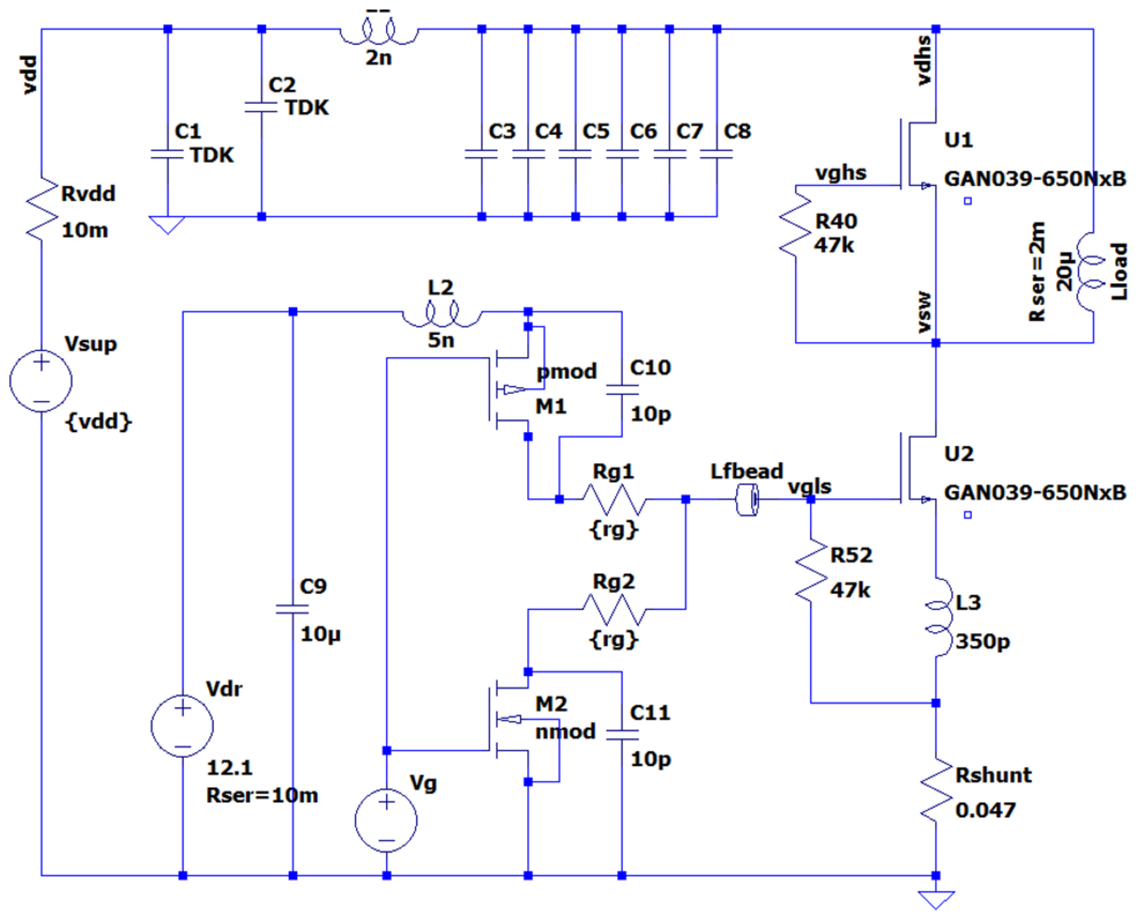
AN90053

Rev. 1.0 — 31 May 2024

Application note

Advanced SPICE models for Nexperia cascode Gallium Nitride (GaN) FETs



Abstract: This application note presents Nexperia's advanced SPICE models for cascode GaN FETs. Details of the model versions and structures are included together with their application in circuit simulations. Comparisons of the simulation results and actual measurement data are provided, showing an excellent fit. Guidance on using the models is given e.g. solving convergance issues.

Keywords: SPICE, GaN, electrothermal

1. Introduction

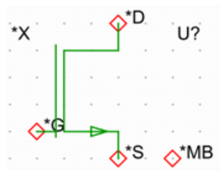
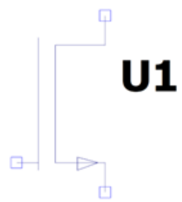
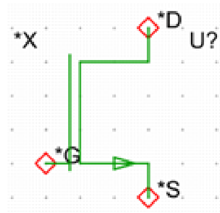
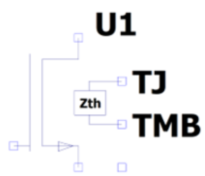
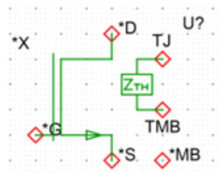
Recently Nexperia released an advanced SPICE model for the new GAN039-650NBB device with extended precision. The advanced SPICE model has the following features:

- High quality fitting to experimental DC, AC and transient characteristics
- Frequency-dependent parasitic inductances of package interconnections (skin-effect)
- Leakage currents and drain-source breakdown modeling
- Model is fitted across full temperature range: -55 to 175 °C
- Smooth current characteristics, including their derivatives, across the whole voltage and temperature range
- Dynamic electrothermal version of the model – first GaN cascode electrothermal model on the market.

2. Available versions

Models are currently available for using in two simulators – LTspice and SIMetrix. They are encrypted due to advanced modelling techniques used to create them. For all simulators there are 3 versions of the same model:

Table 1. Model versions

Version	Description	Name format	LTspice symbol	SIMetrix symbol	Pins
Isothermal	Electrical model with global temperature dependence, local temperature of device equal to the global temperature and remains constant during transient analysis.	*part_name*.asy			4
Isothermal without package stray inductances	Electrical model with global temperature dependence	*part_name*_NO_IND.asy			3
Electrothermal	Electrical model with dynamic thermal capability that includes self-heating of the device	*part_name*_ETH.asy			5

All information provided in this document is subject to legal disclaimers.

3. Internal structures of the model versions

Isothermal electric model.

Fig. 1 shows the basic variant of the model, and it should be used at the main design stage with global fixed temperature. It consists of a GaN FET connected in series with a low-voltage power MOSFET, *including body diode*, and frequency-dependent (skin effect) parasitic inductances and resistances of the package. The mounting base terminal, MB, is internally connected to the source and may be left unconnected in the simulation schematic, or it may be connected to impedances which model the physical MB connection.

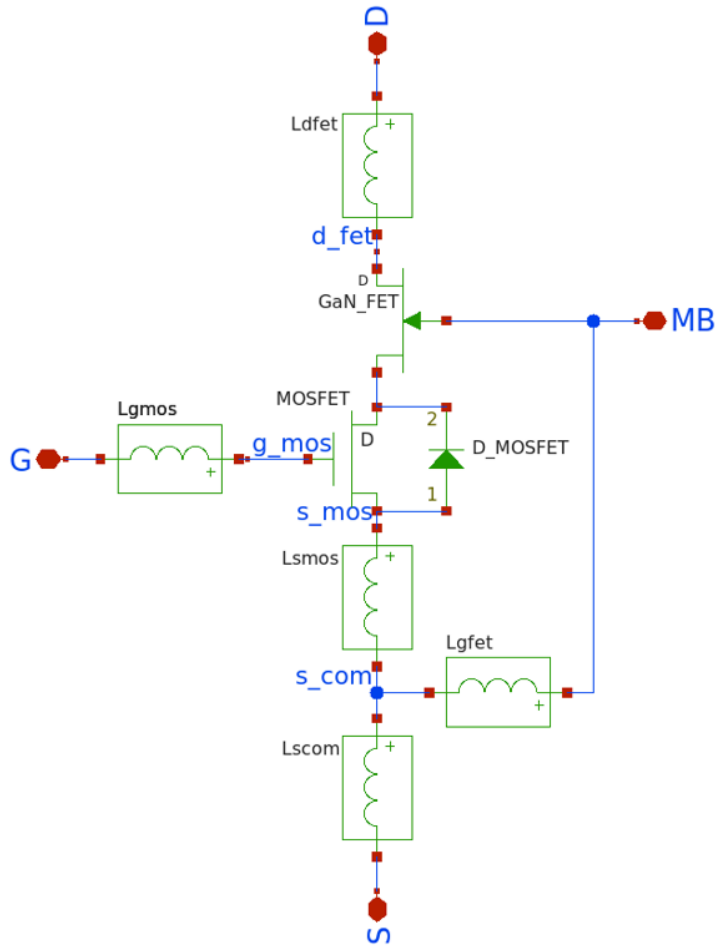


Fig. 1. Internal structure of the cascode Isothermal electric model

Cascode Isothermal electric model without parasitic inductances.

This model version allows you to speed up simulation time as well as define package parasitics externally to get access to internal nodes of the device, see Fig. 2.

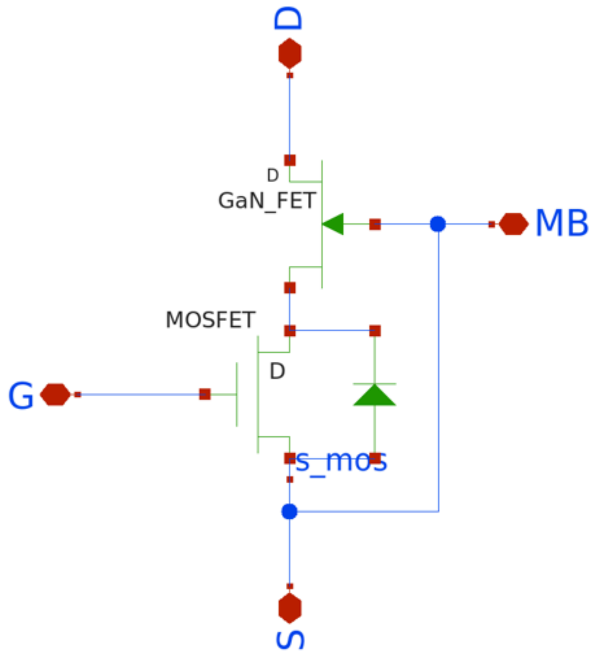


Fig. 2. Internal structure of the cascode Isothermal electric model without parasitic inductances

Electrothermal model

Fig. 3 shows the Electrothermal version which has a built-in Cauer thermal network the values of which are fitted to Z_{th} experimental data. Each internal device has its own source of heat, that produces real active heat loss power.

