## User Guide PRD-09035

# Power Module RC Thermal Models User Guide

## User Guide **PRD-09035**

# Power Module RC Thermal Models User Guide

The thermal behavior of SiC power modules is a key contributor to their overall capabilities in a power electronic system. Predicting the junction temperature of SiC MOSFETs during operation is important for understanding the safe-operating-area (SOA), modeling device lifetime, and building system level models. This user guide gives a brief overview of practical implementations of the RC Cauer models provided for Wolfspeed power modules.

## Contents

# 1. Overview

This document discusses how to install and use Wolfspeed's RC thermal SPICE model library and associated data file. Wolfspeed provides 4th order Cauer network models for all Wolfpeed power module devices. These models aim to accurately describe the transient thermal impedance of the device, such that a relationship between power loss and junction temperature can be simulated. Thermal models are useful for predicting how short transient events (1 µs – 1 s) will affect die temperature; understanding such relationships is important for short-circuit, surge, or heavy-load conditions where momentary power spikes may cause the die to exceed its SOA, where using the steady-state thermal resistance is not appropriate.

The .zip folder containing the SPICE models and data file are available for download under 'SPICE RC Thermal Models' located on Wolfspeed's website. The data file defines the parameters for a 4th order Cauer network for all Wolfspeed power modules. See Section 4 for more detailed information.

# 2. Thermal Impedance Background

During power module operation, the conduction of current through the SiC MOSFETs results in power losses that increase the temperature of the semiconductor and surrounding packaging. Temperature has two important implications in design. First, if the temperature of the device exceeds the maximum rating listed in the datasheet, then the device may fail or have its lifetime significantly reduced. Second, changes in temperature impart thermo-mechanical stresses on the packaging that slowly degrade the interfaces.

It is helpful first to understand the general structure of a power module and how it physically relates to datasheet parameters. Consider the cross-sectional diagram of an XM3 power module in Figure 1. The SiC MOSFET is located on the top of the substrate, which is comprised of a ceramic material that is surrounded by a copper metallization on both sides. The ceramic material provides an electrical isolation between the MOSFET and the baseplate and has good thermal conductivity. For modules with baseplates, the bottom metallization of the ceramic is then attached to the baseplate of the module. The general term for the bottom module layer where heat is removed is the *case*. In this document, all diagrams and discussion will assume a module with baseplate. The baseplate can then be attached to a heatsink to remove heat energy from the system.

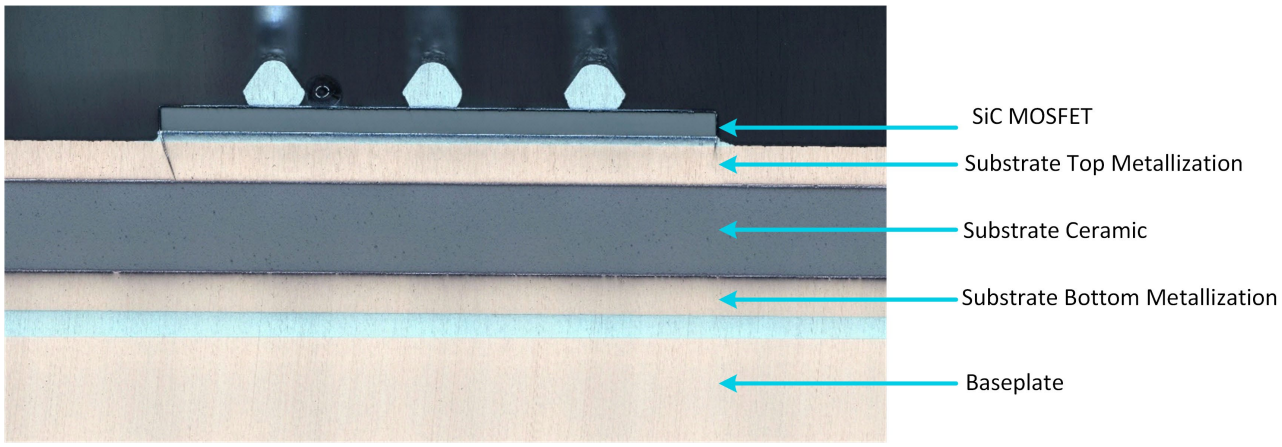



*Figure 1: Cross-sectional picture of a typical power module stack up*

Because the heat energy is generated at the *die*, and the heat energy is removed at the *baseplate* (or case), a critical thermal impedance is defined between them. Effectively, any heat energy generated by the die must propagate through the substrate and interface materials before being removed by the heatsink. This forms a temperature gradient across the module stack-up that limits how much power can be dissipated in the device before it exceeds the rated temperature. In Wolfspeed power module datasheets, this quantity is referred to as the FET thermal resistance from junction to case, or $R_{TH\ JC}$, provided in units of °C/W. So, for example, consider a module with an $R_{TH,JC}$ of 0.1 °C/W. If 100 W of power is dissipated at the device, then the MOSFET temperature will be 10 °C hotter than the baseplate. If the baseplate is 50 °C, then the MOSFET will be at 60 °C.

