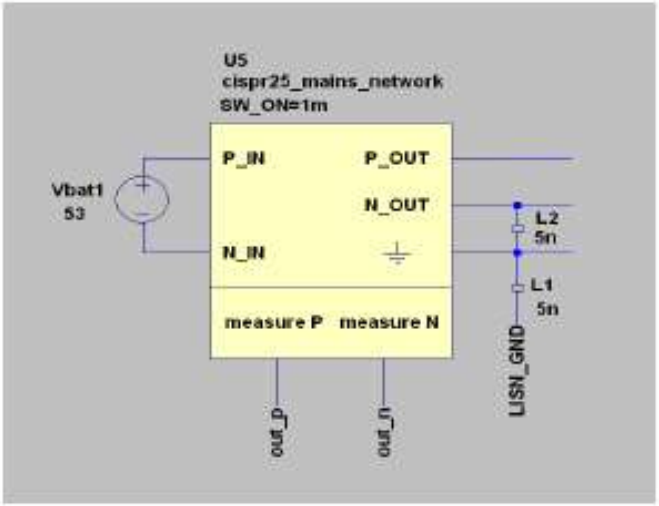
Simulation Results

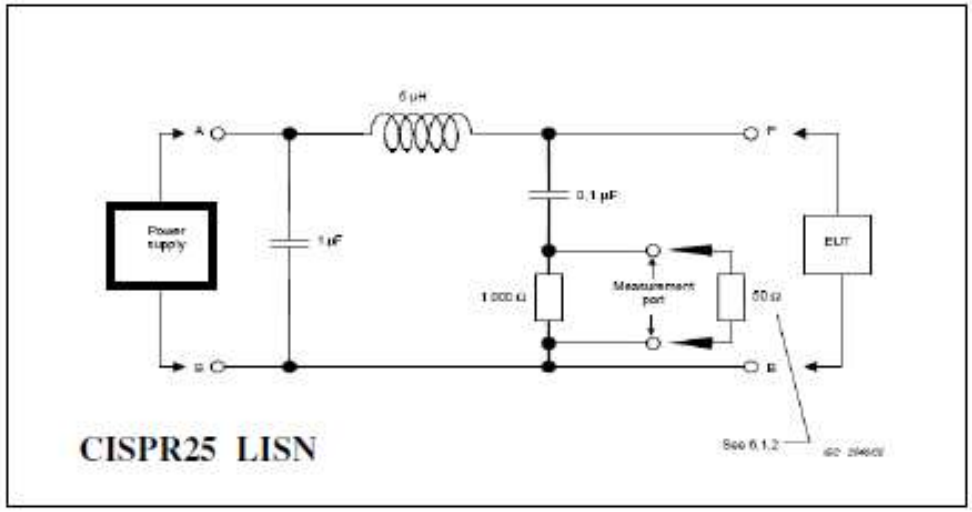
Frequency

Conducted EMI Simulation with LTspice

LISN Model

How to set up an EMI simulation





Simulation model of CISPR25 LISN :

- 2 CISPR25 channels modelled
- 4th order lowpass filtered measurement outputs (anti aliasing)
- Startup with inductance zero, switch on at time specified with SW_ON

FFT Analysis

- The FFT, or Fast Fourier Transform, is a method of calculating harmonics using a special algorithm.
- The FFT requires much less processing power than a DFT for the same number of harmonic results.
- Requires that the number of samples N being analyzed are multiples of 2.

FFT Analysis

- N = number of samples
- ΔT= time increment between samples
- Fs = sampling frequency = 1/ΔT
- T = total sampling time = N.ΔT = N/Fs
- **ΔF**= frequency resolution = **1/T** = Fs/N
- **Fmax** = Folding frequency = Fs/2 = **1/(2.ΔT)**

- Higher Fmax -> lower ΔT (maximum timestep simulation value)
- Lower ΔF -> higher T (stop time - time to start saving data simulation values)

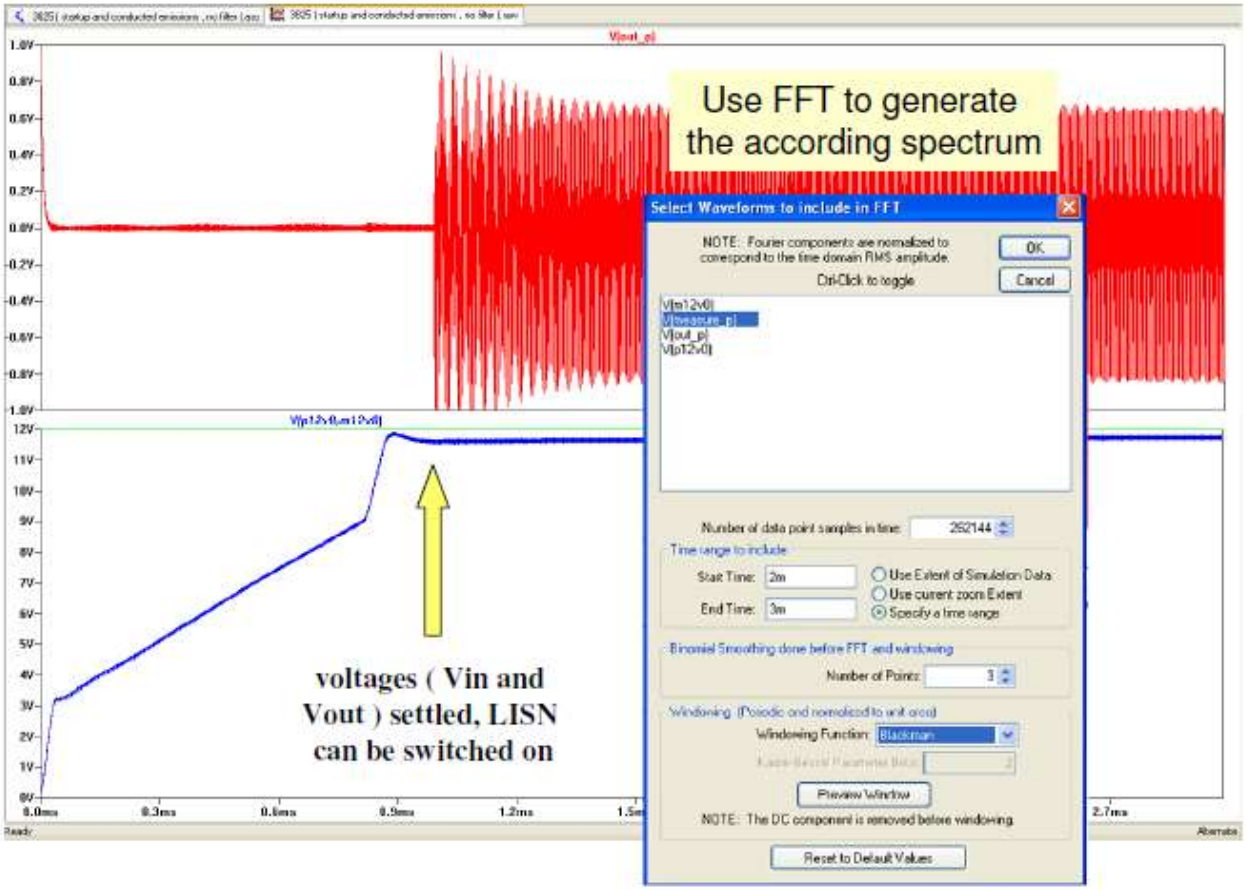
- Example:
  - transient simulation T = 100ms, N = 131072
  - -> Frequency resolution = 1/T = 1/100ms = 10Hz
  - -> Fmax = Fs/2 = ½ N/T = 655kHz

FFT Analysis

- When you do a Fourier Transform (DFT or FFT), you only take a **snapshot** from a signal.
- The Fourier transform then delivers the Fourier coefficients of a signal which is equivalent to an infinity number of concatenated snapshots.
- If your snapshot does not contain **exactly full periods** of all your frequencies (signal), there will be a discontinuity in the assumed concatenated signal from one snapshot (window) to the next.
- As a result, your FFT gives you broad peaks.
- That's where the **FFT windows** come into play. They force the signal and their derivatives to zero at the ends of your snapshot (window). This helps, but is by far not as good as a **well chosen time frame which contains only full periods of your signal**.

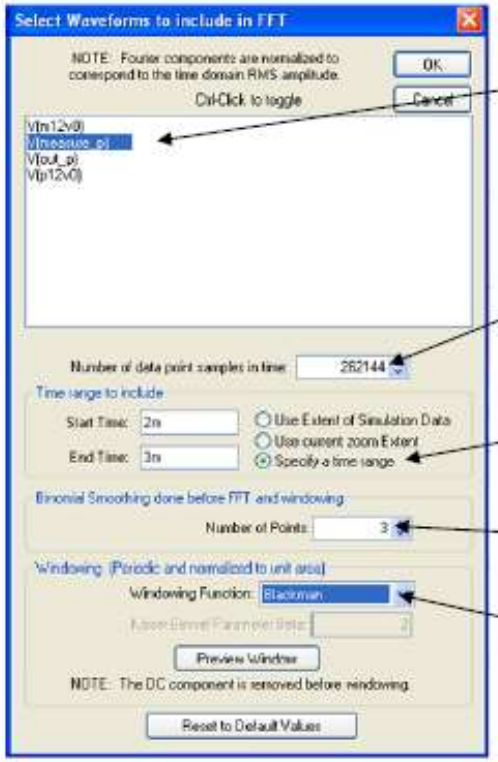
FFT Directive

How to set up an EMI simulation



FFT Directive

How to set up an EMI simulation



postprocessed signal (divided by 1uV)

→ gives you dBuV scaling in FFT spectrum

that is a good number to do FFT of 1ms of transient signal

choose 1ms slice of the data in the settled area

use 1 or 3 here

Lowpass filter, higher N for more filtering. N=1 -> no filtering.

