



PLECS

Tutorial

Introduction to the Thermal Domain in PLECS

Tutorial Version 1.0

www.plexim.com

- ▶ Request a PLECS trial license
- ▶ Check the PLECS documentation

1 Introduction

In this exercise you will learn how to use the thermal modeling features of PLECS to create a combined electrical-thermal simulation of a buck converter. The specific learning goals for this exercise are listed as follows:

- Learn which components are needed for a thermal simulation.
- Extend an existing thermal model for the MOSFET with conduction loss and thermal impedance data using the Thermal Editor and built-in Curve Import Wizard tool.
- Establish the steady-state operating junction temperature of the MOSFET.
- Calculate the individual and total losses of the switches and overall system efficiency.

Before you begin Ensure the file `thermal_domain_start.plecs` and the directory `Thermal_Lib` are all located in your working directory. You should also have the reference files that you can compare with your own models at each stage of the exercise. A datasheet `c3m0120090d.pdf` is also included for your reference as part of the exercise.

2 Circuit

The circuit you will use for this exercise is shown in Fig 1. It is a buck converter with a SiC MOSFET switch that operates at 20 kHz. The electrical parameters are given in Table 1.

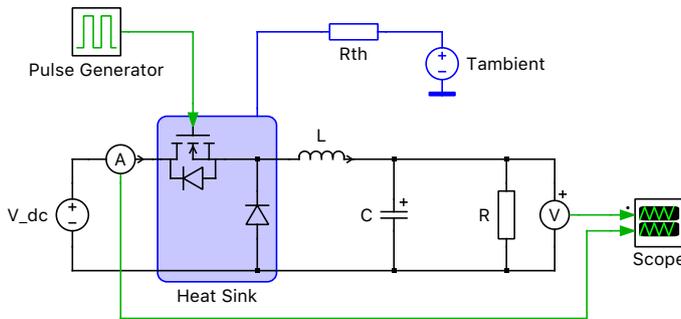


Figure 1: Combined electrical-thermal simulation of a buck converter

3 Background

To perform a thermal simulation using PLECS, three components are needed: a Heat Sink, semiconductor component(s) with a thermal description and an Ambient Temperature. The thermal simulation takes place at the same time as the circuit simulation but the thermal components are represented in PLECS in a separate domain. The thermal quantities and their analogous electrical equivalents are given in Table 2. In the analogous electrical circuit, conduction losses of a component are modeled as an ideal current source, the thermal impedance between the component and air as a simple resistor or RC chain, and the ambient temperature as a voltage source. This equivalent circuit is shown in Fig. 2.

The Heat Sink block is the fundamental component for modeling thermal circuits in PLECS. It provides a surface of equal temperature that can represent an actual heat sink in a system but more

Table 1: Buck converter electrical parameters

Component	V_{DC}	f_{sw}	Duty cycle	C	$V_{C,initial}$	L	$I_{L,initial}$	R
Value	500 V	20 kHz	50 %	200 μ F	250 V	10 mH	8 A	31.25 Ω

generally is an individual thermal layer. The heat sink block absorbs the switching and conduction losses of all components contained within its boundary. All semiconductors located on the heat sink will have the same case temperature. In PLECS, the switching energy pulses are modeled as a dirac-type pulses, having a zero-width and an infinite height. Therefore, either the thermal capacitance of the heat sink must be defined or a thermal chain with capacitance must be used to prevent the switching energy pulses from creating an infinite temperature across the thermal resistance.



Note: The Heat Sink captures *all* intrinsic components that dissipate thermal losses, meaning that in addition to semiconductor losses, ohmic resistors can also be monitored. Ohmic losses are calculated as $i^2 \cdot R$, or V^2/R , and are associated with all electrical resistor components found in the library.

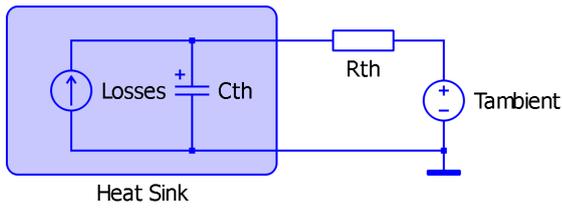


Figure 2: Electrical equivalent of thermal circuit

4 Setup the Thermal Simulation

4.1 Create the thermal model



Your Task: Open the PLECS model `thermal_domain_start.plecs` and add a Heat Sink, Thermal Resistor and Constant Temperature (Grounded) block to the circuit, as shown in Fig. 1. These components can be found in the Thermal section of the PLECS component library. The parameters for these components are as follows:

- The ambient temperature is 25 °C.
- The thermal capacitance of the heat sink is 0.01 J/K. Also set the initial temperature to the same value as the ambient temperature.
- The thermal resistance between the heat sink and ambient temperature is 0.5 K/W.