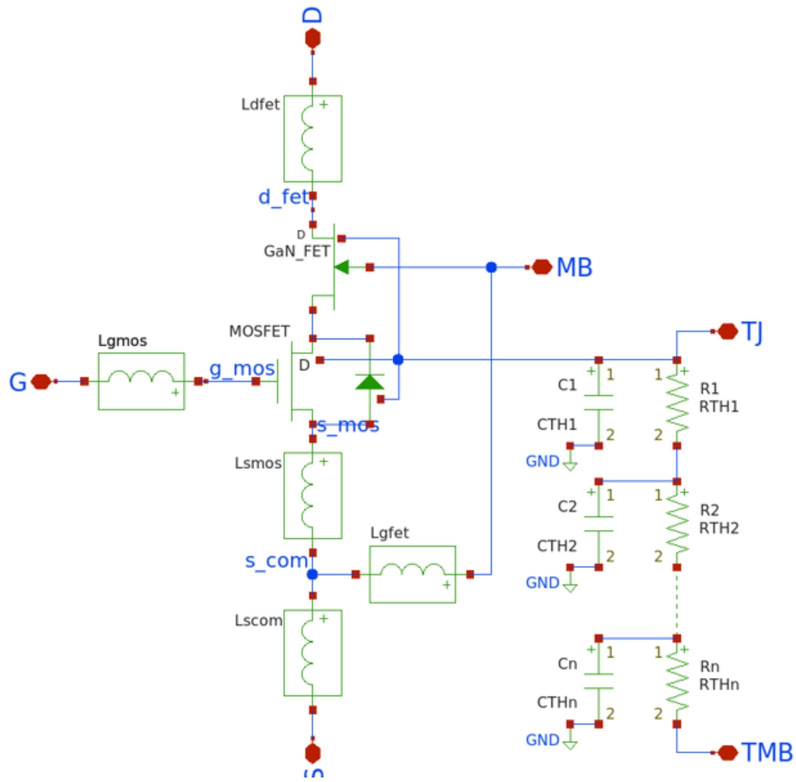


Fig. 3. Internal structure of the cascode Electrothermal model

4. Fitting accuracy of the new advanced models

The characteristic curves generated using the new advanced models show an excellent match with the measured values. See Fig. 4, Fig. 5, Fig. 6 and Fig. 7.

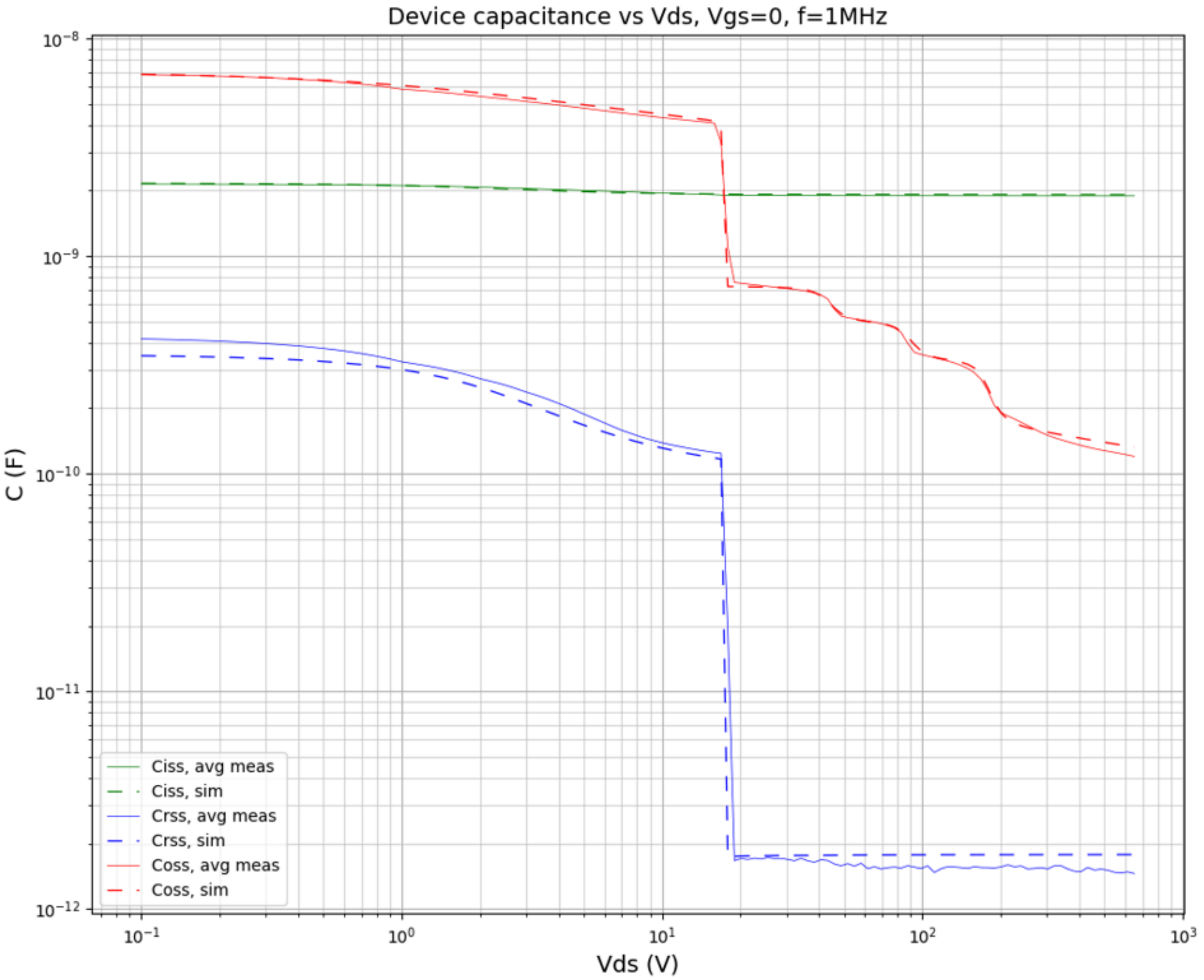


Fig. 4. C_{iss}, C_{rss} and C_{oss} as a function of drain-source voltage; solid = experiment, dashed = model

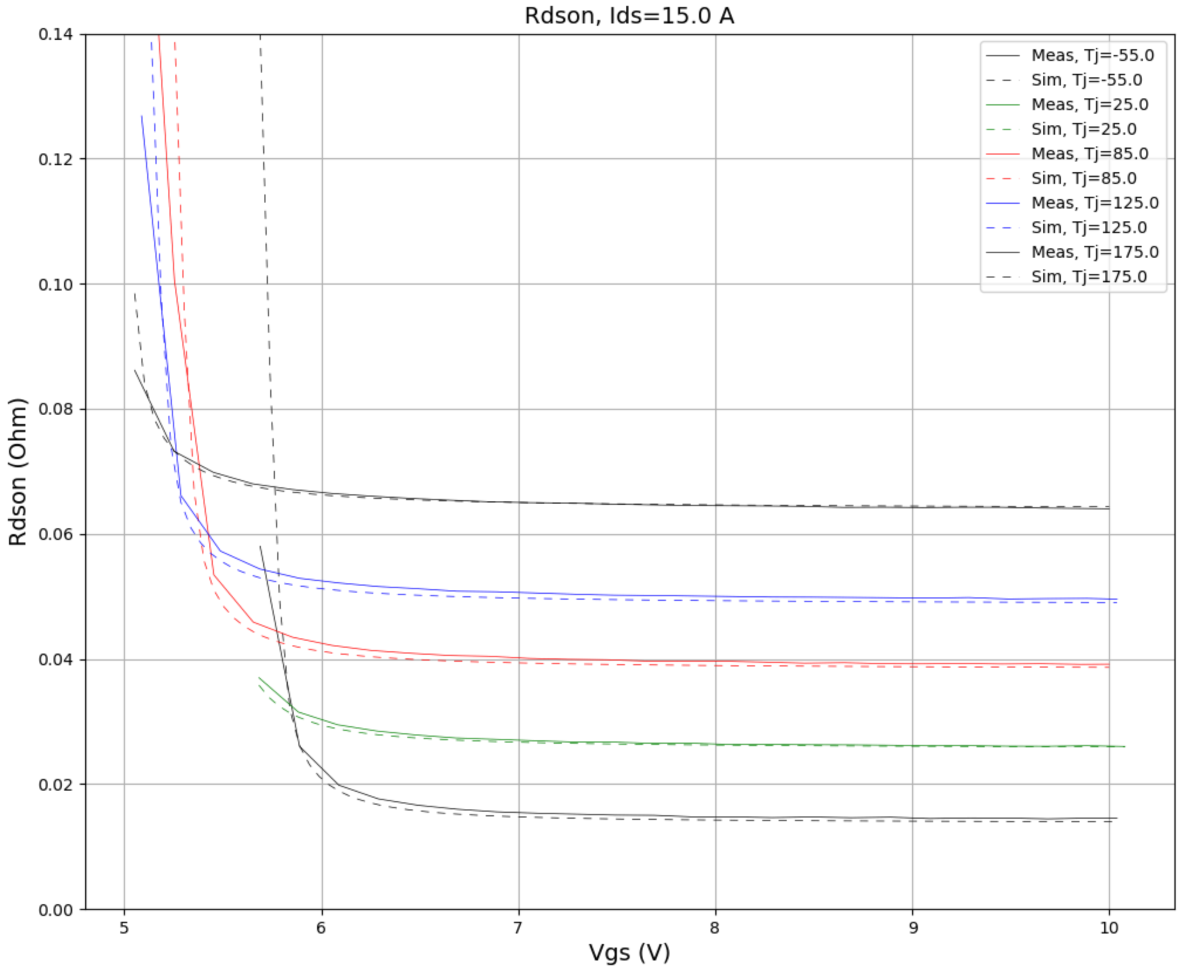


Fig. 5. Drain-source on-state resistance as a function of gate-source voltage at different temperatures (-55, 25, 85, 125, 175°C); solid = experiment, dashed = model