Windowing function with best amplitude accuracy

Example: Low Noise LTM4606

Reduces ripple and HF noise

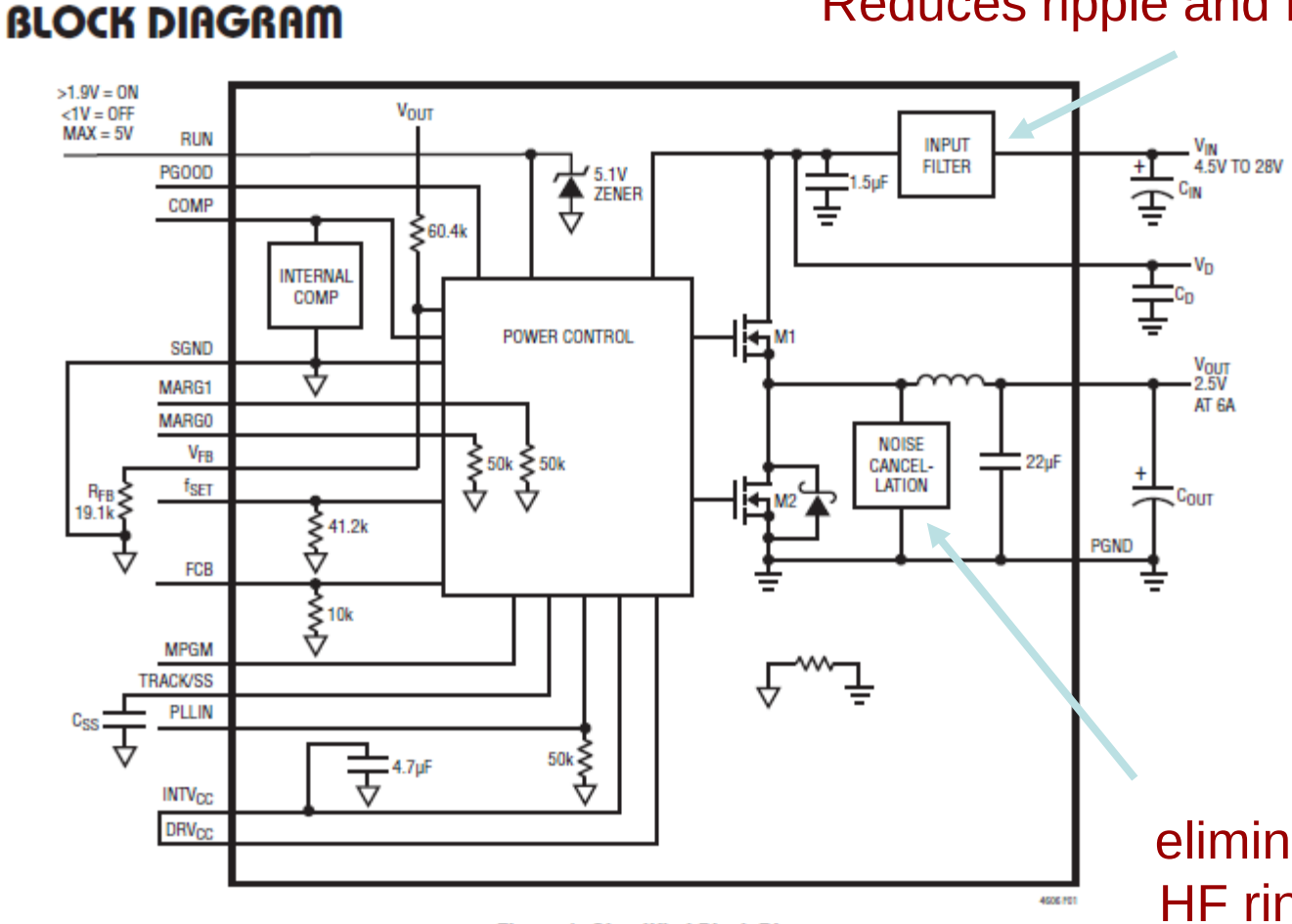
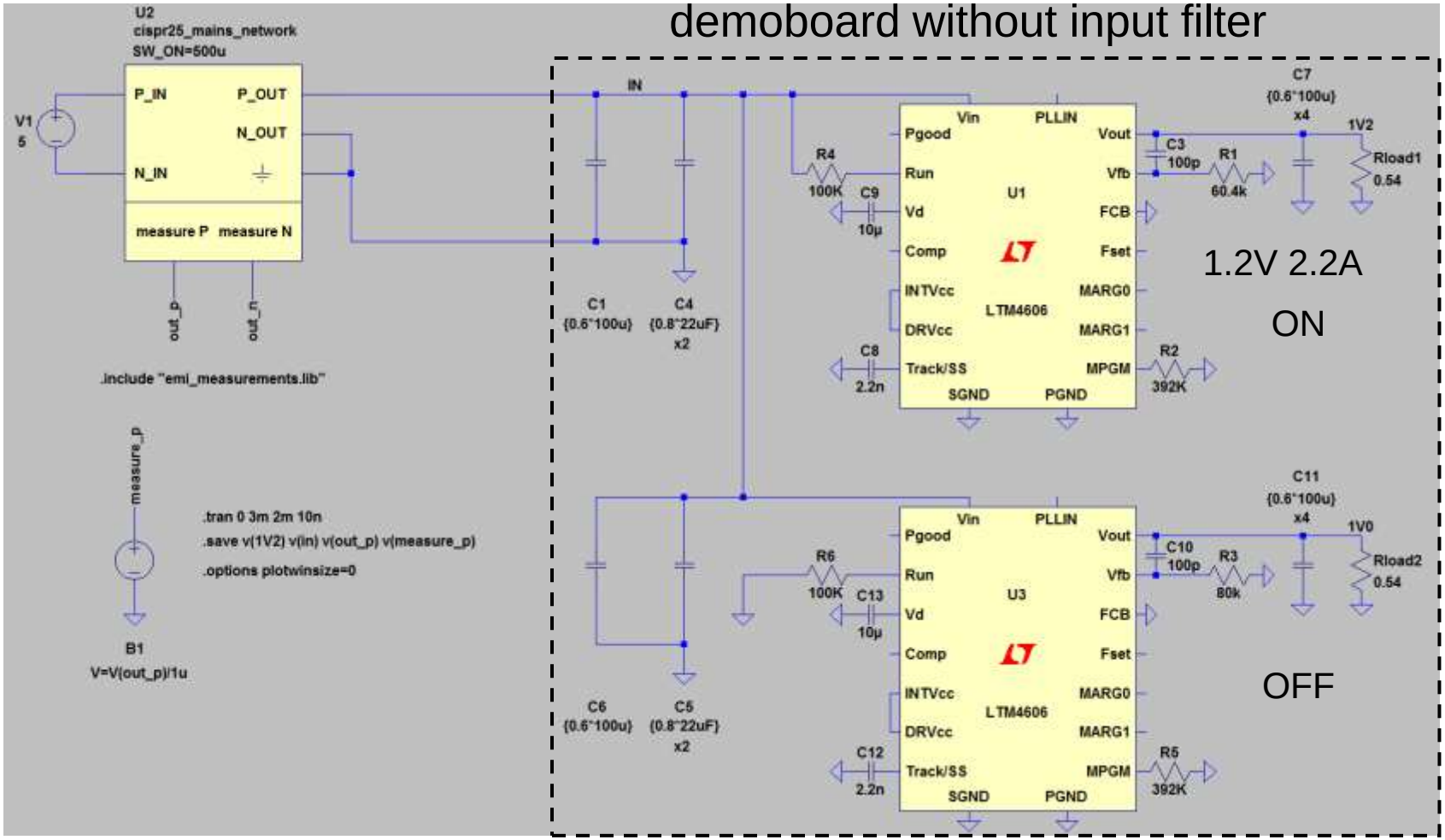


Figure 1. Simplified Block Diagram

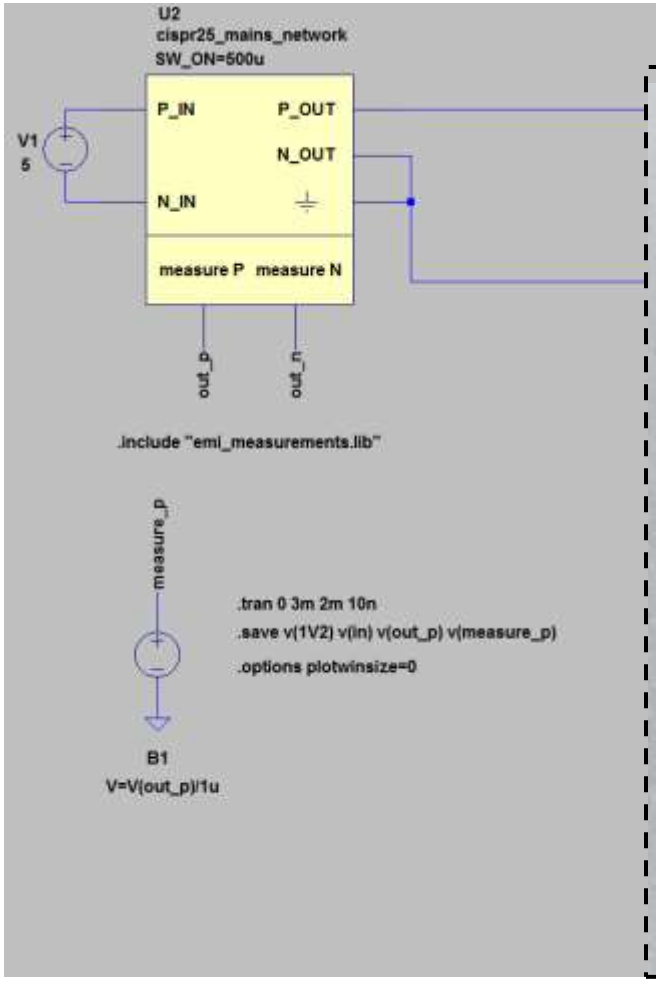
eliminates
HF ringing

Low Noise LTM4606 (EMI simulation)