In most cases, the $R_{TH,JC}$ is sufficient to calculate the expected junction temperature of a device under average loss conditions to determine if the system is operating within SOA. However, $R_{TH,JC}$ is only valid when considering average losses on longer time scales (typically > 1 s) and will not provide accurate estimations when applied to instantaneous power. For example, consider a normal switching event, where the instantaneous power can easily exceed 50 kW. A module with an $R_{TH,JC}$ of 0.1 °C/W would calculate a temperature rise of 5000°C, but this clearly is incorrect. This is because thermal resistance ignores the fact that materials can store thermal energy. Consider the notional module diagrams in Figure 2. The module is attached to a heatsink which, for simplicity, is imagined as a constant temperature. Initially, the system is inactive, and thus the power module stack-up is at a uniform temperature. As the device begins conducting current, heat is generated at the die. As the thermal energy propagates through the stack-up, the substrate absorbs some of that energy as it increases in temperature. The amount of energy absorbed is defined by the thermal mass of the layer. Eventually, the substrate materials reach equilibrium, where the heat flux being received from the die is the same as the heat flux leaving into the surrounding substrate layers (and its temperature is now at steady-state). Here, the substrate, the baseplate, and, in fact, the SiC die itself act as a thermal reservoir, where each can immediately

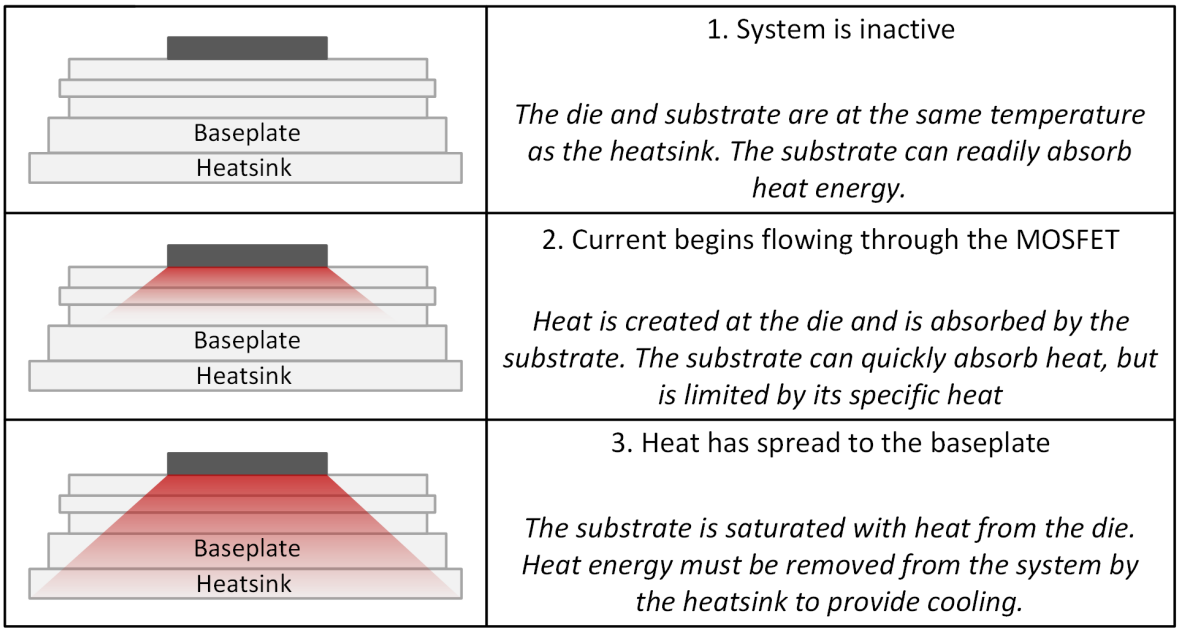



*Figure 2: Notional diagram of power module stack-up*

absorb some amount of thermal energy, limiting the peak temperature during short, high-power loss events such as those that occur during switching.

This example demonstrates the importance of the thermal *impedance,* or $Z_{TH,JC}$, when calculating the die temperature during transient events. In datasheets, thermal impedance plots are provided as a function of duty cycle and pulse time, such as in Figure 3. Here, the x-axis is the duration of the pulse, the y-axis is the thermal impedance, and each line represents a different duty cycle. In this document, duty cycle is ignored; only the *single pulse* line, which describes a simple square wave, is discussed.

The $Z_{TH,JC}$ plot has two distinct regions. Between 1 µs and 100 ms, the thermal impedance increases with pulse time. So, for example, if the die is subject to a single 1 kW power pulse with a width of 10 µs, then the temperature rise caused by that pulse would be 1 kW * 0.0015°C/W = 1.5 °C. If the pulse width were increased to 1 ms, then the temperature rise caused by that pulse would be 1 kW * 0.01°C/W = 10°C. In these examples, the pulses are short enough that the thermal mass of the substrate itself absorbs most of the thermal energy. As the pulse time increases, the substrate area closest to the die reaches an equilibrium temperature where additional heat flux is only transferred to deeper substrate layers, leading to higher impedance. Above 100 ms, the thermal impedance plateaus at $R_{TH,JC}$ because the thermal energy reaches the baseplate and is removed by the thermal management system. Thermal energy may still be spreading out laterally along the substrate layers, but this is insignificant in regard to removing heat from the die. A continuous 1 kW pulse for this same part would result in a temperature rise of 1 kW * 0.094°C/W = 94°C – significantly higher than would be observed for a transient pulse.