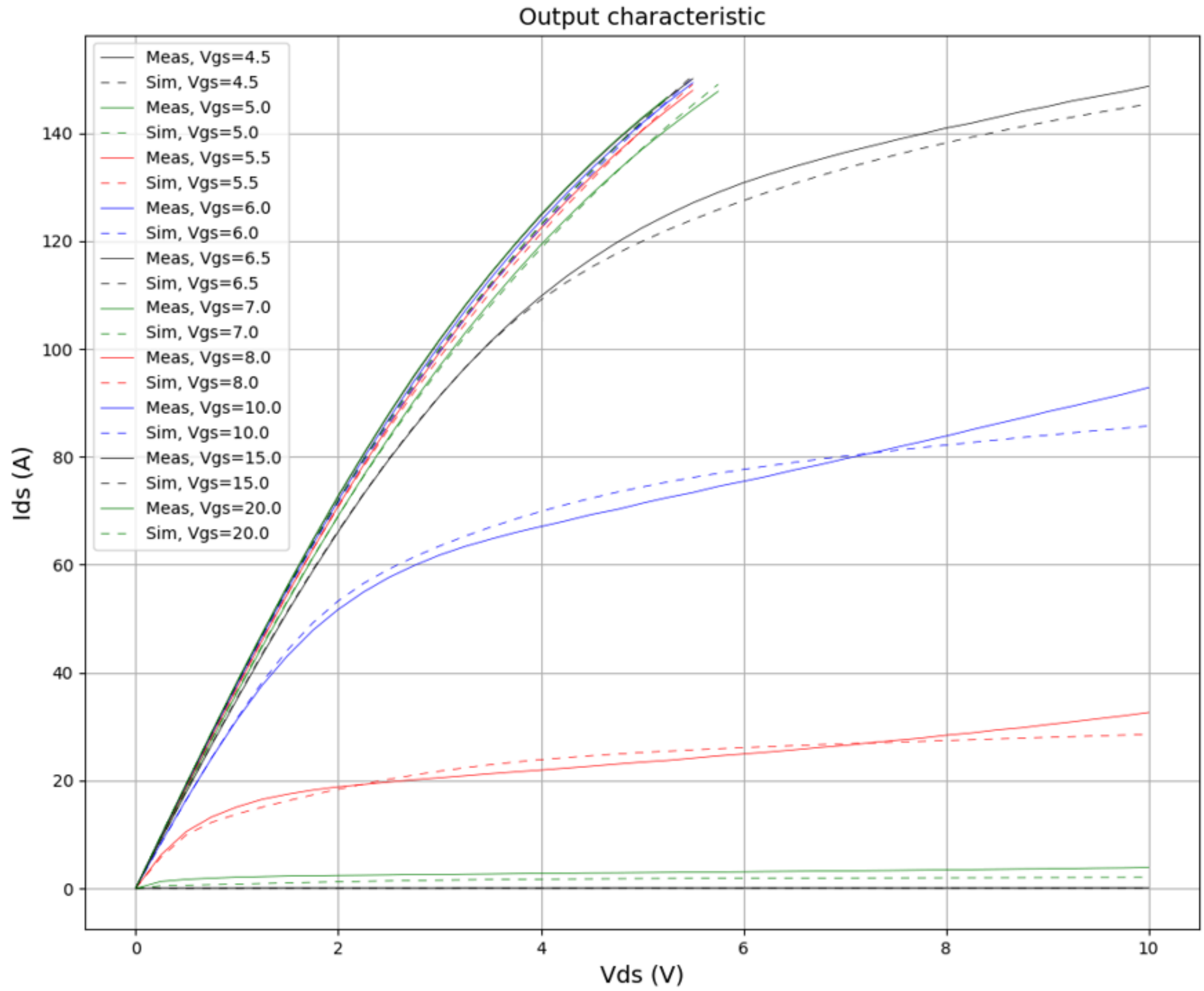


Fig. 6. Drain-source current as a function of gate-source voltage (pulsed); solid = experiment, dashed = model

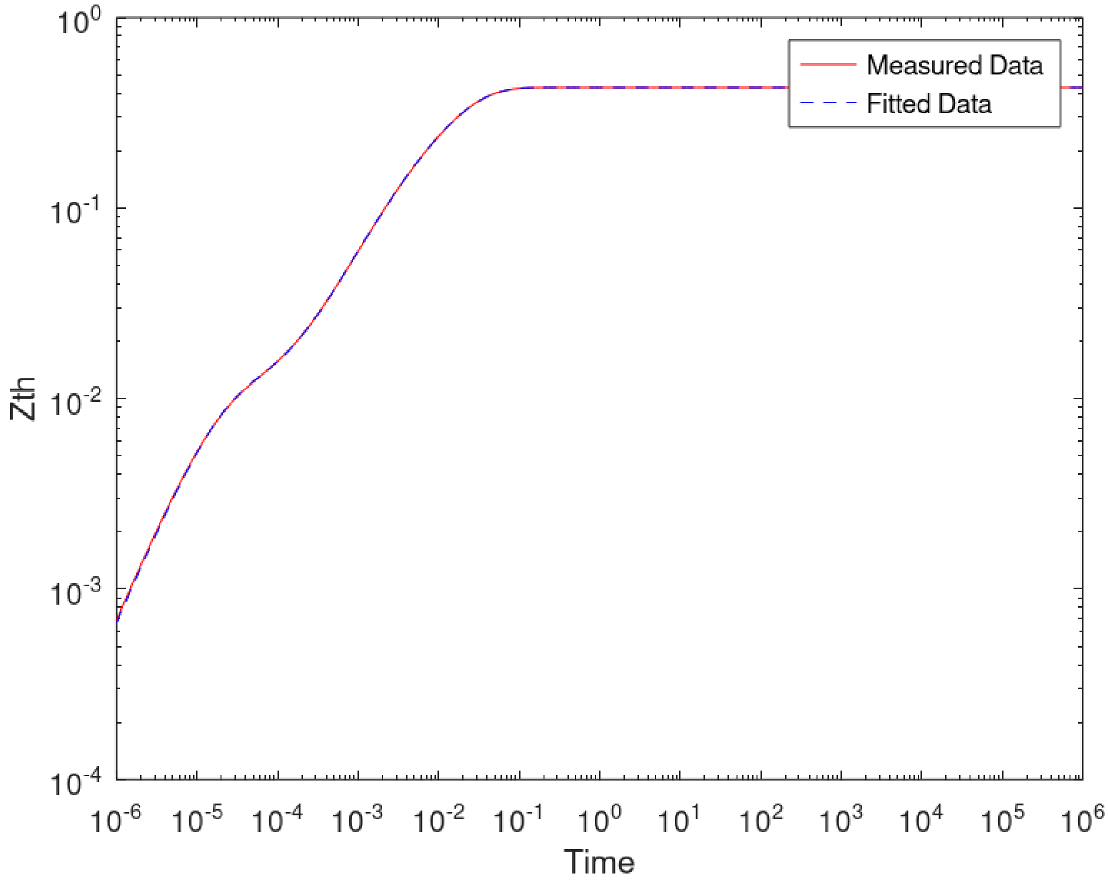
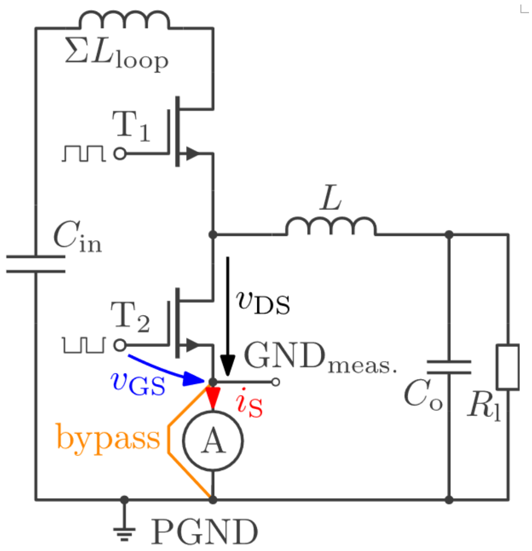


Fig. 7. Transient termal impedance as a function of time; solid = experiment, dashed = model

4.1. Application tests of the new advanced model

The measured and simulated power loss and efficiency curves are shown in Fig. 8 and Fig. 9, using the circuit and conditions given in Table 2.

Table 2. Buck-mode configuration

Parameter	Value / Range	Circuit diagram
output power	150 - 4800 W	 Simplified schematic of the test circuit used for the efficiency sweep See UM90028 for details.
test inductor	330 µH MPP toroid	
V_{DD}	400 V	
Vout	230 V	
V_{GS}	0 - 12 V	
R_G	15 Ω	
gate ferrite bead	30Ω @ 100 MHz (std. BLM type)	
T_{amb}	22 °C	
$f_{switching}$	100 kHz	
t_{dead}	100 ns	
Note: Simulation uses experimental temperature data of each cascode device and also takes into consideration the load inductor core losses in its final calculations. Core losses are calculated using the core manufacturer's method that takes into account nonlinear magnetization and current ripple oscillation frequency.		

