cādence®

Allegro PCB Editor Flow – Initial PCB Design Creation

Rapid Adoption Kit (RAK)

Product Version 17.2
October 2018

Copyright Statement

Contents

Purpose

This RAK outlines the initial steps used to create a PCB, including padstack generation, symbol generation, design parameter import, technology file import, netlist import, and cross-section creation.

Audience

This document is intended for PCB designers, hardware designers, and engineers who use Allegro PCB Editor to create PCBs.

Download

RAK testcase database, Scripts and References can be found at 'Attachments' and 'Related Solutions' sections below the PDF.
This RAK pdf can be searched with the document 'Title' on https://support.cadence.com

Terms

CSet   Constraint Set

DE HDL Allegro Design Entry HDL

Module 1: Allegro PCB Editor Overview

Start Page

A Start Page may appear when you invoke Allegro PCB Editor. If Allegro is opened without a design specified, the Start Page will be displayed.
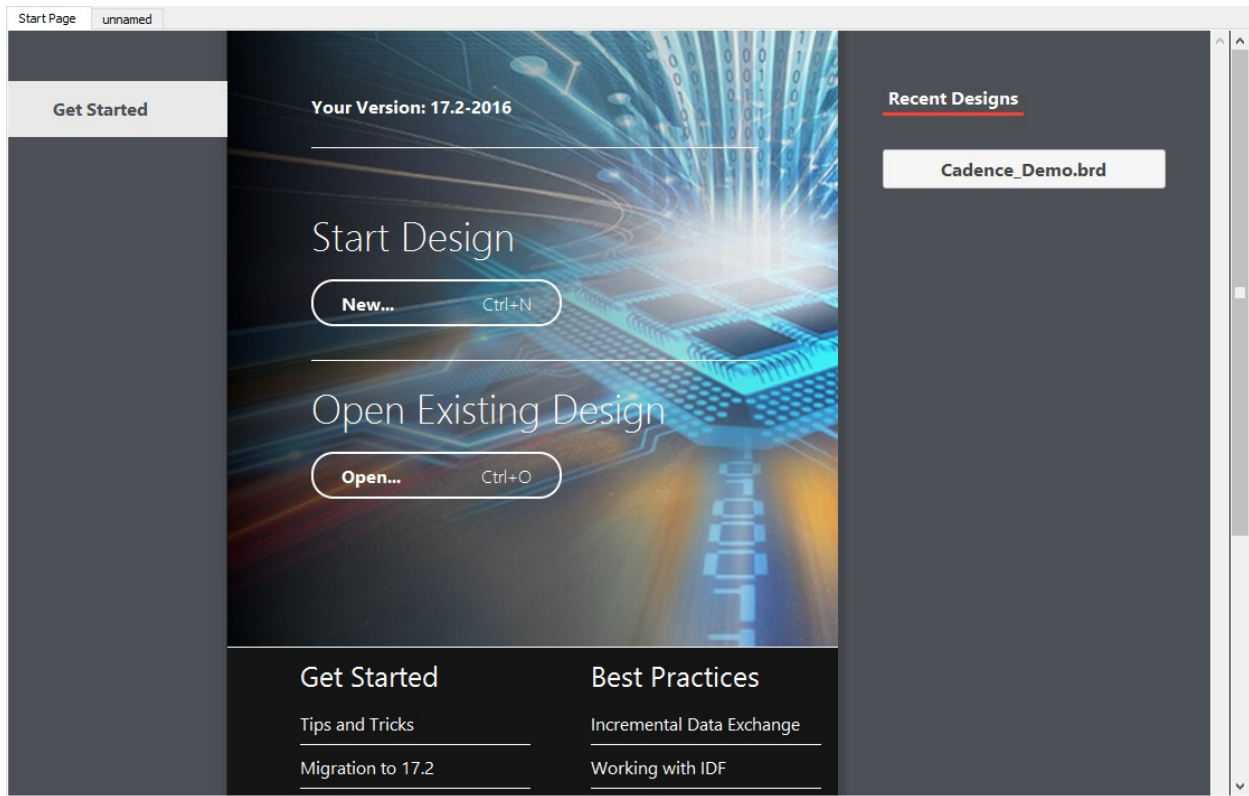
If Allegro is opened with a design, the design is displayed in the canvas and the Start Page appears as a tab.
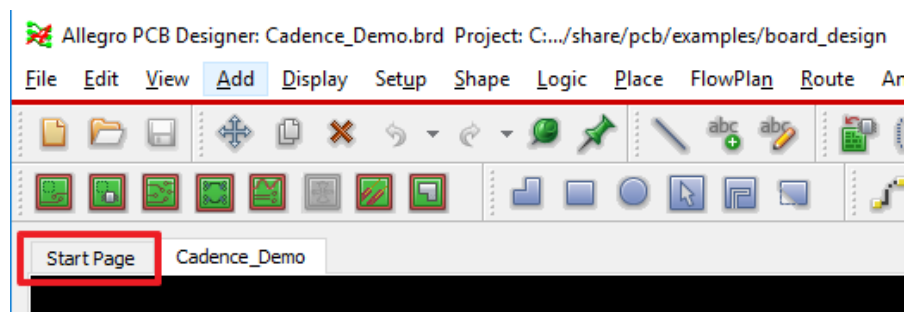
The Start Page provides easy access to best practice papers and migration information, as well as access to recently opened designs.

Design Workflow Pane
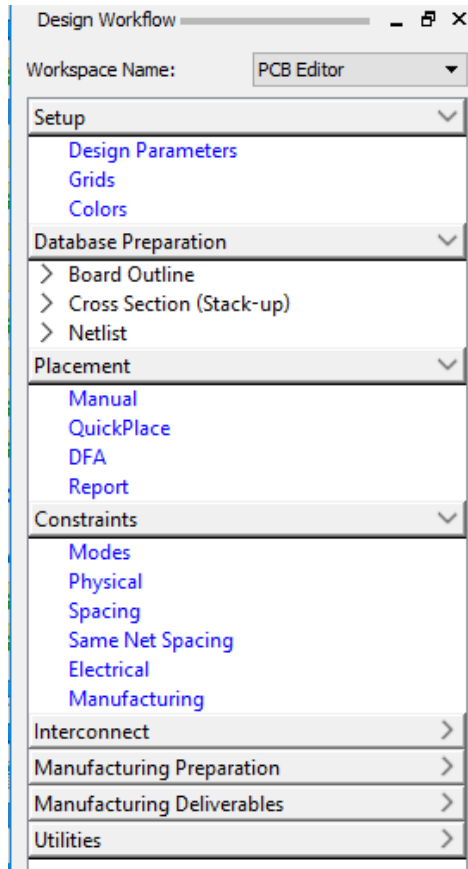
The **Design Workflow** pane guides users in performing the basic tasks of board design using Allegro PCB Editor. The steps in the workflow eliminate the need to search the canvas for icons or menus to perform tasks. Selecting any choice will bring up the proper dialog.

Workflow files are XML files that are customizable. The default workflow files are available at:

<installation_directory>/share/pcb/text/workflows

Using Technology and Parameter Files

Both technology and parameter files are essential components in the process of leveraging reusable design information during the database creation stage of the design flow. Technology files (techfiles) are used to enter cross-section, drawing, and constraint settings into the database, while parameter files are used to enter settings for global and application-based functions.

Working with Technology Files

Techfiles contain the following types of neutral design data:

- Drawing parameters (including units and design extents)
- Layout cross section
- DRC modes
- Spacing, physical, electrical, manufacturing, and design constraints
- User property definitions

Techfiles can be exported from either Allegro PCB Editor (**File > Export > Techfile**) or from Constraint Manger (**File > Export > Technology File**). Exporting from Constraint Manager provides more control over what is exported.
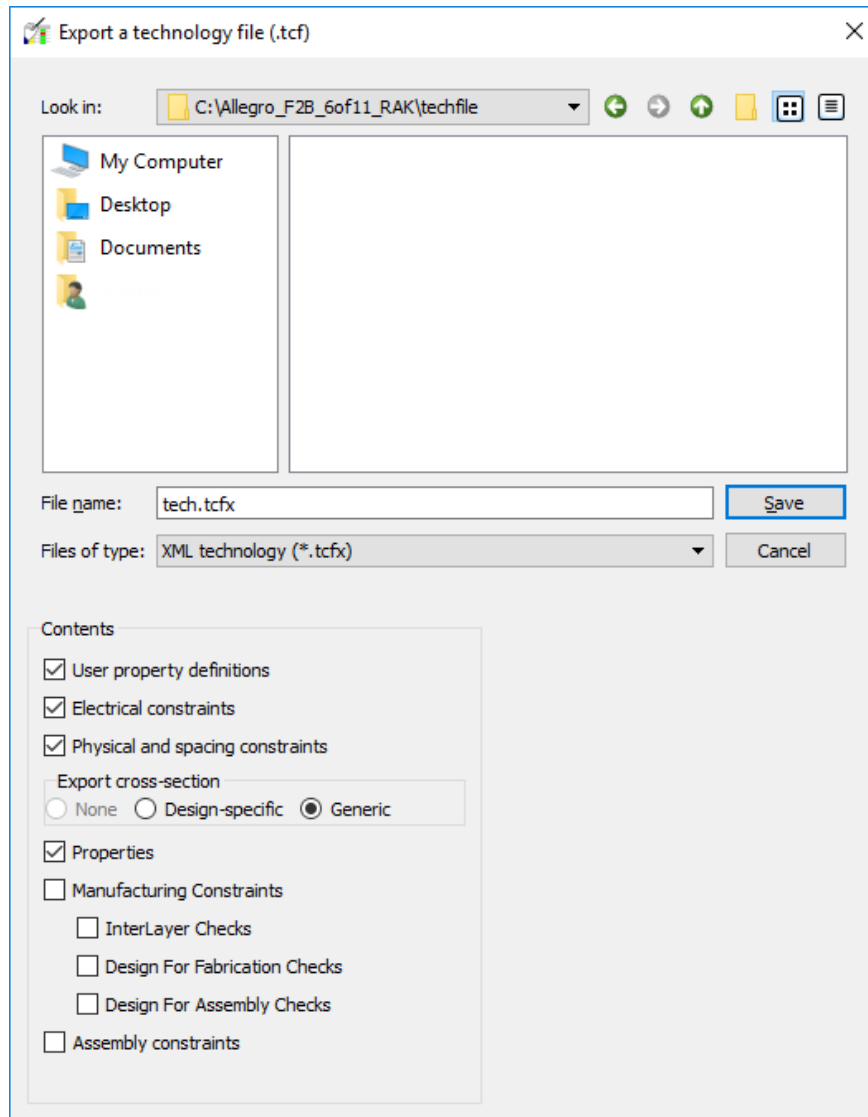
Working with Parameter Files

Design Parameter Editor (**Setup > Design Parameters**) provides a convenient, centralized location for editing parameters saved and stored in the database, such as:

- Display parameters, including net name display, DRC marker size, and grid display and setup
- Drawing parameters, including drawing extents, origin, type, and size; database accuracy; and user units
- Text size
- Dynamic and static shape parameters
- Route parameters for Add Connect, Delay Tune, Slide, Gloss, and Auto-Interactive Tuning
- Manufacturing applications, including Testprep, Thieving, Silkscreen, and Drafting
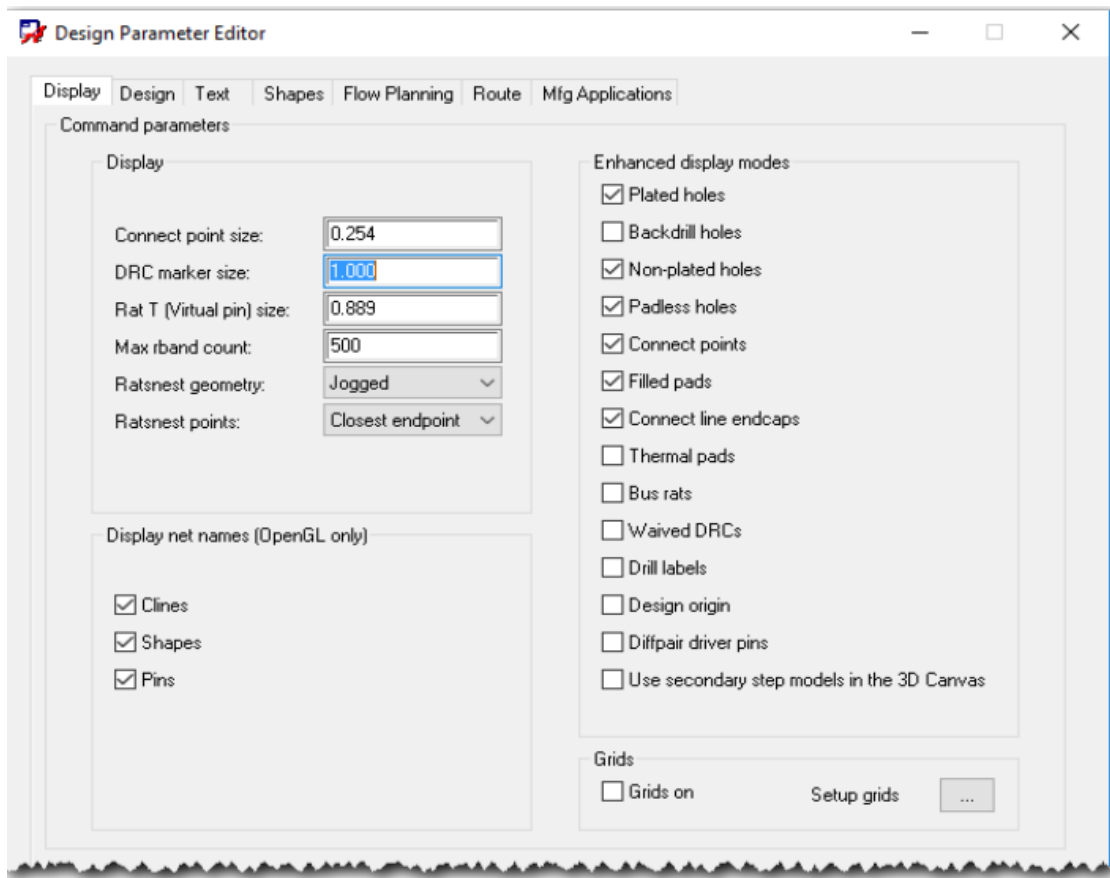

Customized parameter settings from a design can be exported to a database parameter file (.prm) using the **File > Export > Parameters** command. This parameter file can then be imported into a new design using **File > Import > Parameters**.

