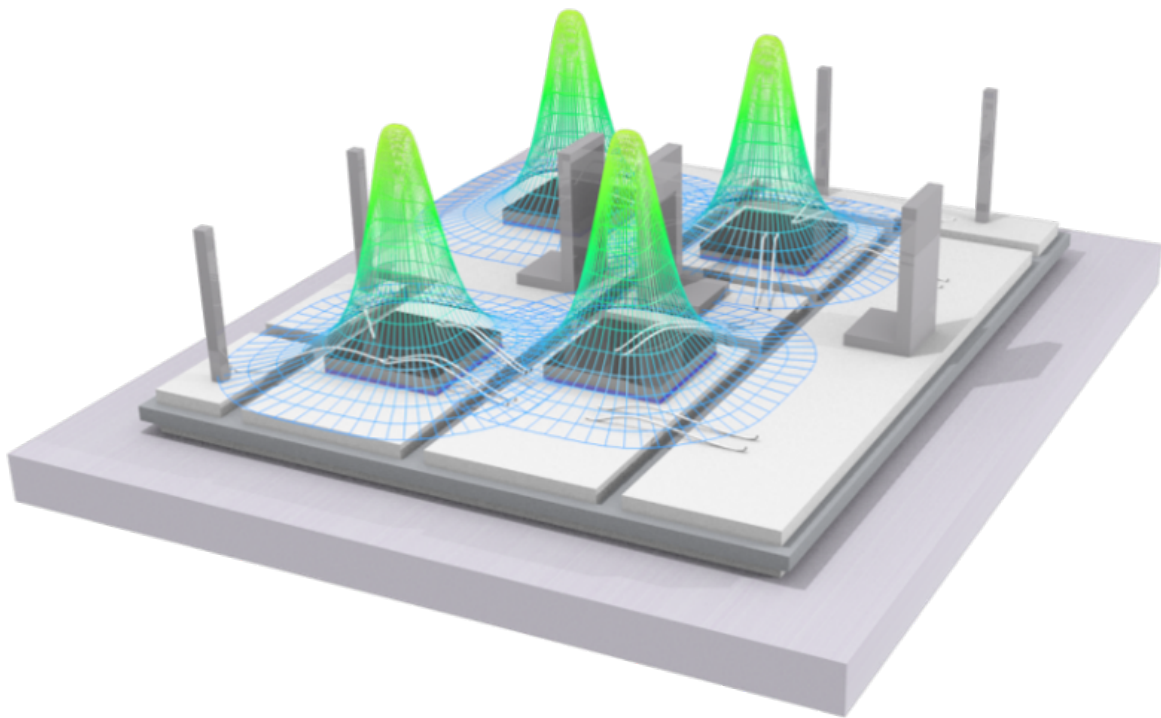




Mixed-Signal Computer-Aided Design Laboratory

PowerSynth User Manual

February 27, 2019



Contents

1	Introduction	3
1.1	Executive Summary	3
1.2	Organization	3
1.3	PowerSynth Overview	3
2	Using PowerSynth	6
2.1	Installing PowerSynth	6
2.2	Creating a New Project in PowerSynth	6
2.2.1	User Interface	6
2.2.2	Symbolic Layout File	6
2.2.3	Create a New Project	8
2.3	Defining the Module and Components	8
2.3.1	Module Stack	8
2.3.2	Layer Stack File	9
2.3.3	Component Selection	10
2.3.4	Device and Lead Selection	12
2.3.5	Assigning Devices and Leads	13
2.3.6	Assigning Virtual Wire Connections	14
2.4	Defining Correlations and Constraints	15
2.4.1	Design Variable Correlation	15
2.4.2	Constraint Creation	15
2.5	Design Performance Identification	16
2.5.1	Thermal Measurement	16
2.5.2	Electrical Measurement	17
2.5.3	Inductance and Resistance Measurement	17
2.5.4	Device Connection Setup (Example)	18
2.5.5	Electrical Modeling	19
2.5.6	Capacitance	21
2.6	Optimization and Results	22
2.7	Post-Layout Optimization	25
2.8	Exporting Saved Solutions	27
2.8.1	Export to ANSYS Q3D	27
2.8.2	Export to SolidWorks	27
2.8.3	Export SPICE Netlist	28
2.8.4	Export to Keysight EMPro	29
3	Libraries and Editors	29
3.1	Technology Library Editor	29
3.1.1	Device Information	30
3.1.2	Add Die Attach	30
3.1.3	Add Lead	31
3.1.4	Add BondWire	31
3.1.5	Add Substrate	32
3.1.6	Add Substrate Attach	32
3.1.7	Add Baseplate	33
3.2	Process Design Rules Editor	34

3.3	Layout Editor	35
4	PowerSynth-Related Publications	36
5	Appendix 1: EMPro Export	37
5.1	Introduction	37
5.1.1	Functionality	37
5.1.2	Caveats and Limitations in Current Release	37
5.2	Exporting from PowerSynth	38
5.2.1	Selecting a Design	38
5.2.2	Saving the EMPro Script	38
5.3	Importing to EMPro	38
5.3.1	Creating a New Project	38
5.3.2	Running the Script	39

1 Introduction

1.1 Executive Summary

PowerSynth currently performs multi-objective optimization to produce Pareto-front solutions to the proper placement of power semiconductor device die and the routing of metal traces on ceramic substrates. The tool accounts for temperature distributions and electrical parasitics as a function of the layout geometries that it considers. This tool has been hardware validated. Continued research on this project will further elaborate the capabilities by extending the work to greater fidelity in both the thermal and electrical domains.

Some of the major features of this tool include:

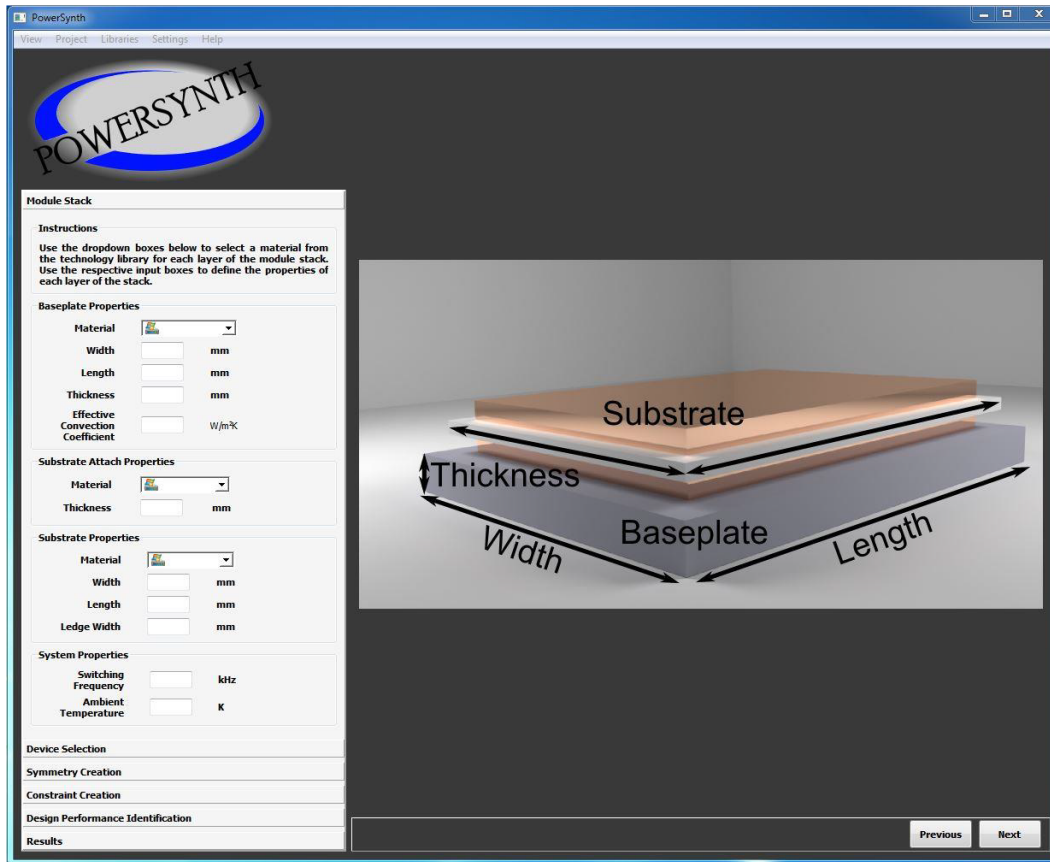
- Fast, accurate models for calculating electrical parasitics and thermal performance
- Multi-objective optimization for layout synthesis
- Able to synthesize and evaluate hundreds of layouts per minute
- User can select multiple performance metrics for optimization
- Built-in technology library including devices, substrates, attachment materials, bondwires, and more.
- Manufacturer Design Kit for incorporating packaging house design rules and tolerances
- Export designs to Ansys Q3D, FastHenry, or SolidWorks
- Post-layout optimization for trace corner filleting
- Easy to use GUI

1.2 Organization

After a brief introduction to PowerSynth, this document walks the user through various features of PowerSynth by creating a new project in a step-by-step presentation. Following the walkthrough, some additional information on the Technology Library, Process Design Rules, and Layout Editor are presented.

1.3 PowerSynth Overview

PowerSynth is an EDA tool with a friendly, graphical user interface for Multi-Chip Power Module (MCPM) layout synthesis. The tool is capable of importing a symbolic layout, which is an abstract layout representation of an MCPM, to automatically synthesize and generate multiple, real physical-layout solutions. This interface allows users to define technology libraries for power module materials, set up design constraints, and establish performance metrics for optimization. Additionally, export of optimal solutions is supported both graphically, on a 3D Solution Browser, and to modeling and analysis tools such as SolidWorks, Ansys Q3D, FastHenry, and Keysight EMPro.



The layout optimization of PowerSynth is based on fast thermal and electrical parasitic models developed specifically for this tool. The thermal model uses spatial superposition of temperature distributions to determine the temperature of devices in a module. The temperature and heat flux distribution of each device is characterized from a single run of an automated FEM simulation. In this simulation, a single die is placed in the center of the module atop an un-patterned substrate. The top surface of this device is treated as the source of power dissipation and a heat transfer coefficient is applied to the backside of the module heat spreader.

A compact thermal model (CTM) is then constructed from the results of this simulation. Thermal resistance values for the CTM are found by placing the characterized temperature distribution for each device in superposition with neighboring ones. Heat flux distributions obtained in the characterization step are used to account for the interaction among devices given their positions on the current trace layout. Once the CTM is constructed, linear algebra techniques similar to those found in circuit analysis are used to predict the temperatures of each device. The model can predict changes in temperature over variations in trace layout and device positioning with a maximum error of less than 10%.

The electrical parasitics in a module are approximated by closed-form solutions to micro-strip transmission line models. Since all power modules share the same basic geometry, the micro-strip model provides a reasonably accurate representation of a power module's trace parasitics. This is many times faster to execute when compared with finite element analysis. The electrical parasitic extraction works by breaking down a given layout into a lumped element network or graph with resistance and inductance values for each element or edge, approximated by micro-strip solutions. Capacitance from trace to baseplate (effective ground-plane) is evaluated at each lumped element node. These values can be used either directly in optimization or written into a netlist for simulation.

purposes.

PowerSynth operates on an imported symbolic layout. This is a simple drawing of the topology of an initial module layout and is easy for a designer to create. A symbolic layout is comprised of three basic elements: lines, points, and rectangles. The line elements represent traces or bond wires. The point elements represent particular devices or leads, and the rectangle elements represent traces which span multiple traces vertically or horizontally in topological space. A designer chooses a set of performance metrics, in this case electrical or thermal, based on the symbolic layout. A multi-objective optimization problem is formulated based on the symbolic layout by allowing each layout line-element a variable width, which constitutes a set of design variables. The Non-dominated Sorting Genetic Algorithm II (NSGA-II) is used to perform the optimization procedure.

After the optimization routine is run, a particular layout can be selected from a set of trade-off solutions for the MCPM design, allowing the designer easy access to the entire, viable design space. This system also allows a designer to quickly test many different layout solutions while maintaining layout quality. The fast electrical and thermal models both predict temperature and parasitic values accurately with respect to FEM tools.