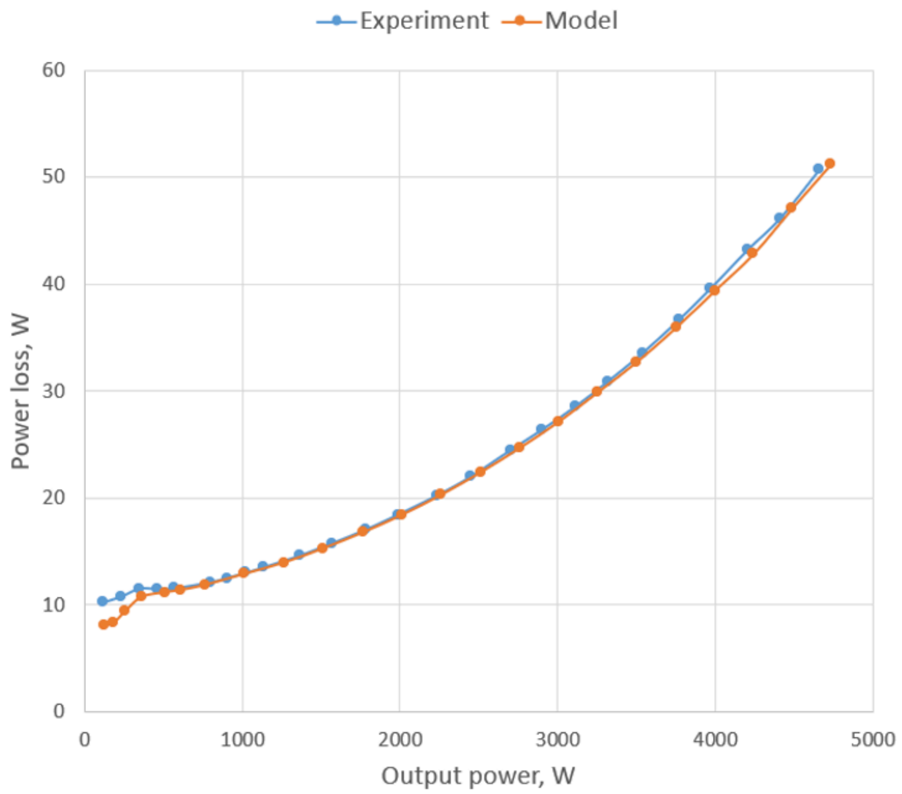


Fig. 8. Power loss as a function of output power; experiment vs model

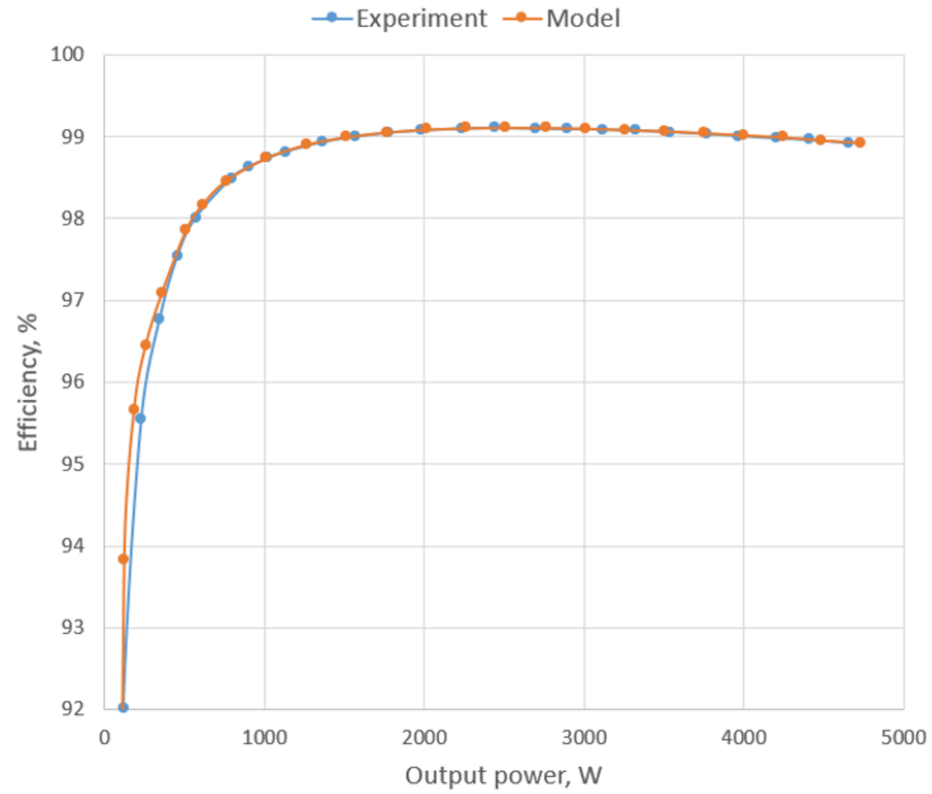


Fig. 9. Efficiency as a function of output power; experiment vs model

5. Adding a model to the simulator

5.1. LTspice

Models can be used in two ways in LTspice:

1. Place the symbol .asy and library .lib files into the same folder as your circuit, see Fig. 10:

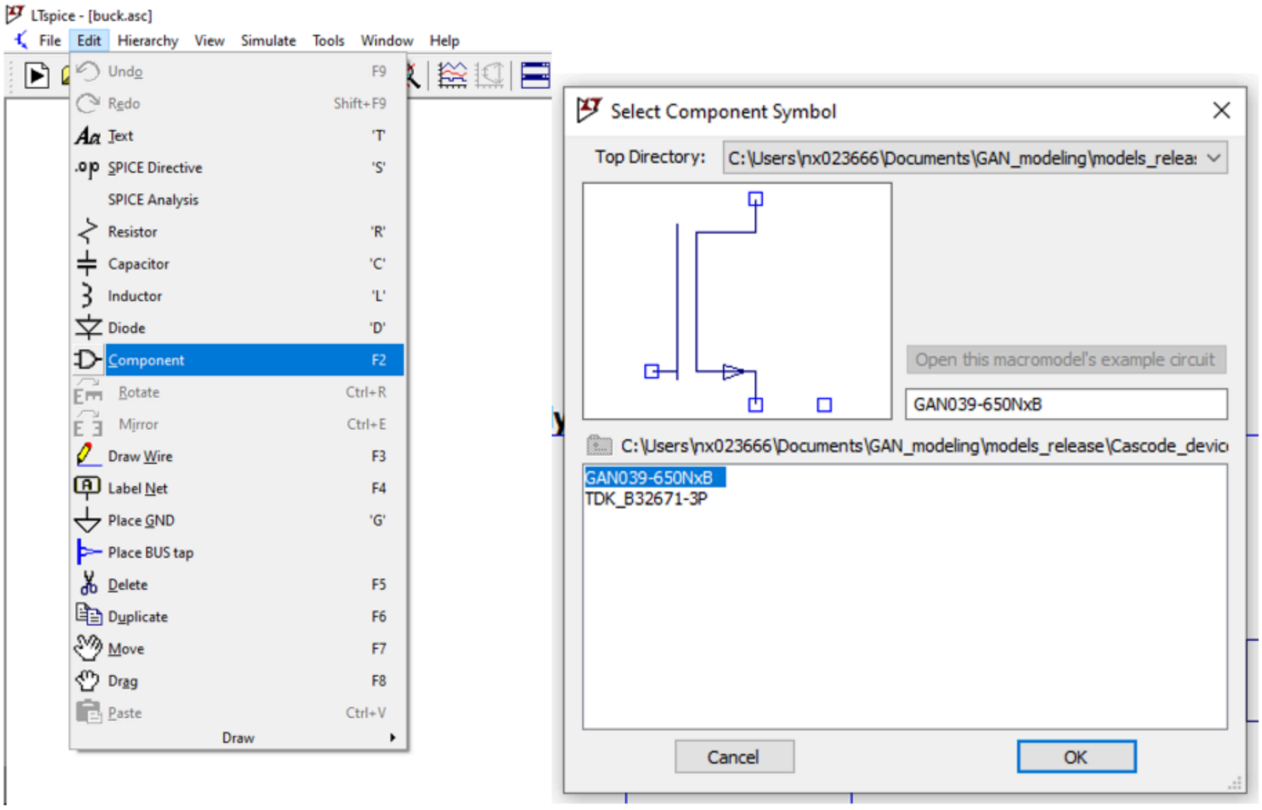


Fig. 10. Menus in LTspice, selecting the component using the symbol browser

Place symbol via menu Edit->Component->Select your folder in Top Directory menu-> Select symbol you want to add and place onto schematics.
Include the library file by using the .include statement with this syntax:

a. include <filename> where filename is the full path to library

b. if .lib files are placed into the schematic directory, including .lib extension, entering the full name is enough.

2. Place symbol .asy and library .lib files into any directory you want and then add that directory into search paths for symbols and libraries in menu Tools->Control Panel->Sym. and Lib. Search Paths, see Fig. 11:

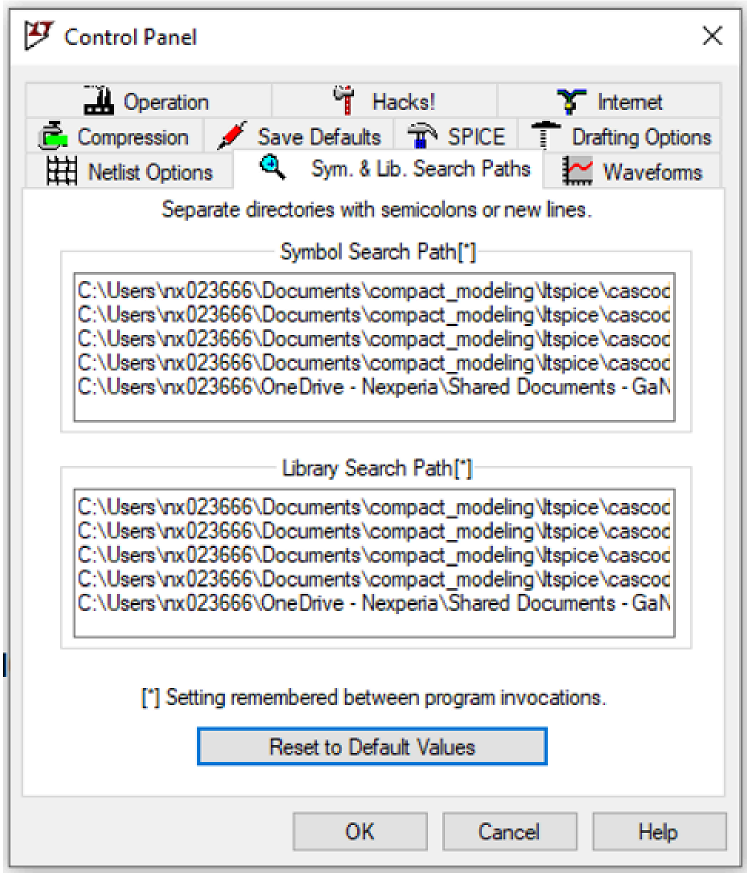


Fig. 11. Menus in LTspice, adding symbol and library search paths

Now you can access symbols in the same way as method **1)**, but now you don't need to explicitly put ".include" statement onto schematic because the library file is already in search path of LTspice and the name of that file is already in symbol attribute 'ModelFile', see Fig. 12.