- Conducted Emissions LTM4606 demoboard.asc

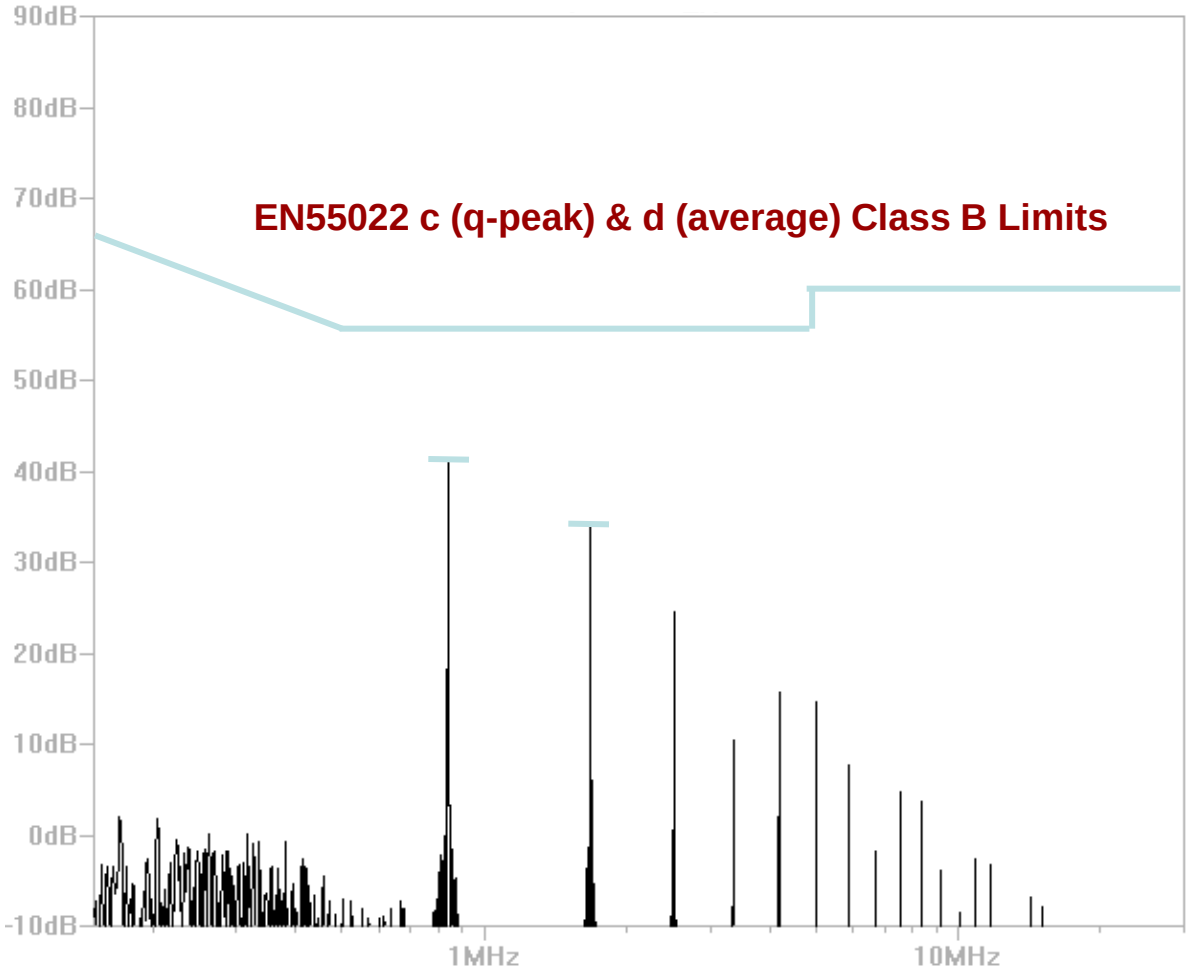


Low Noise LTM4606 (EMI simulation)



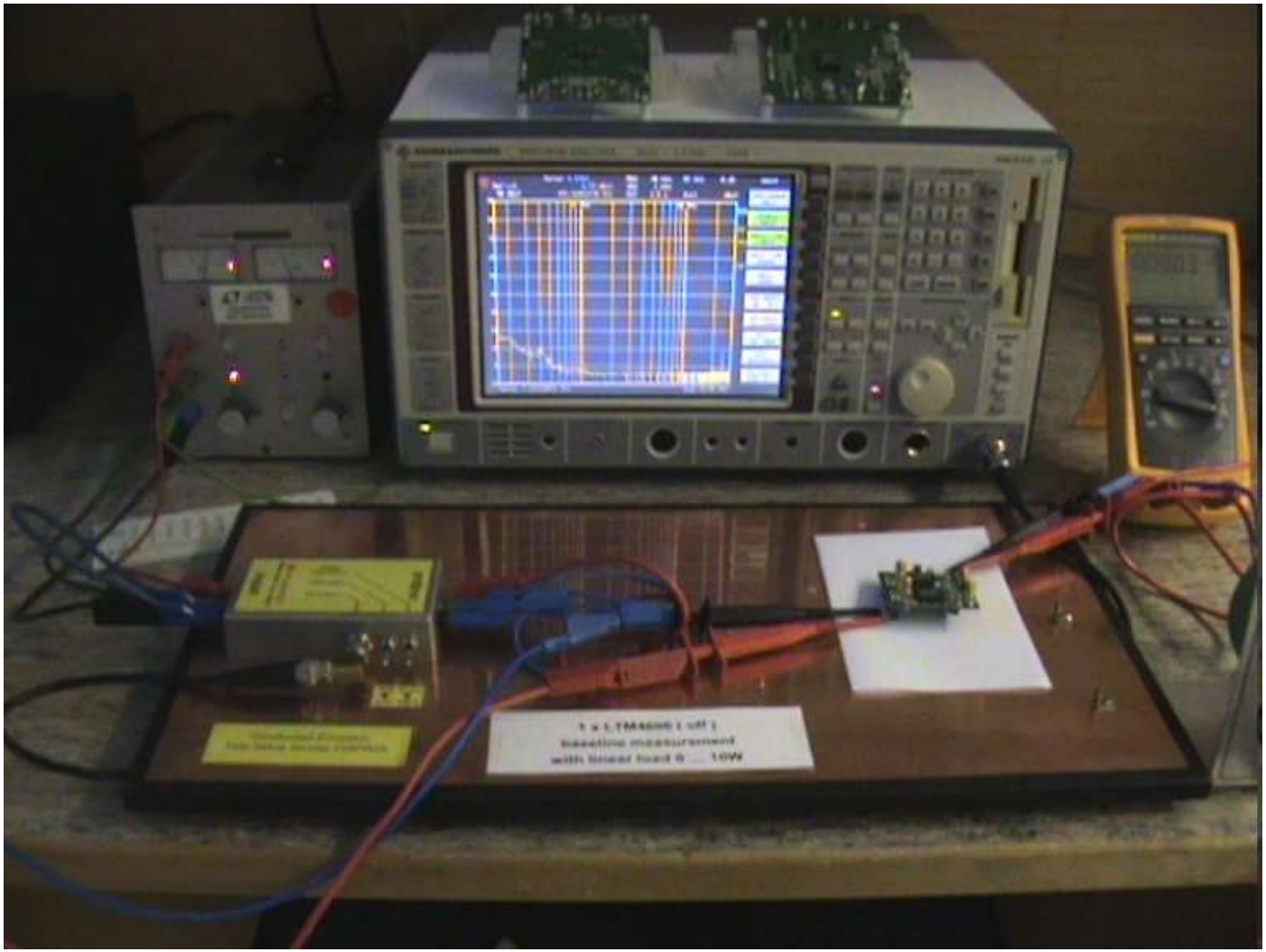
demoboard without input filter

Example 1: Low Noise LTM4606 (EMI Simulation)