2 Using PowerSynth

2.1 Installing PowerSynth

1. Run the PowerSynth install wizard
2. Run PowerSynth.exe

Note: It is STRONGLY RECOMMENDED that you install PowerSynth in the C drive (the default installation directory) and NOT in Program Files. Installing PowerSynth in Program Files can result in various issues relating to administrator permissions.

2.2 Creating a New Project in PowerSynth

2.2.1 User Interface

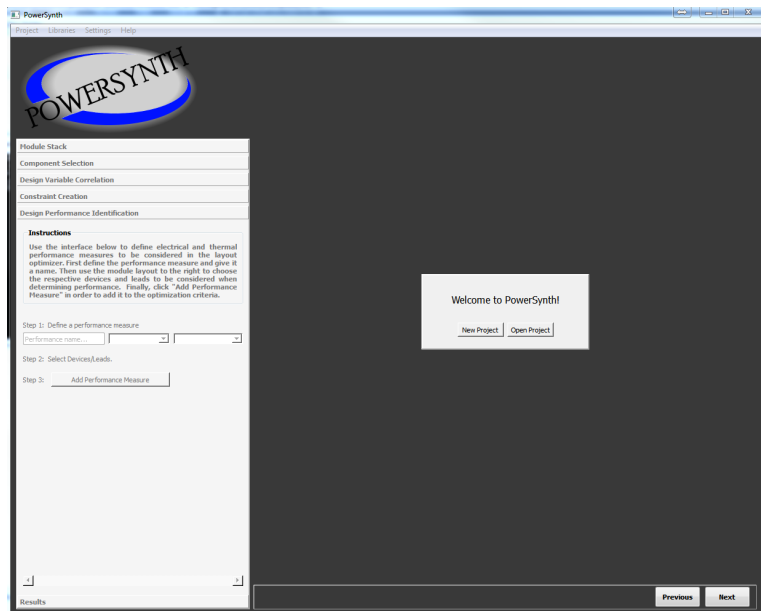
The welcome screen is shown on the right. The navigation pane on the left is used to develop a project once a new project is created or an existing project is opened. Before we click “New Project”, we need to define the layout template that will be imported.

2.2.2 Symbolic Layout File

A symbolic layout is a simple stick diagram used to represent a simple power module’s abstract layout. In the previous version, PowerSynth relies on InkScape to create a SVG file. This SVG file represents a stick diagram which is then imported into PowerSynth for a new project. In this version, SVG feature has been removed due to some unstable functionalities. Instead, the user can provide a script (text file) to define the stick diagram. Each point on this stick diagram represents either a device or a lead, while each line represents a trace. Bondwires information will need to be provided later in the setup (see Assigning Virtual Wires Connections). This script must be created before creating a new project in PowerSynth. Open a text file and define the layout template using lines and points with the syntax as shown below. Lines denote traces, while points denote devices and leads. Below is a sample layout script. It is of the format:

Name Type (x1,y1) (x2,y2)

Note: The element name (first column) must be four characters long. For a point, (x1, y1) is the coordinate of the point. For a line, (x1, y1) is one of the end points. (x2, y2) is the other endpoint. Save this file in either .psc or .txt format.

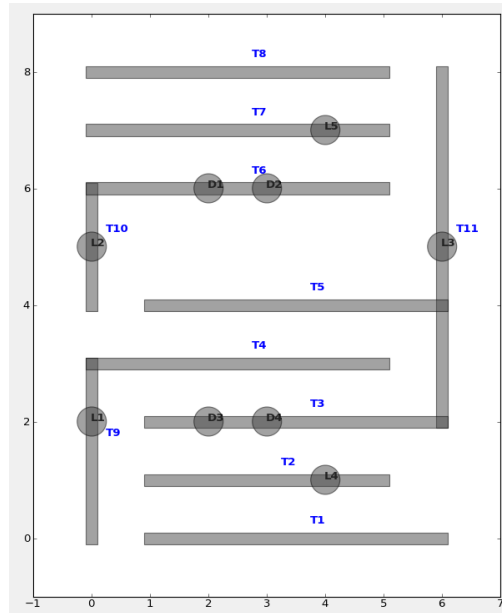


The example script for the symbolic layout is shown below:

```

T1 l (1,0) (6,0)
T2 l (1,1) (5,1)
T3 l (1,2) (6,2)
T4 l (0,3) (5,3)
T5 l (1,4) (6,4)
T6 l (0,6) (5,6)
T7 l (0,7) (5,7)
T8 l (0,8) (5,8)
T9 l (0,0) (0,3)
T10 l (0,4) (0,6)
T11 l (6,2) (6,8)
D3 p (2,2)
D4 p (3,2)
D1 p (2,6)
D2 p (3,6)
L1 p (0,2)
L2 p (0,5)
L3 p (6,5)
L4 p (4,1)
L5 p (4,7)

```



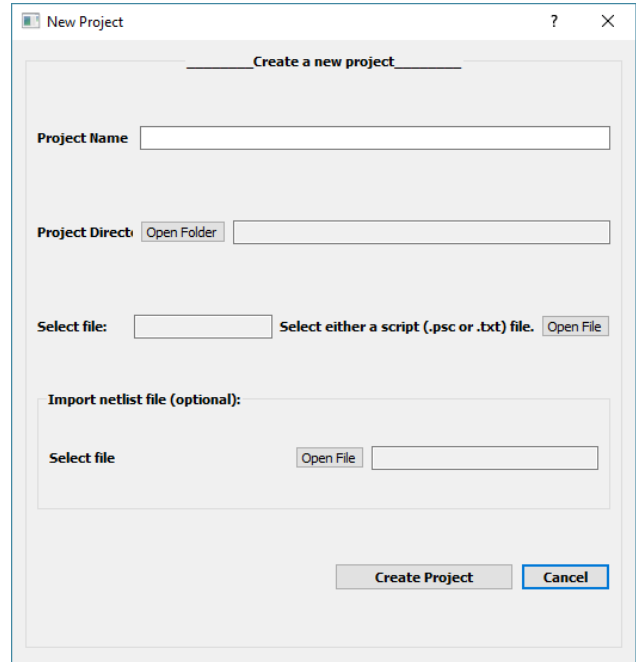
Note that these coordinate values are not real coordinates and thus have no dimensional units. These coordinates are used to find the relative position of each line and point to each other. Once this file is imported to the software, an automatic normalization algorithm will run to count the number of rows and columns. These coordinates are then normalized between zero and the maximum number of rows/columns. The corresponding symbolic layout for the abovementioned script is represented on the left. Once the project is loaded, the layout can still be modified using the Layout editor under **Project -> Open Layout Editor**

2.2.3 Create a New Project

Selecting the “New Project” button on the welcome window brings up the dialog for a new project as shown on the right. First, give the project a name. Then, select the folder where it will be saved. Import the symbolic layout file as shown above using “select file”. Select the directory where you saved the text file. Finally, click “Create Project”.

Optional: The user can also specify a netlist file that have same circuit topology with the symbolic layout. This can be later used to map the net names with the symbolic layout object. A sample netlist format for a 4 switches half bridge is shown below:

```
*Sample Netlist file:*
M1 DC_plus G_High Out Out NMOS
M2 DC_plus G_High Out Out NMOS
M3 Out G_Low DC_neg DC_neg NMOS
M4 Out G_Low DC_neg DC_neg NMOS
.model NMOS NMOS
.model PMOS PMOS
.end
```



2.3 Defining the Module and Components

2.3.1 Module Stack

The module stack tab on the left allows the user to input module stack information for the design. This includes dimension and material information of the substrate, substrate attach and baseplate. Operating conditions such as ambient temperature, effective convection coefficient of baseplate and switching frequency are also required for thermal and electrical evaluations. The user can update the tech files (.p) using *** Libraries->Tech lib.**

Module Stack

Use the dropdown boxes below to select a material from the technology library for each layer of the module stack. Import a layer stack CSV file or use the respective input boxes to define the properties of each layer of the stack.

Baseplate Properties

Material:

Width: mm

Length: mm

Thickness: mm

Effective Convection Coefficient: W/m²K

Substrate Attach Properties

Material:

Thickness: mm

Substrate Properties

Material:

Width: mm

Length: mm

Ledge Width: mm

System Properties

Switching Frequency: kHz

Ambient Temperature: K

Another way to import this module stack is through a csv layer stack file. The description for this file is explained in the next section.

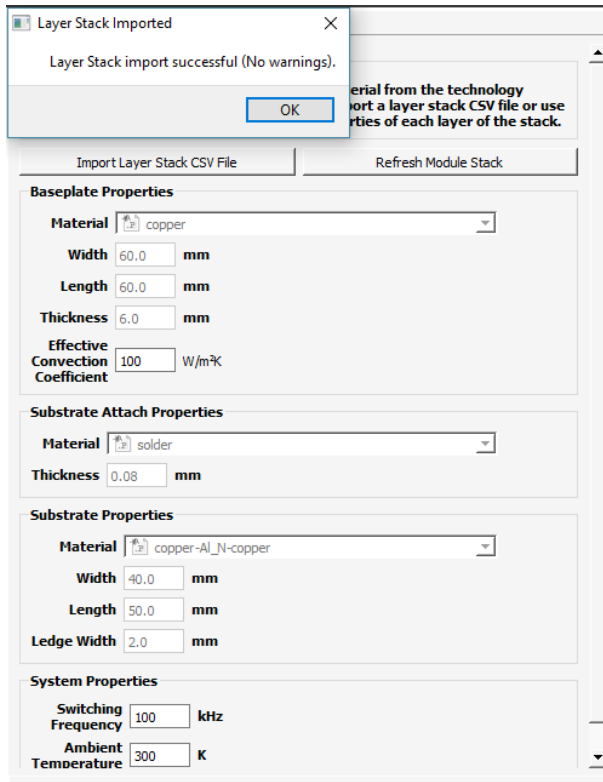
2.3.2 Layer Stack File

Abbrev	LayerID	Name	Layer Position	Width	Length	Thickness	Material
B:	1	B1	1	45	45	6	copper
SA:	2	SA1	2	40	40	0.08	solder
M:	3	M1	3	38	38	0.41	copper
D:	4	D1	4	40	40	0.64	Al _N
I:	5	I1	5	28	28	0.41	copper
C:	5	C1	6	2.5	4	0.2	SiC

The Table above shows the module stack structure. The user can specified this in an excel (.csv) and load it using the **Import Layer Stack File** button in the UI.