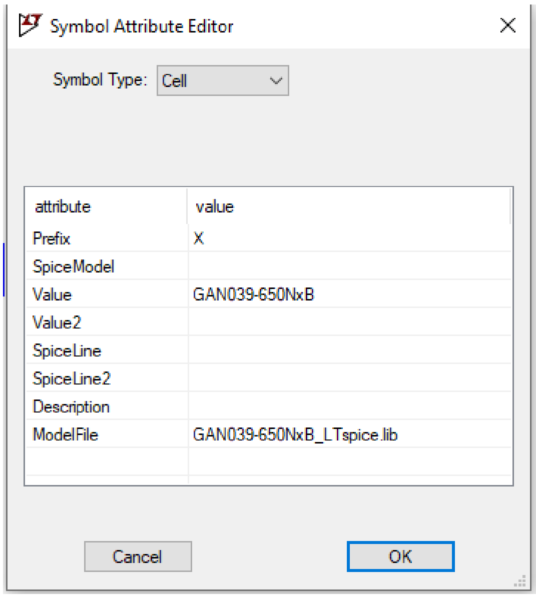


Fig. 12. Menus in LTspice, attribute properties of device symbol

5.2. SIMetrix

To use the models in SIMetrix follow these steps:

1. Import lib file: File -> Model Library -> Add/Remove Libraries -> Select Spice File folder -> Ok (or directly drag the model file into the command window of SIMetrix which is located in bottom-left corner by default), see Fig. 13:

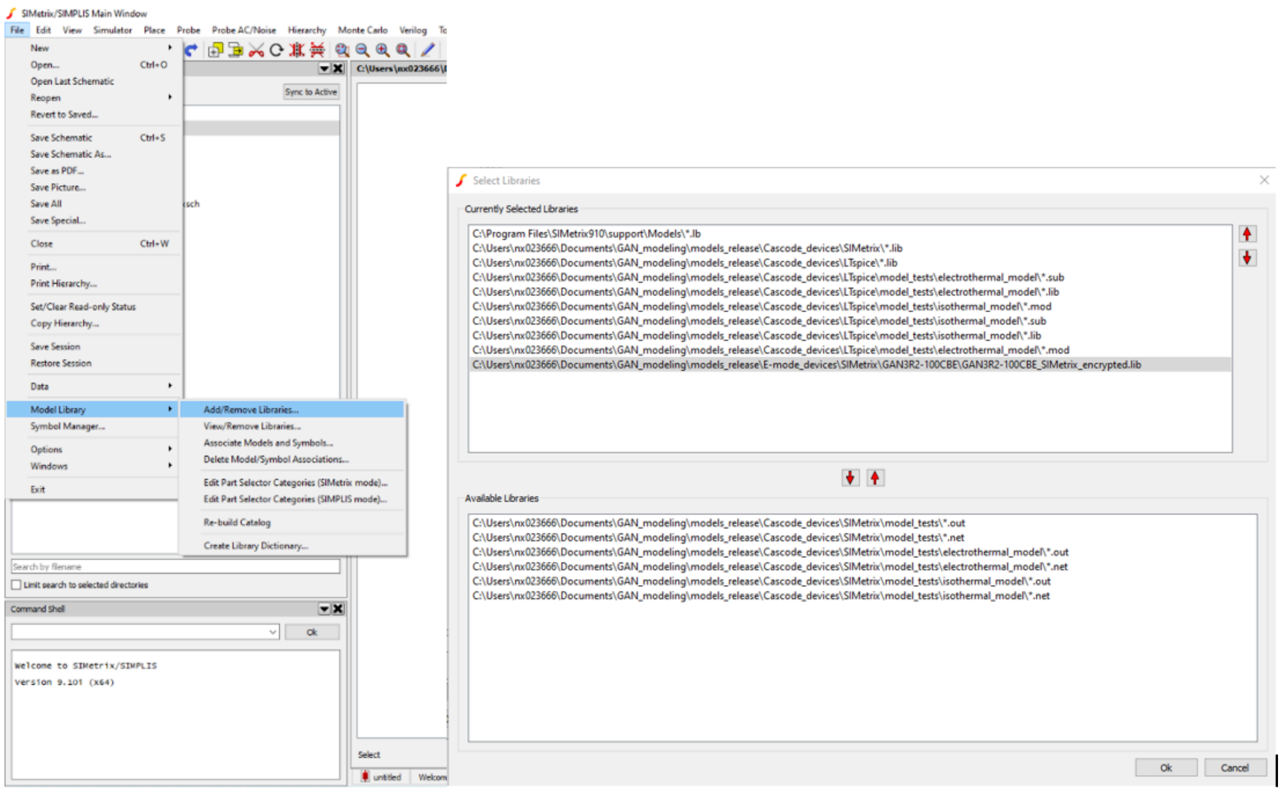


Fig. 13. Menus in SIMetrix, adding library into SIMetrix.

2. Import Symbol file: File -> Symbol Manager -> Add -> Select *.sxslb File -> Ok (or use file dragging as in previous step). see Fig. 14 and Fig. 15:

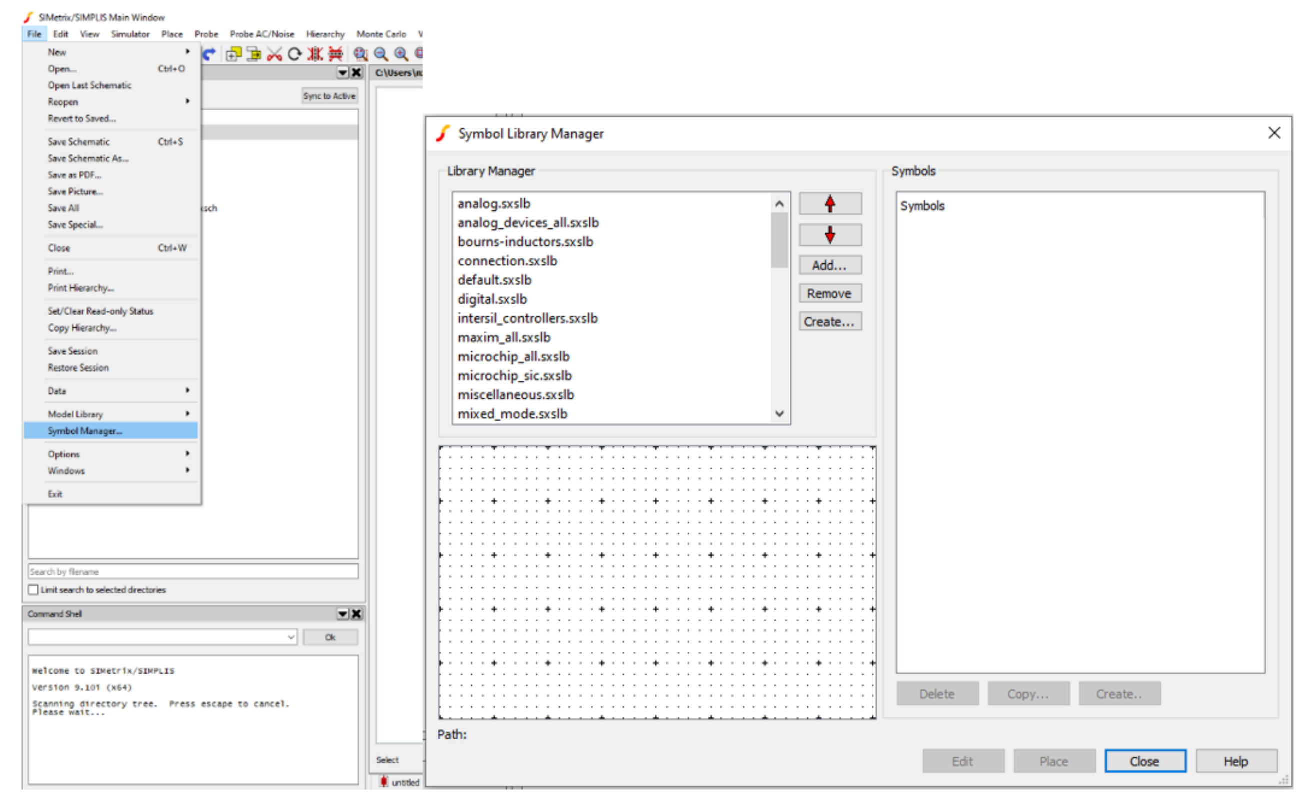


Fig. 14. Menus in SIMetrix, adding symbol into SIMetrix

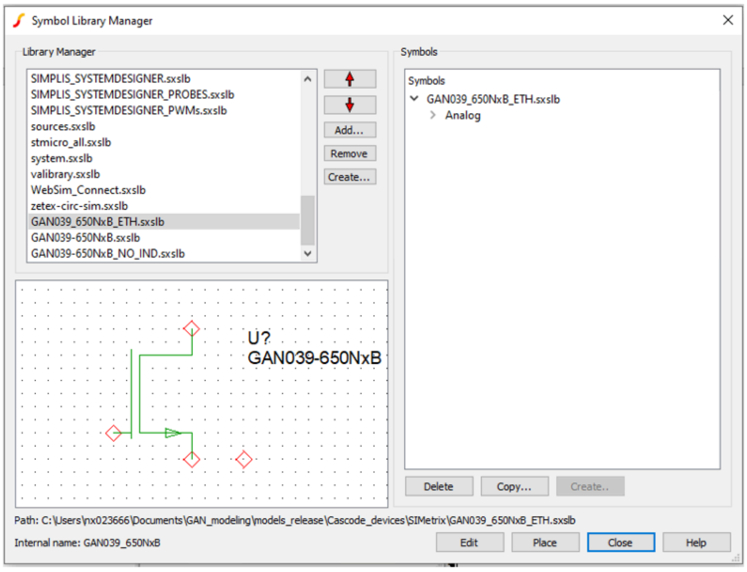


Fig. 15. Menus in SIMetrix, adding symbol into SIMetrix; 2

3. Associate Symbol file: File -> Model Library -> Associate models and symbols -> New Category -> Define Symbol -> Ok. See Fig. 16.