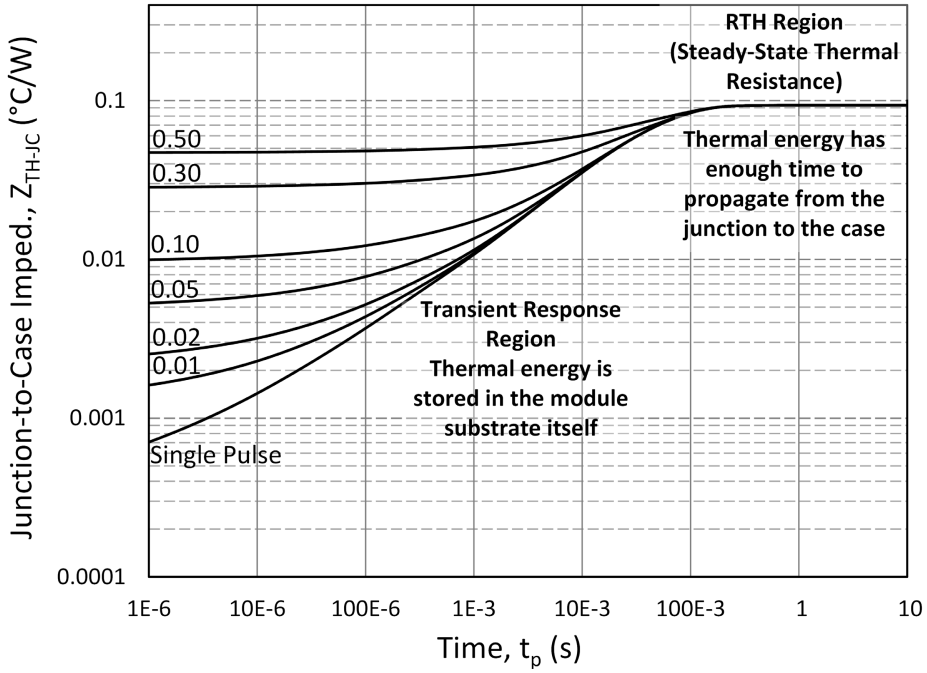



*Figure 3: Annotated thermal impedance plot from the CAB450M12XM3 datasheet*

# 3. Thermo-Electrical Circuit Models

While isolated calculations using thermal impedance curves are simple, the problem becomes much more complex when prior conditions are considered. For example, say that a 100 W, 10 µs pulse is immediately followed by a 200 W, 100 µs pulse. After the first pulse, the die and substrate will have heated to some temperature that will affect the response of the second pulse. While it is possible to solve this analytically, it is overly challenging and complex. Fortunately, the thermal behavior of mechanical structures can be easily solved by linearizing the complex thermal system using electrical circuit elements as an analogue. These circuits can be solved directly by hand or easily implemented into SPICE simulation software. These circuits only involve passive components and basic voltage/current sources and are thus very easy to implement into SPICE and solve quickly. Complex power loss inputs and conditions can be easily input into SPICE and solved with great detail.

## 3.1 General Thermal Circuit Theory

While thermal circuits are solved and behave the same as electrical circuits, the units differ to reflect the physical meaning of the model. **Voltage** relates to temperature in units of °C or K. **Current** relates to power in units of Watts. **Electrical Resistance** (Ω, or V/A) then relates to thermal resistance in units of degrees Celsius per Watt (°C/W). **Electrical Capacitance** (C/V) translates to thermal capacitance (J/°C or J/K). Using these translations, the basic building blocks of thermal circuits are summarized in Figure 4. Current sources are used to define power flow in a circuit; the most common use for these elements in a circuit is to define heat sources, such as the power losses in a MOSFET during operation. Voltage sources force the temperature of nodes in a system – this is often used to define ambient air or coolant fluid temperatures. Resistors are used to define paths where heat energy can flow, and the value of the resistance determines how quickly the heat can transfer. Finally, capacitors define the thermal mass of materials that can store and release heat energy.

A basic thermal circuit example is provided in Figure 5. On the left, a current source injects 50 W through four resistances, which are terminated by a temperature source set to 25°C. The temperature at each node (T1 – T5) can be determined as follows. First, T5 is 25°C, as defined by the voltage source. T4 is equal to the temperature at T5 plus the temperature drop across Rt4, which is equal to 25°C + 50 W * 2°C/W = 125°C. This can be repeated to determine T3, T2, and T1, as annotated in Figure 5.

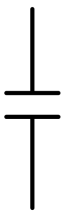
| A **current source** describes the generation or removal of heat energy in units of Watts (W). | A **voltage source** forces a temperature differences across two nodes (K or °C). | A **resistor** describes the generation or removal of heat energy in units of degrees per Watt (°C/W or K/W). | A **capacitor** describes the storage of heat energy in units of Joules per degree (J/°C or J/K). |
|---|---|---|---|
|  |  |  |  |

*Figure 4: Basic building blocks for thermal circuits*

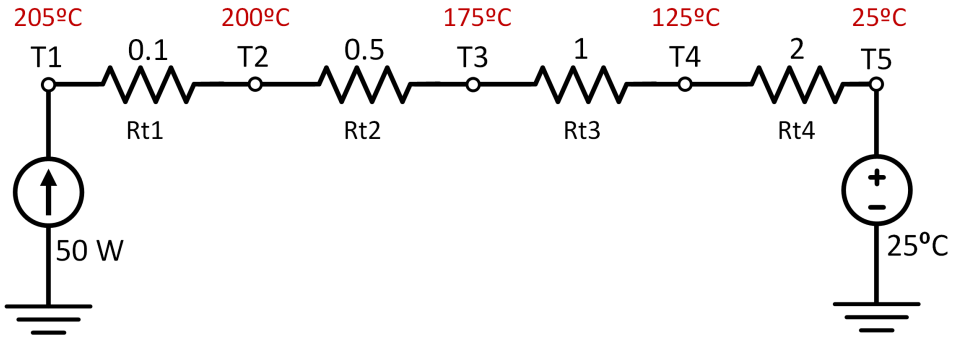



*Figure 5: Basic example thermal circuit*

## 3.2 Semiconductor RC Thermal Models

Two circuit models are commonly used to represent heat transfer through the multiple layers of semiconductor modules. Foster network models use a series of paralleled RC elements to model the thermal impedance of a structure. These systems can easily be fit to match a thermal impedance analytically, but the RC elements do not correlate to any physical elements of the structure. Often, Foster models are converted to Cauer thermal models (which are challenging to fit to a transient thermal impedance directly), the general structure of which is shown in Figure 6. The Cauer model itself is defined by the repeating RC elements; the heat source and heatsink elements are added to show how they are typically implemented in circuits. The resistance and capacitance terms can be repeated indefinitely to improve the accuracy of the model (at the cost of complexity).