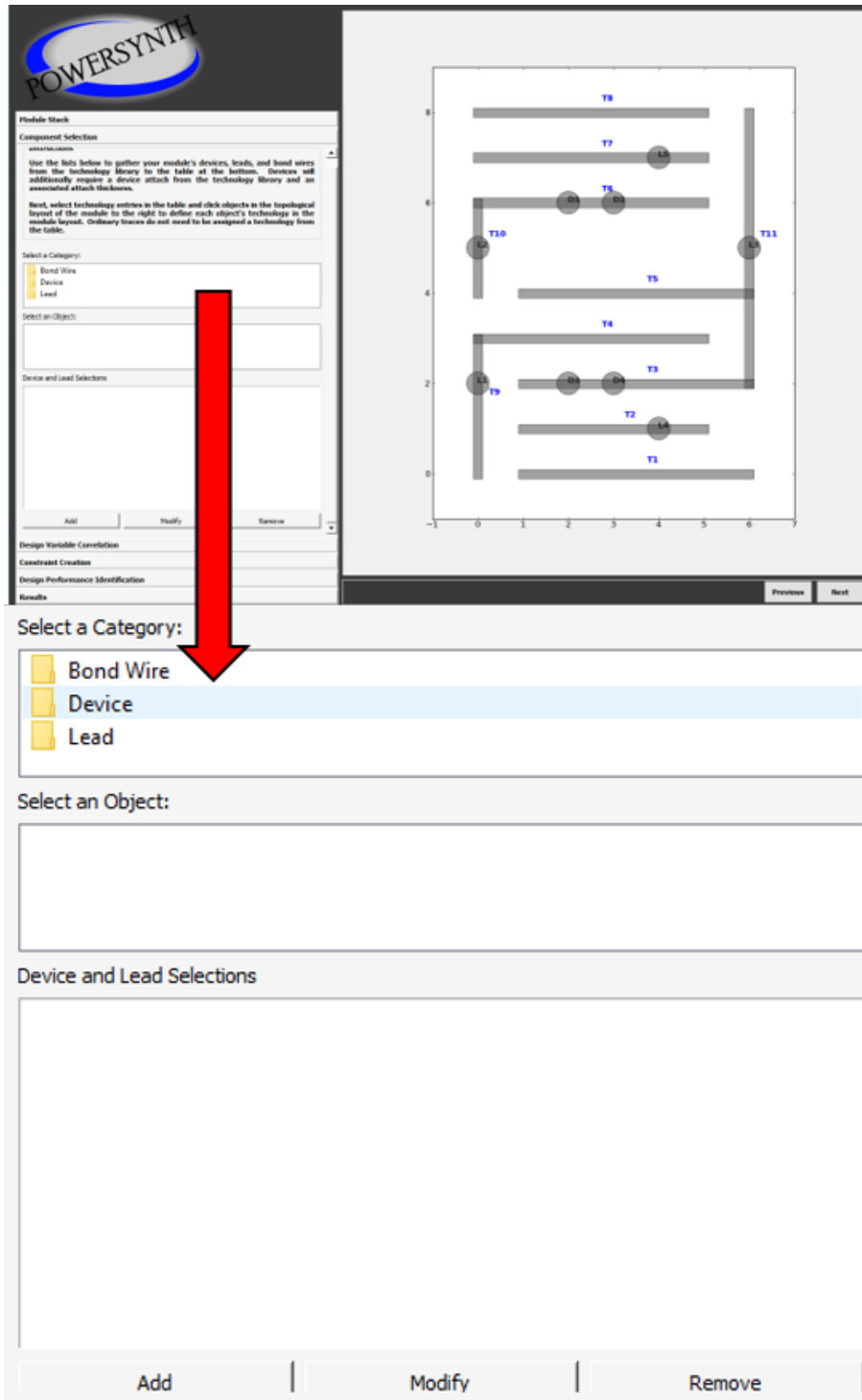
Some Abbreviation: B: Baseplate, SA: Substrate Attach, M: Metal, D: Dielectric, I: Interconnect (Metal), C: Component (this is optional for this version).

Below shows the UI after successfully importing the layer stack. To go back to the normal mode, press the **Refresh Module Stack** button.



2.3.3 Component Selection

The symbolic layout is shown on the right side of the interface. This allows users to insert component technology files. These files contain information about the bondwires, leads, and devices the user wishes to include. Notice that the names from the layout script for each traces and Devices are shown on the right-hand side. (Symbolic Layout)



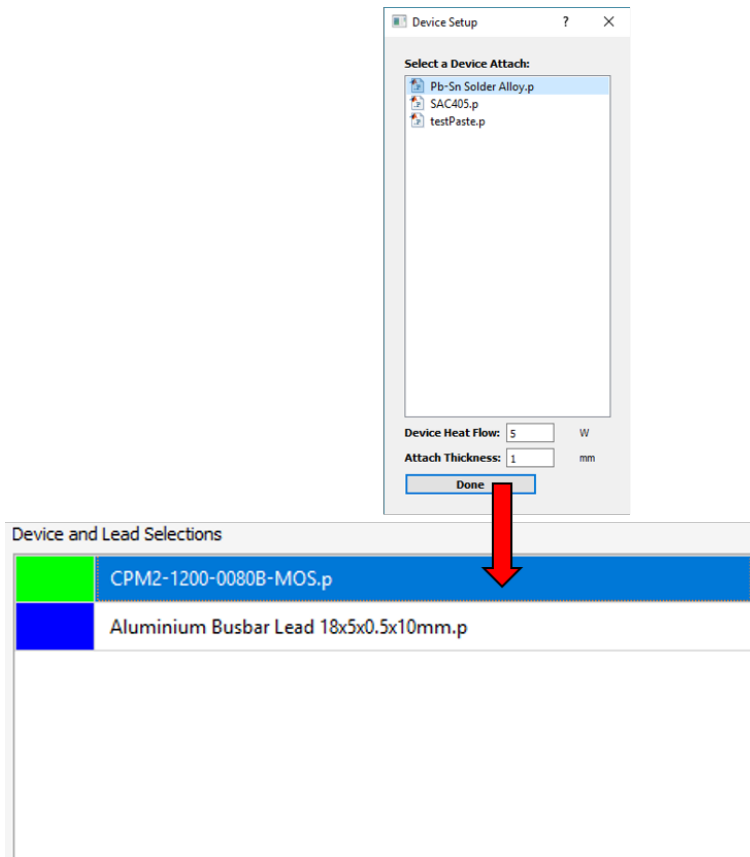
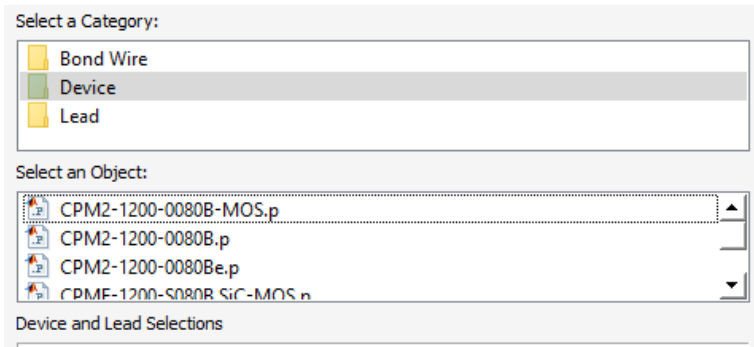
The user can assign devices and leads to the circles on the symbolic layout. The unassigned lines will represent copper or aluminum traces (depending on substrate selection). Wires bond will be treated as virtual connections. However, it is recommended to finish the devices and leads selection first before assigning virtual wire bonds to ensure correct connection.

2.3.4 Device and Lead Selection

Click on “Device” or “Lead” from the Category list, a list of objects under object selection will be listed out. This can be specified using the Tech-Lib (See Technology Editor section)

Once an object is selected, press the “Add” button. In case of device, this will pull up a dialog on the right to define the solder type (tech lib), solder thickness and average power of the device. This information will be used for thermal optimization. Click Done

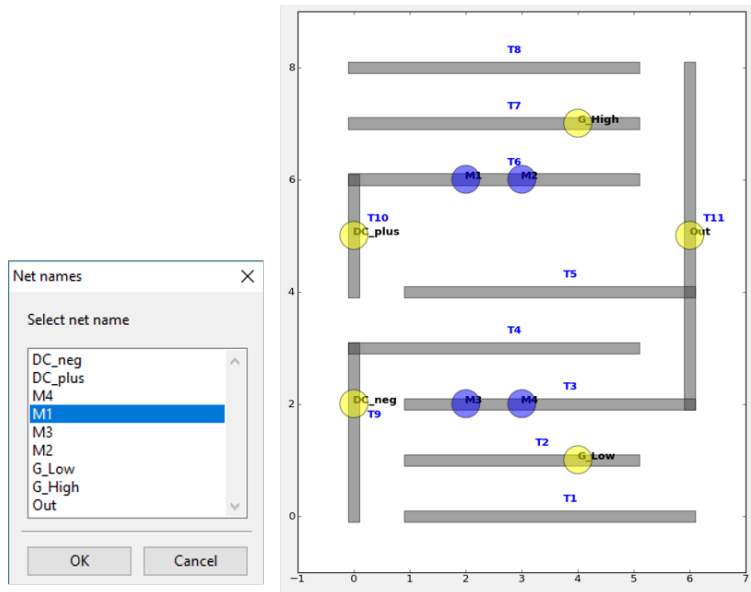
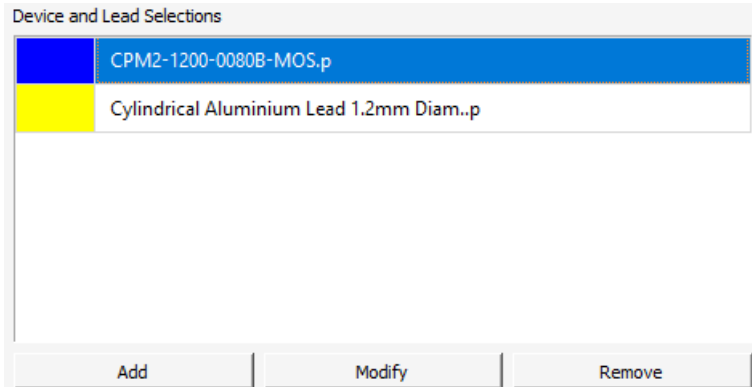
to apply this for the device. This will update the device and lead selection table. The user can do the same for the lead selection to fill the Device and Lead Selection table on the right.



2.3.5 Assigning Devices and Leads

First, select a device or lead object from the table.

Assigning these selections on the Symbolic Layout by clicking on the circles. To deselect, click on the selected circle one more time. The circles will be colored based on the random generated color on the Device-Lead Table. In case the user imported a netlist (Create new project section), a small dialog will be triggered to select the corresponded net name.

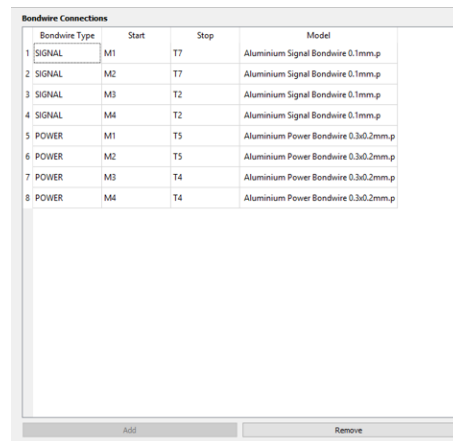
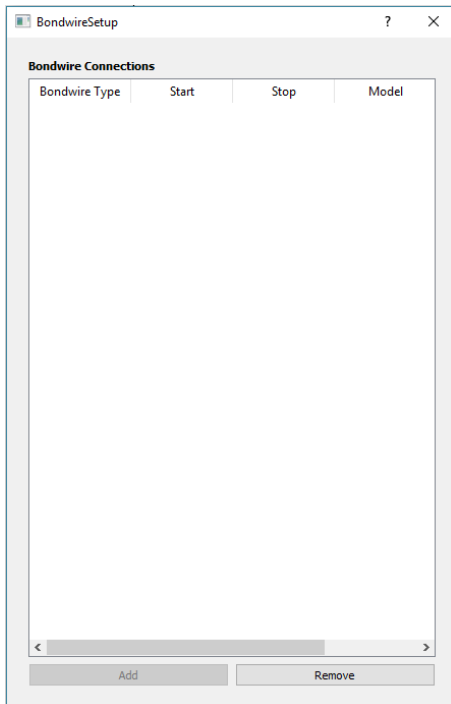
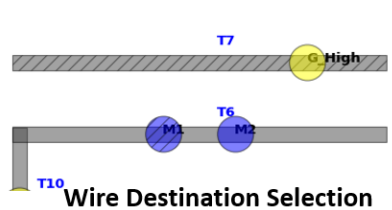
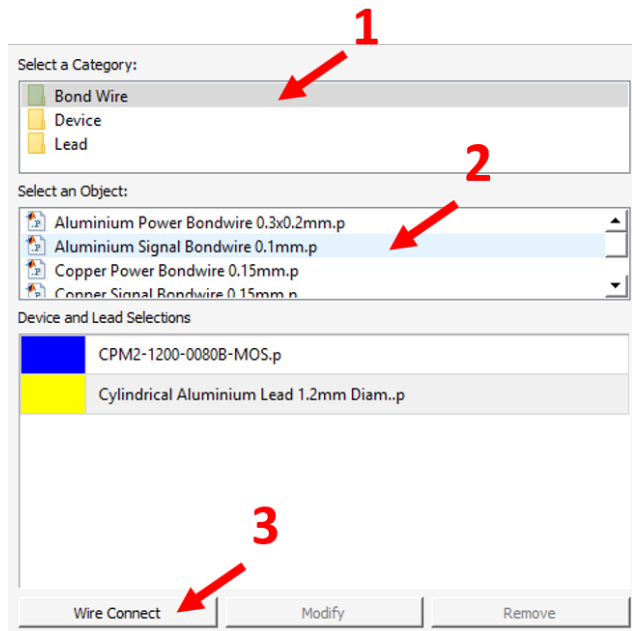


2.3.6 Assigning Virtual Wire Connections

Once all of the devices and leads are setup like above, the next step is to setup the virtual wires connections. To do this, first click on “Bond Wire” from the categories selection list. Then, select a bondwire object. The “Add” button will change its text to “Wire Connect” as seen on the right:

A bond wire Setup dialog will open as seen on the right. The symbolic layout is now in wire selection mode. The user will need to select 2 traces or a trace and a device (device to device bondwire is not supported yet) for wire connections. For a MOSFET, it is required to have signal and power wires for signal and source sides. To select 2 objects, simply click on the symbolic layout again. The selected object will be hatched as shown below.