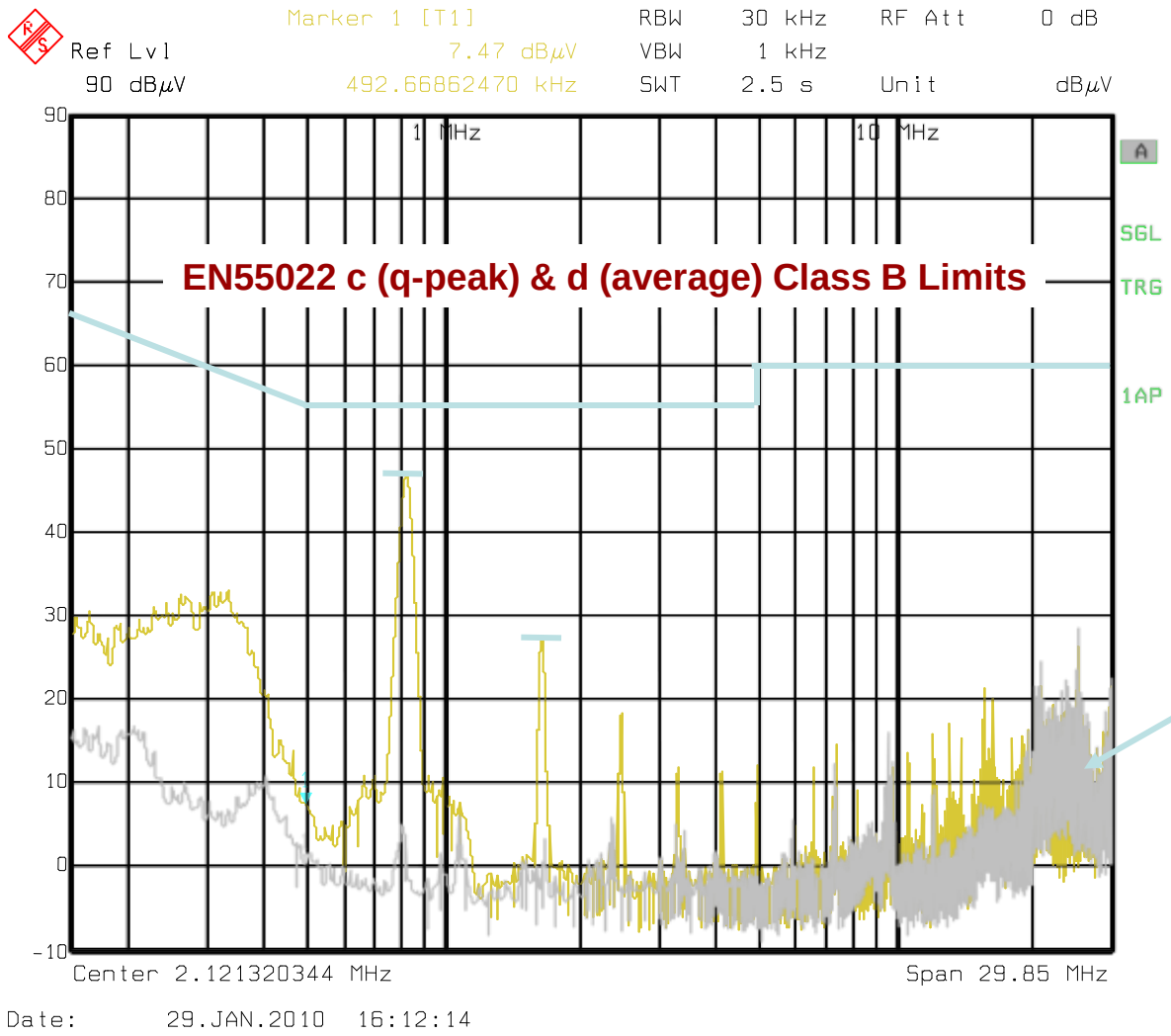


1 x LTM4606, 5V in, 1.2V 2.2A out

LTM4606 (EMI measurement - baseline)



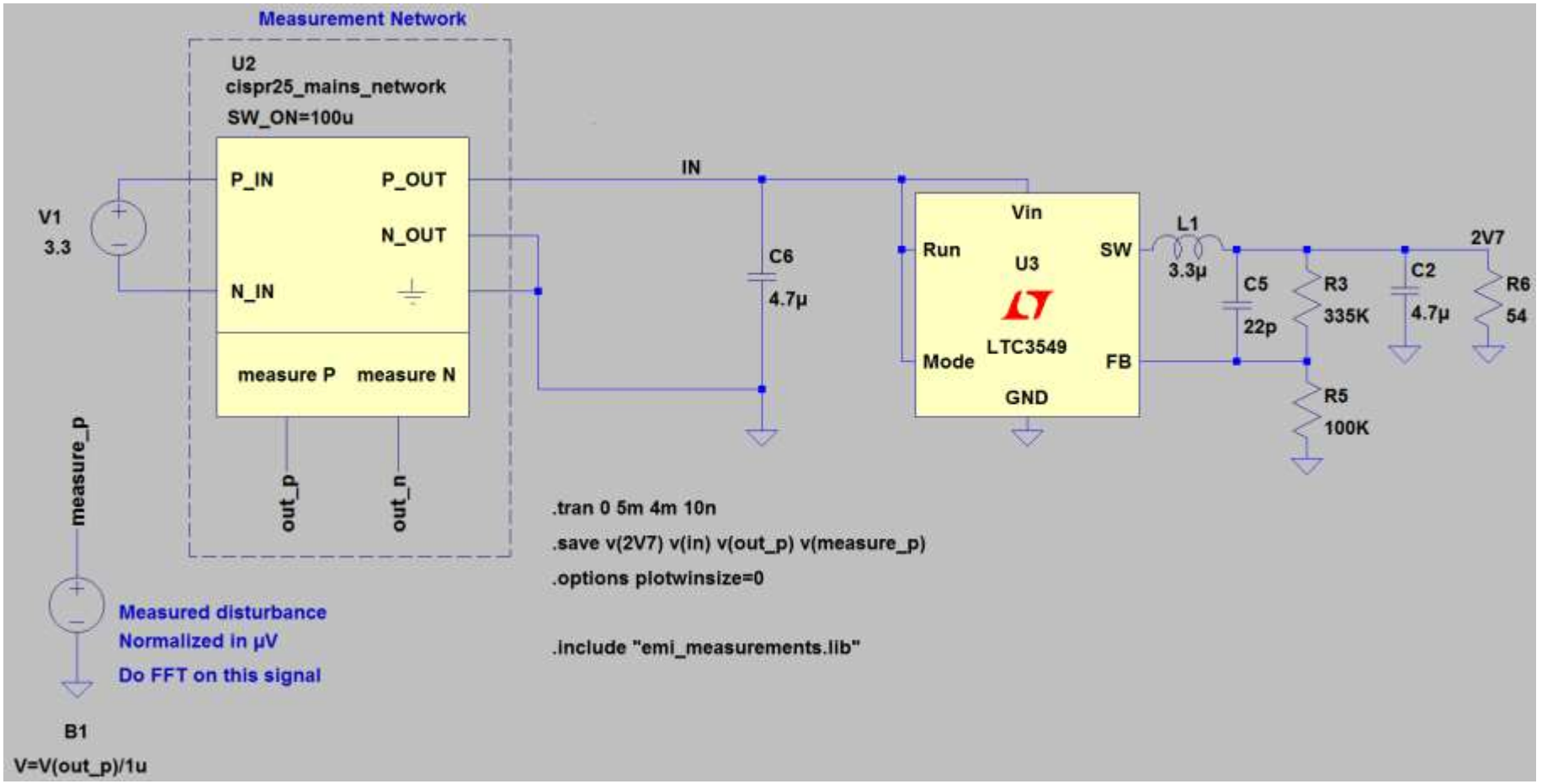
LTM4606 (EMI measurement)



grey : baseline measurement

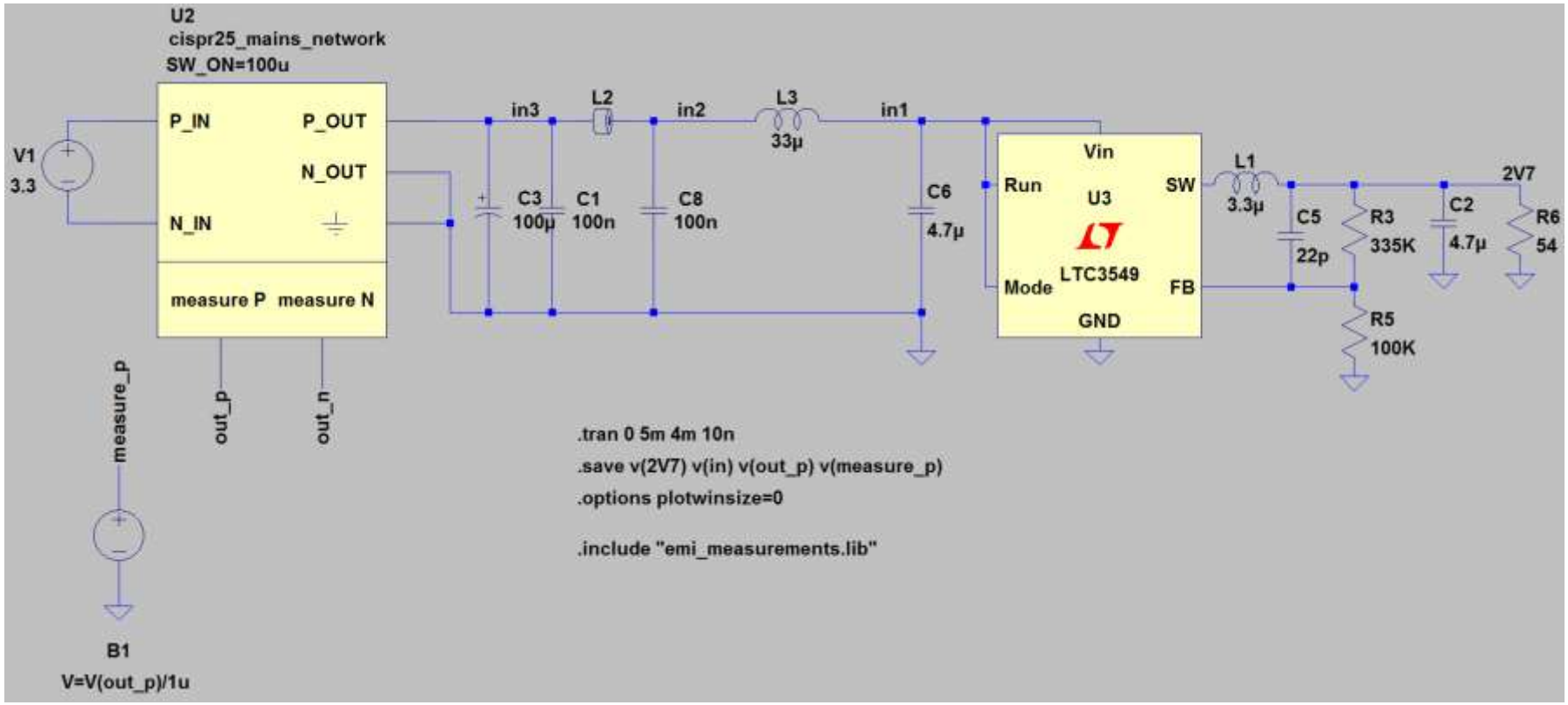
LTC3549 (EMI without Input Filter)