Cauer models have two main advantages. First, each RC pair can physically represent a layer or area of the power module's stack-up. This is useful because the temperature of each layer can be measured on the circuit (e.x. T1). Junction temperature, Tj, represents the temperature of the semiconductor, and case temperature, Tc, represents the temperature at the bottom of the module. Second, because the capacitors are tied to a common node, the temperatures in the stack-up can be measured relative to a common node (usually ground). For these reasons, Cauer networks are more commonly used when simulating semiconductor devices.

The current and voltage source elements in Figure 6 show how these RC thermal models are typically used in practical circuits. The current source on the left represents the power losses of the semiconductor, and the voltage source on the right represents the temperature of the heatsink/coldplate. The resistor in series with the voltage source represents the thermal resistance between the case of the module and the attached heatsink/coldplate.

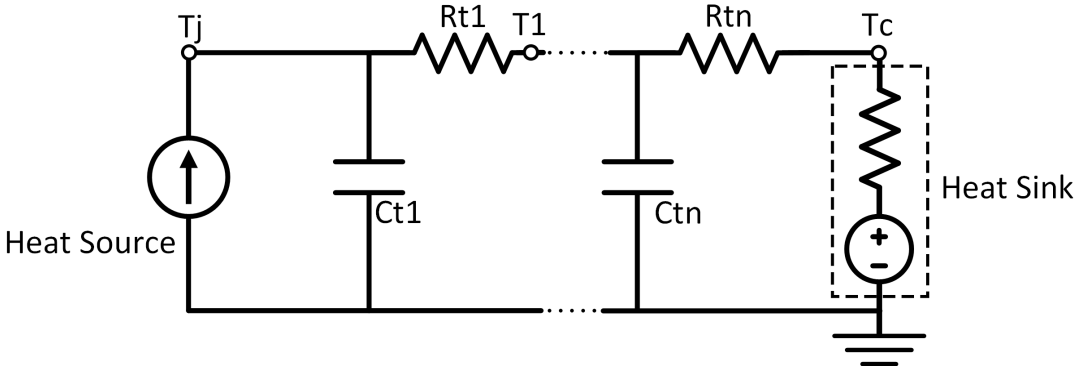



*Figure 6: General Cauer thermal model circuit with additional heat source and heatsink*

# 4. Wolfspeed SPICE RC Thermal Models

The circuit for the Wolfspeed SPICE RC thermal models is provided in Figure 7. The model uses a 4th order Cauer network; it should be noted that this is not an exact physical representation of the module stack-up, but simply that a 4th order circuit provides good matching with the measured thermal impedance. The model has three pins: a Tj pin for measuring the die temperature and injecting power losses, a Tc pin for measuring the case temperature and applying a heatsink, and a Ta pin to define a common node (usually ground).

There are two methods to implement the Cauer model from Figure 7 in a practical circuit. The first method is to

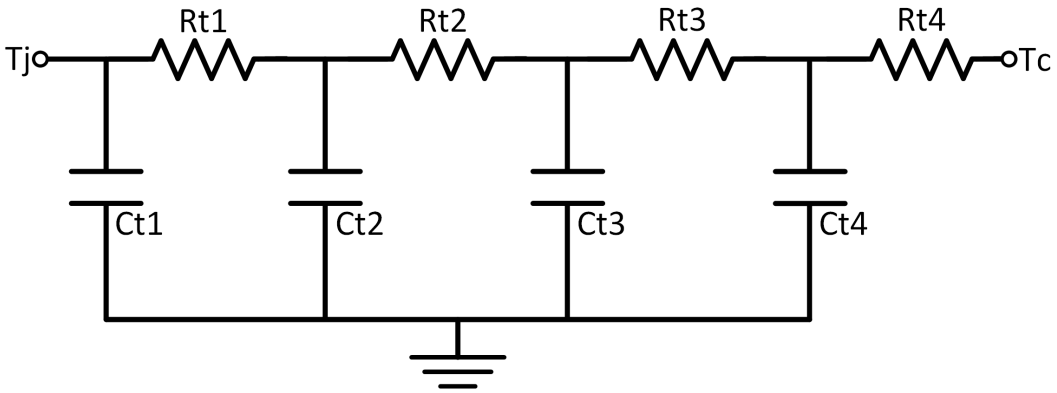



*Figure 7: Wolfspeed Cauer thermal model circuit*

manually insert the passive elements into a SPICE simulation circuit. The values Rt1 – Rt4 and Ct1 – Ct4 are provided in a data file on Wolfspeed's website for all power module products. See Section 4.1 for download instructions.

The second method is to use the provided Wolfspeed Cauer model library developed for use in LTspice. A screenshot of the LTspice model symbol is provided in Figure 8. The LTspice model has several advantages. First, several example simulations are provided to demonstrate how to use the model appropriately. Second, a symbol is provided to easily add the model to a simulation circuit. This symbol has pre-defined attributes that automatically include the associated CauerModels.lib file in the simulation. To access the attributes window,