The **Add** button on the dialog will be enabled if 2 objects are selected (click again if you want to deselect an object). On the main window’s left pane, re-select the type of bondwire. Click the **Add** button to add the wire connection to the table. Doing so, we can have a complete table (on the left). Close the dialog, the table can be saved and edited any time.

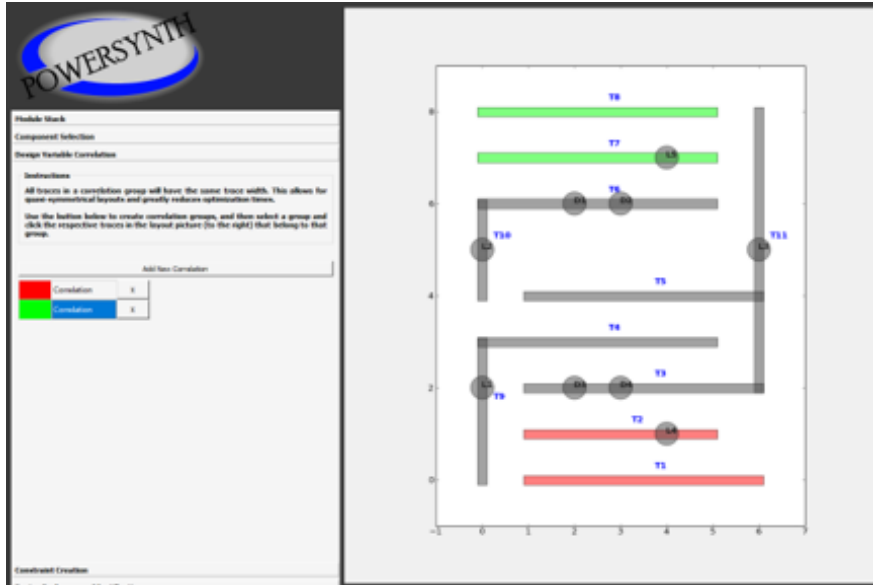


Full Virtual Connection Table

2.4 Defining Correlations and Constraints

2.4.1 Design Variable Correlation

If symmetrical traces or device positions are desired in the design, correlation between devices or traces can be assigned. This will ensure that selected traces will have the same width (length) or that selected devices will have same position. Click “Add New Correlation” and a new row will appear in the table. Click on a row then click on traces or devices to specify a correlation.



2.4.2 Constraint Creation

Here, the user can click on each line (trace) to specify a min and max or fixed width for a trace. If there are symmetry constraints, these will be applied to all symmetric traces.



2.5 Design Performance Identification

In this step, the user can choose the performance metrics that are to be optimized. The next two sections below show the process for assigning performance metrics. Once the performance metrics are selected, the user can go to the next tab to run the design optimization. Currently, PowerSynth supports thermal and electrical measurement of a power module.

2.5.1 Thermal Measurement

On the navigation tool, a user can define a performance measure. There are 2 available thermal models in PowerSynth: Rectangle Flux (analytical) and Fast Thermal (Characterized by FEM simulation). In case of thermal measurement, follow the steps below:

The image shows a software interface for defining performance metrics. It includes a 'Design Performance Identification' window with instructions and a step-by-step process. A separate box lists six instructions for thermal measurement, with red arrows pointing to the corresponding UI elements. Below the instructions is a diagram of a power module layout with various components labeled (T1-T11, C1, C2, C3, C4, C5, C6, C7, C8, C9, C10, C11, C12, C13, C14, C15, C16, C17, C18, C19, C20, C21, C22, C23, C24, C25, C26, C27, C28, C29, C30, C31, C32, C33, C34, C35, C36, C37, C38, C39, C40, C41, C42, C43, C44, C45, C46, C47, C48, C49, C50, C51, C52, C53, C54, C55, C56, C57, C58, C59, C60, C61, C62, C63, C64, C65, C66, C67, C68, C69, C70, C71, C72, C73, C74, C75, C76, C77, C78, C79, C80, C81, C82, C83, C84, C85, C86, C87, C88, C89, C90, C91, C92, C93, C94, C95, C96, C97, C98, C99, C100). The diagram shows a central component with various leads and connections, and a temperature profile overlaid on it.

Design Performance Identification

Instructions

Use the interface below to define electrical and thermal performance measures to be considered in the layout optimizer. First define the performance measure and give it a name. Then use the module layout to the right to choose the respective devices and leads to be considered when determining performance. Finally, click "Add Performance Measure" in order to add it to the optimization criteria.

Step 1: Define a performance measure
Average_TEMP [Thermal] [Select...]

Step 2: Electrical Model:
Thermal Model: [Response Surfs] [Fast Thermal (FEM)]

Step 3: Select Devices/Leads.

Step 4: [Setup device states] [Add Performance Measure]

Thermal (read the instructions carefully):

1. Name your performance metric
2. Choose "Thermal" as the measurement.
3. Select a computation type: average, max, Std deviation...
4. Select a thermal model.
5. Click on the devices you want to apply this measurement to (only devices can be selected)
6. Select "Add Performance Measure" button, new measurement will be added to the performance table

Step 4: [Setup device states] [Add Performance Measure]

Average_TEMP	Thermal	Max	X
--------------	---------	-----	---

2.5.2 Electrical Measurement

Parasitic capacitance, resistance, and inductance can each be specified as design performance metrics to be optimized. The next few sections will detail the steps necessary to define them and build response surface models for quick evaluation.

2.5.3 Inductance and Resistance Measurement

Instructions

Use the interface below to define electrical and thermal performance measures to be considered in the layout optimizer. First define the performance measure and give it a name. Then use the module layout to the right to choose the respective devices and leads to be considered when determining performance. Finally, click "Add Performance Measure" in order to add it to the optimization criteria.

Step 1: Define a performance measure

L1 Electrical Inductance

Step 2: Electrical Model: Response Surfs Thermal Model: Fast Thermal

Step 3: Select Devices/Leads

Step 4: Setup device states Add Performance Measure

Step 4: Setup device states Add Performance Measure

L1	Electrical	Inductance	X
----	------------	------------	---

Electrical (read the instructions carefully):

1. Name your performance metric
2. Select "Electrical" as the performance measure
3. Select from: inductance, resistance
4. Select an Electrical model (see Electrical Modeling)
5. Here you can select two leads and devices objects (only two at a time for each path you want to measure). If the user clicks on a device type, the user will be asked to specify a target pin (e.g: D, S, G ...)
6. Hit "Setup device states" to enter the loop selection mode (see examples in next section). A dialog is opened where user can setup the connections between devices pins
7. Finally, hit "Add Performance Measure" to add the measure to the table

Device State

Device ID	Connection 1	Connection 2	Connection 3
1 M3	<input type="radio"/> D-S	<input type="radio"/> G-S	<input type="radio"/> G-D
2 M4	<input type="radio"/> D-S	<input type="radio"/> G-S	<input type="radio"/> G-D
3 M1	<input type="radio"/> D-S	<input type="radio"/> G-S	<input type="radio"/> G-D
4 M2	<input type="radio"/> D-S	<input type="radio"/> G-S	<input type="radio"/> G-D

OK

2.5.4 Device Connection Setup (Example)

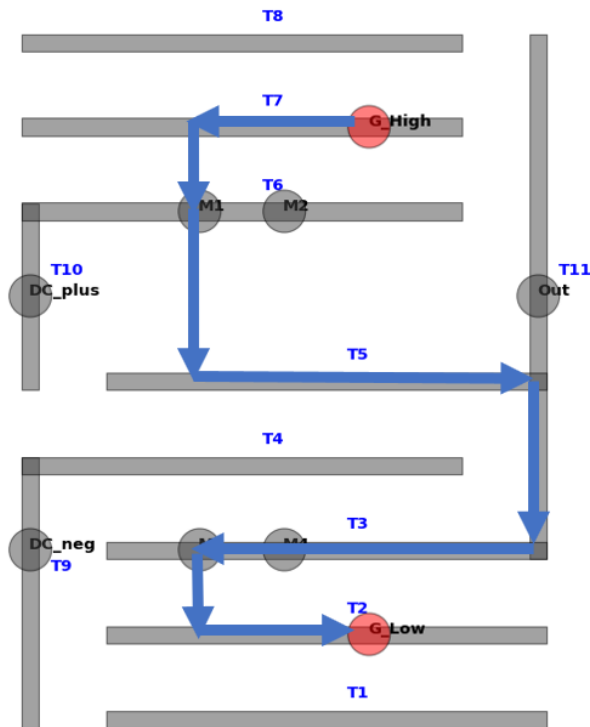
Although on the symbolic layout devices are represent as nodes (circles), in a real power module layout, these devices have real dimension and several connection pads. By default, there are no connections between these pads (usually a capacitance if this is a MOSFET). To give the user the flexibility to select different loops, the setup device connection feature is used to select virtual connections between any two pins of a given devices. For example, from the same electrical measurement above, we can setup 2 different loops.

Loop 1:

Setup the device connections as follows:

Device ID	Connection 1	Connection 2	Connection 3
1 M3	<input checked="" type="radio"/> D-S	<input type="radio"/> G-S	<input checked="" type="radio"/> G-D
2 M4	<input type="radio"/> D-S	<input type="radio"/> G-S	<input type="radio"/> G-D
3 M1	<input checked="" type="radio"/> D-S	<input type="radio"/> G-S	<input checked="" type="radio"/> G-D
4 M2	<input type="radio"/> D-S	<input type="radio"/> G-S	<input type="radio"/> G-D

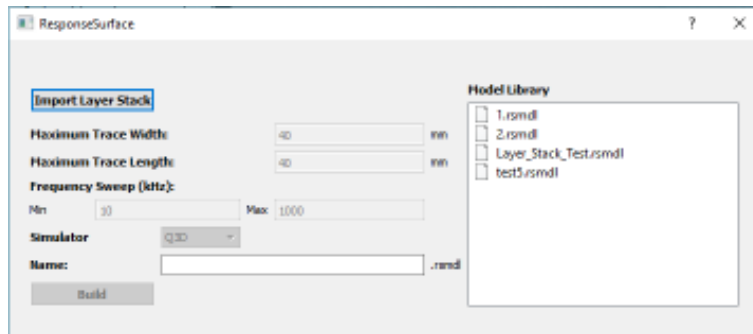
We now have the following loop:



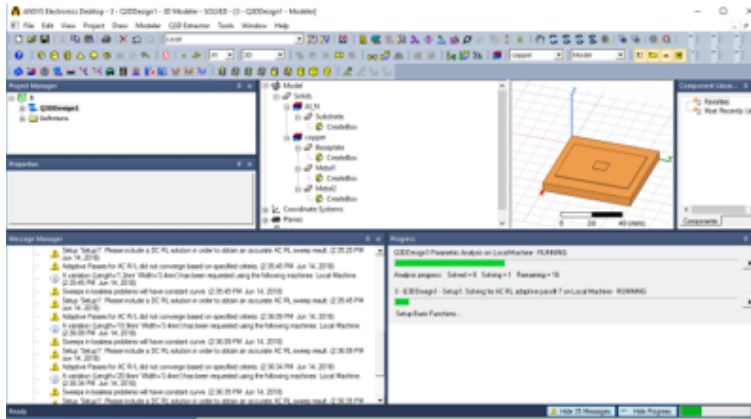
Name this loop as L1 and add this to the table.