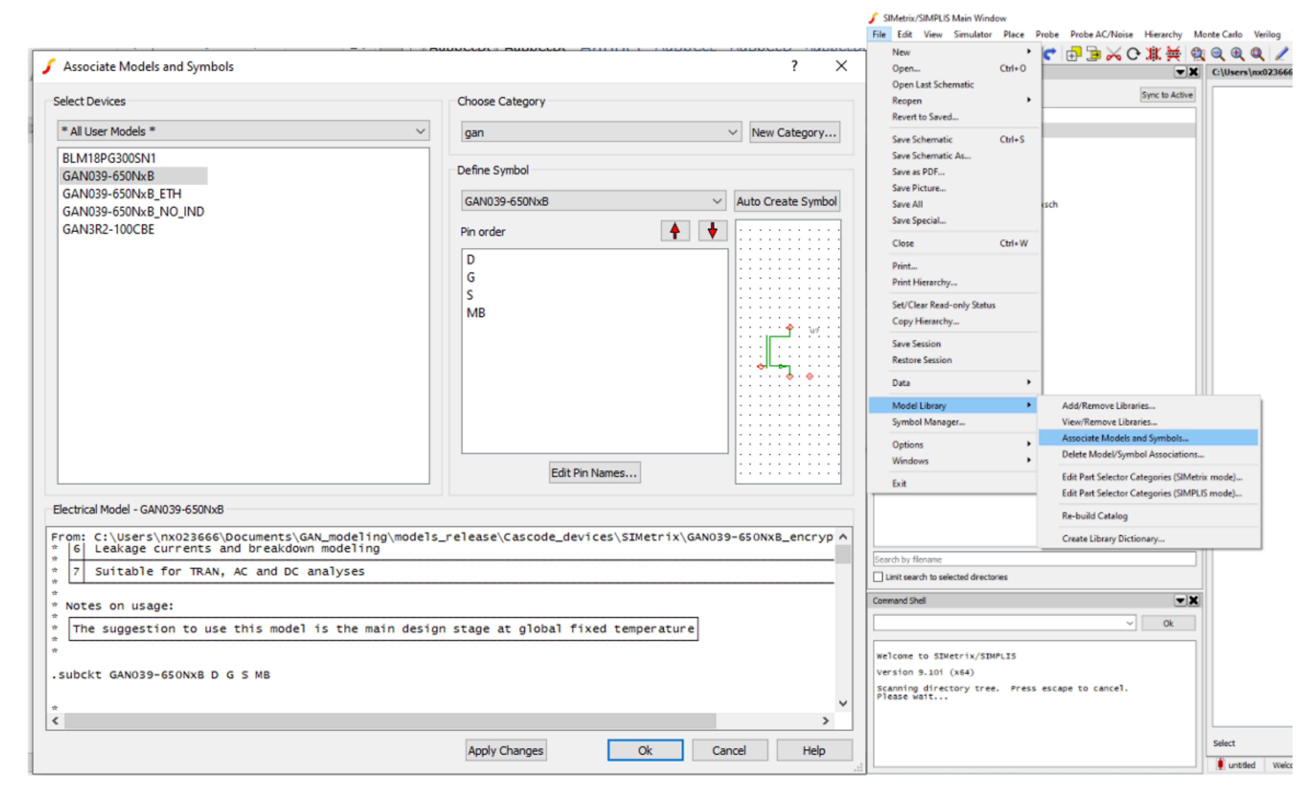


Fig. 16. Menus in SIMetrix, associate symbol with library file

6. Example of an application simulation using the base model

Fig. 17 shows a double-pulse test circuit using a low-side switch. Here we use the standard model without the dynamic thermal part. The simulation results are shown in Fig. 18.The default test conditions are:

- V_{DD} = 400 V
- L_{power_loop} = 2.35 nH
- I_F = 20 A
- R_G =15 Ω
- 30 Ohm BLM ferrite bead

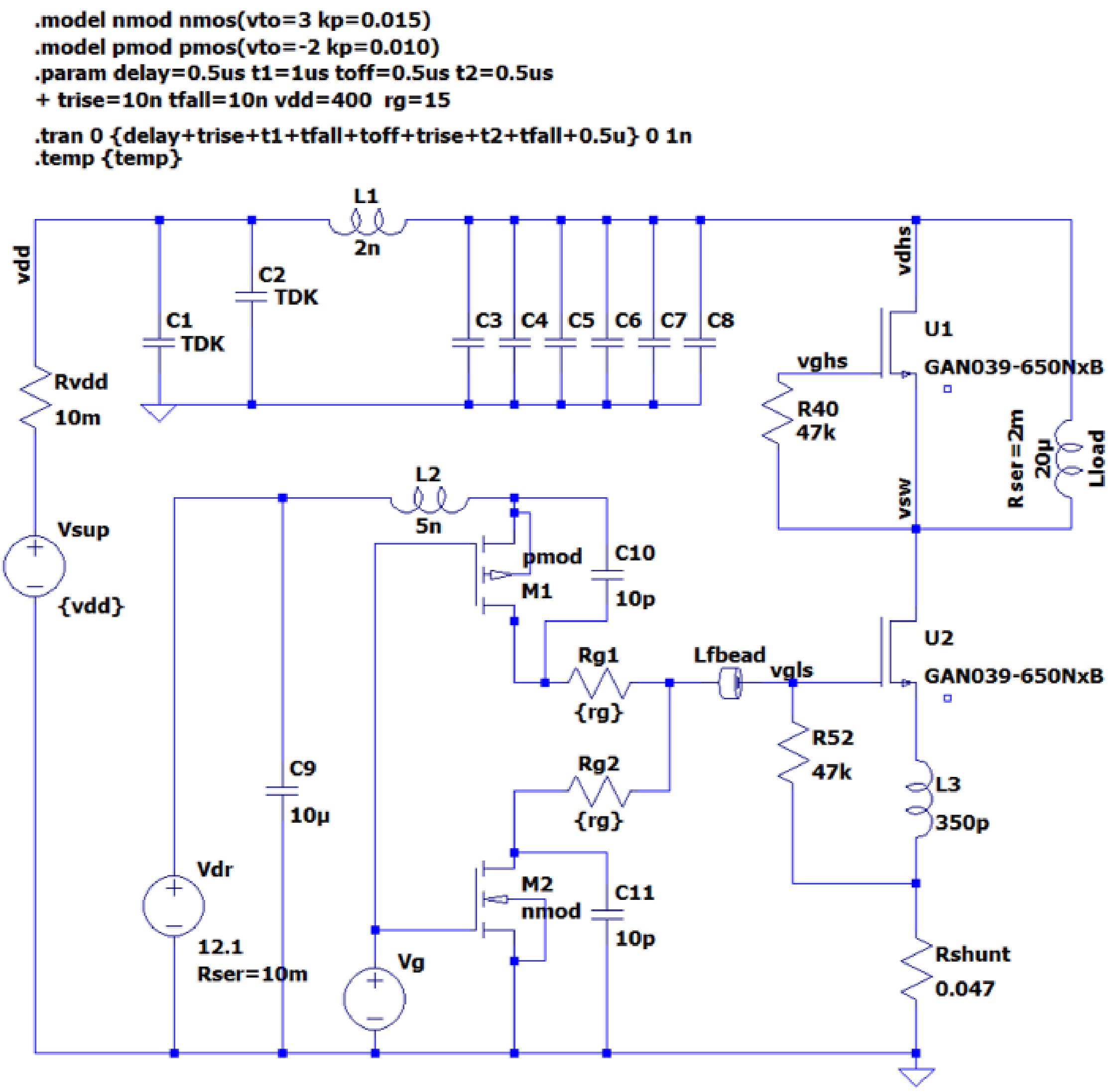


Fig. 17. Double pulse circuit in LTspice

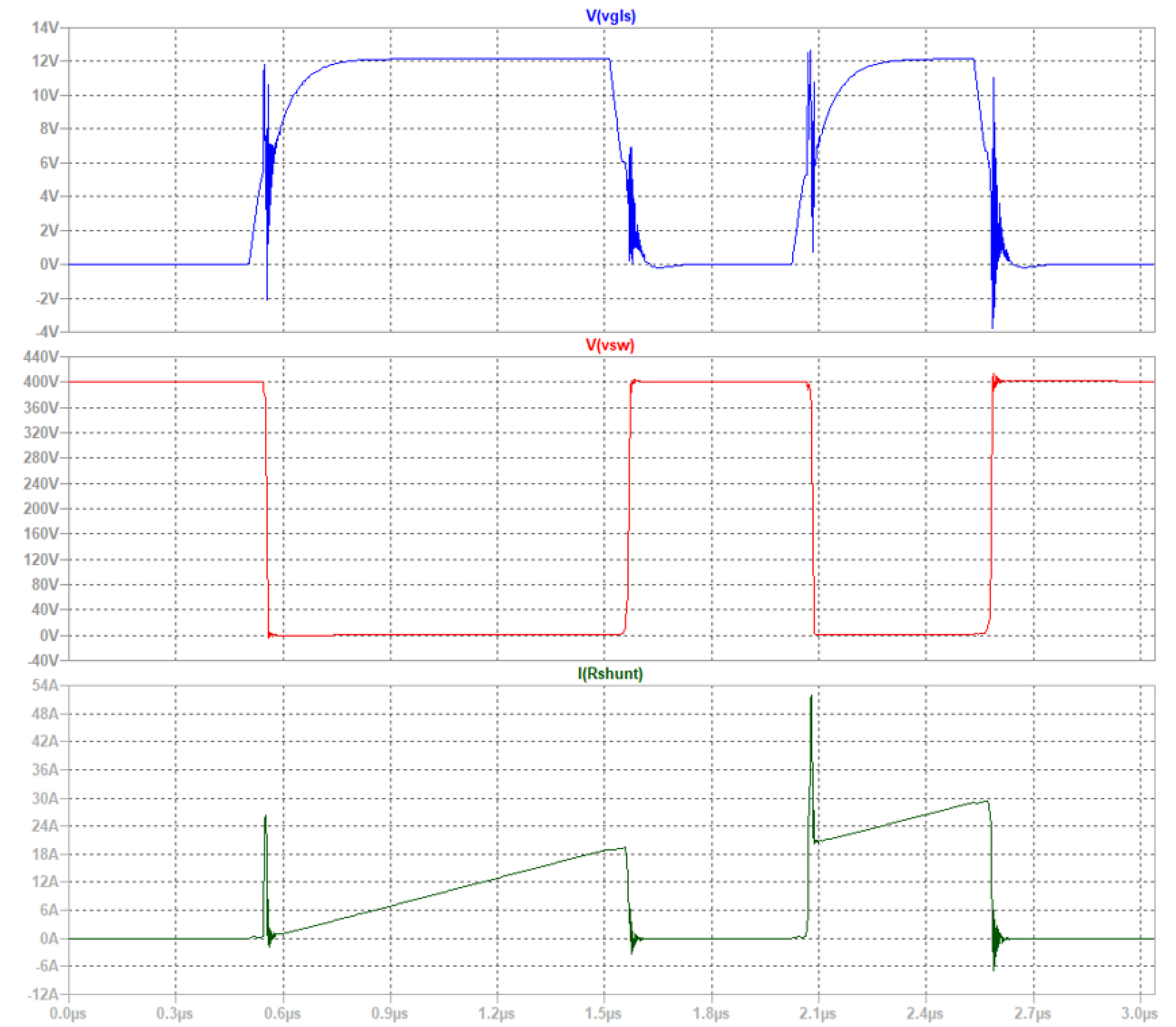


Fig. 18. Results of circuit simulation in LTspice

7. Using electrothermal models

The electrothermal models can be used in different ways, but you must remember that thermal and electrical circuits are different from each other and can't be mixed.