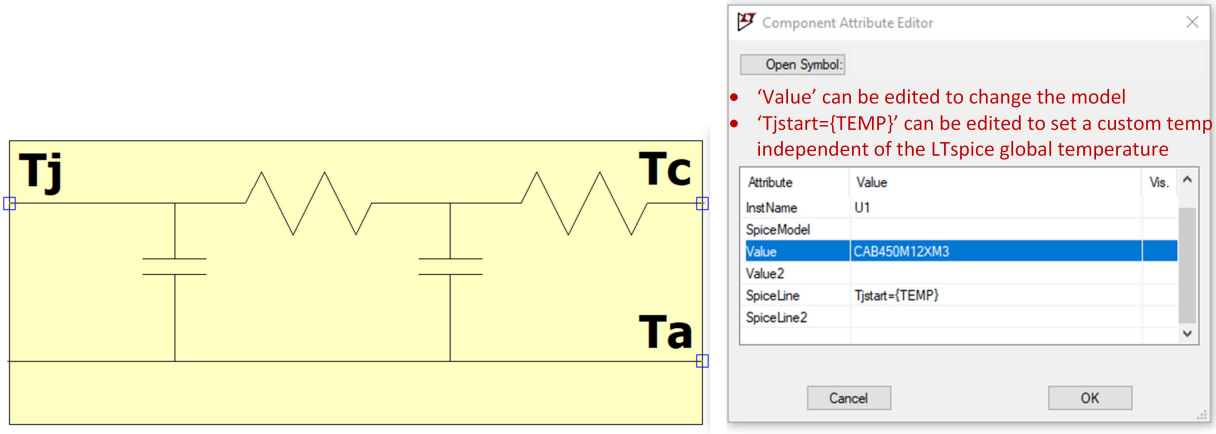



*Figure 8: LTspice Cauer thermal model symbol and attributes window*

alt + right-click the model symbol. The 'Value' field defines the model being used from the .lib file. The part number in this field can be changed to simulate a different device. For modules with Schottky diodes, the Schottky diode can be simulated by appending the part number with "_Schottky" (for example, "WAS175M12BM3_Schottky"). The 'SpiceLine' field can be edited to change any of the model parameters. By default, the *Tjstart* parameter is defined to {TEMP}. *Tjstart* defines the starting temperature of the model, and {TEMP} represents the LTspice global temperature value. It is recommended to change the {TEMP} parameter in LTspice to change the starting temperature (try ".TEMP 50" to change the temperature to 50°C, for example). However, if it is desired to have multiple Cauer models in the same simulation have different starting temperatures, then Tjstart can be redefined in the SpiceLine to another value or parameter.

Each set of parameters represents the thermal model for a single switch position in a module. The thermal model is for a virtual junction temperature, which describes the average temperature of the entire switch position. For devices with Schottky diodes, a separate thermal network would be used for losses within the MOSFETs and the Schottkys. Note that cross heating between the MOSFETs and Schottky diodes is not modeled.

## 4.1 Download Instructions

To begin using Wolfspeed's thermal SPICE models, first download the .zip folder containing the SPICE models under 'SPICE RC Thermal Models' located on Wolfspeed's website. Extract the folder to view the contents. A data file, "Cauer_Network_Parameters.xlsx", contains all of the parameters for a 4th order Cauer network circuit for every Wolfspeed power module. A snippet of this data file is shown in Figure 9. The 'Module' column describes the name of the part. If the module name is appended with (Schottky), then it indicates that this is a separate thermal model for any anti-parallel Schottky diodes within the module.

| Module | R1 (K/W) | R2 (K/W) | R3 (K/W) | R4 (K/W) | C1 (J/K) | C2 (J/K) | C3 (J/K) | C4 (J/K) |
|---|---|---|---|---|---|---|---|---|
| CAR600M12HN6 | 0.00059 | 0.00306 | 0.02038 | 0.03897 | 0.00230 | 0.05165 | 0.31945 | 1.68052 |
| CAR600M17HN6 | 0.00049 | 0.00220 | 0.01406 | 0.03125 | 0.00235 | 0.05388 | 0.32763 | 1.48394 |
| CAS110M12BM2 | 0.00270 | 0.01256 | 0.05705 | 0.10069 | 0.00067 | 0.00605 | 0.02837 | 0.18876 |
| CAS110M12BM3 (Schottky) | 0.00242 | 0.01172 | 0.07643 | 0.12243 | 0.00072 | 0.00799 | 0.03936 | 0.18349 |

*Figure 9: Snippet of Cauer network parameters file*

The "LTspice Library" folder contains the LTspice library file, symbol, example circuits, and .pwl files necessary to get started with the SPICE Cauer model library.

- The "Cauer_Network.asy" file is the provided symbol file for the Cauer model
- The "CauerModels.lib" file contains all of the .SUBCKT models for each product. The list of parts available in the library are provided in the header of the file.
- The ".asc" files are example circuits that can be opened and simulated to get started with the RC thermal models
- The ".pwl" files are sample data used by the example simulation circuits
- The ".plt" files define plotting information for each example circuit

**In order for the models to work properly, the .asc circuits need to be in the same directory as the .asy and .lib files. If the directories are different, the configuration will need to be adjusted so LTspice knows where to find the files.**

## 4.2 LTspice Example Circuits

The following two subsections will briefly overview the provided example circuits.

### 4.2.1 Zth_Extractor.asc

A screenshot of the "Zth_Extractor.asc" simulation circuit is provided in Figure 10. Different aspects of the circuit are annotated to explain their purpose. This circuit is intended to extract the single pulse thermal impedance characteristics of the specified module. This is useful for extracting tabulated $Z_{TH}$ characteristics without needing to digitize datasheet plots (there will be some difference to the datasheet values, but the differences will be minimal).

To extract $Z_{TH}$, a current source set to 1 A is injected into the Tj port. This represents 1 W of losses at the die. Then, because thermal impedance is defined by °C/W, the measurement at Tj yields the $Z_{TH}$ characteristics. For example, if Tj increases by 1°C after 10 µs, then the thermal impedance would be 1°C/1W = 1 °C/W at a pulse width of 10 µs. For this simulation, the Tc and Ta pins are set to zero (note the .TEMP definition in the bottom-right that defines the Vcoolant voltage source). The first 1 µs of the simulation is ignored, as that is the limit of the measured data provided in the module datasheets.