Module 2: Pad Editor

Pad Editor is used to generate multiple types of pads that will be used in a design. The Pad Editor user interface includes tabs that are flow-designed, beginning on the left with the Start tab, and progressing to the right. The tabs are "smart-enabled", meaning that only the tabs applicable for the type of pad being created will be available.
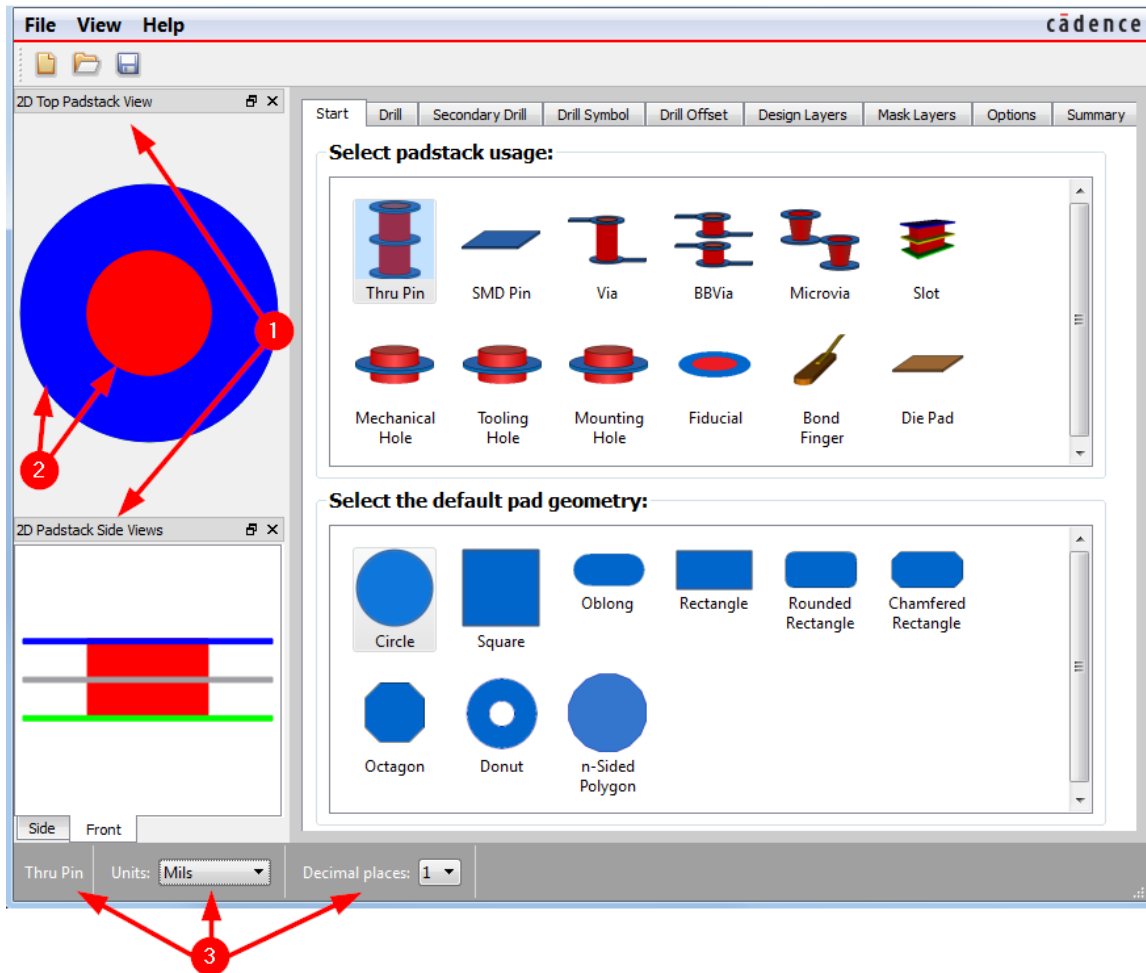

There are two 2D views located on the left side of the Pad Editor user interface (1) – 2D Top Padstack View and 2D Padstack Side Views. Two tabs provide a graphical view from the side and front of the padstack definition.

The Top view (2) shows a two-dimensional graphical representation of the drill and pad layers of the padstack and the relationship between the hole, if defined, and pad geometries.

The Pad Type Status bar at the lower left (3) indicates the padstack type that is currently defined, as well as the units and accuracy.


To begin, use the Start tab to select the padstack type. The padstack type is an attribute used when exporting IPC-2581 data for manufacturing. Based upon the padstack type selected, other tabs may be disabled to prevent an incorrect padstack definition.
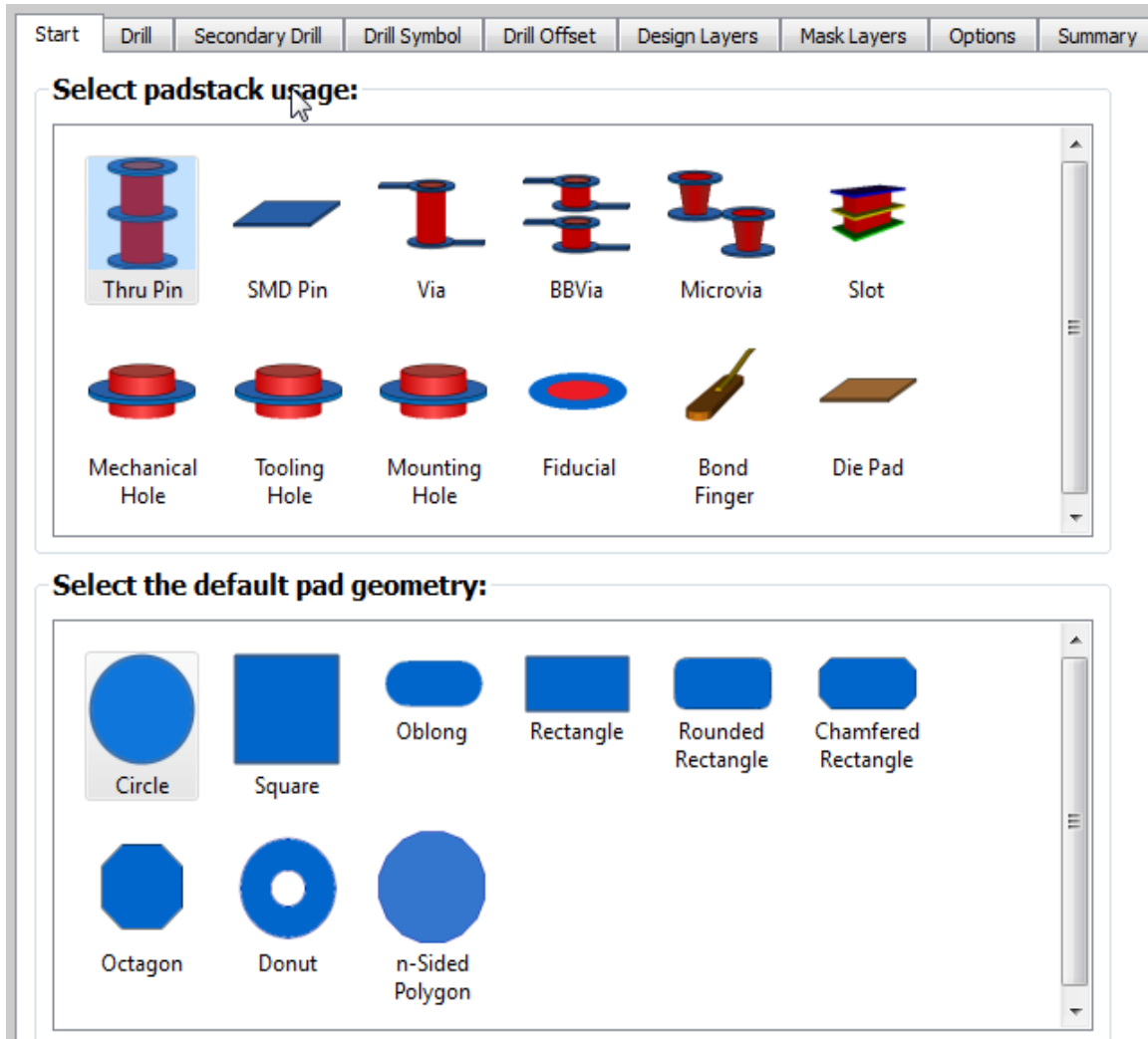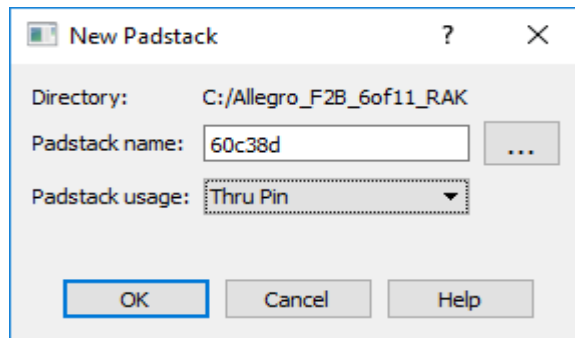
Lab 1: Creating a Thru Hole Padstack

This lab will demonstrate how to create a thru hole padstack using Pad Editor.

1. Open Pad Editor and select **File > New**.
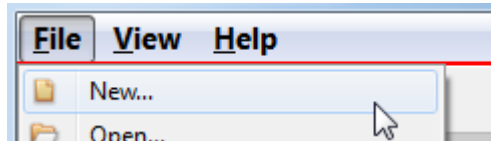
2. Name the padstack **60c38d**, select **Thru Pin** for the **Padstack usage**, and select **OK**.

New attributes are assigned to the padstack based on the intended usage type. The usage type attribute is used internally in the database and is also passed as a label to exported formats, such as IPC-2581.

3. In the status area of Pad Editor, set the **Units** to **Mils** and the **Decimal places** to **2**.
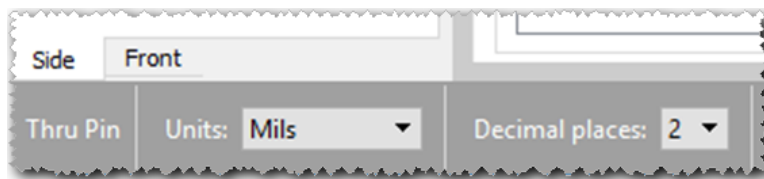
Defining NCDrill Requirements

1. Select the **Drill** tab.

2. Fill out the **Drill Hole** section of the form as follows:

Hole type	Circle
Finished Diameter	38
Tolerance	+0, -0

3. In the **Hole plating** section, set the **Hole/slot plating** to **Plated**.

4. Leave the default values for **Define the drill rows and columns**.

5. Select the **Drill Symbol** tab.

6. Fill out the **Define a drill symbols** section of the form as follows:

Type of drill figure	Circle
Characters	A
Drill figure diameter	60

Defining BEGIN LAYER Pad

1. Select the **Design Layers** tab.

2. Left-click in the **Regular Pad** column for the BEGIN LAYER.

3. Select **Circle** from the **Geometry** pulldown menu and enter a **Diameter** of **60**.
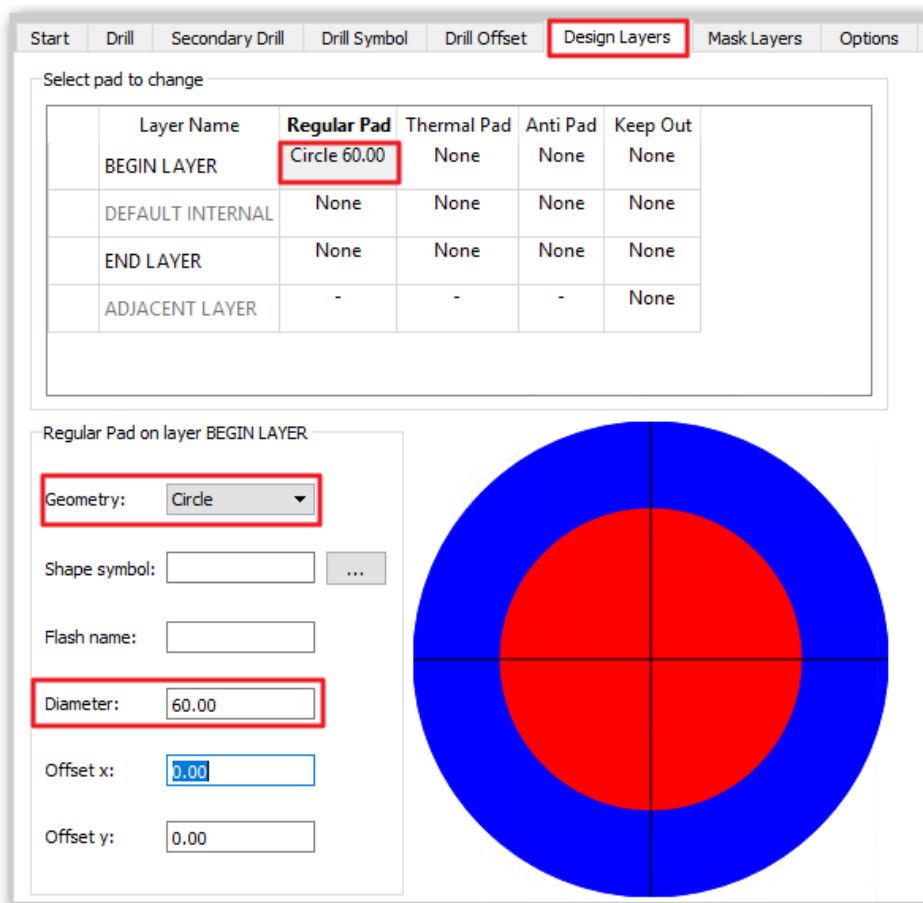
4. Click the **BEGIN LAYER > Thermal Pad** cell.

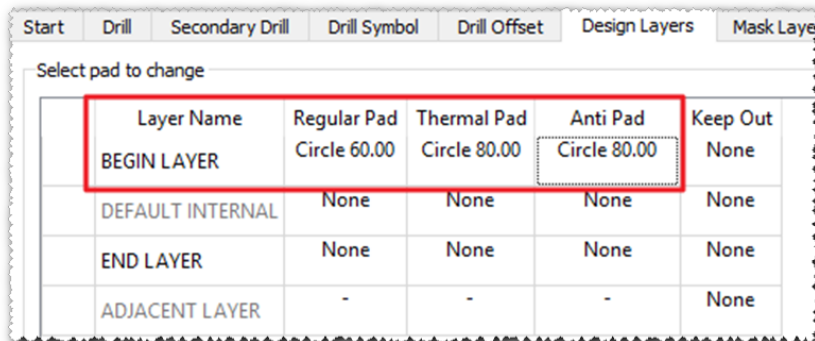5. Set the following for Thermal Pad:

   Geometry   Circle
   Diameter   80

6. Click the **BEGIN LAYER > Anti Pad** cell.

7. Set the following for Anti Pad:

   Geometry   Circle
   Diameter   80

The BEGIN LAYER is now defined.