Note: You can move a heat sink terminal to another free slot on the border by dragging it while holding down the Shift key. When you release the mouse button, the terminal is moved.

Table 2: Equivalent thermal and electrical quantities

Therm. Domain	Elec. Domain
Temperature (K)	Voltage (V)
Heat (J)	Charge (C)
Heat flow (W)	Current (A)
Therm. cap. (J/K)	Elec. cap. (F)
Therm. res. (K/W)	Elec. res. (Ω)

4.2 Link and extend the thermal description for the MOSFET

The switching losses for the MOSFET have been predefined in the file C3M0120090D_tutorial.xml. To add this thermal description to the MOSFET you need to add the path of this file to the thermal description search path. You will then input the conduction loss information for the device. After the loss description is complete, you will then use the PLECS Probe to monitor signals associated with the MOSFET.



Your Task: First you will modify the conduction loss data by manually importing the data.

- 1 Under the **File + PLECS Preferences...** menu, navigate to the **Thermal** tab and add the directory that contains the MOSFET thermal description to the search path. Press the **Rescan** button to activate the change.
- 2 Double-click the MOSFET component. Under the **Thermal description** parameter on the **Thermal** tab, select “C3M0120090D_tutorial” using the **From library...** option. Click **Apply** to save the selection.
- 3 Use the **Edit...** option to open the Thermal Editor and look at the switching and conduction loss definitions. Note that there is component switching loss data already included using an **Energy scale** of μJ , as well as conduction loss data for 150°C . You will add datapoints for -55°C and 25°C , using **Figures 1** and **2** in the device datasheet, respectively. For the -55°C data, you will manually enter voltage values for the existing current columns, and then you will use the Curve Import Wizard tool to graphically acquire datapoints for the 25°C table.
- 4 Using the **Edit + New temperature values...** menu entry or by right-clicking on the 150°C cell or in the plot area, select **New temperature values...**, and add the two new temperatures in vector format in the dialog box like so: -55 25. These can either be separated by a space or comma. Note that when an existing row of voltage values exists, the cells for any new temperatures will acquire the nearest existing value by default (the new cells won't necessarily show 0 V or 0 J for switching losses).
- 5 Using **Figure 1** of the C3M0120090D device's datasheet, estimate voltage values for the positive current columns headers already chosen (i.e., 5 A, 15 A, 25 A, 35 A, and 45 A). It is recommended by the manufacturer to refer to the “ $V_{\text{GS}} = 15$ ” curve unless you are not driving the gate with full voltage (note that the data at 150°C uses the “ $V_{\text{GS}} = 15$ ” curve).
- 6 Ensure there is a value of 0 V at 0 A for all temperatures. It is also possible to enter voltages for negative current values, but since the MOSFET will not be conducting in the reverse direction for the buck converter topology we will ignore this region. However, should you need to extend the data into the negative current region, **Figures 15**, and **16**, and **17** provide these plots for the same temperatures.



Your Task: Now you will import further conduction loss data graphically from a datasheet plot.

- 1 For this next step you will need to take a screenshot image of **Figure 2** for importing into the Thermal Editor. An image file C3M0065090J_datasheet_figure2_conduction_loss_25C.png has been provided already. Highlight the 25°C temperature row and click on the  button in the top-right corner of the plot area.
- 2 Next, drop the image file into the window. You will see that this adds the image along with three temperature curves. You can deselect the curves for all three temperatures temporarily to avoid confusion. This particular datasheet uses the convention of the X-axis representing voltage, which is the opposite of the format used by PLECS. To rotate the image's orientation to accommodate this,

you will need to click the **Mirror axes** button for the **Orientation** selection in the **Configure axes** widget to the right of the Thermal Editor window.