Figure 11 shows a comparison between the simulated and datasheet $Z_{TH}$ characteristics of a CAB450M12XM3 power module. The predicted impedance of the model matches the datasheet closely, but there are small errors, particularly at low-time pulses. It should be noted that these models may be inaccurate at pulse times below 1 µs. All provided Cauer models will match the datasheet $R_{TH}$ exactly.

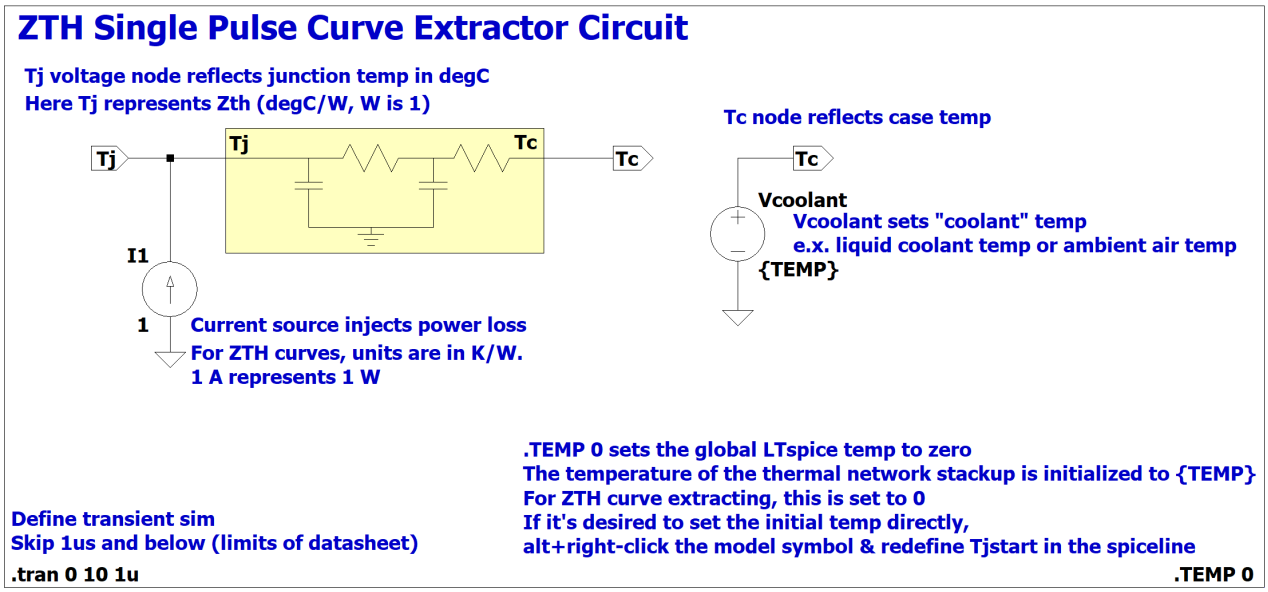



*Figure 10: $Z_{TH}$ single pulse curve extractor circuit*

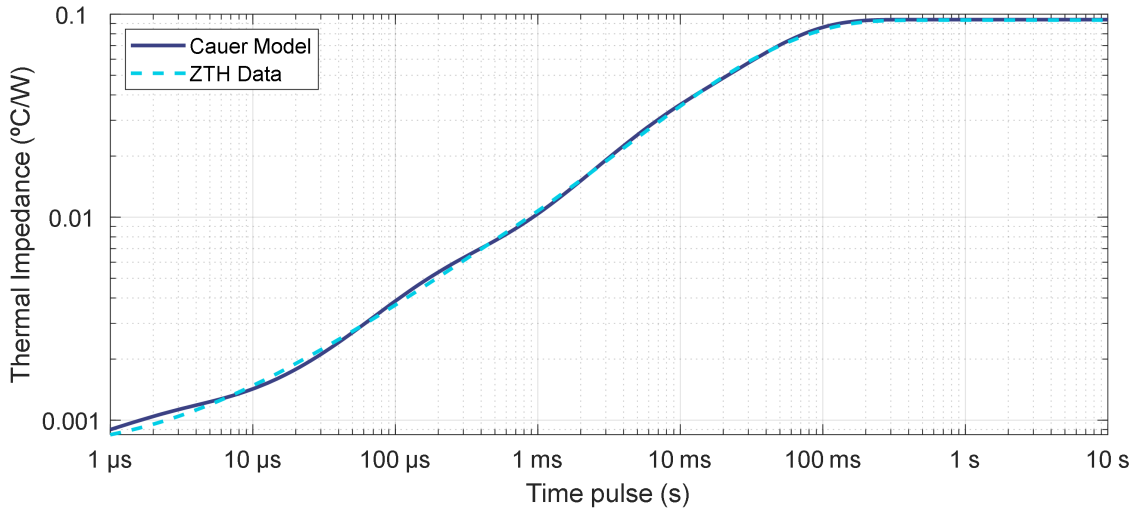



*Figure 11: Comparison of CAB450M12XM3 thermal model impedance to measured data*

### 4.2.2 Inverter_Example.asc & Surge_Example.asc

The "Inverter_Example.asc" and "Surge_Example.asc" files demonstrate one method for using the Cauer model library for practical applications. The example circuits describe an example loss profile for a 3-phase inverter and for a current surge event, respectively. A screenshot of the inverter simulation is shown in Figure 12. The circuit is similar to the $Z_{TH}$ extractor circuit, but with a few differences. First, the current source is defined to load data from a provided .pwl file that contains the loss profile during an inverter's operation. Second, the .TEMP parameter is increased to 50°C to represent a heatsink with a 50°C coolant, and a 0.1 Ω $R_{CA}$ parameter (resistance from case to ambient) was added. Figure 13 (a) shows the simulation results for the surge example circuit, and Figure 13 (b) shows the simulation results for the inverter example circuit. Three values are provided. The red waveform is the power loss loaded from the .pwl file, the green waveform is the junction