Defining DEFAULT INTERNAL and END LAYER Pads

Because the DEFAULT INTERNAL and END LAYER pads are generally the same size and shape as the BEGIN LAYER, the Copy command can be used to save time.

1. To copy the BEGIN LAYER pad definition, click in the Regular Pad cell and drag across to select the Thermal Pad and Anti Pad cells.

2. Right-click on one of the selected cells and select **Copy**.
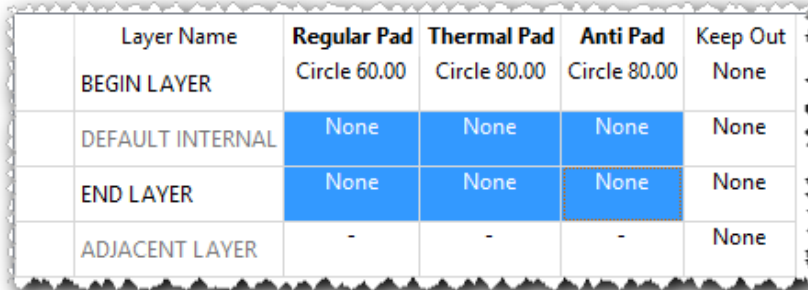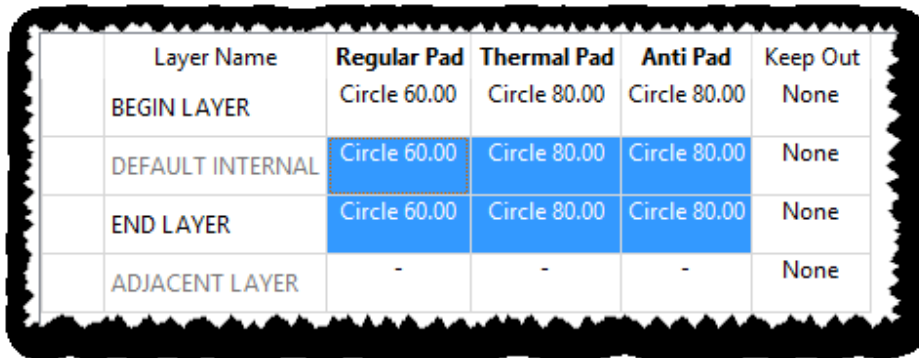
3. Select all Regular Pad, Thermal Pad, and Anti Pad cells for the DEFAULT INTERNAL and END LAYER cells.

Layer Name	Regular Pad Circle 60.00	Thermal Pad Circle 80.00	Anti Pad Circle 80.00	Keep Out None
BEGIN LAYER				
DEFAULT INTERNAL	None	None	None	None
END LAYER	None	None	None	None
ADJACENT LAYER	-	-	-	None

4. Right-click on one of the selected cells and select **Paste**.

Layer Name	Regular Pad Circle 60.00	Thermal Pad Circle 80.00	Anti Pad Circle 80.00	Keep Out None
BEGIN LAYER				
DEFAULT INTERNAL	Circle 60.00	Circle 80.00	Circle 80.00	None
END LAYER	Circle 60.00	Circle 80.00	Circle 80.00	None
ADJACENT LAYER	-	-	-	None

Defining Thermal Flash

1. Click on the **Thermal Pad** cell for the DEFAULT INTERNAL layer.

2. In the **Thermal Pad on layer DEFAULT INTERNAL** section of the form, click the **Browse** button next to the **Flash symbol** field.
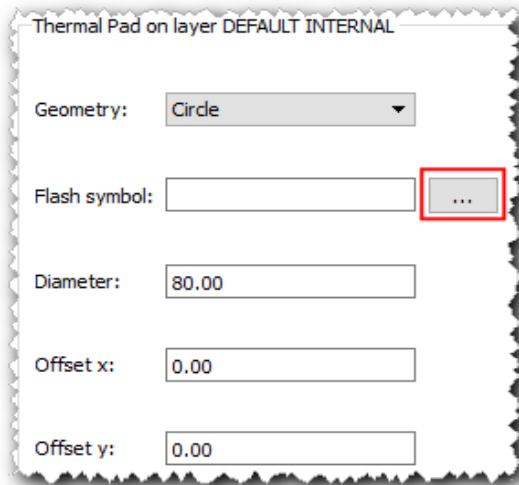
3. Select **Create New Flash Symbol** in **Library Shape Symbol Browser**.
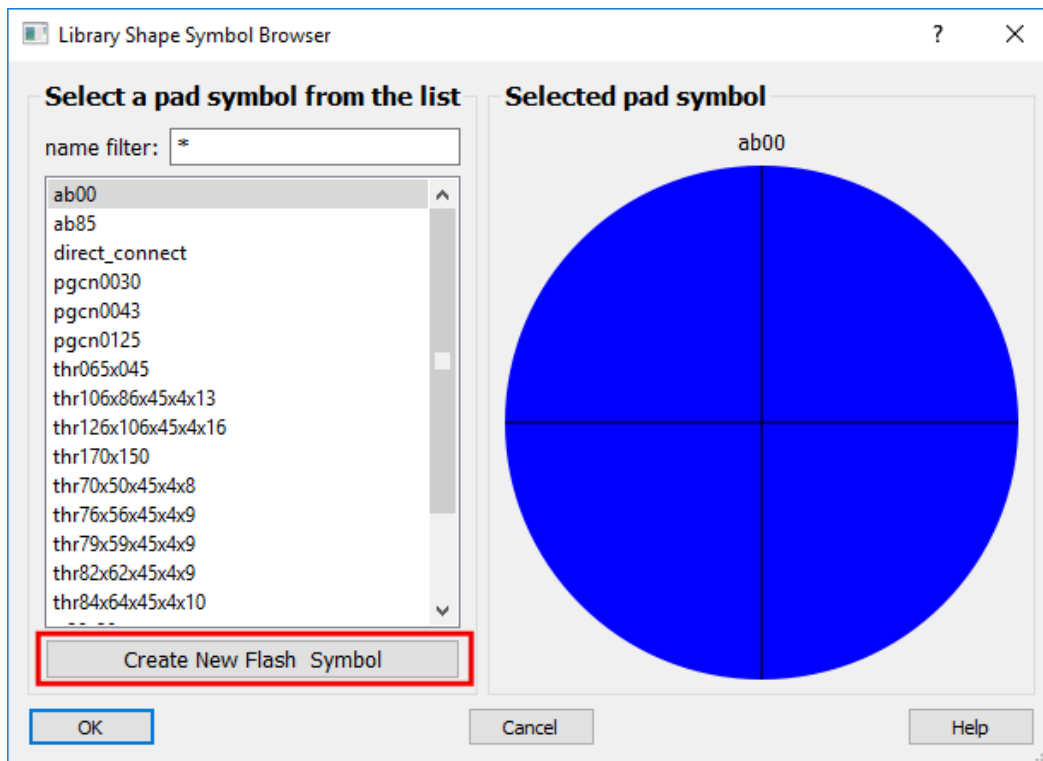
4. Enter a flash name of **tr_80_60** and select **OK**. Allegro PCB Editor opens in a mode to create flash symbols.

5. In Allegro, select **Add > Flash**.

6. Set the following:

Inner Diameter	60
Outer Diameter	80
Spoke Width	15
Number of spokes	4
Spoke angle	45

7. Click **OK**.

8. Select **View > Zoom Fit** to zoom in around the thermal relief.

9. Select **File > Save** to save the flash symbol.

10. Select **File > Exit** to close Allegro.

11. Select **tr_80_60.fsm** in **Library Shape Symbol Browser**, then select **OK**.

Defining SOLDERMASK

1. Select the **Mask Layers** tab in Pad Editor.

2. Set the following values for **SOLDERMASK_TOP**:

Geometry	Circle
Diameter	70

3. Copy the SOLDERMASK_TOP definition to SOLDERMASK_BOTTOM.

4. Select the **Summary** tab and verify the padstack information is correct.

5. Select **File > Save** to save the padstack.

Lab 2: Creating a Surface Mount Padstack

This lab outlines how to create a padstack for a surface mount device. This is a 76 mil x 24 mil rectangular pad with no drilled hole. It is assumed that Pad Editor is still open.

Naming the Padstack

1. In Pad Editor, select **File > New**.

2. Name the padstack as **76x24smd**, change the **Padstack usage** to **SMD Pin**, and select **OK**.
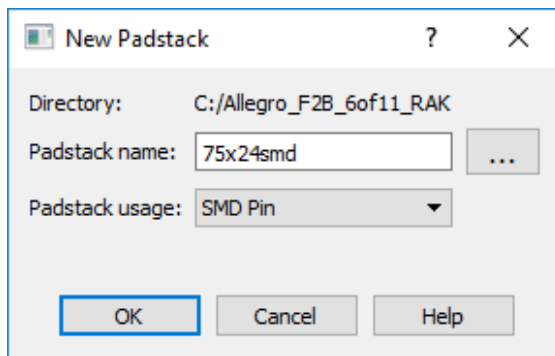
Defining BEGIN LAYER Pad

1. Select the **Design Layers** tab.

2. Select the **BEGIN LAYER > Regular Pad** cell.

   The bottom of the form displays the current definitions for Regular Pad on BEGIN LAYER.

3. Fill out the following values for the BEGIN LAYER row:

Geometry	Rectangle
Width	76
Height	24

Defining Mask Layers

1. Select the **Mask Layers** tab.

2. Define SOLDERMASK_TOP as follows:

Geometry	Rectangle
Width	86
Height	34

3. Define PASTEMASK_TOP as follows:

Geometry	Rectangle
Width	76
Height	24

Layer Name	Pad
SOLDERMASK_TOP	Rectangle 86.00x34.00
SOLDERMASK_BOTTOM	None
PASTEMASK_TOP	Rectangle 76.00x24.00
PASTEMASK_BOTTOM	None

4. Select **File > Save** to save the padstack.
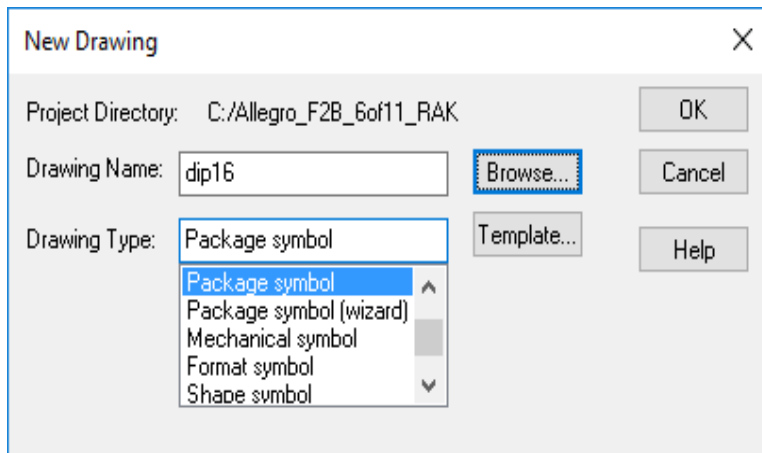
5. Select **File > Exit** to exit Pad Editor.

Module 3: Creating Package Symbols

Lab 3: Creating a Thru Hole Package

Creating a New Drawing

In this lab, a 16-pin DIP will be created using Package Symbol Editor.

1. Open Allegro PCB Editor.

2. Select **File > New**.

3. In the **New Drawing** form, name the drawing **dip16** and select a **Drawing Type** of **Package symbol**, then click **OK**.

New Drawing		✕
Project Directory: C:/Allegro_F2B_6of11_RAK		OK
Drawing Name: dip16	Browse...	Cancel
Drawing Type: Package symbol	Template...	Help
Package symbol ∧		
Package symbol (wizard)		
Mechanical symbol		
Format symbol		
Shape symbol ∨		

4. Select **Setup > Design Parameters**. In the **Design** tab, set the following:

User units	Mils
Size	A
Accuracy	2
Left X	-5000
Lower Y	-3000

5. Click **OK**.

Adding Pins

1. Select **Layout > Pins**.

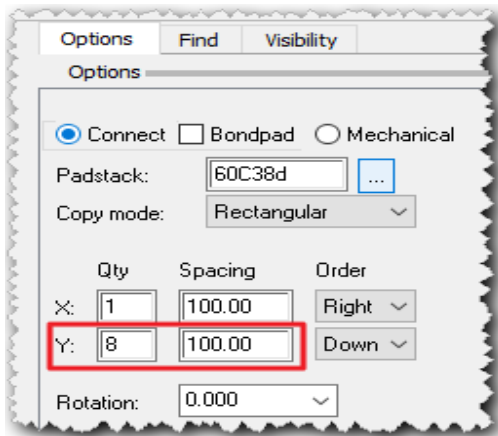2. In the **Options** pane, select the **Browse** button.
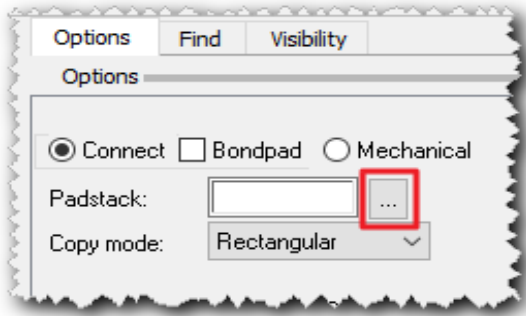
3. Select the **60c38d** padstack, then select **OK**. You should see the following message in the Allegro command window:

```
Using '60C38D.pad'
```

This means that the padstack was found. The padstack is now attached to the cursor.

4. In the **Options** pane, set the **Qty** field for the **Y** direction to **8** and the **Spacing** to **100**.
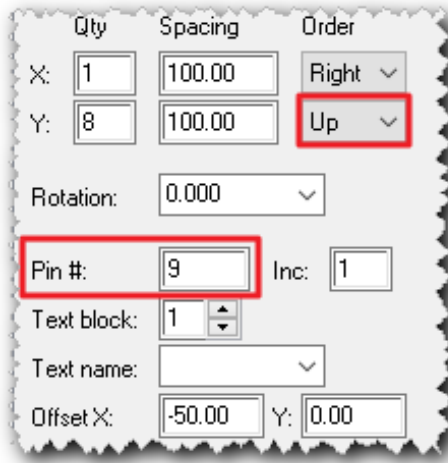

5. In the Allegro command window, type:

```
x 0 0
```

This will place pins 1 through 8 in a single column of 8 rows. The spacing between the rows is 100 mils.