- 3 You will then need to align the green axis guides with the axes and their endpoints on the plot. It is helpful to use the zooming option to align the two sets of axes more accurately by holding down the **Control** key if you are using Windows, or the **Command** key if you are using MacOS, and scrolling up with the mouse wheel. Try and place the upper limit of the Y axis as close as possible to value provided.
- 4 Then, reselect the 25 °C row and for each current column, drag the blue dot up from 0 V to the appropriate location on the “ $V_{GS} = 15$ ” curve. Rename the imported image if desired and click **Save**.



Your Task: (Optional) Import thermal impedance data from a datasheet curve.

- 1 The thermal impedance for the device has already been specified using a 4-element Cauer chain, but if you have time you can replace these values using information from the datasheet. Datasheets will often provide Foster or Cauer network coefficients, but in this case, only the heating curves are provided. The Curve Import Wizard tool has a fitting algorithm to calculate Foster network coefficients from the Single-Pulse test curve. Click on the **Therm. impedance** tab and then the  button in the top-right corner of the table area. Drop a screenshot image of **Figure 21** into the window. An image file C3M0065090J_datasheet_figure21_thermal_impedance.png has been provided already. You will need to enter the limits in the **Configure axes** widget to the right and line up the green axis guides with the axes on the plot as you did before. It is recommended to use $100e-6$ as the lower limit of the X-axis (**X-min**) to avoid that very small Foster element coefficients are calculated resulting in stiff differential equations. You should then start setting points to recreate the Single-Pulse curve by carefully double-clicking along it. When you have selected the points with sufficient accuracy, increase the **Number of elements** to 1. PLECS calculates a set of Foster coefficients and displays the result as an orange curve on top of the graph. Increase the **Number of elements** until the orange curve graphically indicates a close match to the Single-Pulse curve (~4 elements should be appropriate). If the current fit is not satisfactory, you can calculate a new one by pressing the **Recalculate** button or provide more resolution to the fit by adding additional points along the curve. Fig. 3 shows a complete curve fit and resultant coefficients. When you are satisfied with the fit, click the **Accept** button, followed by the **Save** button. Note that doing this will overwrite the original thermal impedance data provided with the file.



Your Task: Now we can monitor the MOSFET junction temperature and run a simulation.

- 1 Close the Thermal Editor window and return to the schematic. Add a Probe block from the library to your schematic and use it to monitor the MOSFET temperature. The easiest way to associate a component with a Probe is to drag it directly on to the Probe and release the mouse button when the pointer shape changes to crosshairs. After doing this, select the option to monitor the junction temperature from the list on the right.
- 2 A complete thermal description has already been provided with the diode (Wolfsped’s C3D08060A), but you will need to associate it from within the component’s parameter window as you did with the MOSFET. You can choose to also monitor the diode’s junction temperature with the PLECS Probe if you are interested.
- 3 This topic is not covered in this introductory tutorial, but a thermal description can also use custom variables, such as for allowing gate resistance-dependence in switching loss data. The MOSFET data already includes a variable “Rg” in the equations for **Turn-on losses** and **Turn-off losses**. In order to run a simulation we need to provide a value for the gate resistance. Double-click the MOSFET component. Under the **Thermal description** parameter on the **Thermal** tab, enter a value of 2.5Ω for the **Gate resistance**.

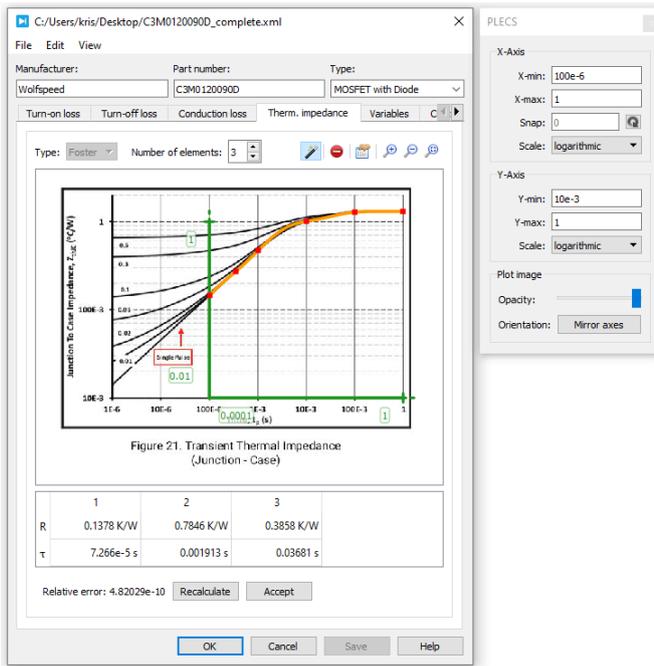


Figure 3: Thermal impedance curve import

- 4 Enter a value of 25°C for the **Initial temperature** on that same tab for the MOSFET, and also for the diode.
- 5 Run the simulation for 0.5 s by pressing **Ctrl + T**, leaving all other simulation parameters as their defaults.