2.5.5 Electrical Modeling

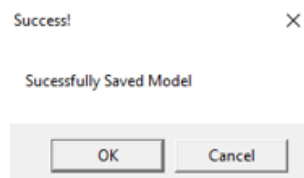
Under the electrical model selection on the UI, the analytical microstrip option can provide quick estimation of the inductance and resistance results. However, there are several drawbacks for this model such as not-frequency dependent, and usually less accurate for big traces. To overcome this problem, a response surface option is also provided. This model can be characterized using FastHenry or Q3D. To open the model creator, the user can go to: **Model -> Response Surface Setup**. A dialog as seen below is opened:



1. **Import Layer Stack:** this is explained in the Module Stack section below, this defines the baseplate and substrate dimensions. Different baseplate structures can affect the inductance calculation.
2. **Defining Maximum Trace Width and Length:** minimum trace width and length are based on minimum trace width in the Design Rule Check from PowerSynth (See Process Design Rules Editor). Define the maximum length and width here to characterize the model
3. **Defining Frequency sweep:** Specify the minimum and maximum frequency in KHz. The maximum frequency will define the minimum mesh in the characterization.
4. **Simulator:** 2 simulators are available now. These include ANSYS Q3D and FastHenry, the default ANSYS directory is: C:\Program Files\AnsysEM\AnsysEM18.2 (Versions up to 18.2 are supported). Another option is FastHenry (recommended), this is an open source parasitic extraction tool from MIT. The exe file for FastHenry is included in the package.
5. **Model name:** Model name can be specified by user. This will be later added to the model library on the right.
6. **Build:** Once everything is setup, hit “build” button. Depending on the Simulator choice, and maximum frequency, the model will be built within 15~30 minutes.

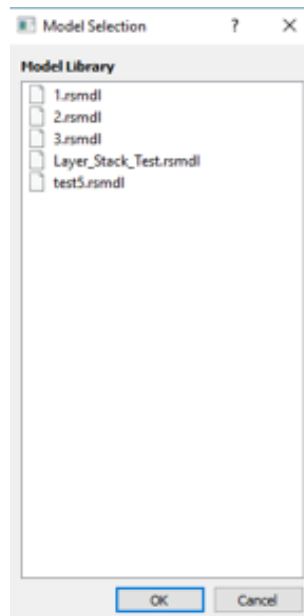


The simulation is automatically run in Q3D to characterize the model. If FastHenry simulation is selected this will be run in the back-end (Command Prompt) of the tool.



A dialog shows up when the simulations are done. Click Ok to go back to PowerSynth UI.

If the model is selected in the design performance selection, a dialog will be opened. The user can select the models from the list:



2.5.6 Capacitance

Electrical (read the instructions carefully):

1. Name your performance metric
2. Select "Electrical" as the performance measure
3. Select "capacitance"
4. Capacitance are calculated using double plate equations. So this won't affect the result
5. Select the traces by click on them, the result will be the sum of parasitic capacitance for all traces
6. Finally, hit "Add Performance Measure" to add the measure to the table

Step 1: Define a performance measure

total capacitance Electrical Select...

Step 2: Electrical Model: Micro Step

Thermal Model: Fast Thermal (FT)

Step 3: Select Device(s), Leads.

Step 4: Setup device states Add Performance Measure

Step 4: Setup device states Add Performance Measure

G_S1	Electrical	Inductance	X
TEMP	Thermal	Max	X
total capacitance	Electrical	Capacitance	X

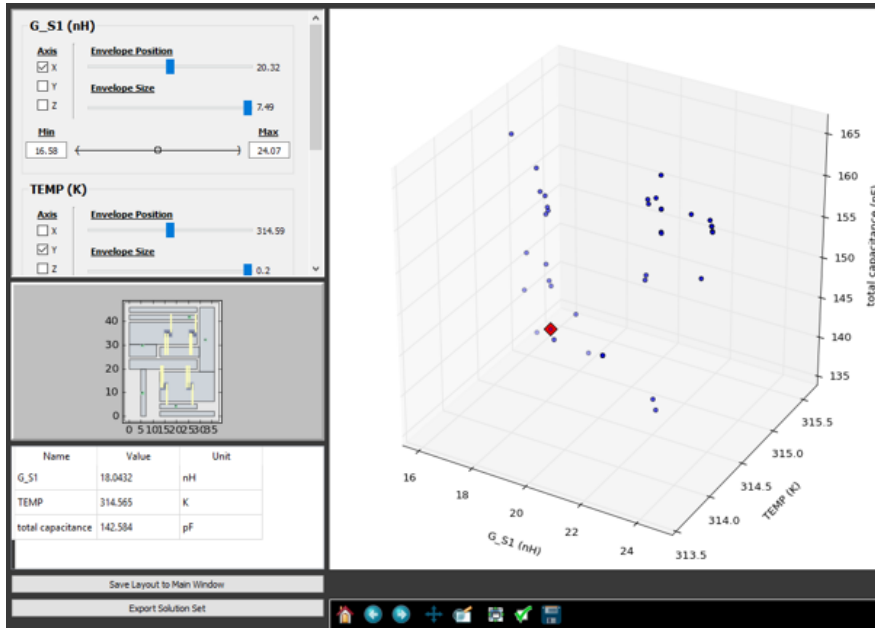
2.6 Optimization and Results

In this step, the user will need to specify number of generations. Since PowerSynth is based on a heuristic genetic algorithm, a higher number of generations will give you more solution choices. For this simple example 100 should be sufficient.

Click **Run Optimization**. The software will ask the user to save the project. Navigate to your specified project directory or any other directory to save this information. Note that the information from last run will be erased if it is not saved elsewhere. However, if the setup, number of generations, and the control seed (used for optimizer random generator) are kept the same, the final result will be the same. The runtime depends on number of generations specified. Upon completion, the solution window is opened as shown below.



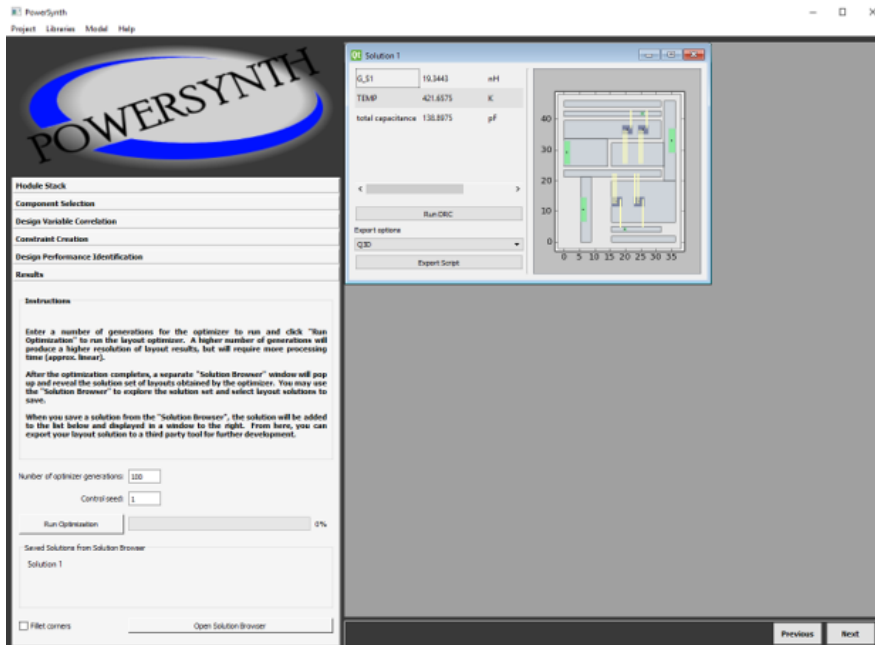
Check the console for progress updates. Once the optimizer routine is over, a new window should appear. Check your task bar if the window does not pop up. This is what the window may look like (depending on your unique solutions).



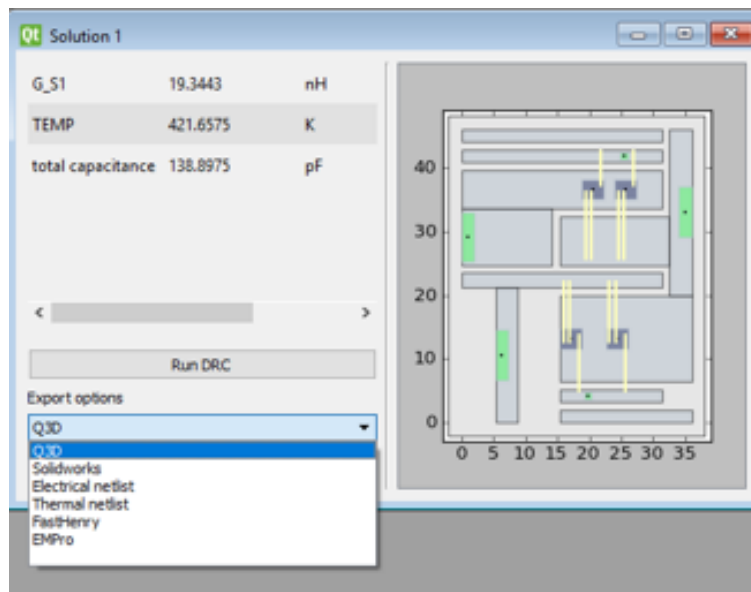
Left click and hold with your mouse inside the graph to rotate the graph. Choose the axes you wish to show using the pane on the left side (the scroll bar on the right will allow you to see more if they are not all visible). Each dot on the graph is a unique solution. Click on a solution and it will be colored red while displaying its preview on the left. Click on a different solution to see how the layout changes. The corresponding values for the objective functions are given below the layout preview.

To save the entire solution set, use the Export Solution Set button and save in .csv format. You can use Excel to open the file.

To save a solution for export into other tools or to see the E/T netlists, click on “Save Layout to Main Window.” You can enter a name for this solution. In this case it is “Solution 1.”



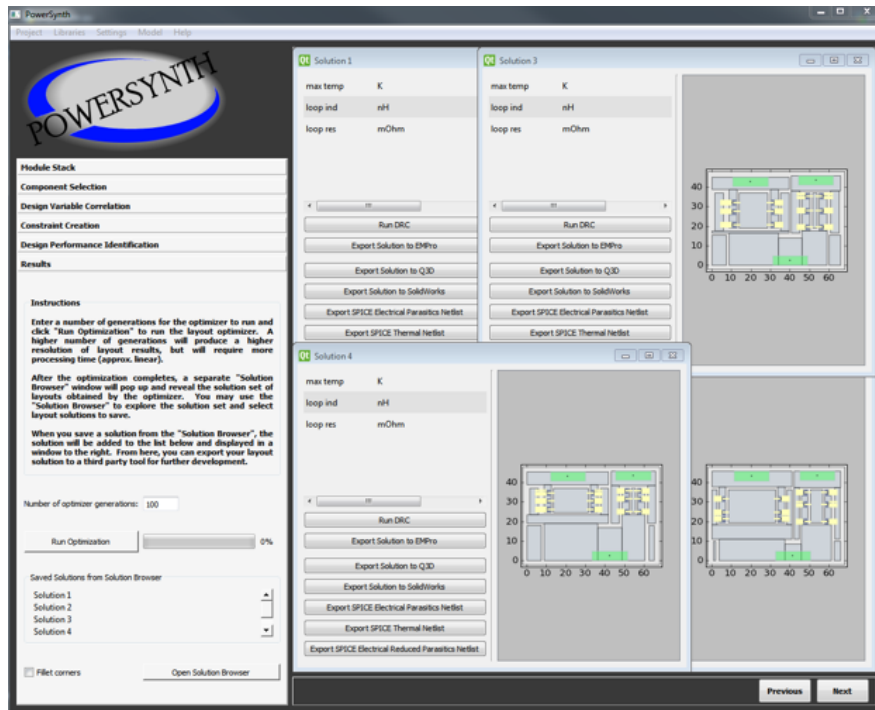
Under “Saved Solutions from Solution Browser” you can see Solution 1. The solution interface allows the user to export the solution to Q3D or SolidWorks. Other options allow the user to obtain the thermal and electrical netlists. The “Run DRC” button allow user to validate the design vs the Process Design Rule. To validate against different Process Design Rules see the section detailing the Process Design Rules Editor.