6. To add another column of pins (9 through 16), change the **Order** for the **Y** direction to **Up**. Verify that the **Pin #** is set to **9**.
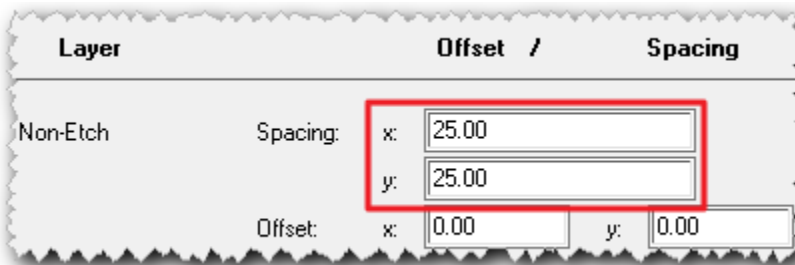

7. In the Allegro command window, type:

```
x 300 -700
```

A column of 8 pins is placed in an upward direction.

8. Right-click and select **Done**.

Adding Assembly Outline

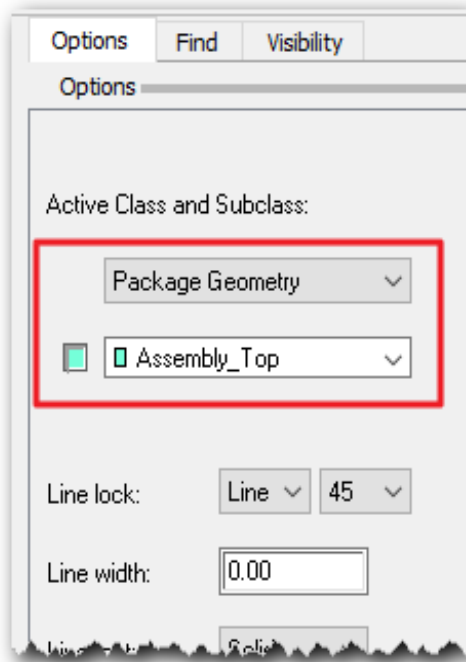1. To make the assembly outline easier to draw, select **Setup > Grids**.

2. Change the **Non-Etch x** and **y Spacing** to **25**, then click **OK**.

3. Select **Add > Line**.

4. Verify that the **Active Class and Subclass** in the **Options** pane are set to **Package Geometry/Assembly_Top**.

5. Draw the assembly outline. Use **Add > 3pt Arc** to add the arc showing where pin 1 is located.
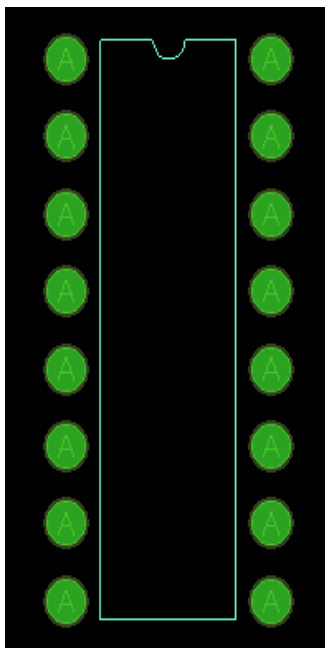
Adding Silkscreen Outline

1. Select **Add > Line**.

2. In the **Options** pane, change the **Active Class and Subclass** to **Package Geometry/Silkscreen_Top**.

3. Draw a silkscreen outline as shown below:

4. To end the **Add Line** command, right-click and select **Done**.

Adding Labels

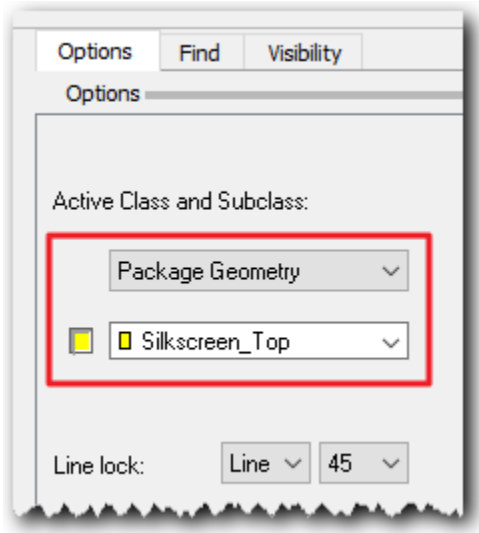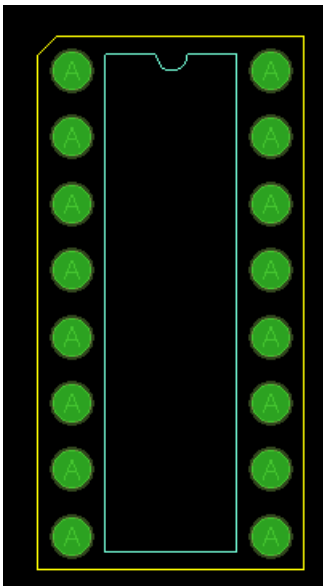Labels are placeholders that are used to display information about a device (reference designator, device type, and value).

1. Select **Layout > Labels > Refdes**.

2. Verify that the **Active Class and Subclass** in the **Options** pane is set to **Ref Des/Assembly_Top**.
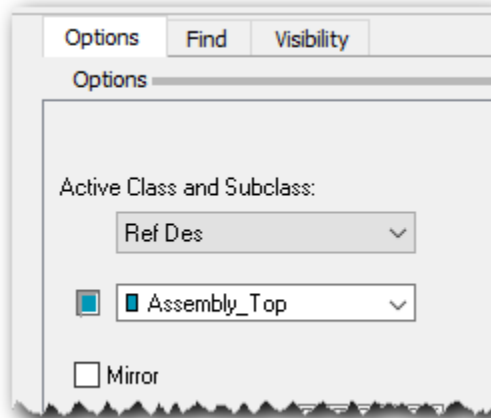
3. Click inside the assembly outline and type **U***.

4. Right-click and select **Done**.

5. Select **Layout > Labels > Device**.

6. Click near the RefDes label and enter:

   ```
   devtype
   ```

7. Right-click and select **Done**.

8. Select **Layout > Labels > Refdes**.

9. Change the **Active Class and Subclass** in the **Options** pane to **Ref Des/Silkscreen_Top**.

10. Click outside the silkscreen outline and type **U***.

Creating Package Boundary

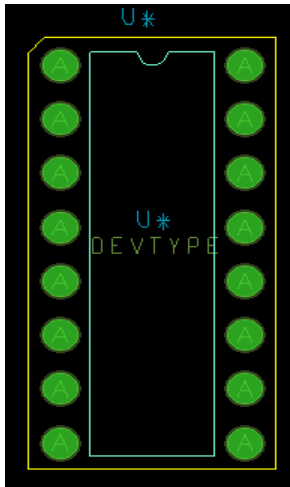A package boundary is used by the DRC system to make sure a package does not overlap another package or package keepouts. If you do not create a package boundary, it will be created when the **Create Symbol** command is used.

1. Select **Setup > Areas > Package Boundary**.

2. Verify that the **Active Class and Subclass** in the **Options** pane is set to **Package Geometry/Place_Bound_Top**.

3. Draw a polygon representing the area required for placement. Typically, this is an outline that is outside all the pins.

4. When you finish drawing the boundary, right-click and select **Done** to close the polygon.
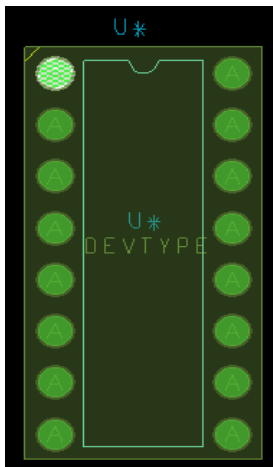
Defining Package Height

The package height is used to make sure a package does not violate a height-restricted area of the board. A default package height for all symbols can be defined in Design Parameter Editor (**Setup > Design Parameters**). To override this default package height, attach a height value to the package boundary.

1. Select **Setup > Areas > Package Height**.

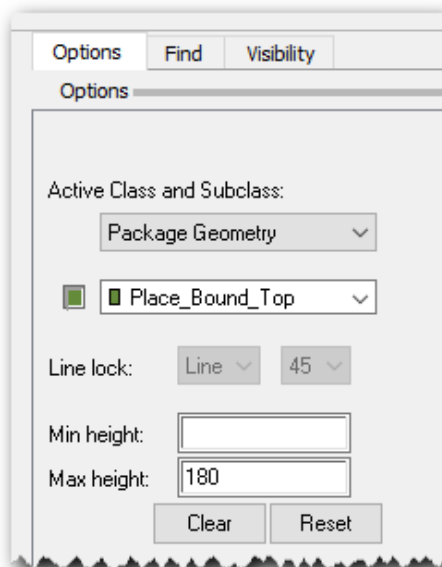2. Verify that the **Active Class and Subclass** in the **Options** pane is set to **Package Geometry/Place_Bound_Top**.

3. Click on the package boundary polygon.

4. In the **Options** pane, set the **Max height** to **180**.

5. Right-click in the canvas and select **Done**.

6. Select **File > Save**. This will save dip16.dra and create a symbol named dip16.psm. The .dra file is used to edit the footprint graphics, and the .psm file is used during component placement.

Lab 4: Creating a Surface Mount Package

Creating a New Drawing

In this lab, a 16-pin SOIC will be created using Package Symbol Editor.

1. Select **File > New**.

2. In the **New Drawing** form, name the drawing **soic16** and select a **Drawing Type** of **Package Symbol**, then click **OK**.

3. Since the pins will be on 50 mil centers, the grid needs to be changed. Select **Setup > Grids**.

4. Change the **Non-Etch x** and **y Spacing** to **25**, then click **OK**.

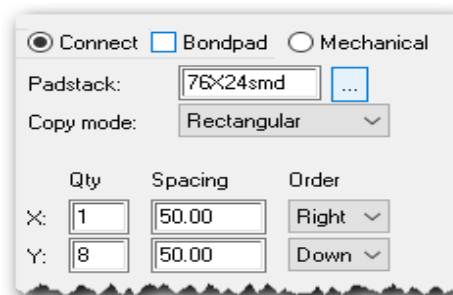| Layer		Offset /	Spacing
Non-Etch	Spacing: x:	25.00	
	y:	25.00	
	Offset: x:	0.00	y: 0.00

Adding Pins

1. Select **Layout > Pins**.

2. In the **Options** pane, select the **Browse** button.

3. Select the **76x24smd** padstack, then select **OK**. You should see the following message in the Allegro command window:

        Using '76X24SMD.pad'

4. In the **Options** pane, set the **Qty** field for the **Y** direction to **8** and the **Spacing** to **50**.
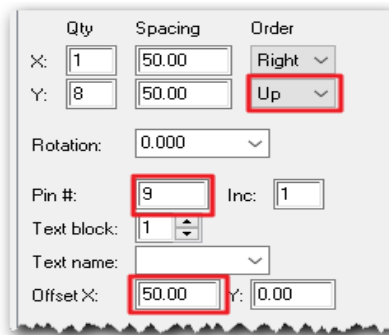
5. In the Allegro command window, type:

    x 0 0

    This will place pins 1 through 8 in a single column of 8 rows. The spacing between the rows is 50 mils.

6. To add another column of pins (9 through 16), change the **Order** for the **Y** direction to **Up**. Verify that **the Pin #** is set to **9**.

7. To set the pin number text to the right of pins, change the **Offset X** field to **50**.

8. In the Allegro command window, type:

    x 225 -350

    A column of 8 pins is placed in an upward direction.

9. Right-click and select **Done**.

Adding Assembly and Silkscreen Outlines

1. Select **Add > Line**.

2. Change the **Active Class and Subclass** in the **Options** pane to **Package Geometry/Assembly_Top**.

3. Draw the assembly outline.

4. Right-click and select **Next**.

5. In the **Options** pane, change the **Active Subclass** to **Silkscreen_Top**.

6. Draw a silkscreen outline.

7. To end the **Add Line** command, right-click and select **Done**.

Adding Labels

1. Select **Layout > Labels > Refdes**.

2. Verify that the **Active Class and Subclass** in the **Options** pane are set to **Ref Des/Assembly_Top**.

3. Click inside the assembly outline and type **U***.

4. In the **Options** pane, change the **Active Subclass** to **Silkscreen_Top**.

5. Click outside the silkscreen and type **U***.

6. Right-click and select **Done**.

7. Select **Layout > Labels > Device**.

8. In the **Options** pane, change **Rotate** to **90**.

9. Click inside the assembly outline and enter:

   ```
   devtype
   ```

10. Right-click and select **Done**.
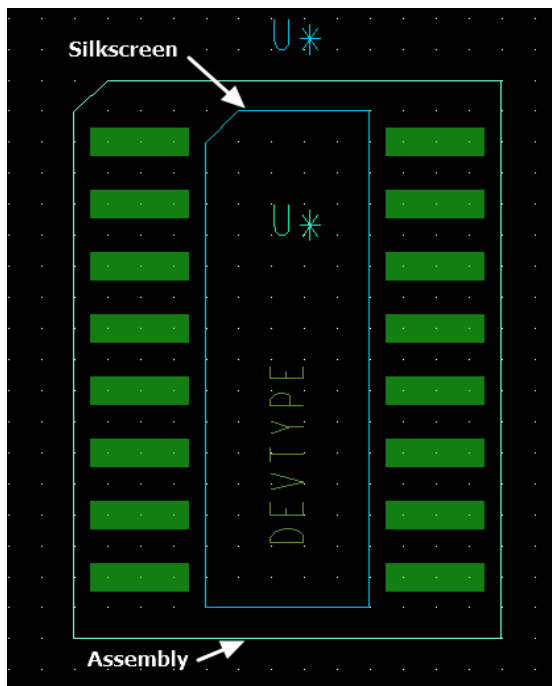
11. Select **File > Save**. This will save soic16.dra and create a symbol named soic16.psm.

# Module 4: Board Outline and Connector Placement Import Using IDX

IDX (Incremental Data Exchange) can be used to import a mechanical board outline and connector component placement. IDX is an intelligent bidirectional interface with mechanical design tools. It allows for each object in the mechanical design tool to be accounted for, resulting in easy incorporation of mechanical design ECOs.