? What is the final temperature of the MOSFET junction?

A $\sim 32^{\circ}\text{C}$

At this stage, your model should be the same as the reference model: thermal_domain_1.plecs.

5 Average Loss Calculation

In many cases, a factor of interest is the average power dissipation of each semiconductor. The average losses for a component can be calculated by summing the losses that occur during a switching cycle and generating an average power pulse during the next switching cycle. This cycle-average loss calculation procedure is summarized in Fig. 4. The PLECS C-Script block is used to perform the per-cycle integral and summing operations on the conduction and switching losses.

Your Task:

- 1 The PLECS component library contains blocks in the Controls > Filters list that can be used to measure the average losses. The Periodic Average block is suited for determining average conduction losses of semiconductors while the Periodic Impulse Average will determine the average switching losses. Place two of each into the model and provide the appropriate loss signals for the MOSFET and diode to them using PLECS Probe blocks. Ensure that you drag each semiconductor

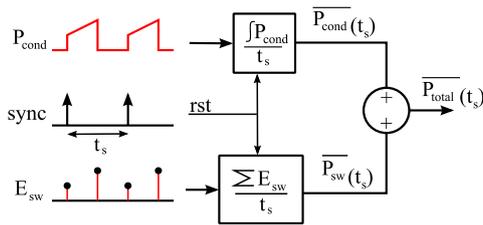


Figure 4: Calculation of total cycle-average losses

into its own probe and select the conduction and switching loss signals. Using Sum blocks, add the signals to obtain the total MOSFET losses, total diode losses, and total overall losses, as shown in Fig. 5.

- 2** Set the **Averaging time** of each averaging block to 0.05 ms to ensure the sync signal is the same as the switching period.
- 3** Place a Heat Flow Meter block between the thermal chain and the ambient temperature to measure the total average losses of both semiconductors. Use a Signal Multiplexer block to display the total losses on the same scope used to display the calculated losses.
- 4** Run the simulation for 1.0 s. Compare the output of the heat flow meter with the total loss calculation.

? Do the results agree? Why is there a difference in the waveforms?

A ~9 W. There is a difference because the heat flow meter measures the heat flow in the thermal circuit, where it takes several seconds for the heat sink to warm up due to its capacitance. The loss probes, however, are connected directly to the components' junctions.

 At this stage, your model should be the same as the reference model: thermal_domain_2.plecs.

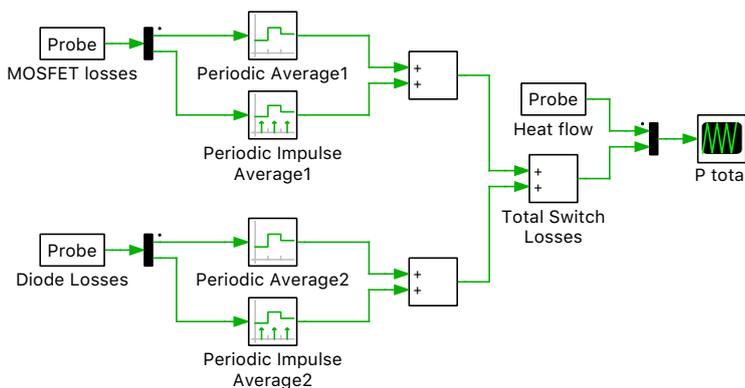


Figure 5: Calculating the average device losses and the total circuit losses using the PLECS averaging components

6 Efficiency Calculation

Once the losses associated with the power semiconductors are known, it is straightforward to calculate the efficiency of the converter. Note that ohmic losses will not be included in this calculation, but one

could add series/parallel resistive elements to the ideal components already in the circuit, and the heat sink could be enlarged to also contain the load resistor.



