cadence®

Embedded Component Design Best Practices

Product Version 23.1
October 2023

Cadence Design Systems, Inc., 555 River Oaks Parkway, San Jose, CA 95134, USA

Contents

Embedded Component Design 5
Introduction 5
Licensing 6
Terminology 7
Front to Back Flow Considerations 8
Drawing-Level Property Alternative 8
Applying a Drawing-level Property 8
Property List 10
PCB Library Considerations 11
Embedded Layer Setup 12
Summary 20
Adding Embedded Properties 21
Embedded Component Parameters 23
Direct Attach Method 23
Indirect Attach Method 25
Embedded Component Constraints 26
Summary 29
Embedded Geometry Subclasses 29
Placement Applications 32
Interactive Placement 32
Quickplace 34
Placement Edit Application Mode 35
Cavities 36
Reports 39
Manufacturing Output 41
SKILL 43

1

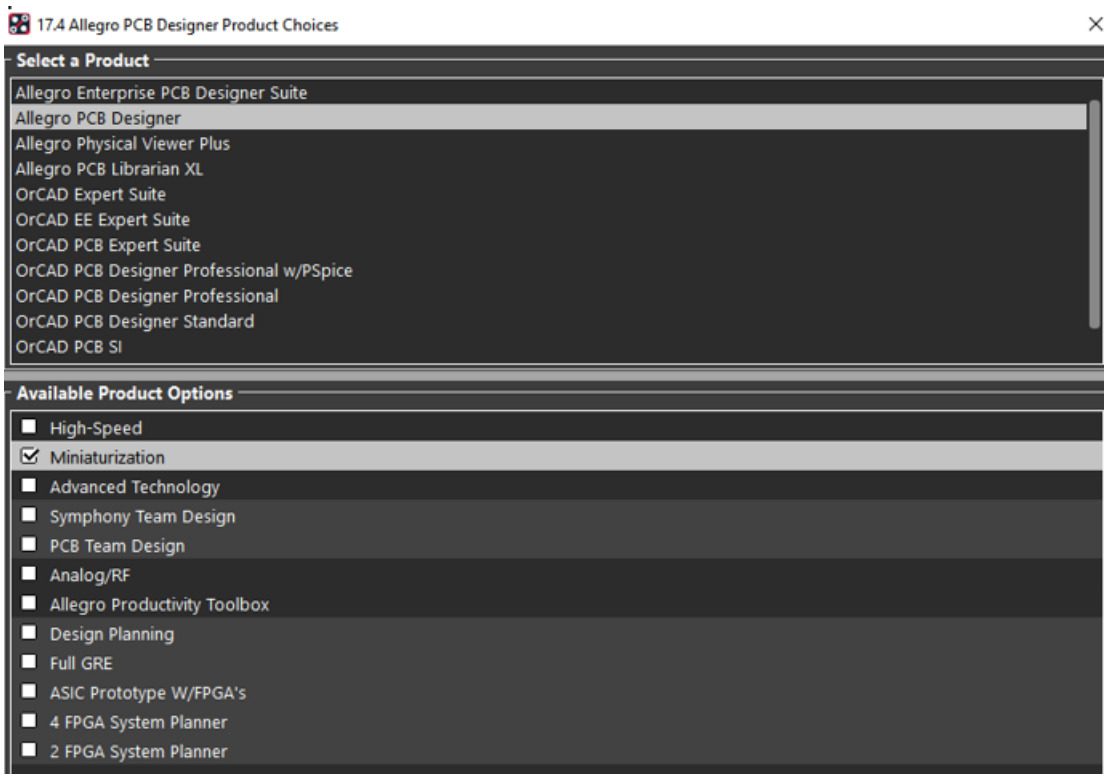
Embedded Component Design

Introduction

With increased market demands for smaller and lighter products, improved performance and higher speeds, it may become necessary to consider the embedding of passive or even active components within the inner substrates of the PCB. If you are designing product that essentially can be held in your hand, perhaps ones used in mobile applications or a consumer electronic device like a digital camera, embedded component technology may be in your product road map plans. This chapter covers the best practices for embedded component design using the Allegro PCB Editor. It may or may not be closely linked to the processes used by the fabricator you partner with. It is likely that your fabricator has patented a process for embedded component manufacturing. The methods on how components are mounted and logically connected to the formation of cavities may differ from vendor to vendor. As always, the best advice for advanced PCB Design whether that be HDI, Flex or Embedded is to work closely with the fabricator who may own the Allegro layout design tools. They can advise on the proper parameter and constraint settings that best accommodate their process flow.