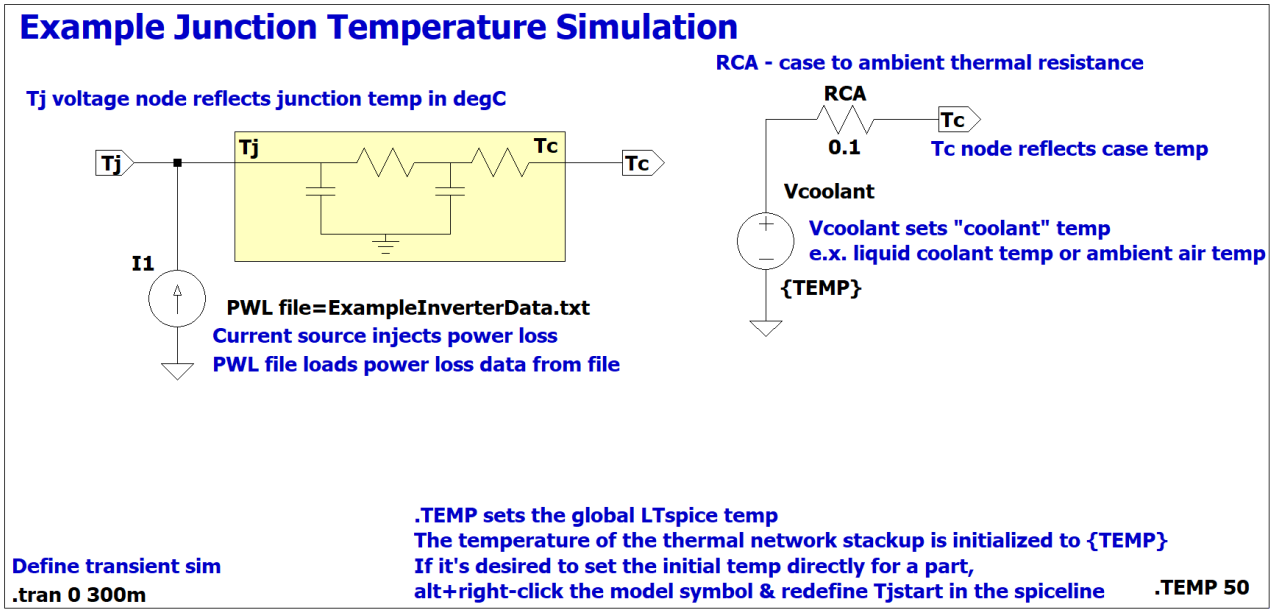



*Figure 12: Example power loss simulation*

temperature, and the blue waveform is the case temperature (amps represents watts, Volts represents °C). In the surge example, the junction temperature of the device rises very quickly, and begins to decay after the surge event. The case temperature does not change, as the duration of the power pulse (~100 µs) is too short for any energy to propagate to the case. For the inverter simulation (with an operating time of 200 ms), the case temperature does increase during the simulation. The model shows the changes in die temperature over short time scales and reflects the change in power levels of the inverter simulation. Empirical power loss data could be used to make mission profiles to understand changes in junction temperature over typical operation.