- Conducted Emissions LTC3549 without filter.asc



LTC3549 (EMI with Input Filter)

- Conducted Emissions LTC3549 with filter.asc

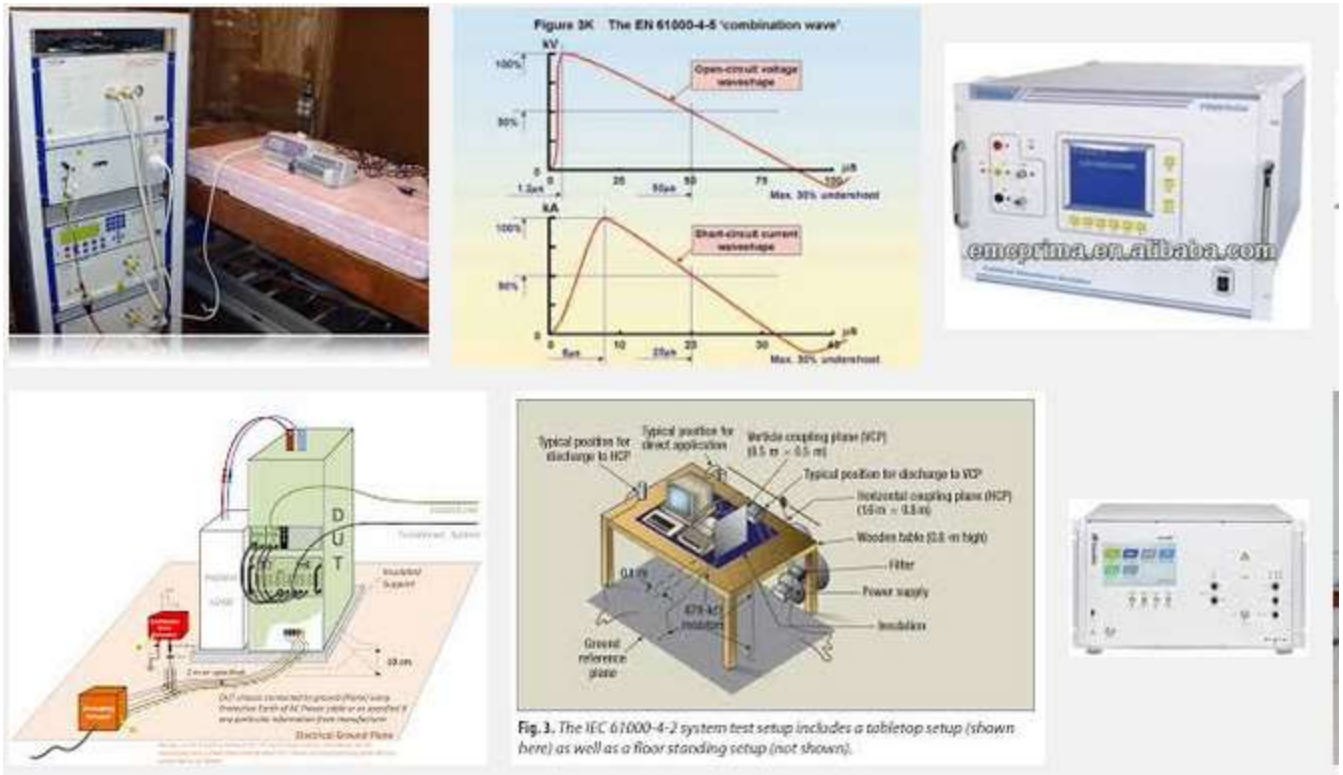


IEC61000-4-5 Surge Testing

Thanks to: *Introduction to Voltage Surge Immunity Testing,* IEEE Power Eelectronics, Denver Chapter, Sep. 18, 2007

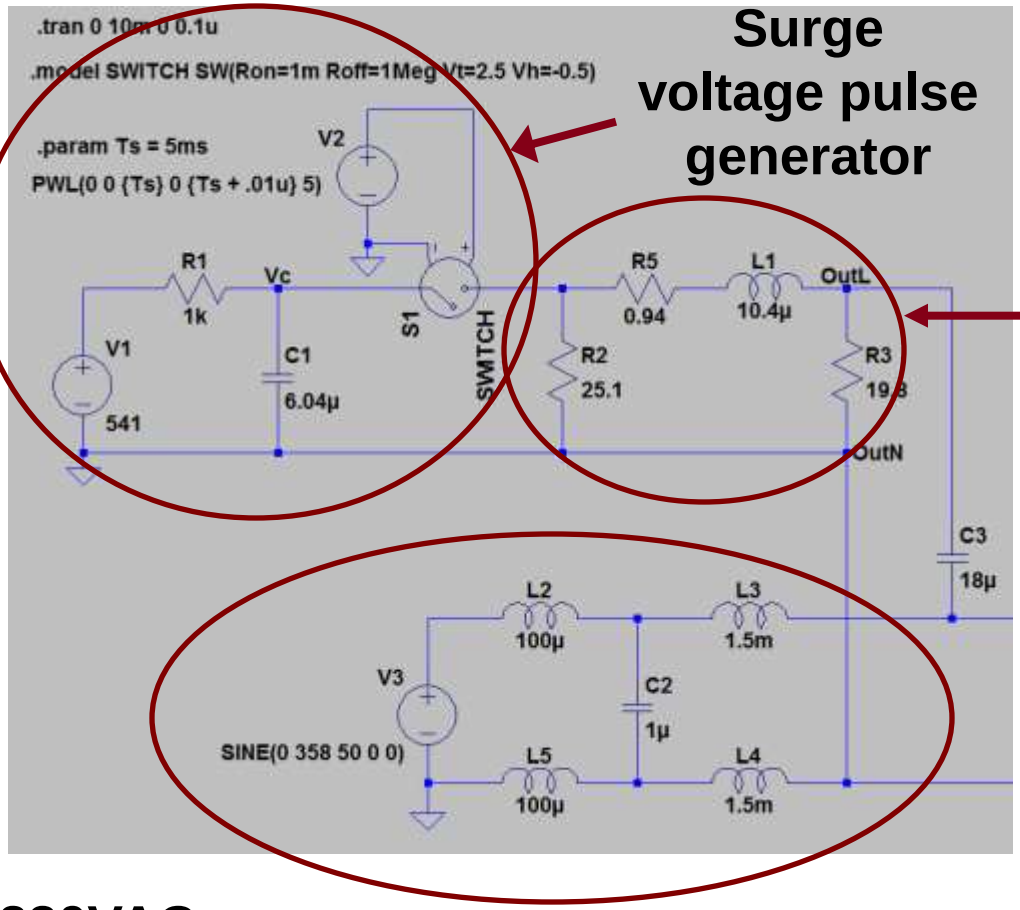
Surge Testing

- Testing equipment for surge is bulky, expensive and complex



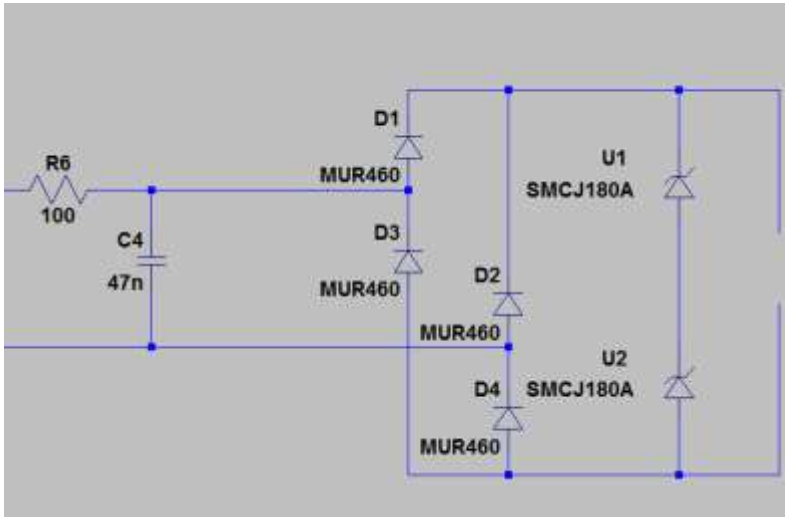
- Section 4-5 of IEC61000 has the highest energy pulses