Licensing

Embedded Component Design is enabled using either the Enterprise, Venture or Miniaturization product option.



Terminology

Direct Attached (soldered)	The manufacturing technology where the components are soldered directly to an internal layer. One way to visualize this is to think of assembling a traditional PCB with the components on the external surface(s) and then laminating more layers on top of the components.
Indirect Attach (glued)	The manufacturing technology where the components are suspended in the dielectric material between the layers. The electrical connections are made by creating holes through the layers to the component pins and then plating those holes.
Body Up	The packaged part body is oriented toward the Top surface of the PCB. Fabricators may call this *Face Up*.
Body Down	The packaged part body is oriented toward the Bottom surface of the PCB. Fabricators may call this *Face Down*
Cavity (closed)	The space around the embedded component in the dielectric between two etch layers. The XY dimensions of the cavity are driven by the size of the component and other manufacturing rules. In most applications, the cavity will be between two adjacent layers; however, multilayer cavities will be supported.
Cavity (open)	A blind hole in the substrate in which components are placed.This hole is open to one of the external substrate surfaces and may be several layers deep. The cavity will often have progressively smaller lengths and widths from the external surface to the depth of the cavity.
IPC-7092	Standard for Embedded Device Printed Circuit Boards (In process)
Hermes	High-density integration by Embedding chips for Reduced size Modules and Electronic Systems. EU-funded project designed to enhance the performance of PCBs by using embedded chips to integrate additional functions.

Front to Back Flow Considerations

The schematic to PCB design flow may differ between companies as well as between PCB and Packaging Design. For that reason, you may choose to enable the use of embedded components from the front-end Cadence tools or to exclude front-end requirements and drive completely within the PCB Editor or Packaging tools.

The overall functionality associated with Embedded Component Design is largely contained in the backend physical products. However, the primary method that enables a component to be an embedded candidate is driven from a component definition or instance level property called EMBEDDED_PLACEMENT. This property can be applied at the schematic level, thus enforcing the front-to-back flow restrictions in the design process. Alternatively, it can be applied with the physical backend editors.

The EMBEDDED_PLACEMENT property supports three values; REQUIRED, OPTIONAL and EXTERNAL ONLY.

- REQUIRED: Use to ensure components targeted for embedded applications are placed on internal layers only. These components may be more expensive and designed for certain embedded applications; for example, copper tipped leads. They may also come with a longer availability lead time and require advance planning. This property can be considered as a “hard” property. A DRC will appear if placed on the surface layers.
- OPTIONAL: The designer can optionally place the components on the surface or internal layers. These components are of the generic type, ones that probably exist in your component library. This property can be considered as a “soft” property.
- EXTERNAL ONLY: This is a limited use property. Consider applying to exception components when using the drawing-level property EMBEDDED_SOFT.

Drawing-Level Property Alternative

If you want to drive the solution entirely from the backend and want to use your own discretion during component placement, consider applying the drawing-level property EMBEDDED_SOFT. With this property enabled at the drawing level, you are free to place any component in the database to an internal layer. This in fact may be the best practice for packaging designers.

Applying a Drawing-level Property

1. Choose *Edit – Properties*.
2. Open *Find* Filter and select *Drawing* in the *Find by Name* field.

The Property dialog displays.