There are three correct ways to use electrothermal models:

1. The TMB node is connected to a constant voltage source with a value that represents constant ambient temperature. If you connect TMB directly to this source, we will assume that the heatsink is ideal and is kept at constant temperature, while the junction temperature of device could be different from ambient, see Fig. 19.

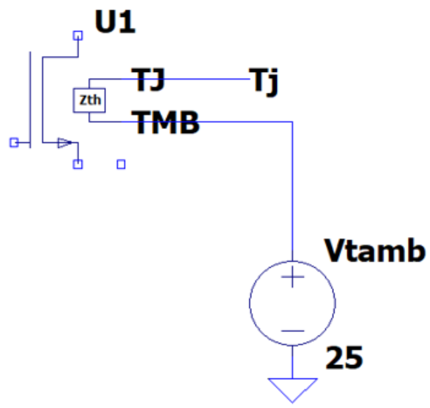


Fig. 19. Standard usage of electrothermal model in LTspice

2. Both TMB and TJ are connected to a constant voltage source. In this case we fix the temperature of the device at the value of the voltage source, see Fig. 20.

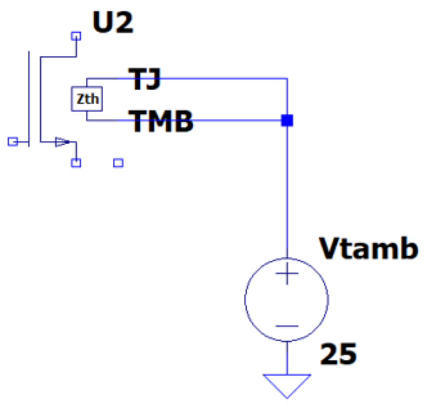


Fig. 20. Fixed temperature usage of electrothermal model in LTspice

3. You can connect TMB node to an external thermal circuit that represents a simple thermal model of the heatsink, see Fig. 21.

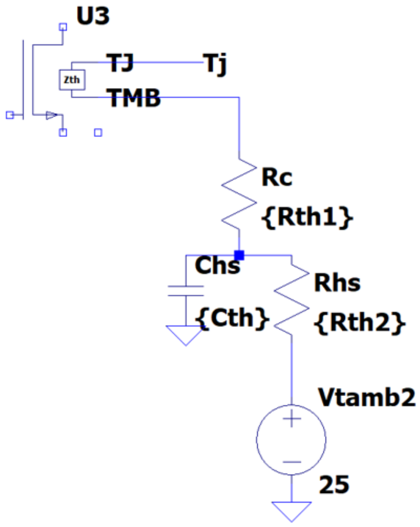


Fig. 21. External thermal part usage together with electrothermal model in LTspice

Also, if you have multiple devices on the board, you can set the individual temperature of each device, see Fig. 22:

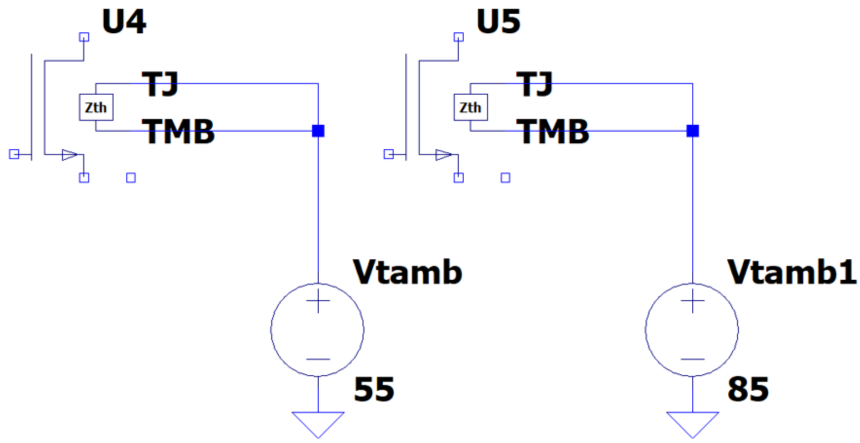


Fig. 22. Setting different fixed temperatures for each device in LTspice

This could be helpful if you have already measured steady-state temperatures of individual devices from the PCB and want to check the simulation result at the same conditions, without the necessity to wait until the transient simulation reaches steady-state condition.

Caution 1: Do not leave TMB floating in transient analysis and connect both TJ and TMB to fixed temperature source in DC/AC characterization analysis to avoid overheating of device.

Caution 2: use the startup option in transient modeling mode with caution for electrothermal models, because in this mode all sources start from zero, including the temperature source, as it takes a long time to reach the actual value due to large time constants in the thermal analysis.

7.1. Example of simulation with electrothermal model

In this example you can see the long simulation with external thermal resistor connected between TMB node and ambient temperature source that represents thermal resistance from mounting base point to ambient, see Fig. 23.

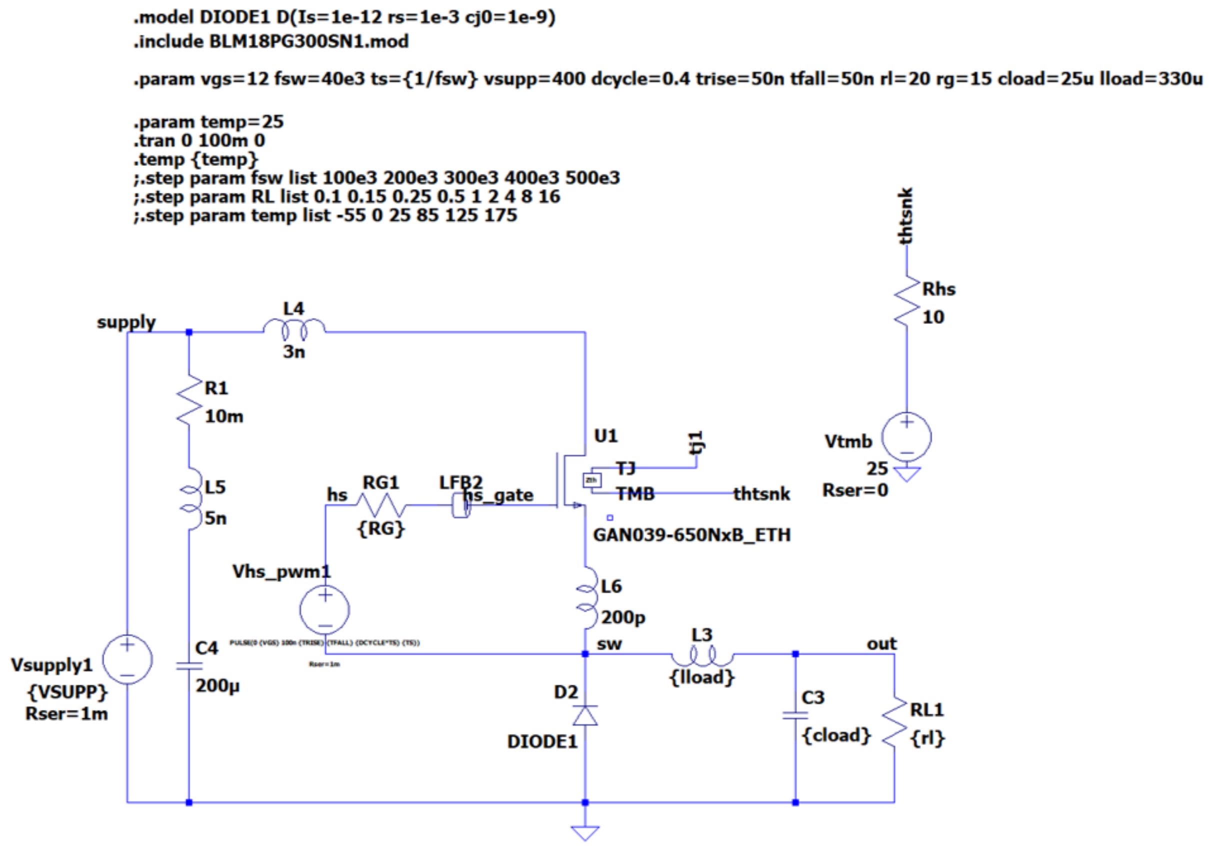


Fig. 23. Example of electrothermal model usage in LTspice

Since the time constant of thermal RC-network is in the order of seconds, we must simulate circuits for a long time to reach the steady-state condition. In this case the temperature has settled after about 2s of simulation time, see Fig. 24. Such simulations could be very time and resource consuming and should be done at the final stage of thermal design verification.