61000-4-5 In LTspice



Surge voltage pulse generator

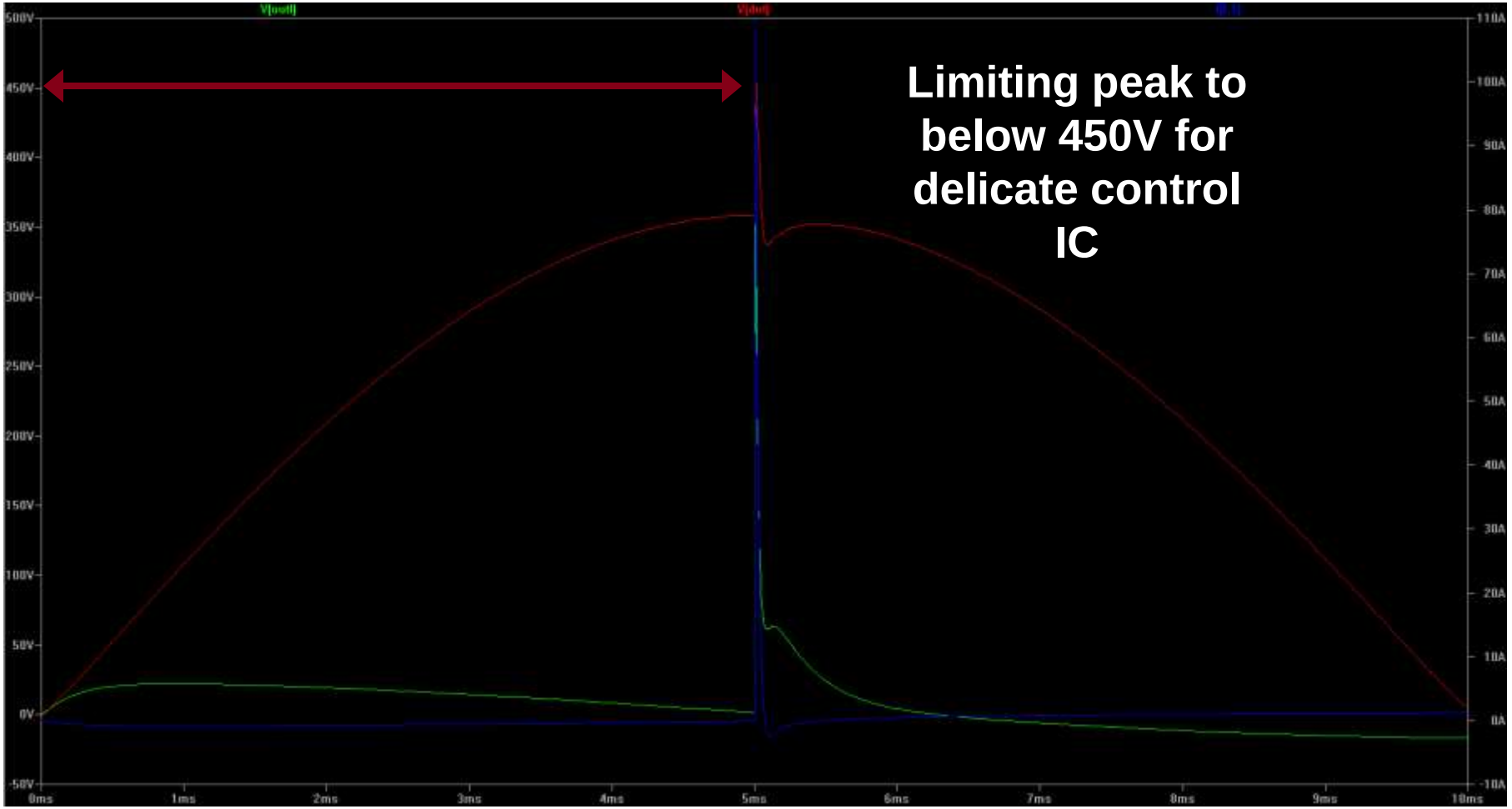
Current pulse shaper



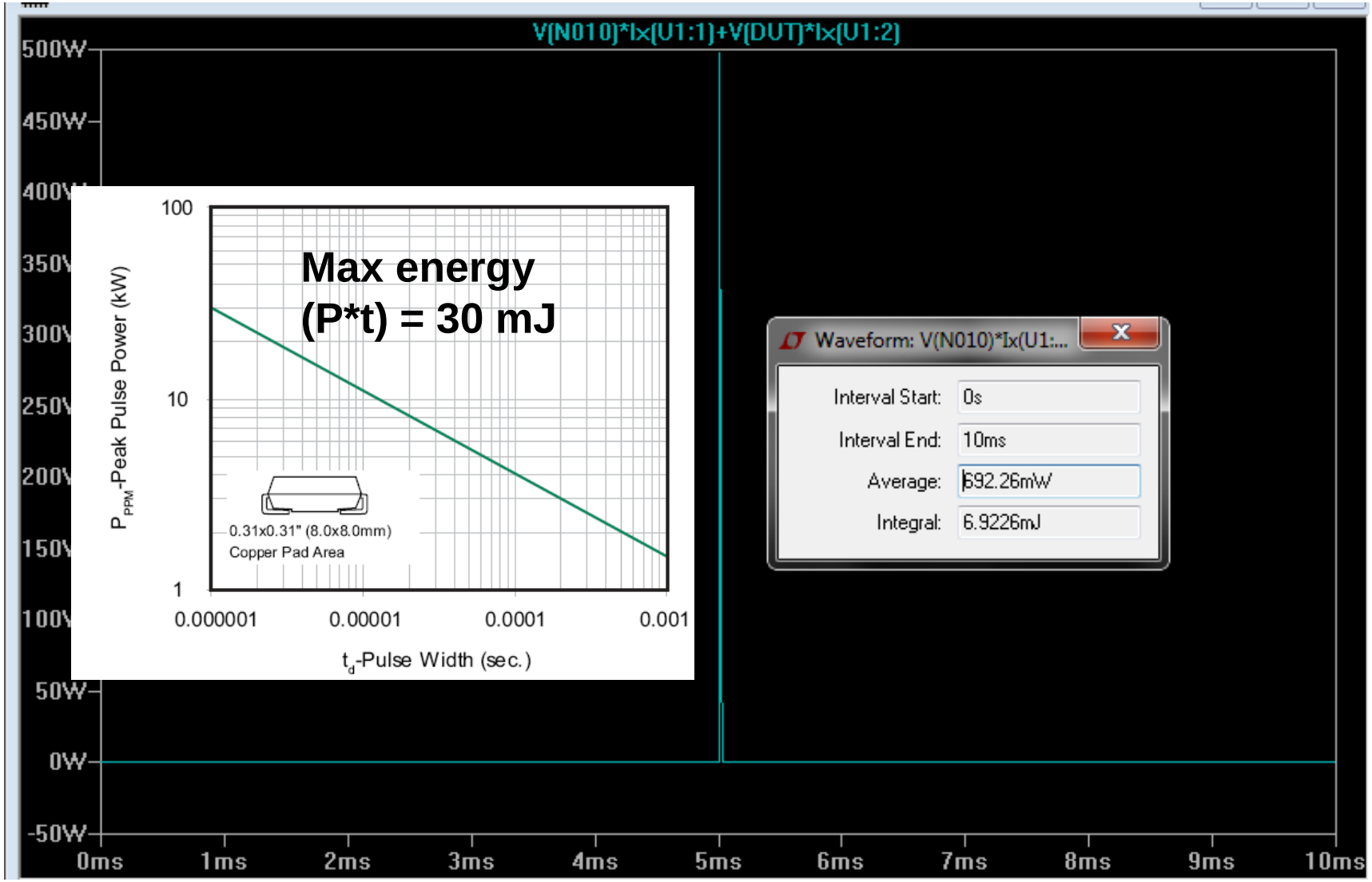
230VAC$_{RMS}$ source with typical impedance