3. Select *Embedded_Soft* from the available list.
4. Select a Value of *True*.
5. Click *OK* to exit the form.

You can create a function/alias key to quickly launch the drawing property editor. The following is an example that you can copy/paste into `env` file:

Example: Function key assignment

```
funckey e "property edit;
setwindow form.find;
FORM find name_type Drawing;
FORM find find_by_name;
FORM findname objlist 'Drawing Select';
FORM findname done"
```

Property List

Property Name	Value	Object	Where Applied	Description
Embedded_Placement	Required Optional External_only	Component Instance or Definition	Schematic Editor PCB Editor Package Editor	Component must be placed on embedded layer(s) only, typical in indirect attach methodology. Components may be specially ordered and more costly. Component can be placed on either the surface or embedded layer. Component is restricted to surface layers; primarily used in the backend as an override state. Not expected to be used on a regular basis.

Embedded_Soft	True/False	Drawing	PCB Editor Package Editor	Designer is permitted to place any component on embedded layers. May be the method packaging designers use.
Emb_indirect_padstack	Via padstack name	Drawing	PCB Editor Package Editor	Adds via padstack to all embedded component pin locations; used in Indirect Attach methodologies only.

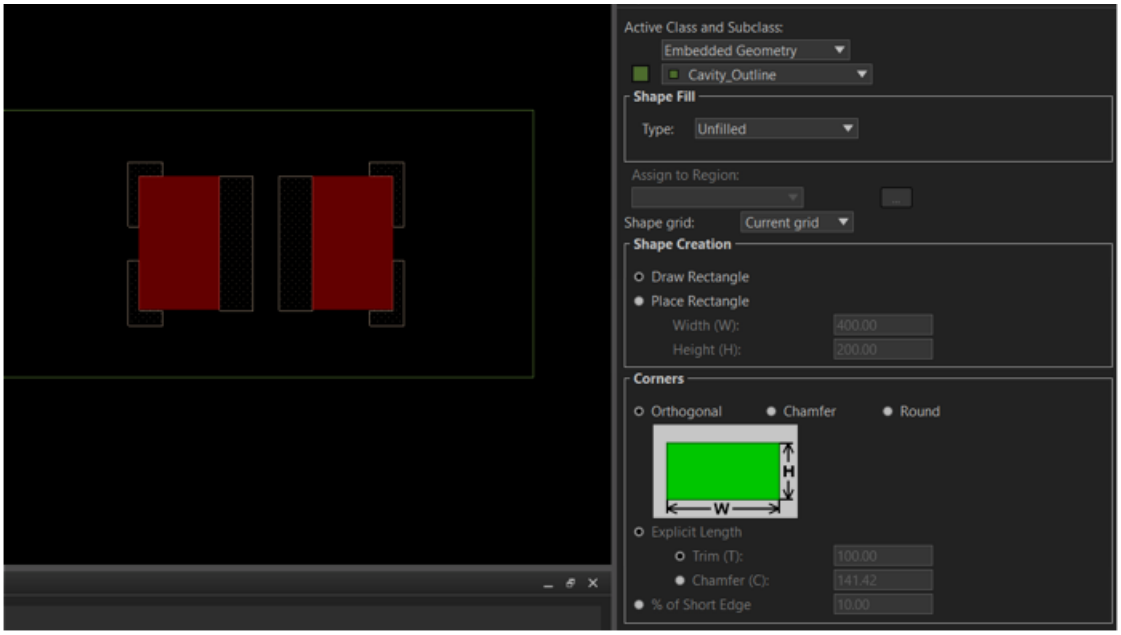
PCB Library Considerations

The Embedded Component flow does not require any modification to your PCB symbol library. It's possible to utilize an existing package symbol such as a 0402 or 0201 capacitor and place the symbol and associated subclass geometry like assembly outline and pastemask to the targeted embedded layer. Cavity formation will be driven from the placebound shape. Controls are provided to extend the cavity clearance beyond the placebound outline.

Some of the factors that can drive custom library symbols for embedded applications are as follows:

- Desire to build cavity outlines into the package symbol instead of using the placebound shape as the source. In the following example, an unfilled shape on Embedded Geometry class and Cavity_Outline subclass is added to the symbol.

- Before signing up for cavity support, contact fabricator and inquire to what degree of accuracy will be used for design intent for cavity milling. The use of the placebound shape may suffice; at least for the initial PCBs you design.