Lab 5: IDX Import

1. In Allegro PCB Editor, select **File > Open** and select **start.brd**.

2. In **Design Workflow**, select **Database Preparation > Board Outline > Import > IDX** (or select **File > Import > IDX** from the Allegro menu).
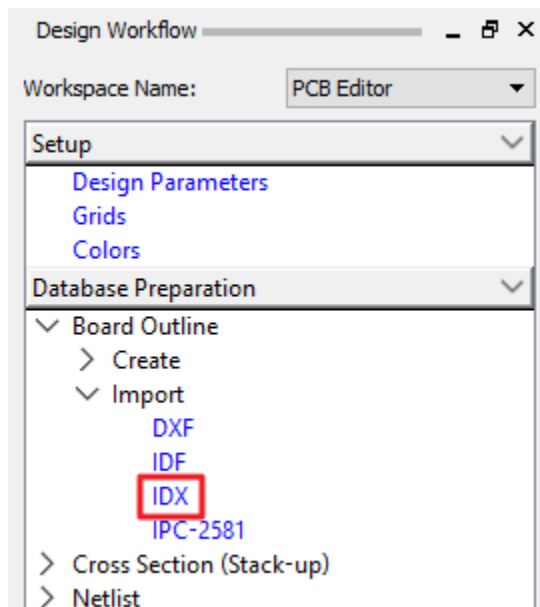
3. Browse to select **Viper_MCAD.idx**. Check **Use as baseline**, then select **Import**.
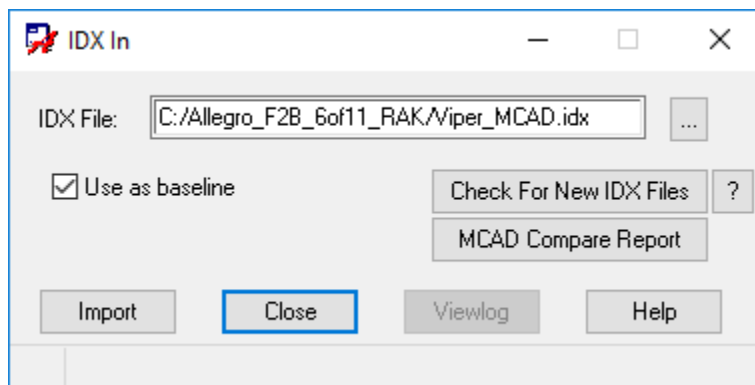
The **IDX Flow Manager Import** form opens. This form lists all objects being imported from the mechanical design tool. The status for all objects is Baseline since this is the baseline import of this IDX data. If this IDX file was an ECO, the status fields would reflect the different changes, which can be reviewed, and accepted or rejected.

4. Check **Select All** at the bottom of the form.

5. With the **IDX Flow Manager Import** box still open, zoom out in the Allegro canvas to see all objects being imported. Select different objects in the **IDX Flow Manager Import** box to zoom-fit them in the design.

6. Select **OK** in the **IDX Flow Manager Import** box to import all objects.

7. Select **Close** in the IDX In form.

8. Save the design as **import.brd**.

Module 5: Cross Section Editor

Cross Section Editor is used to define the cross section, or layer stackup, for the design, which includes all conductor, surface, dielectric, and die stack layers and their characteristics. It contains the ability to set up dynamic unused pad suppression and embedded components.

A graphical image of the stackup is available in a dockable window. The image displays the drilling direction and supports blind/buried layers, embedded layers, and backdrill layers.

Additional features include via label customization, layer-based positive/negative tolerance support, locking functionality to prevent editing of the stackup, and multi-stackup support (for rigid flex designs).

This module will cover:

- Introduction to Cross Section Editor
- Adding layer pairs
- Adding a user-defined number of layers
- IPC-2581 layer functions
- Importing an IPC-2581 stackup
- Adding non-conductor layers to the cross section

Introduction to Cross Section Editor
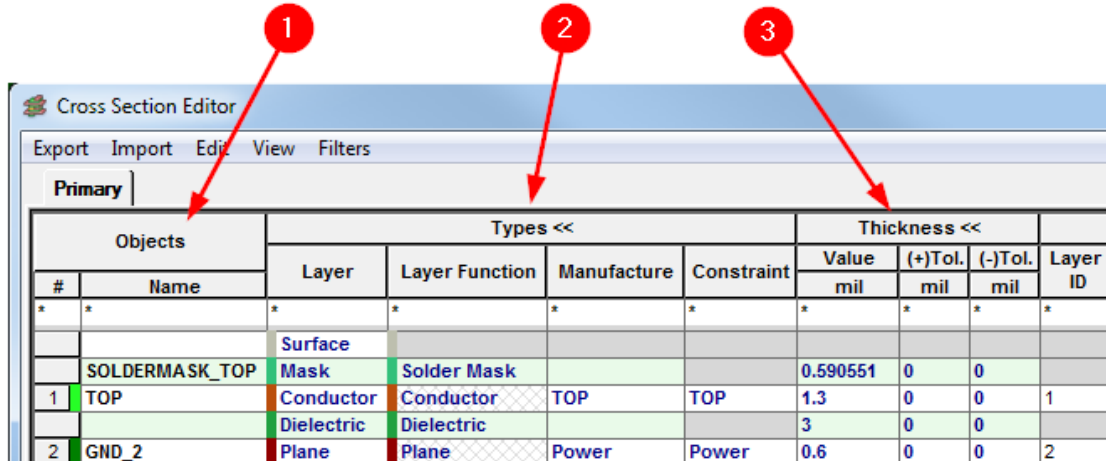
The default view of Cross Section Editor combines the spreadsheet grid with the cross section viewer. The drill display within the viewer is based on actual padstack usage in the database and does not display unused vias in Physical CSets.
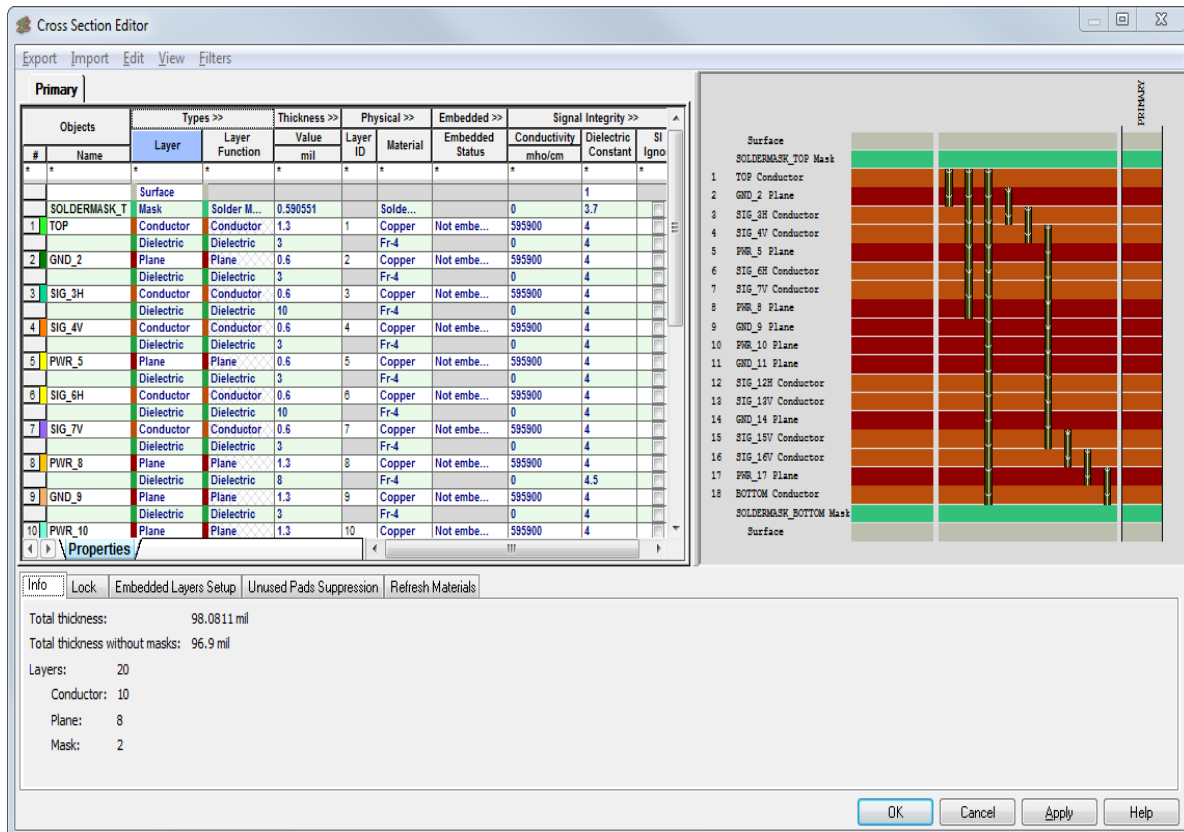
The cross section worksheet displays the layers of the active design using a spreadsheet view, where rows represent the primary layer material and columns represent the various properties of the layer. Cross Section Editor can be resized to display a larger range of layers in the design.

Worksheet Controls

When Cross Section Editor is first opened, a tab named "Primary" is displayed. This is the default stackup that represents the default design stackup in terms of all electrical layers (Conductor, Plane, Dielectric, and non-electrical layers).

Objects (1) – Represents a named layer in the stackup
Types (2)
>   **Layer** – Specifies the layer type – Surface, Mask, Conductor, Dielectric, or Plane
>   **Manufacture** – Assign hierarchical names to signal and plane layers, for example, INNER_SIGNAL or PLANE. This information is supported in the IPC-2581 data schema.
>   **Constraint** – Assign hierarchical names to signal and plane layers. These names are integrated Spacing CSet structures.
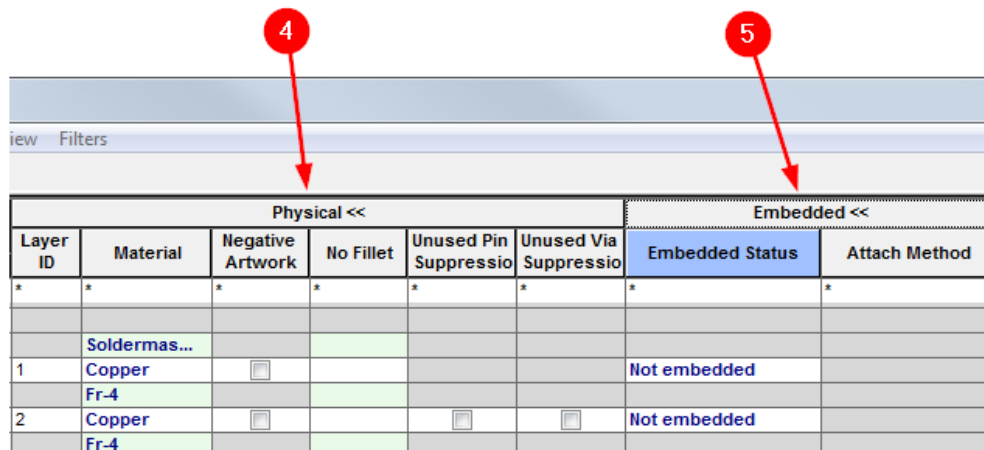
Thickness (3)
>   **Value** – Specifies the thickness of the selected layer
>   **(+)** or **(-) Tol.** – Specifies positive or negative thickness tolerance

Figure 1

Physical (4)

Layer ID – Used for customization of blind/buried via label display

Material – Choose from materials that are specified in the Materials file. Material choices are based on the selected layer type.

Negative Artwork – When checked, creates negative artwork for the selected conductor layer

Unused Pin/Via Suppression – Controls the layer settings for the removal of unused pads on inner signal layers

Embedded (5)

Embedded Status – Specifies whether the layer can be used for component placement and, if used, the orientation of the component on the layer

Attach Method – Specifies the method to be used for connecting components to the embedded layer

Signal Integrity <<										
Conductivity mho/cm	Dielectric Constant	Width mil	Impedance Ohm	Loss Tangent	Shield	Freq. Dep. File	Etch Factor	Diff Coupling Type	Diff Spaci mil	SI Ignore
*	*	*	*	*	*	*	*	*	*	*
	1			0						
0	3.7			0.022						☐
595900	4	4.00	53.680000	0.035			90	None		☐
0	4			0.035						☐

Signal Integrity (6)

Conductivity – Specifies the electrical conductivity for the selected layer

Dielectric Constant – Specifies the dielectric constant for the selected layer

Width – Defines the width of the routed etch line on the layer. The default line width is taken from the active physical rule set. Changes made to Line Width in Cross Section Editor do not affect the line width values in the constraint set.

Impedance – Displays the impedance of etch lines on the layer

Loss Tangent – Specifies the dielectric losses for the selected layer in terms of the tangent of the complement of the insulation power-factor angle

Shield – Designates the selected plane layer as a shield layer

Freq. Dep. File – Specifies the Frequency Dependent File selectable from the files residing in the $MATERIALPATH directory

Etch Factor – Specifies a layer-specific trapezoidal angle for etch on the conductor and plane layers

Diff Coupling Type – Specifies coupling type for differential pairs. Choices are NONE, EDGE, and BROADSIDE.

Diff Spacing – Defines the spacing to use for edge-coupled differential impedance calculations and the trace layer pairing for broadside coupling

Diff Z0 – Specifies the calculated (or user-defined) differential impedance value for the selected layer

Dialogs Pane

Five functional tabs are located near the bottom of the spreadsheet.

Info – Reports total thickness, in database units, and the number of conductor, plane, and mask layers

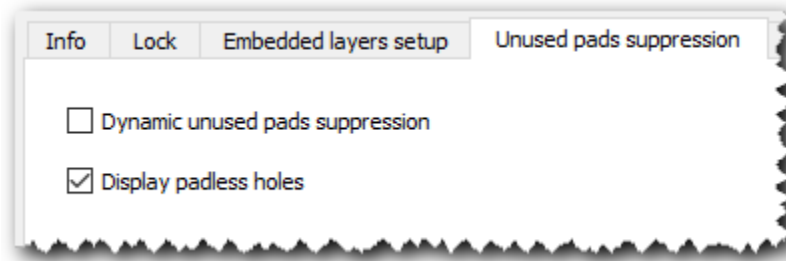**Lock** – Prevents editing, adding layers, or changing values in the spreadsheet

Embedded Layers Setup – Displays the setup form for Embedded Component Design. This tab is only available with the Miniaturization option.