Open-circuit = worst case

Check Maximum Voltage at DUT



Plot Dissipation and Measure Energy in the Varistors/TVS´s



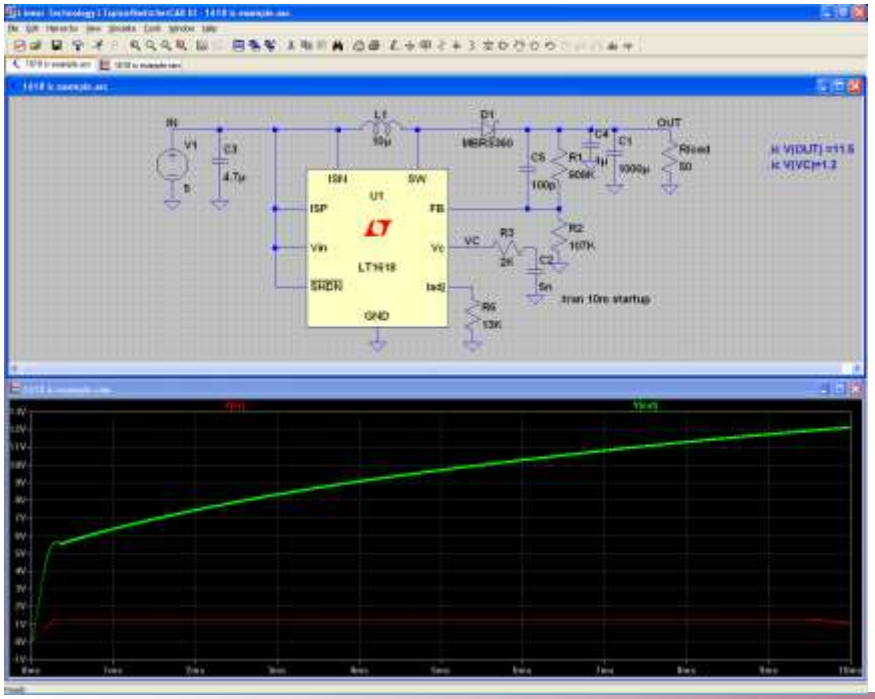
Improving Simulation Speed

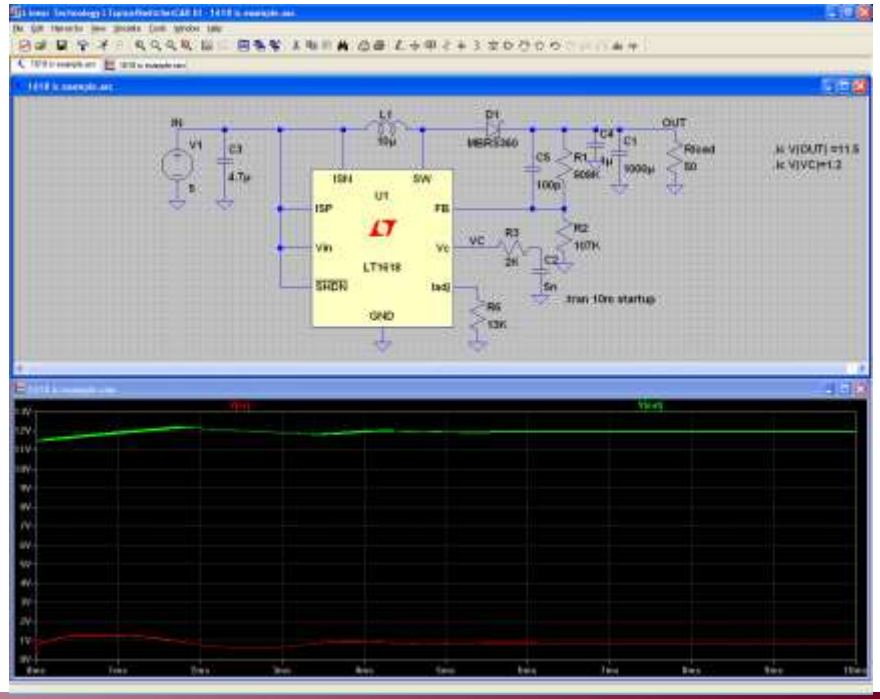
Speed Up Techniques

- For a CPU, get one with as large as possible L2 or L3 cache and as high of clock speed as possible
  - SPICE is sparse matrix math intensive
- RAM speed is also important but pretty much automatically scales with a higher performance CPU
- RAM size is a contributing factor, but cache size has a much larger affect. RAM size primarily helps with regards to system performance when multiple applications are running
- A fast hard drive
- Consider disabling anti-virus scan of the .raw file type

Speed Up Techniques

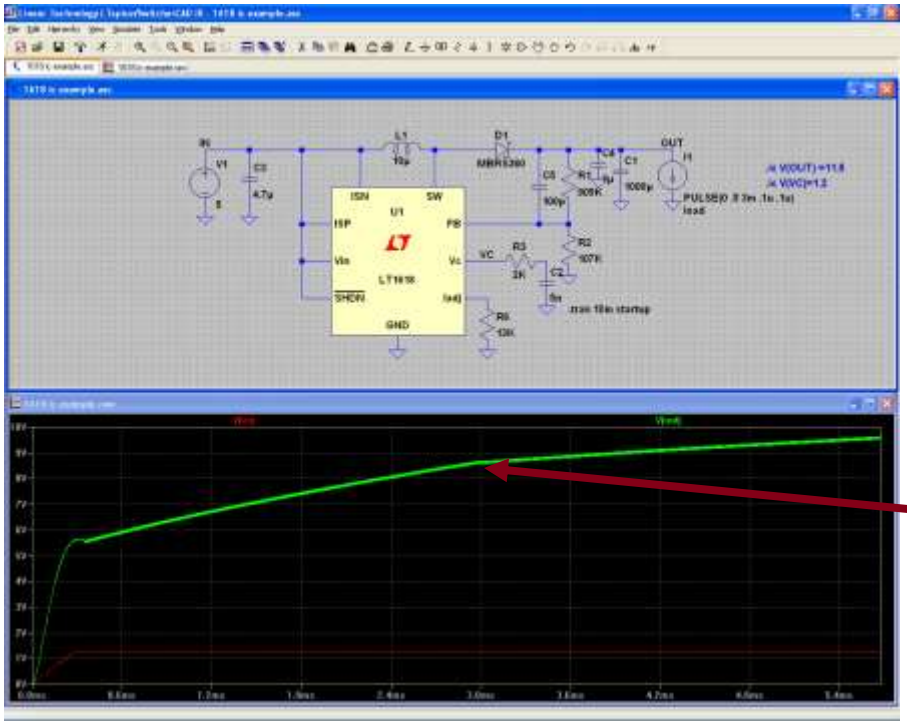
- **LT1618 IC Example.asc**
  - **Initial Conditions**
    - **Use .IC spice directive to set initial conditions**
      - **If output is going to 12V, let's start with it most of the way there**
      - **It is sometimes useful to set other nodes to initial conditions as well. Ex. put an initial condition on the VC node**





Speed Up Techniques

- LT1618 Delay Load Example.asc
- Delay loads using a Pulse Current for a load
  - The time to get the load to the final value can be less if there is no load present during startup (all of the energy is then going into the output caps)
  - Notice the inflection at 3ms when the load turns on