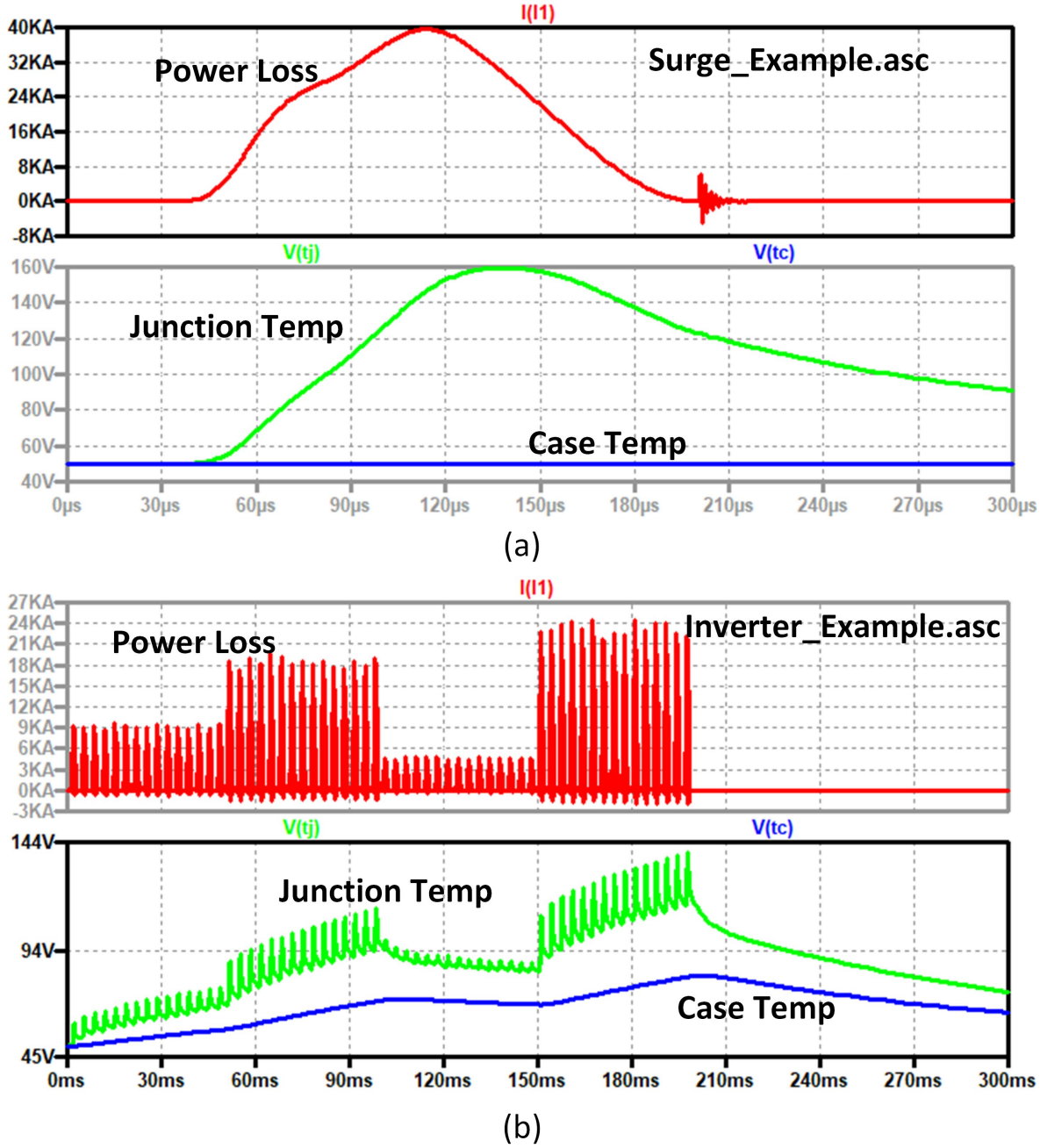



*Figure 13: Simulation results for (a) Surge_Example.asc and (b) Inverter_Example.asc*

## 4.3 Accessing Internal Nodes

As mentioned previously, Cauer models are useful because the node voltages represent temperatures within a depth of the module stack-up. These voltages can be accessed by right-clicking a plot window in LTspice and selecting “Add Traces.” The internal nodes can be accessed from the traces in this window. Figure 14 shows an example of the internal nodes for the inverter simulation. Here, the model instance name is U1, so the relevant nodes are under ‘U1’ in the list. The relevant node names for the model are t1, t2, and t3, with t1 being closest to the die and t3 being closest to the case. In Figure 14, two observations can be made from the simulation results. First, as expected, the deeper portions of the substrate are less affected by short-term changes in temperature due to their distance from the source of the losses and proximity to the stable heatsink. Second, the die temperature is hotter than any of the substrate layers. **Please note that, in the provided models, each RC pair does not represent a physical substrate. While this analysis is useful for evaluating trends and general behavior, it cannot be used to estimate the temperature of specific layers or attachments within the module.**

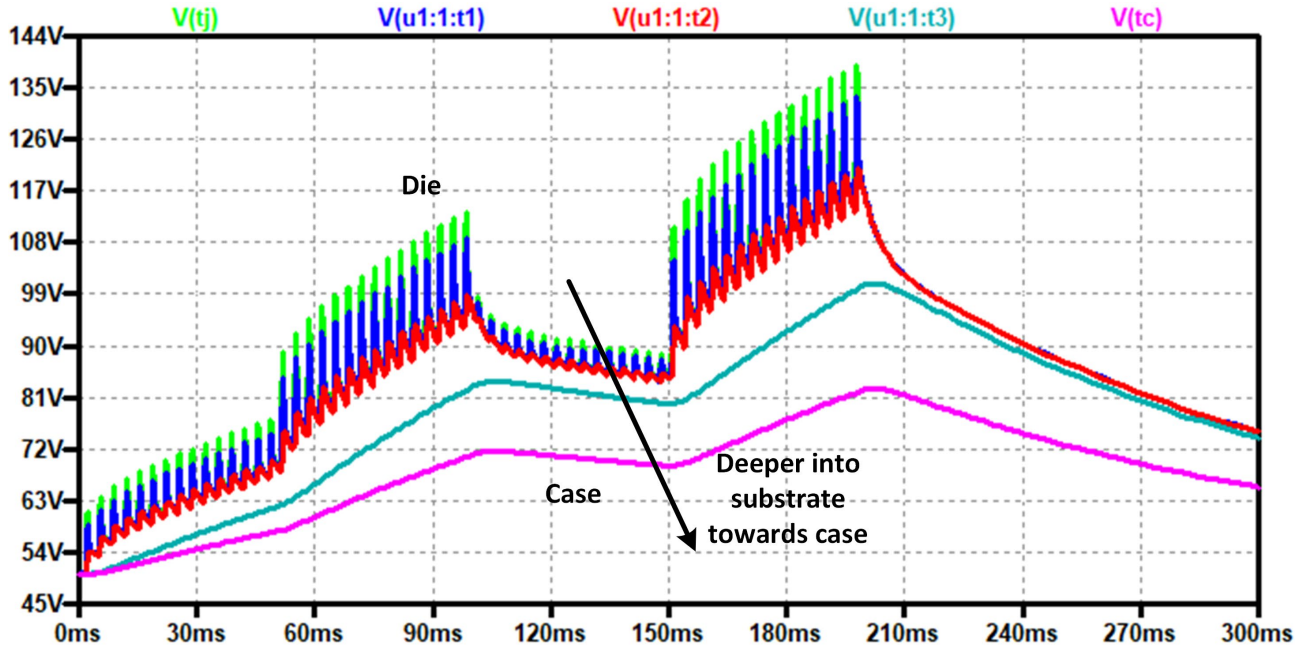



*Figure 14: Inverter simulation – internal node temperatures example*

## Revision History

| Date | Revision | Changes |
|---|---|---|
| **January 2025** | 1 | Initial Release |