Unused Pads Suppression – Displays the setup form for Unused Pads Suppression

Refresh Materials – Refreshes the cross section worksheet with changes that have been made in the materials file (materials.dat)
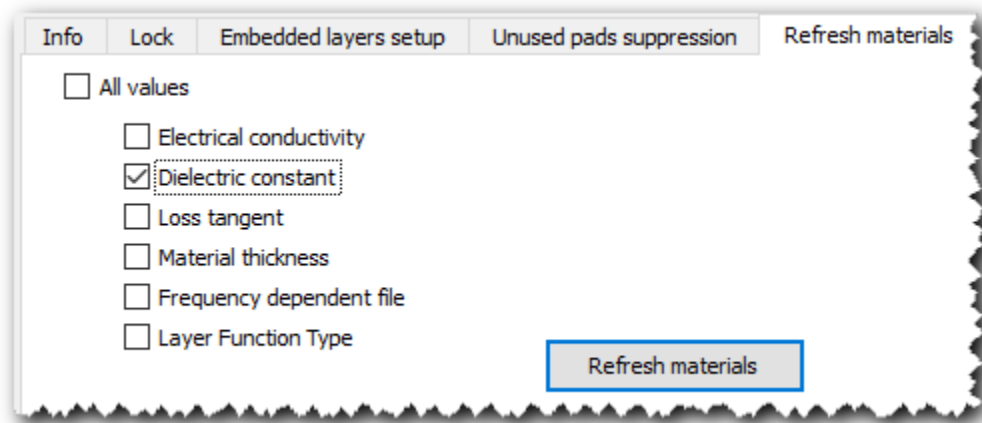
Context-Sensitive Cross Section and Drill Chart

To customize the Cross Section and Drill Chart, right-click on the cross-section image and select **Draw Options**.
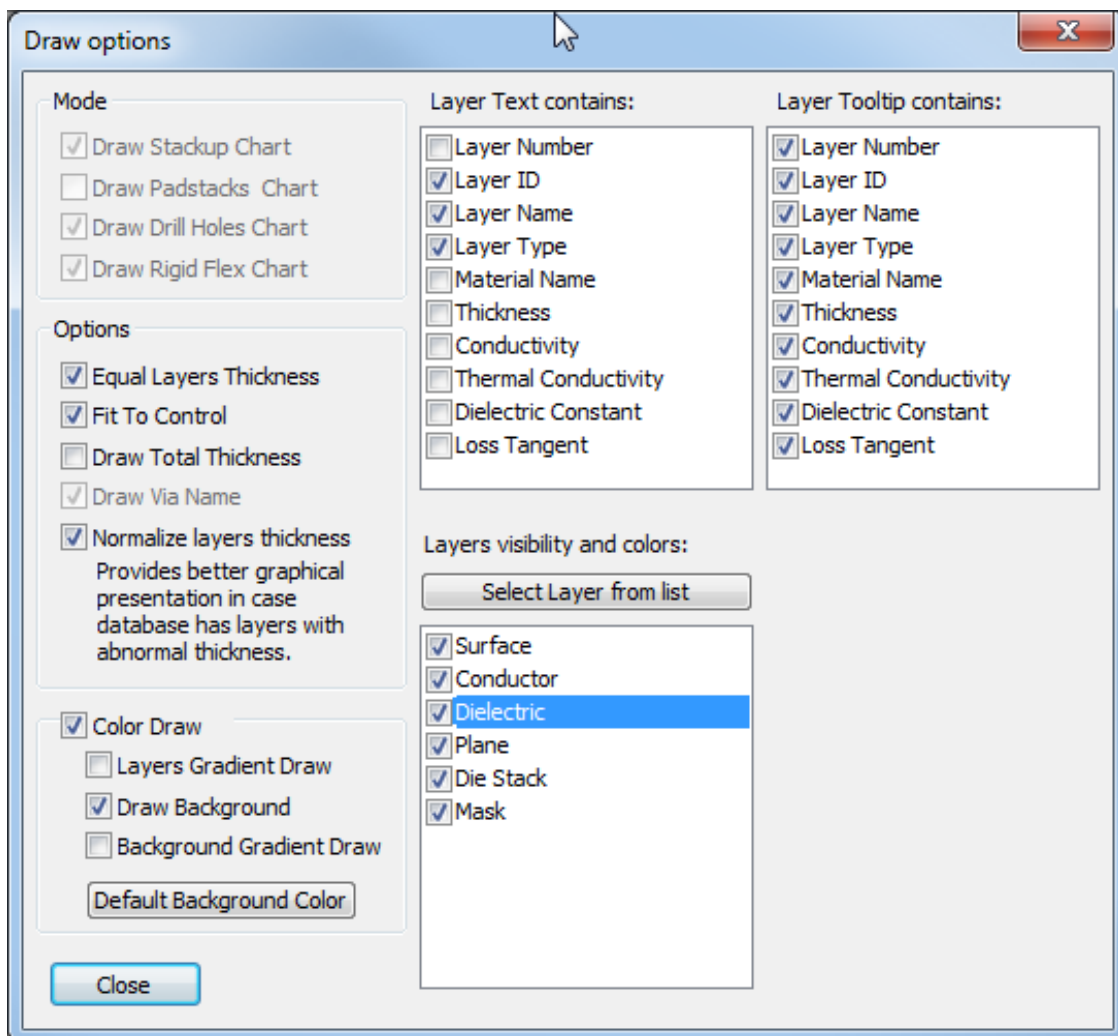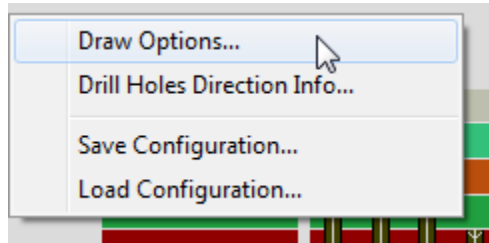
Selecting **Drill Holes Direction Info** displays the **Drill Chart** dialog box, which shows the drill direction for each layer.
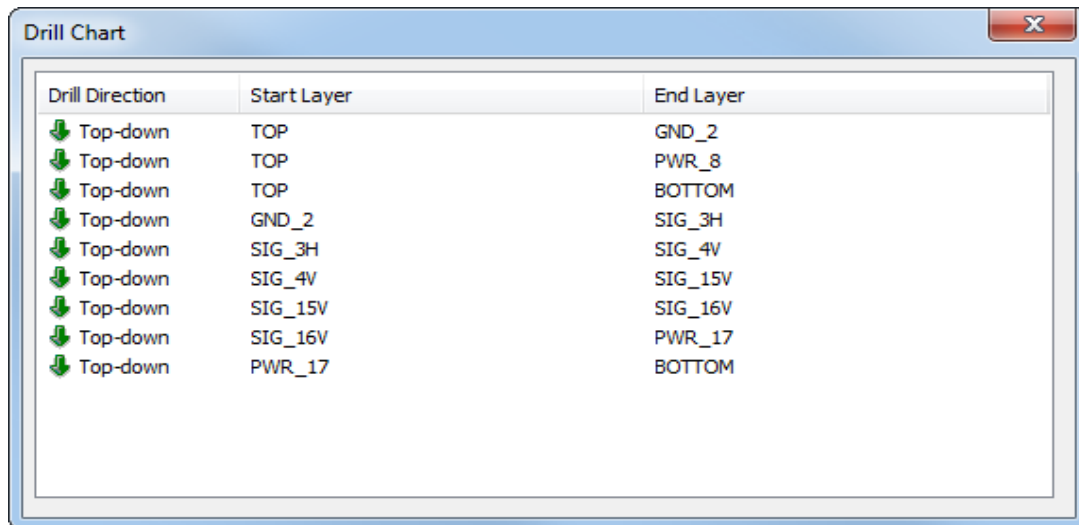
To reverse the drill direction for buried/blind/microvias, right-click on a drill and select **Reverse Drill Direction**. The reversal only impacts the ordering of layers used in NC Drill Legend Chart.
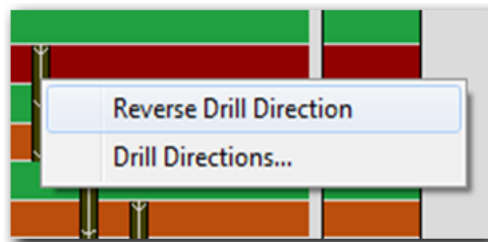
Selecting **Drill Directions** displays the **Drill Chart** dialog box, which shows the drill direction for each drill.

Context-Sensitive Worksheet Controls

Right-click on a Layer object name to display the layer options.
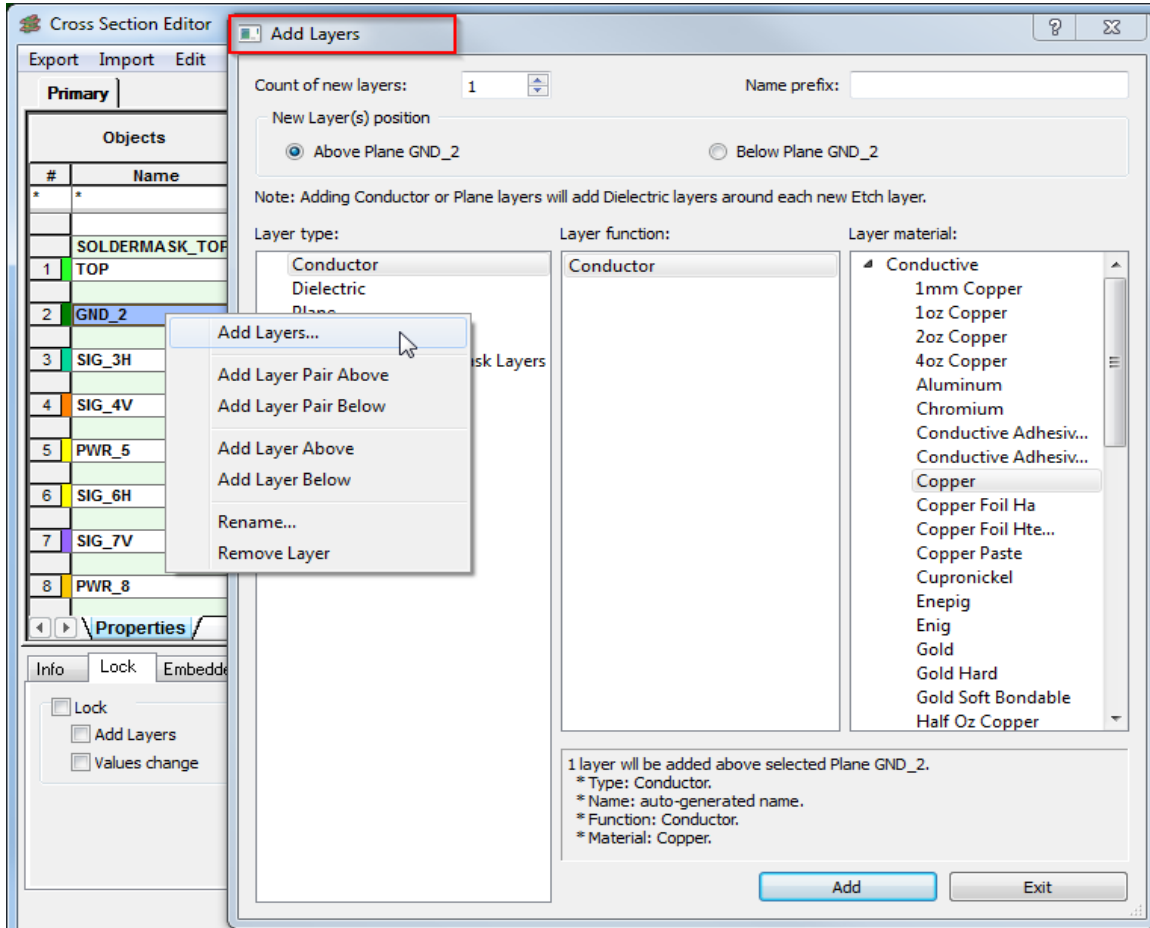
Add Layers – Displays the Add Layers dialog box to add multiple layers
Add Layer Pair Above – Adds a pair of layers to the stackup above the selected layer
Add Layer Pair Below – Adds a pair of layers to the stackup below the selected layer
Add Layer Above – Adds a layer to the stackup above the selected layer
Add Layer Below – Adds a layer to the stackup below the selected layer
Rename – Renames the selected layer
Remove Layer – Removes the selected layer from the stackup

Note: The dielectric layer between two conductive layers or any surface layer cannot be removed.

Lab 6: Cross Section Editor

Adding Layer Pairs

The Add Layer Pair command adds a dielectric and conductor layer pair above or below the selected cell. You cannot add a layer pair above the TOP or below the BOTTOM layers.

1. Continue working in **import.brd**. In the **Design Workflow** pane, select **Database Preparation > Cross Section (Stack-up) > Create** (or select **Setup > Cross Section** from the Allegro menu) to open Cross Section Editor.
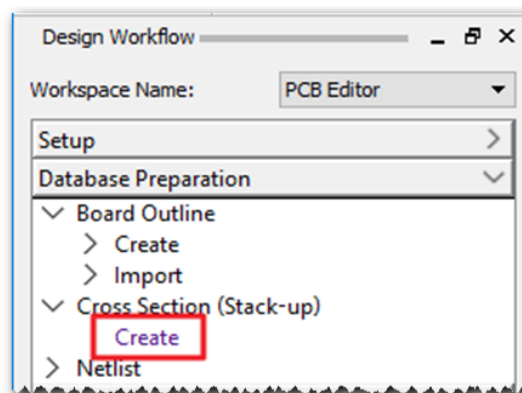
Cross Section Editor displays one row for each layer of the layout cross section. The lines are in the physical order of the layers, from TOP to BOTTOM, as they exist in the layout.

2. In the **Objects** column, right-click on the **TOP** layer and select **Add Layer Pair Below**.
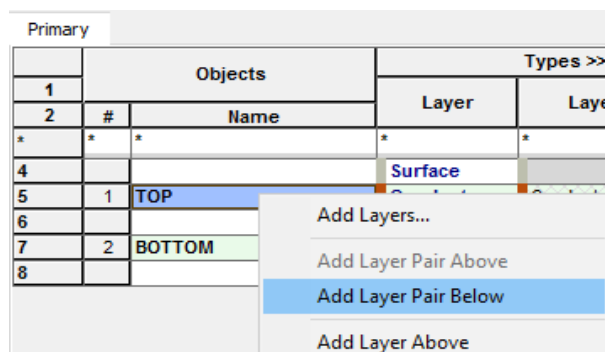
A new dielectric and conductor layer pair is added below the TOP layer.