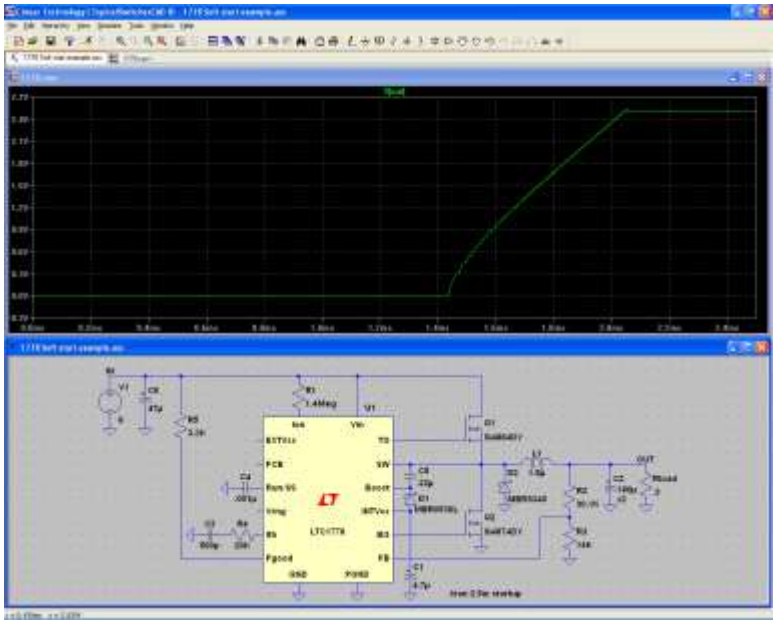
Load turn-on inflection

Speed Up Techniques

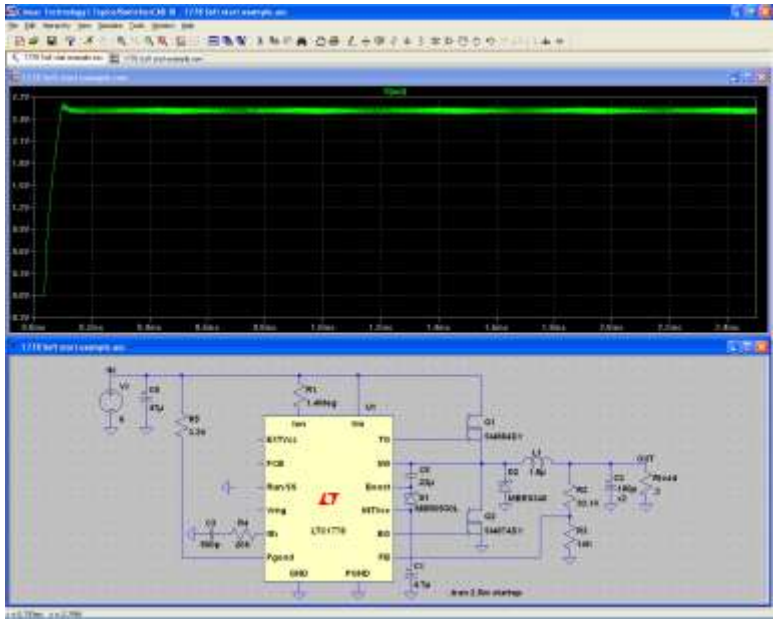
- **LTC1778 Soft Start Example.asc**

- **Remove or reduce the soft start from the circuit (frequently a cap)**

With soft start



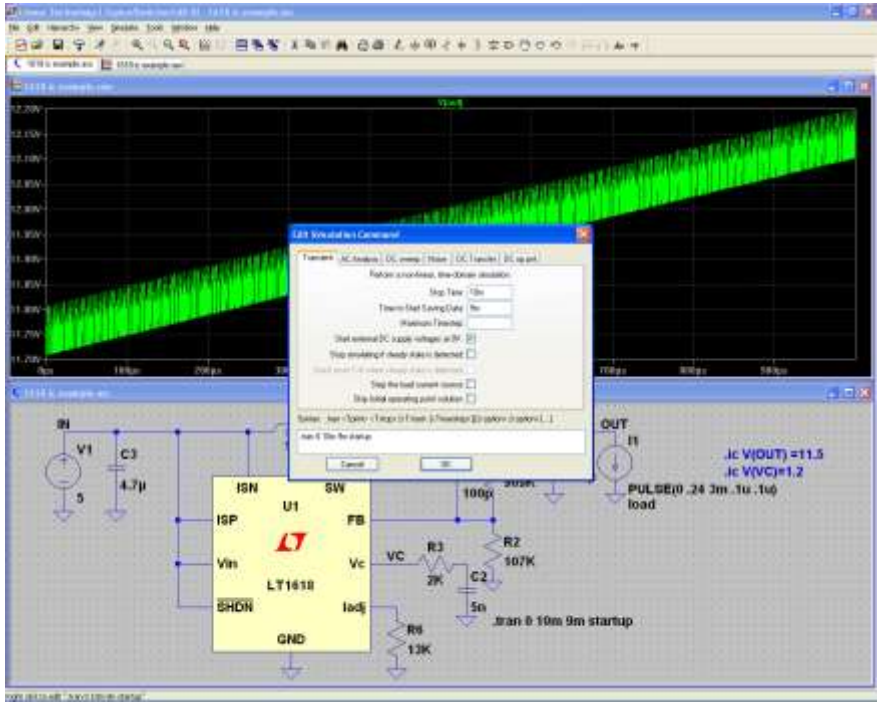
Without soft start



Speed Up Techniques

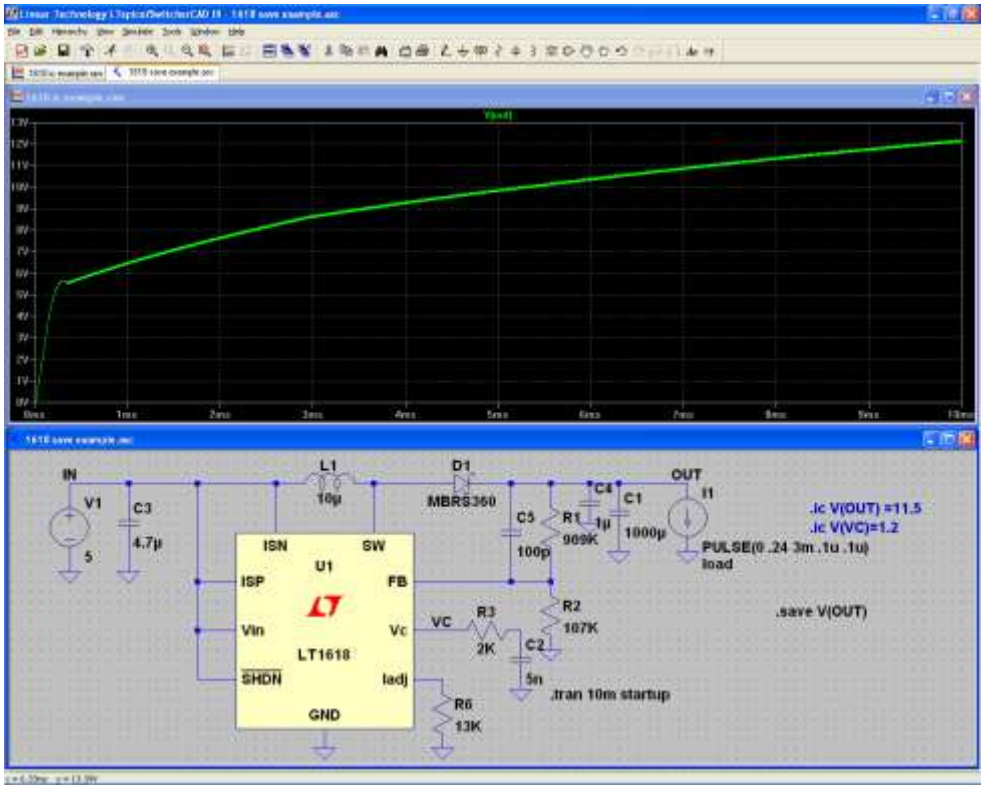
- **LT1618 Save Example.asc**

- **Time to start saving data**
  - **Use this to instruct LTspice to not save any data until the time specified. Fewer things saved to memory/hard drive = more speed.**
  - **Waveforms before "Time to start saving data" time are lost and not viewable and the analysis data has been thrown away.**



Speed Up Techniques

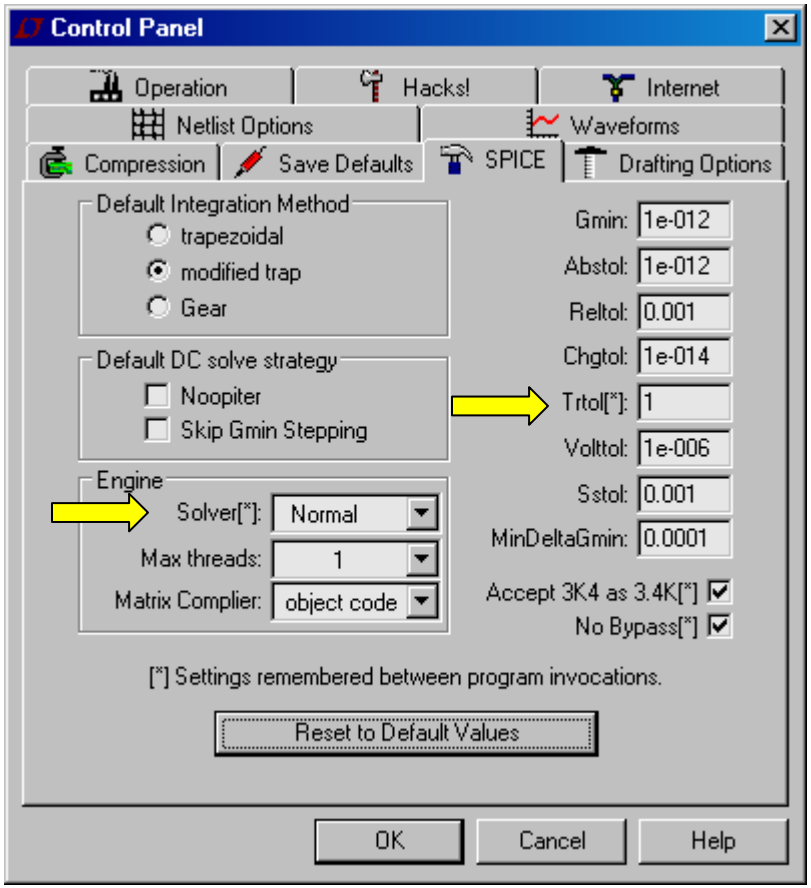
- Use the .SAVE SPICE directive to save only the traces that you need
- Ex.: If you are only interested in Vout, use the .SAVE V(OUT) SPICE directive to only save the OUT node data to the hard drive
- Only the saved nodes are viewable as a waveform, all other simulation data is discarded



Speed Up Techniques

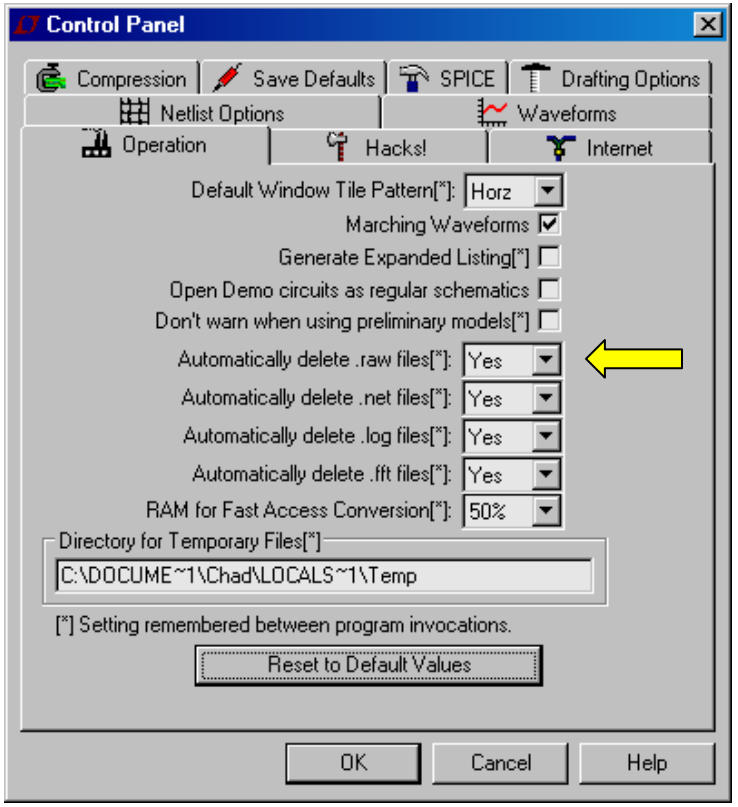
- Control Panel
  - Different Solvers: **Normal** is faster. **Alternate** is more accurate but slower. Default is Normal.

- **Trtol** (Transient error tolerance) - This parameter is an estimate of the factor by which the actual truncation error is overestimated. Most commercial SPICE programs default this to 7. In LTspice this defaults to 1 so that simulations using the SMPS macromodels are less likely to show any simulation artifacts in their waveforms. Trtol affects the timestep strategy more than it directly affects the accuracy of the simulation. For transistor-level simulations, a value larger than 1 is usually a better overall solution. You might find that you get a speed of 2x if you increase trtol without adversely affecting simulation accuracy. Your trtol is remembered between program invocations.



Other System Resource Tricks

- **Pause** – Under the Simulate Menu. Useful when you need to use your computing resources for something else temporarily.
- Have plenty of **defragmented** empty hard drive space. This speeds up a simulation by a factor of 3 in extreme cases!
- **Automatically delete .raw files** (the waveform data file). Raw files can be very large (100's of megs +) depending on the number of nodes in a circuit, and the length of the simulation, etc. This option deletes the current raw file when LTspice is closed. This doesn't necessarily speed up simulations, but it helps to maximize free space on your drive.



Thank you for your attention