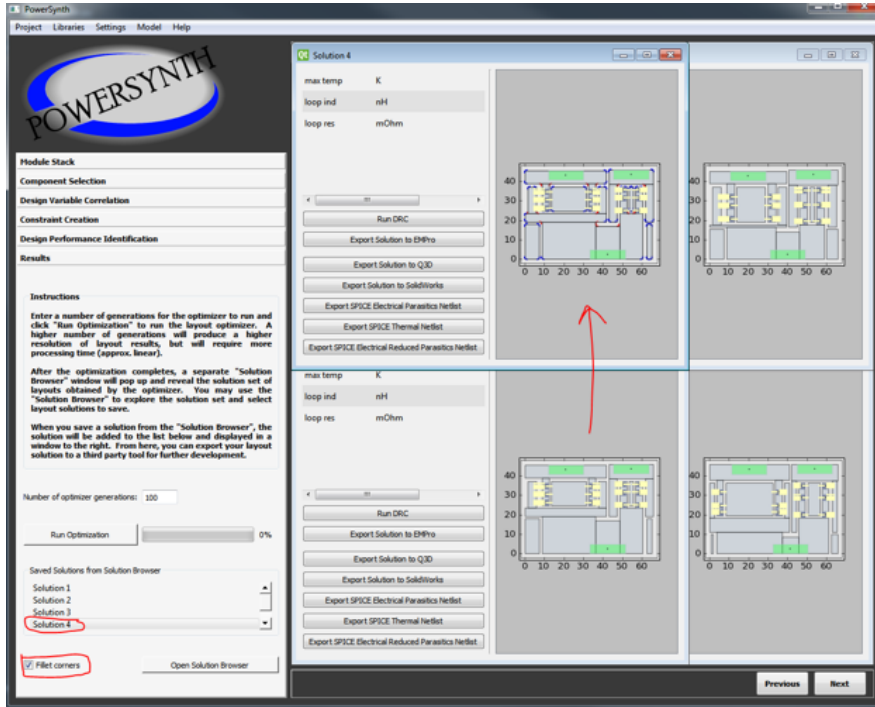
2.7 Post-Layout Optimization

PowerSynth allows the user to optionally apply corner correction to prevent electric field focusing and enhance thermo-mechanical reliability.

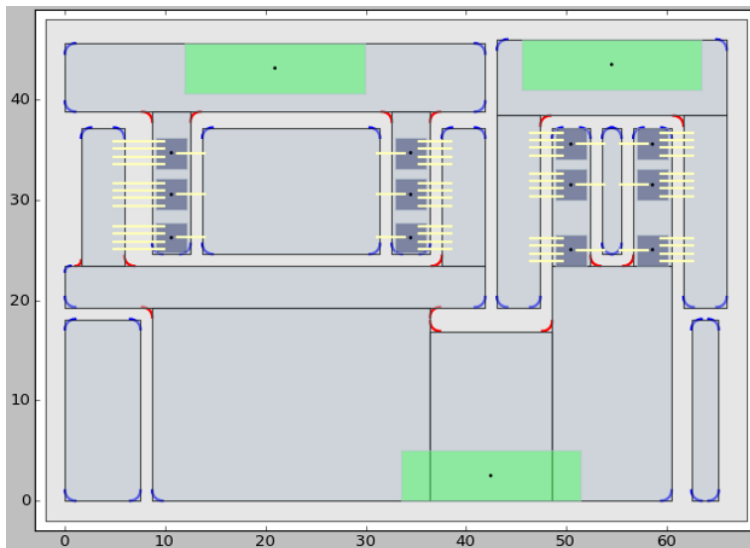
To apply fillets, in the Results tab, one may either check the 'Fillet corners' box, open the solution browser, and then select and save a solution, or alternatively, check the 'Fillet corners' box, and double click an existing solution under the 'Saved Solutions from Solution Browser' section to apply the fillets to an existing solution. For example, before applying fillets to solution 4:



And after applying fillets to solution 4, a new sub-window with the filleted solution appears:



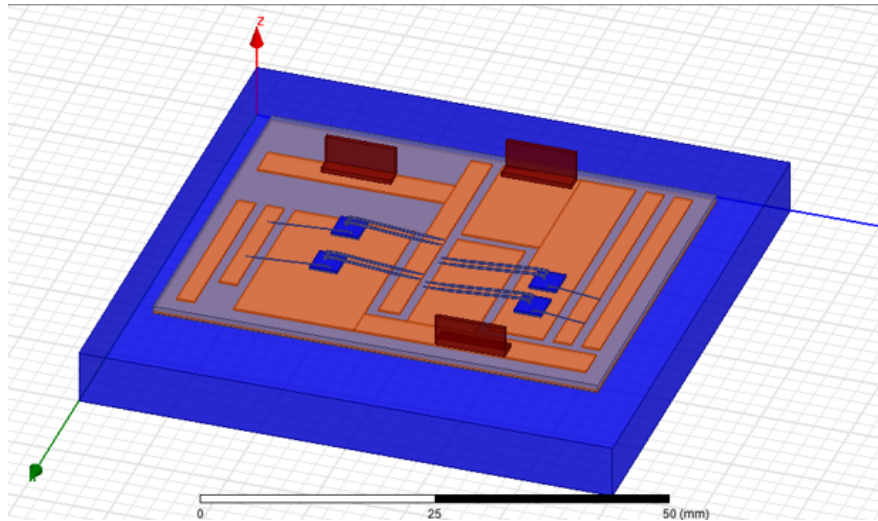
The arrow pointing up shows the same solution without fillets at the tail and with fillets at the head.



2.8 Exporting Saved Solutions

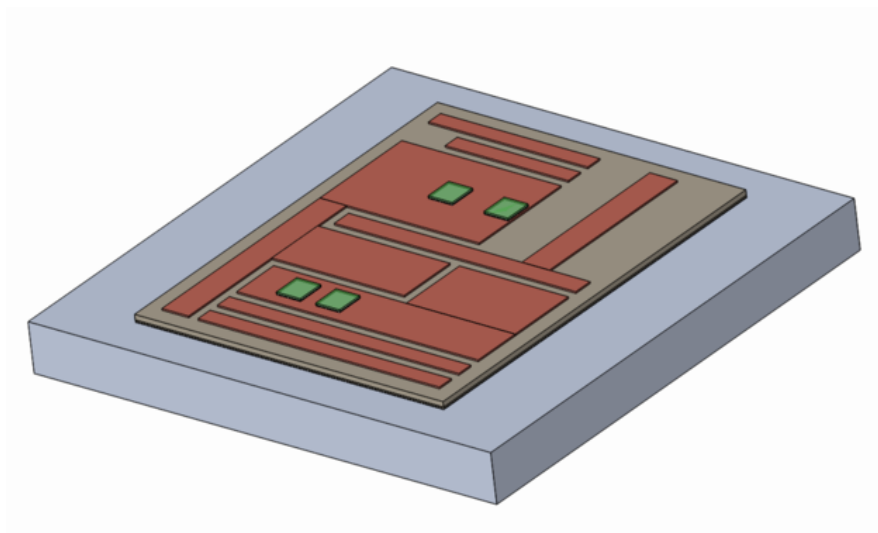
2.8.1 Export to ANSYS Q3D

Select Q3D from the export options list. Choose a directory, and create a file name. This will generate a .vbs file. In ANSYS Q3D open a new Q3D project. Go to **Tool-> Run Script** and load the exported .vbs file, the result is shown as below.



2.8.2 Export to SolidWorks

Choosing this option will open a dialog asking the user for their SolidWorks version. After typing the version (e.g. 2016) for the SolidWorks. Hit ok and choose an output directory. A*.swp file is exported. However, this is not a binary file and can be somewhat troublesome to run this script. In SolidWorks, under Tools-> Macro select “New” (do not choose edit or run since the *.swp from PowerSynth is just a text file, not binary). Copy and paste the text from the exported SolidWorks file and paste into the macro code editor. Finally press “RUN” (green triangular button).



2.8.3 Export SPICE Netlist

This will export a text file for thermal and electrical netlists. An example electrical netlist is shown below. This sub circuit netlist is in a SPICE format which is compatible with many circuit simulators. The user can use this along with any device file to do the analysis.

```
.subckt X1 M3_D M4_D M1_D M2_D G_Low M2_S DC_neg \  
          DC_plus 00out M2_G G_High M3_S M3_G M4_S M4_G M1_S M1_G  
R0 M3_D 0018 0.000128325620465  
L0 0018 M4_D 6.90843508845e-10  
C0 M4_D 0 8.36813864105e-12  
R1 M3_D 0019 6.20960837181e-05  
L1 0019 0020 2.68348440987e-10  
C1 0020 0 4.26631730709e-12  
R2 0021 0022 2.1073578894e-05  
L2 0022 0023 4.59520341299e-11  
R3 0021 0024 1e-06  
L3 0024 0025 1e-12  
R4 0021 0026 0.000395244569043  
L4 0026 0027 1.09343343808e-09  
C4 0027 0 1.08826978588e-12  
R5 0025 0028 0.0291101534176  
L5 0028 M3_G 4.33e-09  
R6 0029 0030 0.000206772456704  
...
```

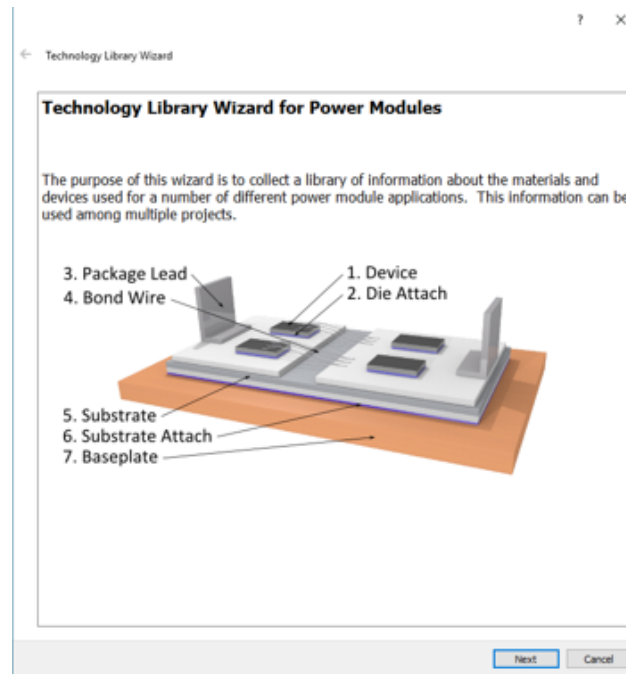
2.8.4 Export to Keysight EMPro

Please see the appendix for more detail.

3 Libraries and Editors

3.1 Technology Library Editor

Under **Libraries**, select **Tech. Lib. Editor**. This allows the user to modify different technology files. The first page of the tech lib editor is shown on the right. Click the “Next” button to go through the pages of this wizard for creating or editing components.



3.1.1 Device Information

This page allows the user to input device information. This includes the dimension and material properties as well as the landing position for bondwires. The user can also input the Verilog A file for transistor/diode, while this information is not used in the optimization process, it will be included once the user exports the electrical netlist for a selected layout.

Click the “Save” button to save the new device to the library.

← Technology Library Wizard

Device Information

Saved Devices: Device Type:

Dimensions of the Device:

Length: mm
Width: mm
Height: mm

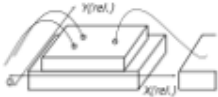
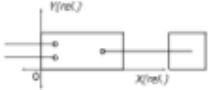
Properties of the Device:

Thermal Conductivity: W/m·K
Specific Heat Capacity: J/kg·K
Density of Material: kg/m³

Bondwire Landing Positions:

Power Signal

X1: Y1:

Save Device As:

Load Verilog-A file:

3.1.2 Add Die Attach

This page allows the user to assign the material properties for the die attach.

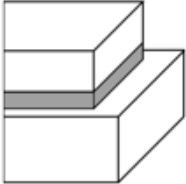
Click the “Save” button to save the new die attach file to the library.