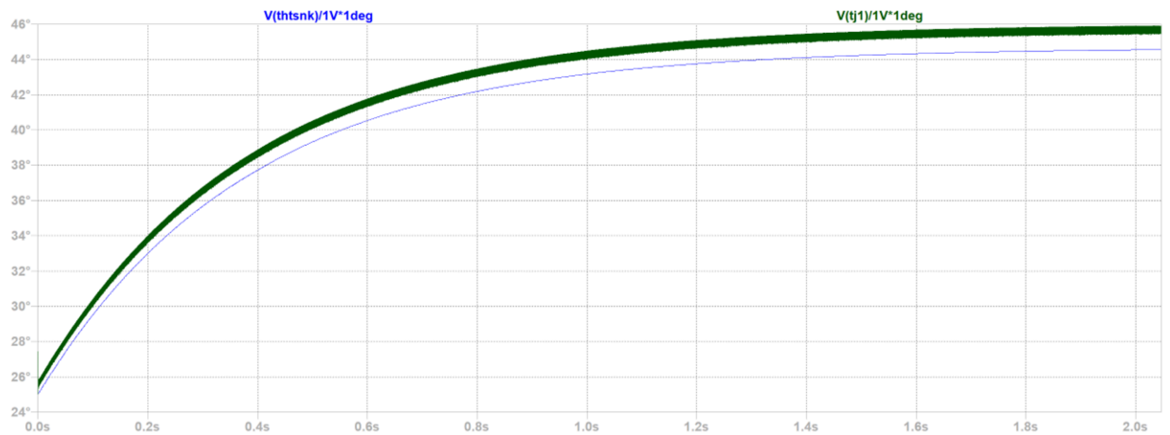


Fig. 24. Result of electrothermal simulation in LTspice

In order to significantly reduce the simulation time, you could loosen some tolerances to achieve faster convergence, for example in this case the following settings were used:

```
.options method=gear reltol=0.003 chgtol=1e-12 abstol=1e-9 trtol=6
 vntol=1e-4 gmin=1e-9 noopiter gminsteps=0
```

You should also keep in mind that these settings could introduce some error in the final settled temperature, but it is usually no more than 1°C. Also, in such long simulations the size of the output data could be very large, so it is helpful to make sure that compressing of transient waveforms is selected on in this tab, see Fig. 25:

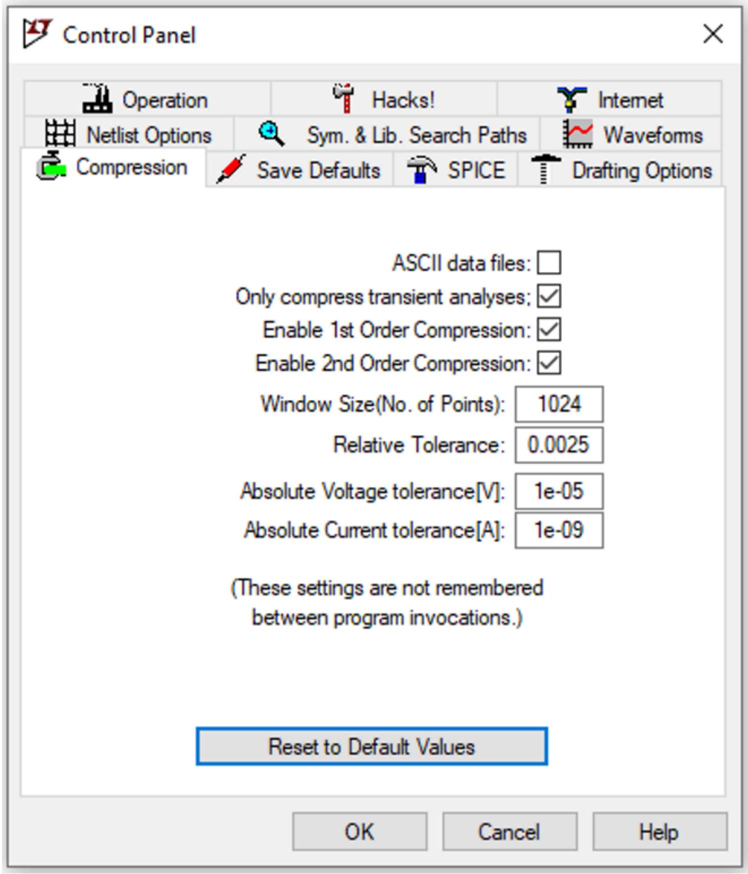


Fig. 25. Compression tab in LTspice

The other option to reduce saved data is set for saving only necessary node voltages, in this case we save only V(thtsnk) and V(tj) with this command:

```
.save v(tj1) v(thtsnk)
```

8. Convergence settings and how to solve convergence issues

8.1. General recommendations (applies to all simulators)

1. Use realistic models for all circuit components as much as possible:
    - Add realistic resistors in series with inductors, even for ones representing parasitic resistance of wires/PCB tracks.
    - Use proper models for decoupling/bypass/load capacitors with resistance (ESR) and inductance (ESL) in series:

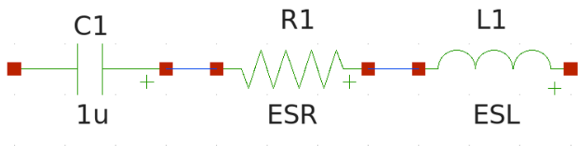


Fig. 26. Possible model of capacitor with parasitics included

    - Add resistance in series with power supply source/waveform generator source such that it models the finite resistance of real power supply.

2. It is good practice to make sure that at start of the transient analysis all gate-controlled devices are in the off state, it gives better performance during DC solution searching before start of transient.

3. In circuits with strong oscillations during switching the convergence could be improved by switch integration method from *trap* (most accurate single-step method) to *gear* (multistep method) that is more stable but provides additional damping. The Gear method is especially helpful when used with electrothermal models.

8.2. LTspice recommendations

1. If you observe non-physical voltage/current spikes or error "*Time step too small*", you could turn on alternate solver that has significantly higher accuracy (you could do this through Tools->Control Panel->SPICE).
2. Increase *abstol* and *vntol* to aid faster convergence during DC analysis, including initial conditions analysis prior to transient analysis, the upper limits are 1e-9 for *abstol* and 1e-4 for *vntol*.
3. Increase *ITL1* and *ITL4* up to 500, these settings represent the number of iterations per solution point for DC and transient analysis correspondingly.
4. Set *noopiter* flag and *gminsteps=0* to skip initial Newton iterations and Gmin stepping algorithm. It helps with the accuracy of the initial DC solution prior to transient simulation. A bad initial solution could lead to bad convergence or error in the transient analysis.
5. If you experience difficulties in finding the DC solution before transient analysis, you could turn on the *startup* option in TRAN settings. Caution: this does not work well with electrothermal

models because temperature voltage source also starts from zero, and overall circuit temperature will be changing very slowly due to large time constant of thermal system

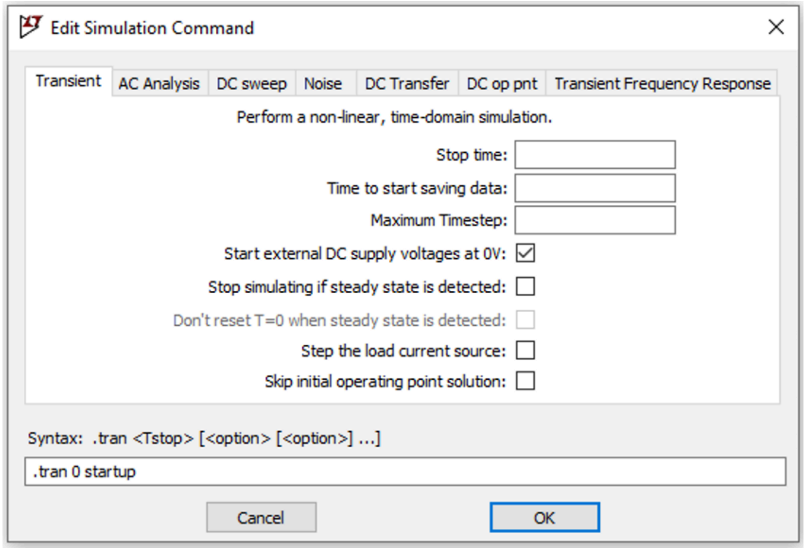


Fig. 27. Selected startup option in transient simulation option menu

6. Increase *chgtol* to get faster simulation during transient analysis, up to 1e-12
7. Add *cshunt* parameter in *.option* statement, value up to 1e-15, it could get rid of “*Time step too small*” error
8. As a last resort you can decrease the maximum time step to force the solver to do smaller time steps – it could improve stability but increases simulation time.

Template of .options setting:

```
.options abstol=1e-10 vntol=1e-4 ITL1=500 ITL4=100 noopiter
 gminsteps=0 chgtol=1e-12 method=trap chgtol=1e-15
```

8.3. SIMetrix recommendations

1. Change default iteration model to *Extended precision*.
2. Set shunt capacitance with value up to 1e-15, it could get rid of “*Time step too small*” error.
3. As a last resort you can decrease the maximum time step to force the solver to do smaller time steps – it could improve stability but increases simulation time.
4. If you experience difficulties in finding the DC solution before transient analysis you could turn on the startup option in the TRAN settings.
   Caution: it does not work well with electrothermal models because temperature/voltage source also starts from zero, and overall circuit temperature will be changing very slowly due to large time constant of thermal process.

9. Revision history

Table 3. Revision history

Revision number	Date	Description
1.0	2024-05-31	Initial version.

10. Legal information

Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. Nexperia does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, Nexperia does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. Nexperia takes no responsibility for the content in this document if provided by an information source outside of Nexperia.

In no event shall Nexperia be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, Nexperia's aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of Nexperia.

Right to make changes — Nexperia reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — Nexperia products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an Nexperia product can reasonably be expected to result in personal injury, death or severe property or environmental damage. Nexperia and its suppliers accept no liability for inclusion and/or use of Nexperia products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. Nexperia makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using Nexperia products, and Nexperia accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the Nexperia product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

Nexperia does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using Nexperia products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). Nexperia does not accept any liability in this respect.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Translations — A non-English (translated) version of a document is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Trademarks

Notice: All referenced brands, product names, service names and trademarks are the property of their respective owners.

List of Tables

List of Figures

Contents

Date of release: 31 May 2024