3. In the **Objects** column, right-click on the **BOTTOM** layer and select **Add Layer Pair Above**.

Adding a User-Defined Number of Layers

4. In the **Objects** column, right-click on **LAYER_2** and select **Add Layers**.
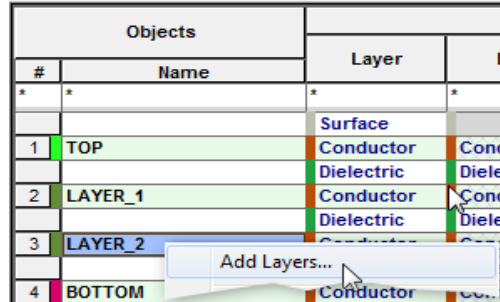
This opens the **Add Layers** dialog box.
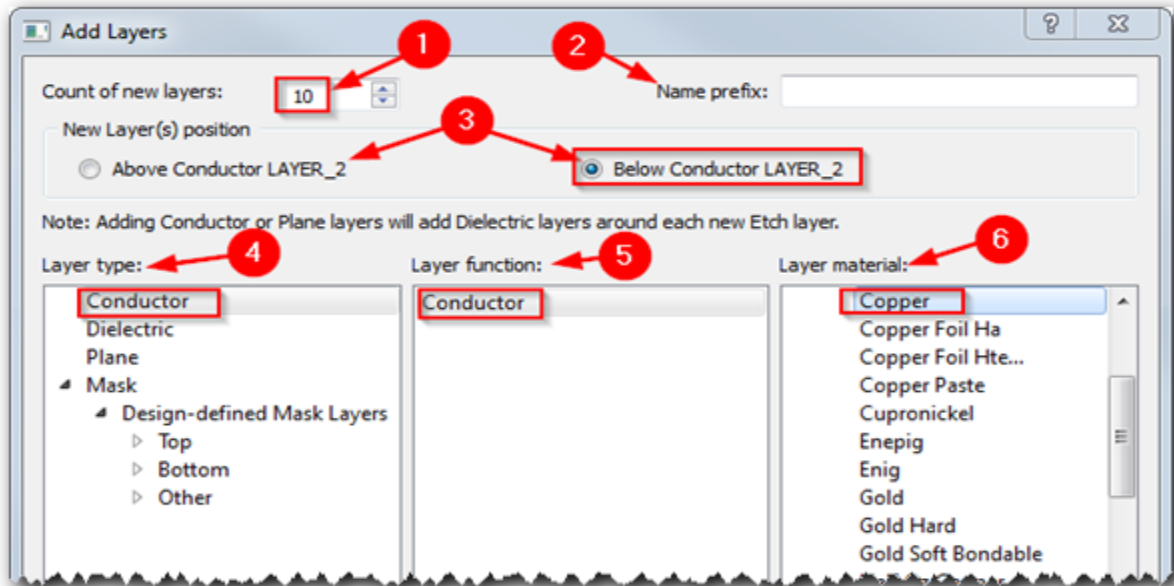
Count of new layers (1) – Number of layers to add
Name prefix (2) – Layer name to be used for a Conductor, Plane, or named Dielectric layer
New Layer(s) position (3) – Add new layer above or below selected layer
Layer Type (4) – Type of layer to be added – Conductor, Dielectric, Plane, or Mask
Layer function (5) – Defines the intended use of the new layers based on IPC-2581 layer type definitions, which are used by the PCB fabricator during the manufacturing process
Layer material (6) – The listed materials are based on the layer type selected.

5. In the **Add Layers** dialog box:

- Set **Count of new layers** to **10**.
- Select **Below Conductor LAYER_2**.
- Set **Layer type** and **Layer function** to **Conductor**.
- Select **Copper** for **Layer material**.

6. Select **Add** to add layers to the stackup, then select **Exit**.

| #	Name	Layer	Layer Function	Manufacture	C
*	*	*	*	*	*
		Surface			
1	TOP	Conductor	Conductor		
		Dielectric	Dielectric		
2	LAYER_1	Conductor	Conductor		
		Dielectric	Dielectric		
3	LAYER_2	Conductor	Conductor		
		Dielectric	Dielectric		
4	LAYER_3	Conductor	Conductor		
		Dielectric	Dielectric		
5	LAYER_4	Conductor	Conductor		
		Dielectric	Dielectric		
6	LAYER_5	Conductor	Conductor		
		Dielectric	Dielectric		
7	LAYER_6	Conductor	Conductor		
		Dielectric	Dielectric		
8	LAYER_7	Conductor	Conductor		
		Dielectric	Dielectric		
9	LAYER_8	Conductor	Conductor		
		Dielectric	Dielectric		
10	LAYER_9	Conductor	Conductor		
		Dielectric	Dielectric		
11	LAYER_10	Conductor	Conductor		
		Dielectric	Dielectric		
12	LAYER_11	Conductor	Conductor		
		Dielectric	Dielectric		
13	LAYER_12	Conductor	Conductor		
		Dielectric	Dielectric		
14	BOTTOM	Conductor	Conductor		
		Surface			

Cross Section Editor — Export Import Edit View Filters — Primary — Objects — Types <<

IPC-2581 Layer Functions

Cross Section Editor supports the IPC-258-defined layer functions. These user-selected layer functions are defined as attributes in the IPC-2581 stackup layer definition for fabrication instructions at the manufacturing level. Consider using these functions if IPC-2581 is used for data transfer. The full set of layer function options are:
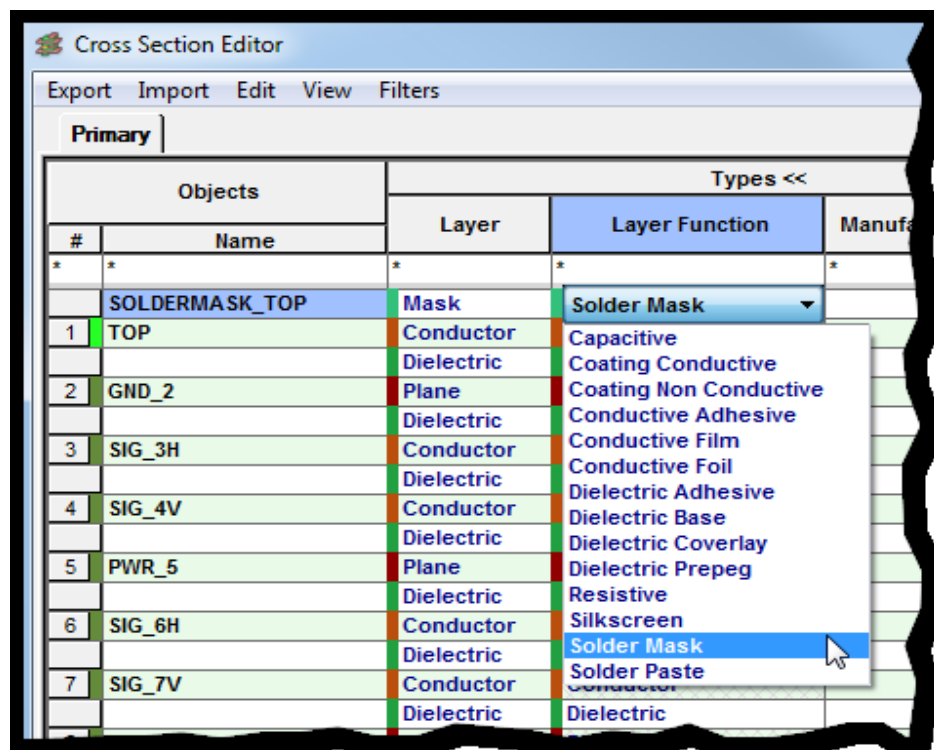
Dielectric layer types supported are:
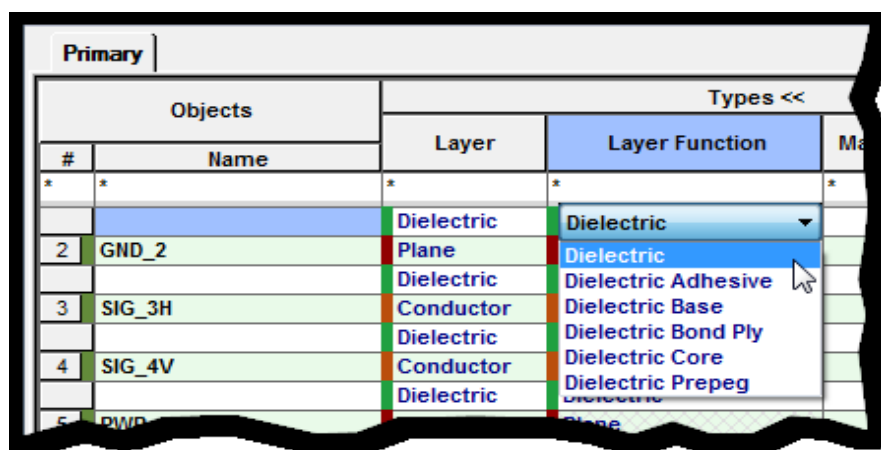
Layer functions do not apply to conductor layers.

Changing Cross Section Layer Name, Type, and Thickness

7. In the **Objects** column, right-click on **LAYER_1** and select **Rename**.
    Alternatively, you can double-click in the cell to make it editable.

8. Enter **GND_2** in the **Rename** dialog box and click **OK**.
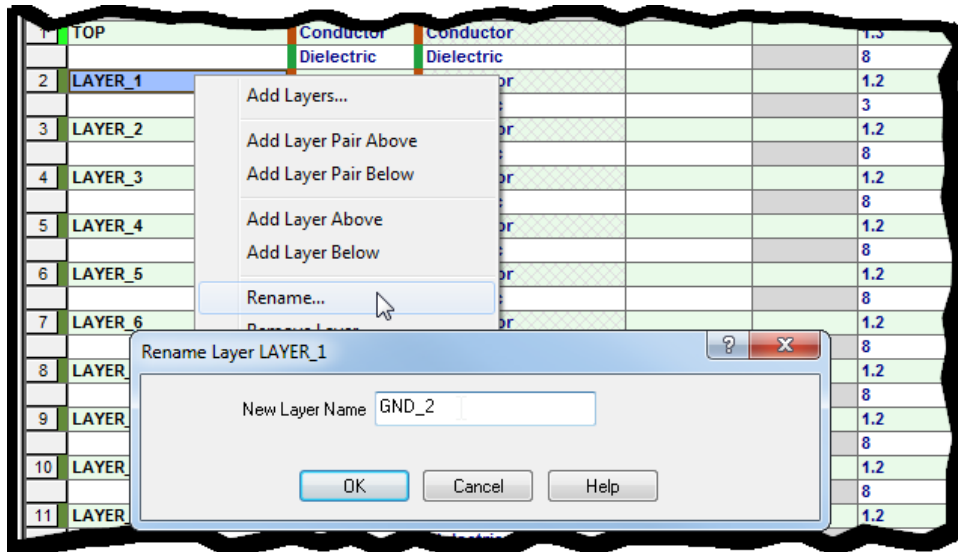
9. Repeat the previous two steps to rename **LAYER_2** to **SIG_3H**, **LAYER_3** to
    SIG_4V, and **LAYER_4** to **PWR_5**.

10. In the **Layer** column for **GND_2**, left-click and select **Plane**. Set the Layer to
    Plane for **PWR_5**.

11. Enter the **Thickness** values for **TOP**, **GND_2**, **SIG_3H**, **SIG_4V**, and **PWR_5** as shown below.

Objects		Types <<				Thickness >>
#	Name	Layer	Layer Function	Manufacture	Constraint	Value mil
*	*	*	*	*	*	*
		Surface				
1	TOP	Conductor	Conductor			1.3
		Dielectric	Dielectric			8
2	GND_2	Plane	Plane			0.6
		Dielectric	Dielectric			3
3	SIG_3H	Conductor	Conductor			0.6
		Dielectric	Dielectric			8
4	SIG_4V	Conductor	Conductor			0.6
		Dielectric	Dielectric			8
5	PWR_5	Plane	Plane			0.6
		Dielectric	Dielectric			8

12. Select **Apply**, then **OK** to close Cross Section Editor.

Importing an IPC-2581 Stackup

IPC-2581 is rapidly gaining support in the PCB design community as an "intelligent" vendor-neutral format that can bring design data into manufacturing in a single file.

A key feature of the IPC-2581 format is the ability to import cross section information from a stackup tool, such as Polar, directly into Allegro PCB Editor. This format will be used to import the cross section for this design.
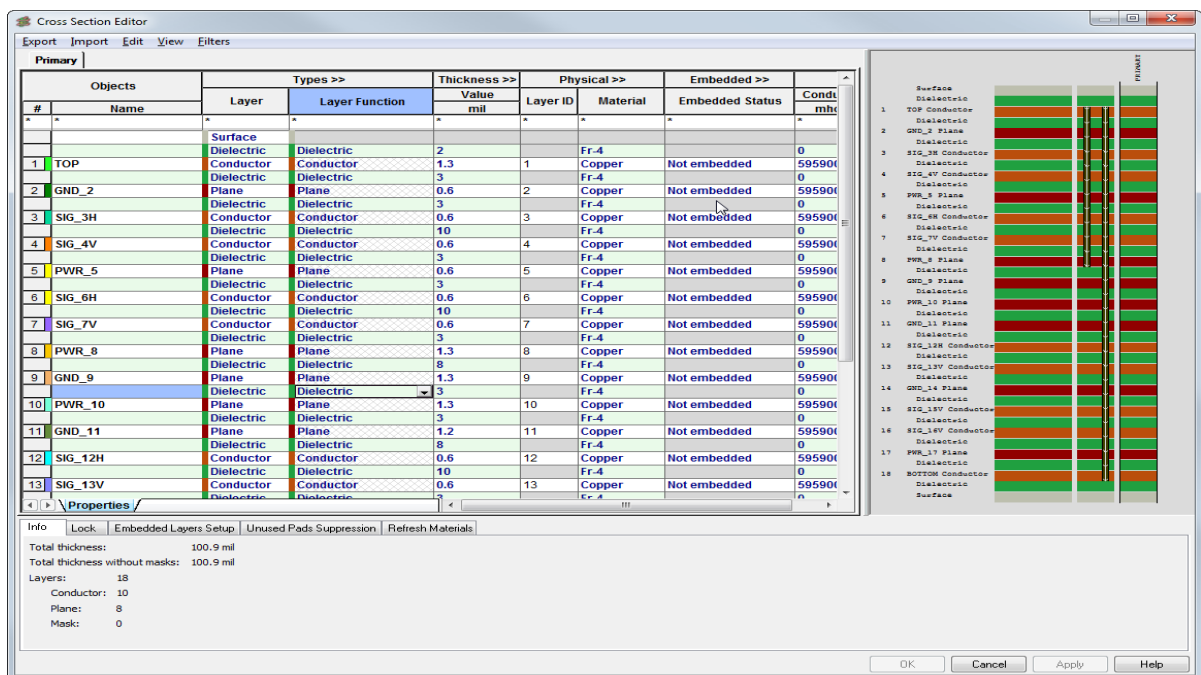
13. In Allegro, select **File > Import > IPC2581**.