Add Die Attach

Saved Die Attach:

Properties of the Attach:

Thermal Conductivity: W/m·K
Specific Heat Capacity: J/kg·K
Density of Material: kg/m³



Save Attach As:

3.1.3 Add Lead


This page allows the user to assign the material properties and dimensions for different lead types. In PowerSynth, this currently falls into two categories: “Round” and “Bus Bar leads.”

Click the “Save” button to save the new lead file to the library.

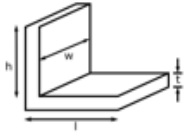
Add Lead

Saved Leads:

Type of lead: Power Signal
Shape of lead: Round Bus Bar



Diameter mm
Height mm



Length mm
Width mm
Height mm
Thickness mm

Electrical Resistivity (ρ) $\Omega \cdot m$

Save Lead as:

3.1.4 Add BondWire


This page allows the user to input dimensions and material properties for different bondwire types. The bondwire model is based on JDEC standards.

Click the “Save” button to save the new bondwire file to the library.


Add Bondwire

Saved Bondwires:

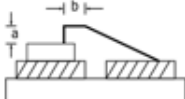
Type of bondwire: Power Signal
Shape of bondwire: Round Ribbon



Diameter mm



Width mm
Height mm



Height of bondwire from the die (a) mm
Fraction of bondwire total length (b) mm

Electrical Resistivity (ρ) $\Omega \cdot m$

Save Bondwire as:

3.1.5 Add Substrate

This page allows the user to input dimensions and material properties for the metal and isolation layers of DBC or DBA substrates.

Click the “Save” button to save the new file to the library.

Add Substrate

Saved Substrates:

Dimensions of the Substrate:

Dielectric Thickness mm

Metal Thickness mm

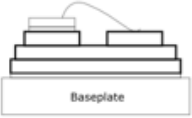
Metal Layer Properties:

Thermal Conductivity W/m K

Specific Heat Capacity J/kg K

Electrical Resistivity Ω m

Density of Material kg/m³



Isolation Layer Properties:

Thermal Conductivity W/m K

Specific Heat Capacity J/kg K

Density of Material kg/m³

Relative Permittivity (ϵ_r)

Relative Permeability (μ_r)

Save Substrate as:

3.1.6 Add Substrate Attach

This allows the user to assign the material properties for the substrate attach.

Click the “Save” button to save the new substrate attach file to the library.

Add Substrate Attach

Saved Substrate Attaches:

Properties of the Attach:

Thermal Conductivity W/m K

Specific Heat Capacity J/kg K

Density of Material kg/m³



Save Substrate Attach as:

3.1.7 Add Baseplate

This allows the user to assign the material properties and dimensions for the baseplate.

Click the “Save” button to save the new baseplate file to the library.

Add Baseplate

Saved Baseplates:

Properties of the Baseplate:

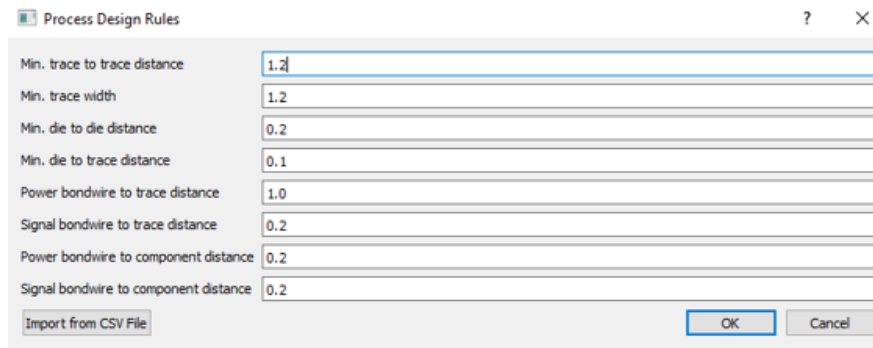
Thermal Conductivity	<input type="text"/>	W/m K
Specific Heat Capacity	<input type="text"/>	J/kg K
Density of Material	<input type="text"/>	kg/m ³
Electrical Resistivity	<input type="text"/>	$\Omega \cdot m$
Relative Permeability (μ_r)	<input type="text"/>	



Save Baseplate as:

3.2 Process Design Rules Editor

The process design rules editor can be opened from **Project -> Design Rule Editors**. The interface below allows the user to input some design rule constraints for the layout. In the optimization process, candidate layouts that violate these rules are deemed as infeasible and will be eliminated from the solution space.



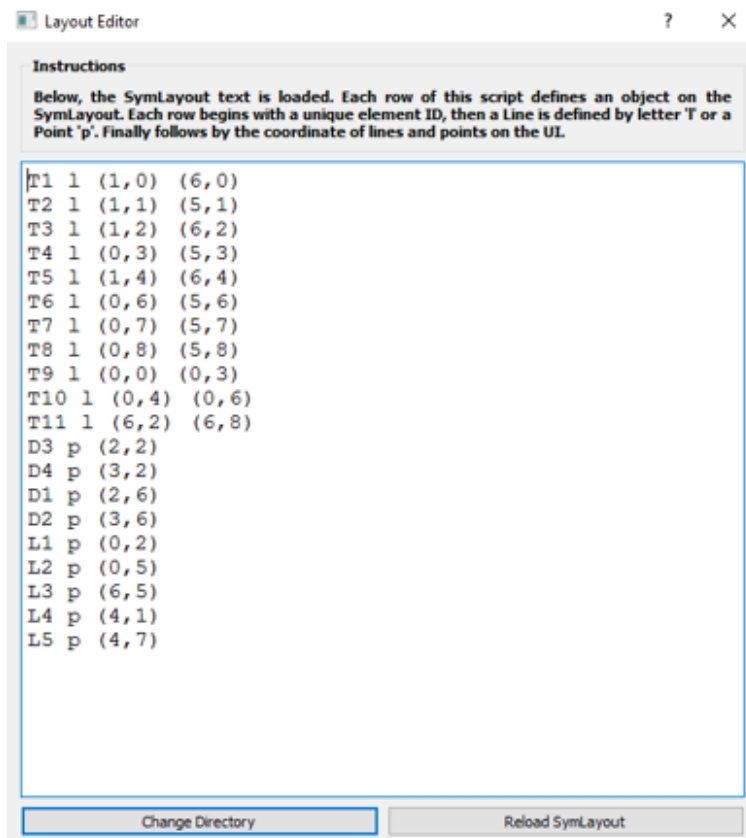
The screenshot shows a dialog box titled "Process Design Rules" with a question mark and a close button (X) in the top right corner. The dialog contains several input fields for design rule constraints:

Min. trace to trace distance	1.2
Min. trace width	1.2
Min. die to die distance	0.2
Min. die to trace distance	0.1
Power bondwire to trace distance	1.0
Signal bondwire to trace distance	0.2
Power bondwire to component distance	0.2
Signal bondwire to component distance	0.2

At the bottom left, there is a button labeled "Import from CSV File". At the bottom right, there are two buttons: "OK" and "Cancel".

3.3 Layout Editor

The layout editor below allows the user to modify the symbolic layout. This can be found under **Project-> Open Layout Editor**. After the script is modified, click “Reload SymLayout” to load the new layout to the workspace.



4 PowerSynth-Related Publications

This section contains a list of all the current publications related to PowerSynth as of the date of this document.

1. P. Tucker, "SPICE netlist generation for electrical parasitic modeling of multi-chip power module designs," 2013.
2. B. W. Shook, A. Nizam, Z. Gong, A. M. Francis and H. A. Mantooth, "Multi-objective layout optimization for multi-chip power modules considering electrical parasitics and thermal performance," in *Control and Modeling for Power Electronics (COMPEL), 2013 IEEE 14th Workshop on*, 2013.
3. B. W. Shook, Z. Gong, Y. Feng, A. M. Francis and H. A. Mantooth, "Multi-chip power module fast thermal modeling for layout optimization," *Computer-Aided Design and Applications*, vol. 9, pp. 837-846, 2012.
4. B. W. Shook, "The Design and Implementation of a Multi-Chip Power Module Layout Synthesis Tool," 2014.
5. J. Main, "A Manufacturer Design Kit for Multi-Chip Power Module Layout Synthesis," 2017.
6. Q. Le, T. Evans, S. Mukherjee, Y. Peng, T. Vrotsos and H. A. Mantooth, "Response surface modeling for parasitic extraction for multi-objective optimization of multi-chip power modules (MCPMs)," in *Wide Bandgap Power Devices and Applications (WiPDA), 2017 IEEE 5th Workshop on*, 2017.
7. Q. Le, S. Mukherjee, T. Vrotsos and H. A. Mantooth, "Fast transient thermal and power dissipation modeling for multi-chip power modules: A preliminary assessment of different electro-thermal evaluation methods," in *Control and Modeling for Power Electronics (COMPEL), 2016 IEEE 17th Workshop on*, 2016.
8. Z. Gong, "Thermal and electrical parasitic modeling for multi-chip power module layout synthesis," 2012.
9. T. Evans, Q. Le, S. Mukherjee, I. Al-Razi, T. Vrotsos, Y. Peng and H. A. Mantooth, "PowerSynth: A Module Layout Generation Tool," Manuscript submitted to *IEEE Transactions on Power Electronics* April 2018.
10. S. Mukherjee et al, "Toward Partial Discharge Reduction by Corner Correction in Power Module Layouts," in *Control and Modeling for Power Electronics (COMPEL)*, 2018.

5 Appendix 1: EMPro Export

5.1 Introduction

The EMPro Export feature in PowerSynth allows a user to export designs generated by PowerSynth for further analysis in Keysight's EMPro and ADS. This document presents an overview of the capabilities available after export as well as the process in doing so. Furthermore, while this document provides a walkthrough of some of the key points for usage, it is assumed that the user is familiar with Keysight EMPro and ADS.

5.1.1 Functionality

This feature of PowerSynth takes a layout design from PowerSynth and generates a script to be run in EMPro. This script automatically defines the simulation type and frequency range, draws all of the design components and assigns materials, and sets up all of the requisite ports for simulation.

Exporting a design generated by PowerSynth to EMPro grants the user several capabilities for further analysis of a given design. These include, but are not limited to:

- S-Parameter modeling of a design
- Near and far field visualization
- Export to ADS for:
 - Incorporation of Verilog-A device models
 - Transient switching analysis
 - Conducted EMI analysis
 - Near field visualization of transient results

5.1.2 Caveats and Limitations in Current Release

Currently PowerSynth assumes a module buildup consisting of:

- Heat spreader
- Substrate attach
- DBC-type substrate (Cu-ceramic-Cu)
- Die attach
- Device
- Wire bonds

This stackup is fixed in the current release and so the resulting model in EMPro reflects this. However, future versions of the tool will allow greater flexibility.

During the export, the thin attachment layers are neglected to allow for easier meshing. Additionally, devices are not physically modeled. Instead, pads corresponding to those of the devices are inserted and wire bonds are connected to them. This is done so that only the parasitics of the module are extracted. Later, in circuit simulators such as ADS, the user can insert a device model of their choosing into the corresponding ports inside the extracted layout parasitics model.

Table 2: Component Material Assignments

Component	Material
Heat Spreader	Aluminum
Traces	Copper
Ceramic	Alumina
Device Pads	PEC
Wire Bonds	Aluminum

Furthermore, materials are automatically assigned to components as shown below, but may be easily changed within EMPro.

As noted above, many of these issues related to the current limitations and assumptions in PowerSynth are currently being revised. As of this writing, a new layout engine is being developed to overcome them and provide even greater flexibility in design. The reader's patience is greatly appreciated as development of PowerSynth continues.

Finally, this feature and document were made using the 2017 versions of EMPro and ADS. Future version compatibility will be ensured. However, compatibility with previous versions has not been tested nor can it be guaranteed.

5.2 Exporting from PowerSynth

Once a design has been selected from PowerSynth, exporting the EMPro script is a straight forward task. This section will cover the steps necessary for exporting only and assumes that PowerSynth has been setup, run, and the user has been presented with design solutions. Please refer to the main PowerSynth manual for information on how to perform these steps.