Your Task:

- 1 Place an additional Probe, Periodic Average, and Multiplexer block into the schematic near where the total MOSFET and diode losses are summed. Probe the input power of the converter by monitoring the power signal of the DC Voltage Source. The **Averaging time** of this block should also be 0.1 ms. Using the Multiplexer, combine the source power measurement and total loss calculation of the switching cell into a single bus with a width of 2 signals, as shown in Fig. 6. As an optional step, add a Saturation block with a **Lower limit** of 1 and an **Upper limit** of inf, which ensures that there are no “Division by zero” issues at the beginning of the simulation.
- 2 Place a Function block in series with the output of the Multiplexer and Saturation blocks. Here you will use the vectorized bus to access both the source power and switching cell loss signals. In the **Expression** box, type the equation $(1 - u[2]/u[1]) * 100$. Assuming that the upper (first) signal you routed to the Multiplexer input is the source power, this efficiency equation is equivalent to

$$\eta = (P_{in} - P_{loss})/P_{in}.$$

Multiplying this result by 100 effectively formats the result as a percentage. Finally, a Display block can be used to show the calculation result in numerical form, which may be preferred over using a Scope to show the calculation as a simulated waveform (and this may also be desired for the previous calculation for the total combined power of the MOSFET and diode losses).

- 3 Rerun the simulation.

? What is the efficiency of the system when operating at steady state?

A ~99.6%



At this stage, your model should be the same as the reference model: thermal_domain_3.plecs.

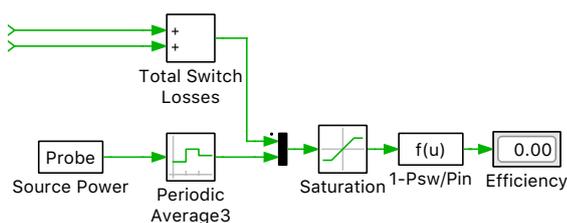


Figure 6: Calculating the converter efficiency using the input power and total device losses



Note: If you specify the Thermal description parameter for a semiconductor component, the dissipated thermal power does not correspond to the electrical power that is consumed by the device. Also, due to the instantaneous nature of the switching transitions, the dissipated thermal energy cannot be consumed electrically by the device. These important factors must be taken into account when you use the thermal losses for estimating the efficiency of a circuit. This means that all on-resistance and forward voltage values specified by a user in the component parameters are fixed during a simulation and only influence the electrical behavior of the system, i.e., the voltage and current magnitudes. Therefore, the efficiency calculation technique described above is the correct method you need to use in PLECS, and simply calculating P_{out}/P_{in} is not equivalent as it doesn't account for the thermal losses in the system.

7 Conclusion

In this exercise, you learned how to form a combined electrical-thermal simulation using PLECS. The fundamental component required for thermal analysis is the heat sink. You also learned how to create a thermal description for a switch by defining the conduction loss as a current-voltage profile, both with manual data entry by reading a datasheet, and using the Curve Import Wizard tool built into PLECS. These methods can also be applied to switching energy loss lookup tables and the thermal impedance network information. Once you have data describing the losses of the semiconductors and their dissipative behavior in the thermal circuit model, you can then calculate losses and system efficiency.

An extension to this exercise is to practice designing more complex, hierarchical thermal models in PLECS. This can be achieved by layering thermal components using subsystems and defining heat sinks underneath the top mask to represent things such as the heat sink(s), hot spots, etc.

References

- [1] Stig Munk-Nielsen, Lucian N. Tutelea, Ulrik Jaeger, *Simulation with Ideal Switch Models Combined with Measured Loss Data Provides a Good Estimate of Power Loss*, IEEE Industry Applications Conference 2000, Vol. 5, pp. 2915-2922, 8-12 October 2000
- [2] Martin Maerz, Paul Nance, *Thermal Modeling of Power Electronic Systems*, Application Note, April 2000
http://www.iisb.fraunhofer.de/en/arb_geb/powersys_pub.htm

Revision History:

Tutorial Version 1.0 First release

How to Contact Plexim:

☎	+41 44 533 51 00	Phone
	+41 44 533 51 01	Fax
✉	Plexim GmbH Technoparkstrasse 1 8005 Zurich Switzerland	Mail
@	info@plexim.com	Email
	http://www.plexim.com	Web

PLECS Tutorial

© 2002–2021 by Plexim GmbH

The software PLECS described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from Plexim GmbH.

PLECS is a registered trademark of Plexim GmbH. MATLAB, Simulink and Simulink Coder are registered trademarks of The MathWorks, Inc. Other product or brand names are trademarks or registered trademarks of their respective holders.