14. Select the browse button and select **18_lyr_stack.xml**. Check the **Layer stackup** and **Layer features** import options, then select **Import**.
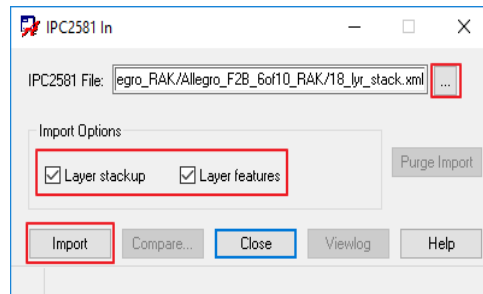
15. Select **Close** to close the **IPC2581 In** form.

16. Open Cross Section Editor and review the stackup imported using IPC-2581.

Support of Non-Conductor Layers

Cross Section Editor supports non-conductor layers, such as masking and coating layers used in rigid flex applications. These layers are usually added above the Top or below the Bottom surfaces but can also be added within the core stackup to accommodate multiple independent flex laminates. The Allegro PCB Editor database represents non-conductor layers as "Mask" or "Dielectric", although they may serve different purposes, such as coating or plating areas.

The next steps outline how to add the TOP and BOTTOM soldermask layers to the cross section.
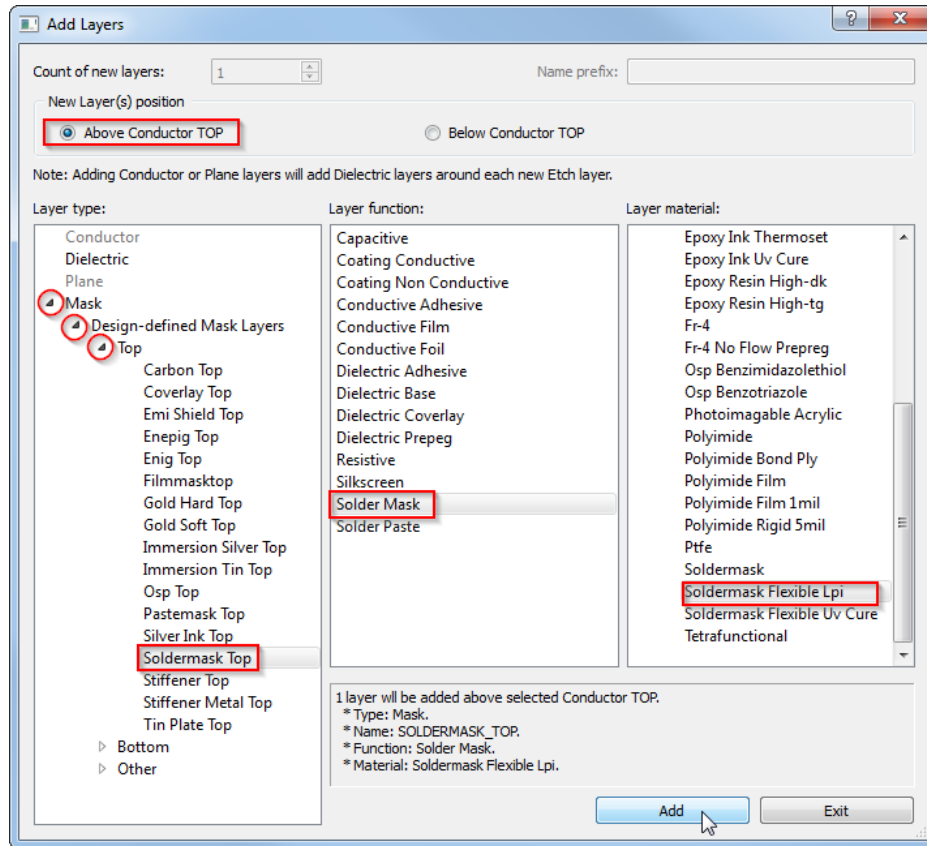
17. In the **Objects** column, right-click on the **TOP** layer and select **Add Layers**.
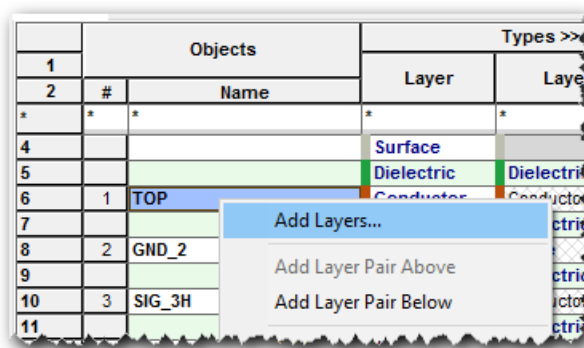
18. Select **Above Conductor TOP**.

19. In the **Layer type** column, expand **Mask > Design-defined Mask Layer > Top**, and select **Soldermask Top**.

20. In the **Layer function** column, select **Solder Mask**.
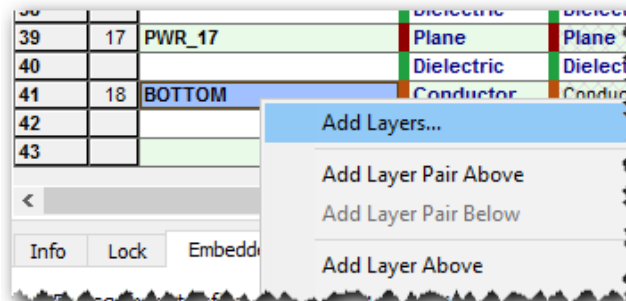
21. In the **Layer material** column, select **Soldermask Flexible Lpi**.

22. Select **Add**, then **Exit**.

23. In the **Objects** column, right-click on the **BOTTOM** layer and select **Add Layers**.

24. Select **Below Conductor BOTTOM**.

25. In the **Layer type** column, expand **Mask > Design-defined Mask Layer > Bottom**, and select **Soldermask Bottom**.

26. In the **Layer function** column, select **Solder Mask**.

27. In the **Layer material** column, select **Soldermask Flexible Lpi**.
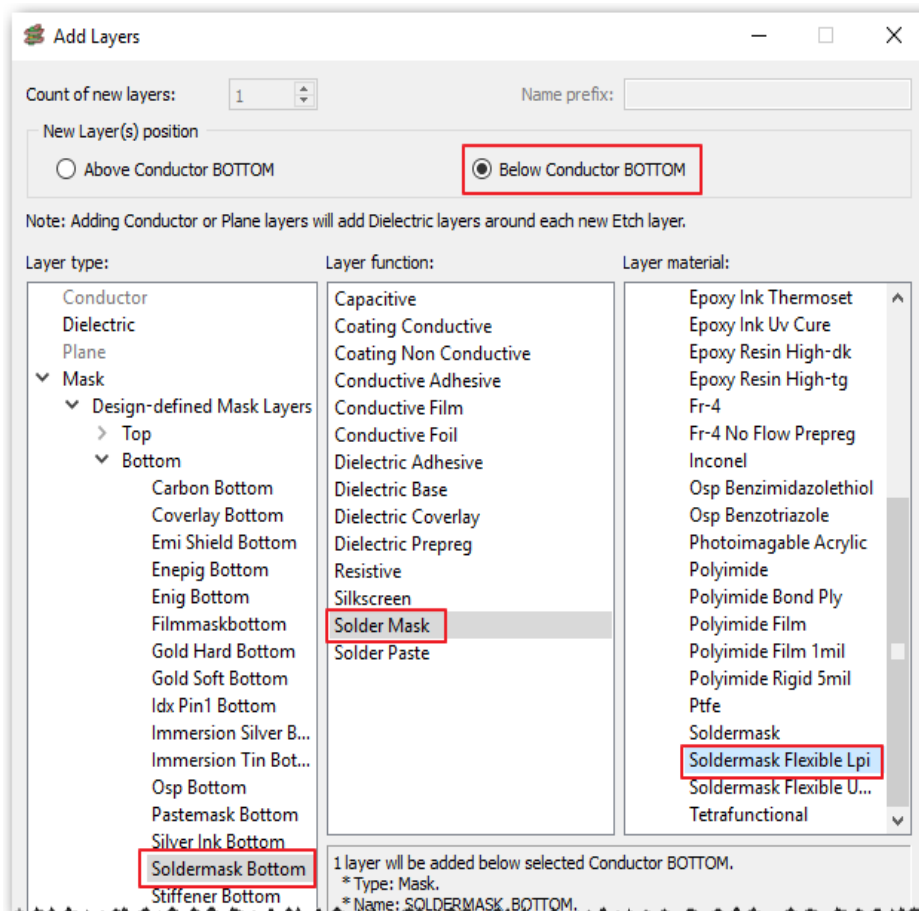
28. Select **Add**, then **Exit**.

   SOLDERMASK_TOP has been added above the TOP layer and SOLDERMASK_BOTTOM has been added below the BOTTOM layer in the stackup.

29. Save the design.

Module 6: Importing Logic Data

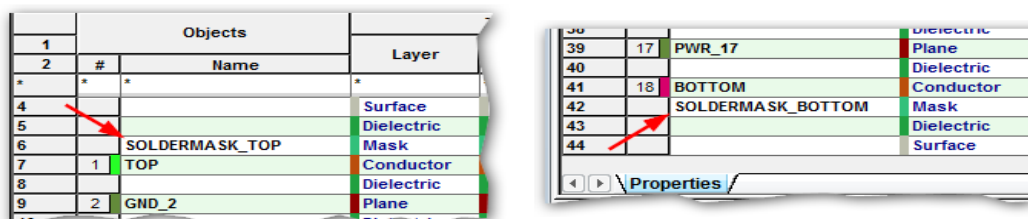Once a schematic has been completed in DE HDL, it must be packaged using Packager-XL, which converts the logic devices into physical packages, assigning a reference designator and physical pin numbers to each symbol in the schematic. The packaged parts and their connections are written into transfer files in the design *packaged* directory. These files are as follows:

- **pstxprt.dat** is a part list file that lists each physical package created by the packager in the schematic, along with its reference designator and device type.

- **pstxnet.dat** is a netlist file that uses keywords to specify the reference designators and pin numbers associated with each net in the schematic. This file may also contain some properties attached to nets in the schematic.

- **pstchip.dat** is a device definition file that contains electrical characteristics such as pin direction and loading, logical-to-physical pin mapping, and voltage requirements. This file also contains the name of the package symbol that represents this device type in the physical layout.

Constraint information can also be entered in the schematic. The packager creates files containing this constraint information to be passed to the board.

- **psdcmdb.dat** contains information about the electrical constraints currently present in the design. This file is generated even if no constraints have been created in the schematic.

- **pstcmdbc.dat** contains the electrical constraint baseline in the schematic data.

Lab 7: Importing Logic Data

1. Continue working in **import.brd**. Select **File > Import > Logic/Netlist**. The **Import Logic/Netlist** form contains two tabs – Cadence and Other. DE HDL/CIS schematics are imported using the **Cadence** tab, while third-party netlists are imported using the **Other** tab.

2. In the **Cadence** tab, browse to set the import directory to the included **packager_files** directory. Set the options as shown below:
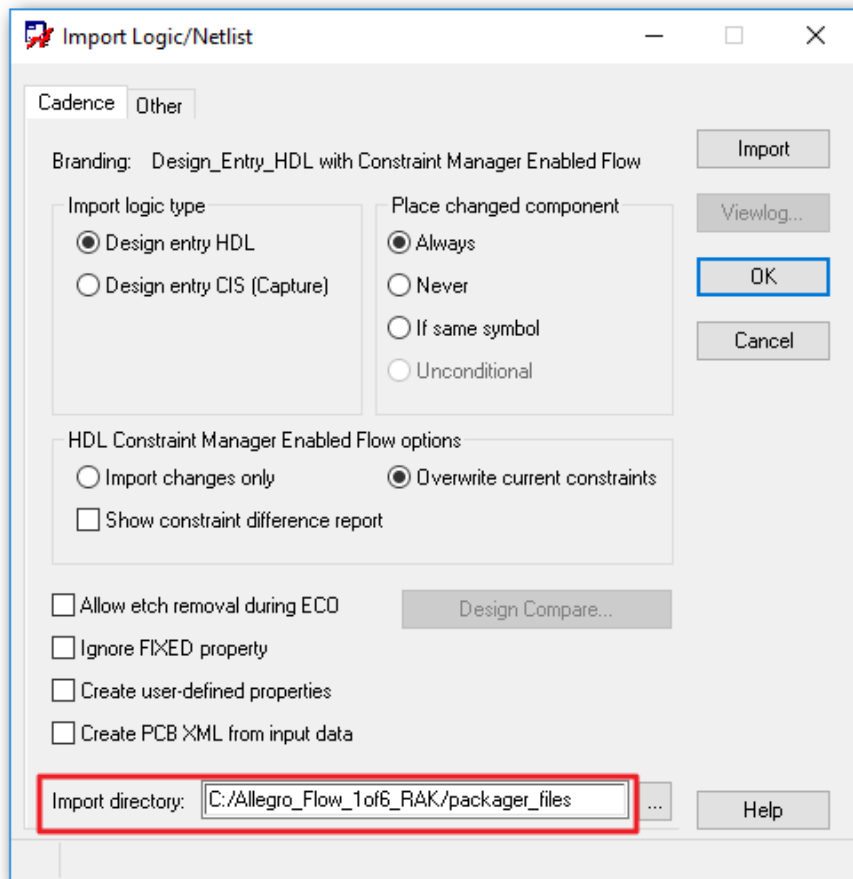
The **Place changed component** section is used during the ECO process to determine how placed parts are treated when there is a schematic change.

The **HDL Constraint Manager Enabled Flow options** section is also used during the ECO process to specify if only changed constraints should be imported or if all constraints in Allegro should be overwritten by the schematic constraints.

3. Select **Import**.

4. Select **File > Viewlog** to view the **netrev.lst** file. This file will include any errors or warnings that were encountered when reading in the netlist.

5. Close the log file.

6. To see the netlist, use **Tools > Quick Reports > Net List Report** or **Tools > Quick Reports > Netin** (non-back).

7. Exit Allegro. You do not need to save the design.

Support

Cadence Support Portal provides access to support resources, including an extensive knowledge base, access to software updates for Cadence products, and the ability to interact with Cadence Customer Support. Visit *https://support.cadence.com*.

Feedback

Email comments, questions, and suggestions to content_feedback@cadence.com.