5.2.1 Selecting a Design

After PowerSynth completes the optimization routine, the user is presented with a solution as shown in Figure 1. From here, select a design and click on the button "Save Layout to Main Window" then return to the main PowerSynth window.

5.2.2 Saving the EMPro Script

In the main window, find the sub-window containing the design that is to be exported. From there, click on the button titled "Export Solution to EMPro" as shown in Figure 2. You will then be presented with a dialog box asking you to specify a name and location for the python script which can be used in EMPro like in Figure 3. The next section will cover how to import the file into EMPro.

5.3 Importing to EMPro

Upon saving EMPro script, the file can be opened with EMPro to setup the model and simulation. The following steps provide more detail on these tasks.

5.3.1 Creating a New Project

First, open EMPro. Then select "New Project" from the file menu. Here, choose "Generic FEM, design in mm" as shown in Figure 4. Set the frequency ranges to be simulated and press "OK."

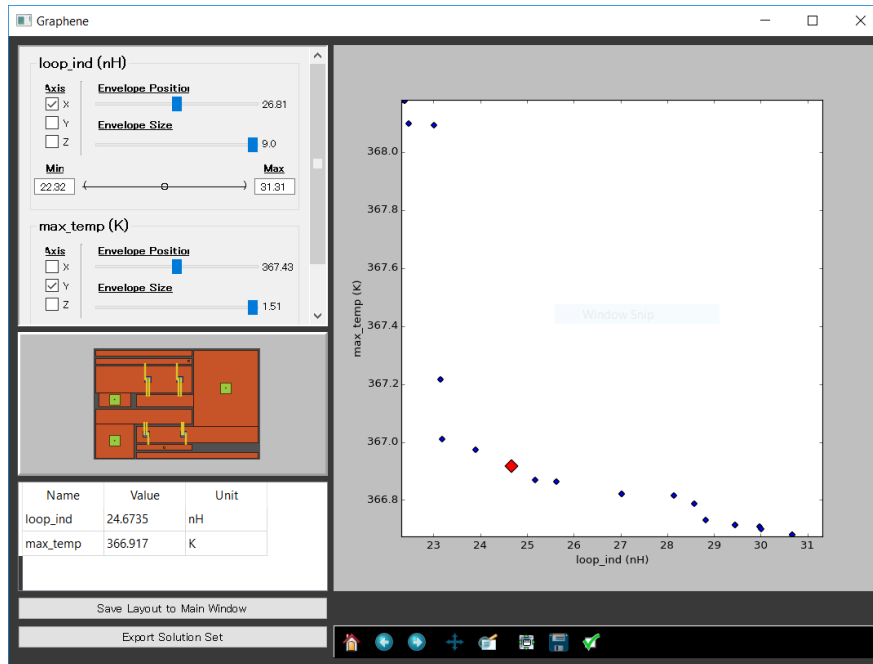


Figure 1: Selecting a design from the solution browser window and saving to the main PowerSynth one.

5.3.2 Running the Script

Next, open the scripting pane in EMPro. From here choose to create a new script from the file menu as in Figure 5. Open the script in an external text editor and copy the entire code, then paste it into the new script window in EMPro that was just created. Finally, press the execute script button (Figure 6) and wait until it completes.

Once the script has finished running, you will see the components fully modeled and populated in the main window like in Figure 7. Now the project is ready for simulation.

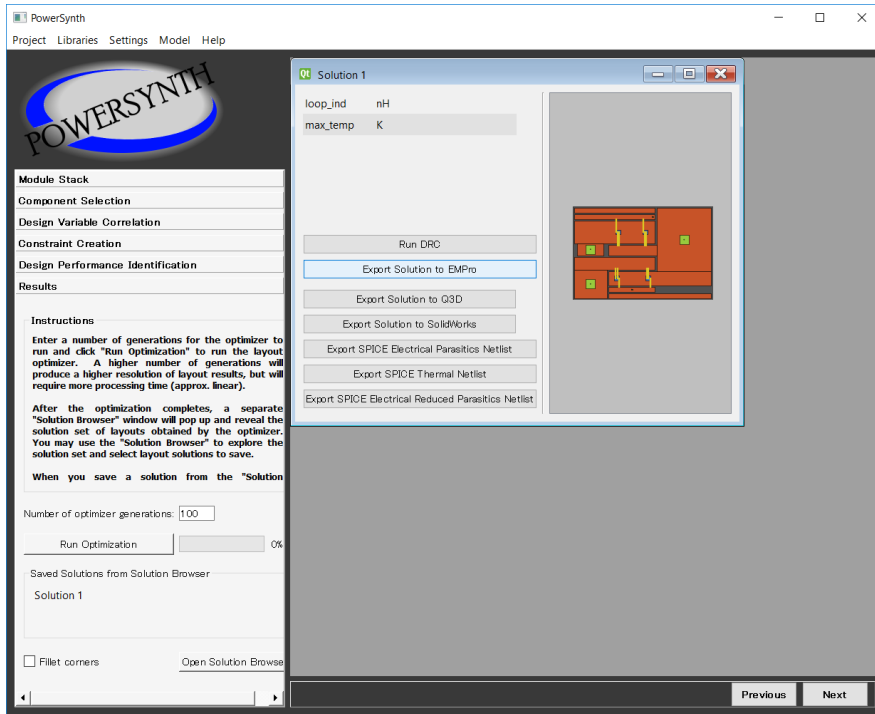


Figure 2: Exporting the design to an EMPro Python script.

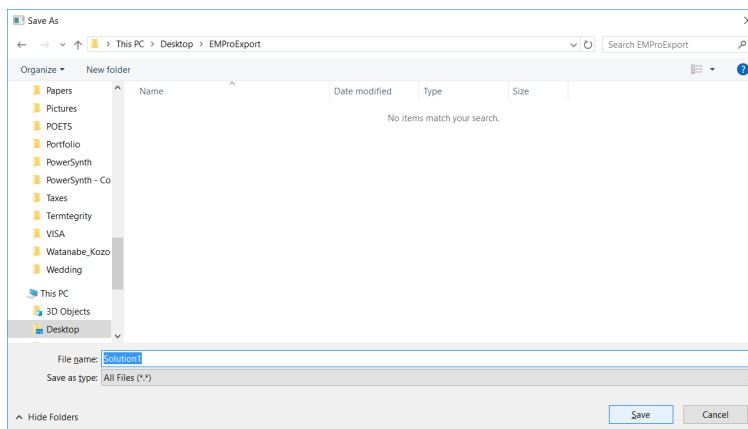


Figure 3: Specifying the file name for export to EMPro.

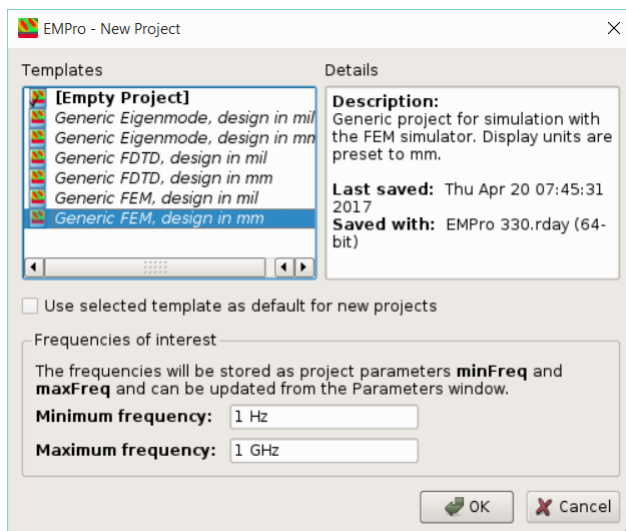


Figure 4: Creating a new EMPro Project.

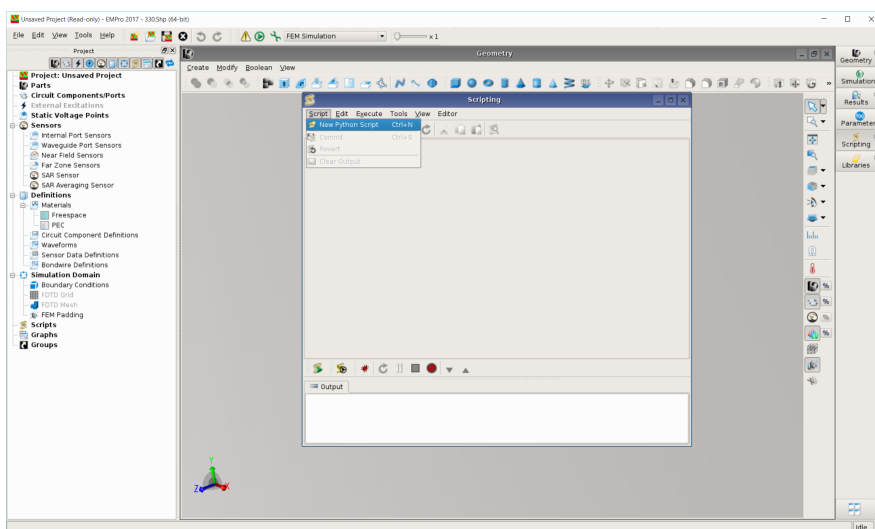


Figure 5: Creating a new script.

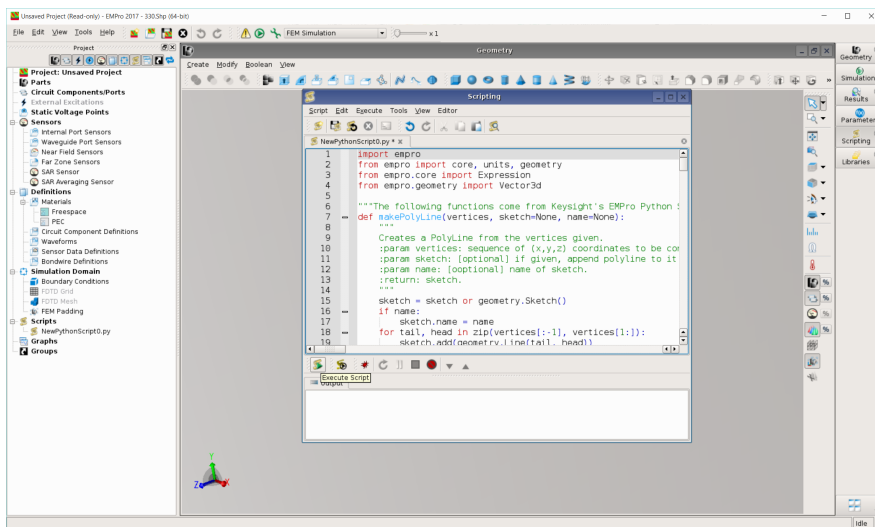


Figure 6: Executing the script.

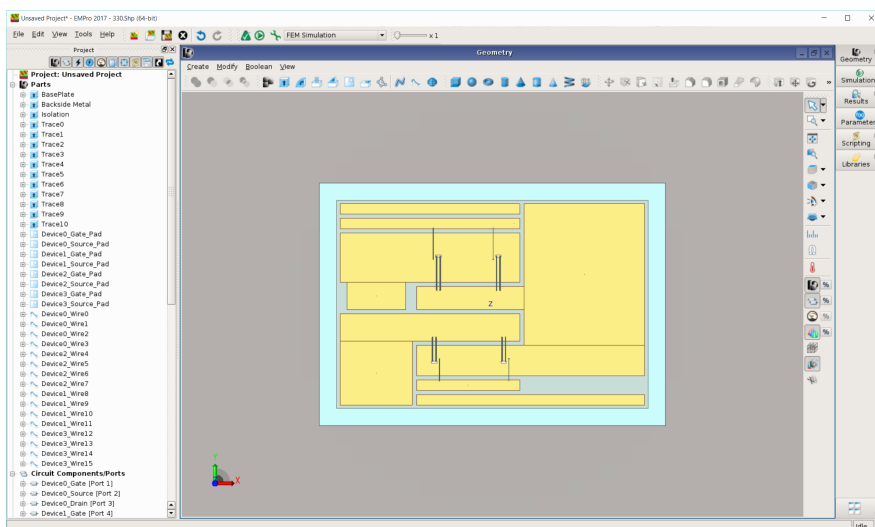


Figure 7: EMPro project creation completed.