- Use of alternate symbols for embedded placement
  - The property syntax for alternate symbols has been modified to include an optional INTERNAL (or I) keyword followed by a list of symbols allowed on an internal layer. This is an optional syntax. By default, any symbol on the any list is allowed on an internal layer. Placement on embedded layers is also controlled with the EMBEDDED_PLACEMENT and the ALT_SYMBOLS_HARD properties.
  - Alt Symbols Hard: When present on a component, ignores JEDEC_TYPE for internal placement. Uses symbols listed in the component's ALT_SYMBOL property sections of any or INTERNAL layer for embedded placement.
- Special order components with unique symbol characteristics.It may be associated with use of *Indirect Attach* methodology.
- All library package symbols contain pin escape vias. Library-driven symbol vias will be removed when the component is placed on an internal layer.

Embedded Layer Setup

Upon completion of your property assignments, the next step in the embedded component flow is to enable layer(s) of the stack-up as embedded. The Cross Section Editor (since 17.2)

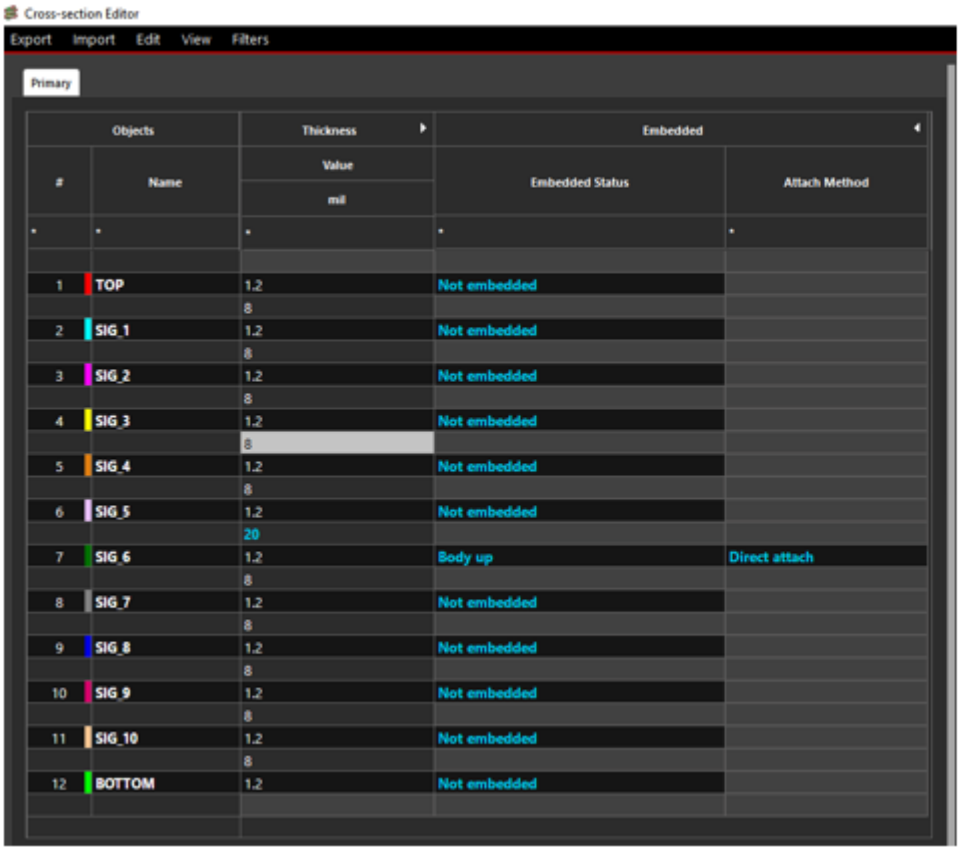
drives the definition of embedded layers as well as component direction and global parameters. Following examples explains some popular embedded component strategies that are supported in the PCB Editor.

Single Layer Mounted

This may be the most common methodology at the time. Components are directly soldered or glued on a single base layer. Dielectric thickness supports the height of the tallest component mounted.

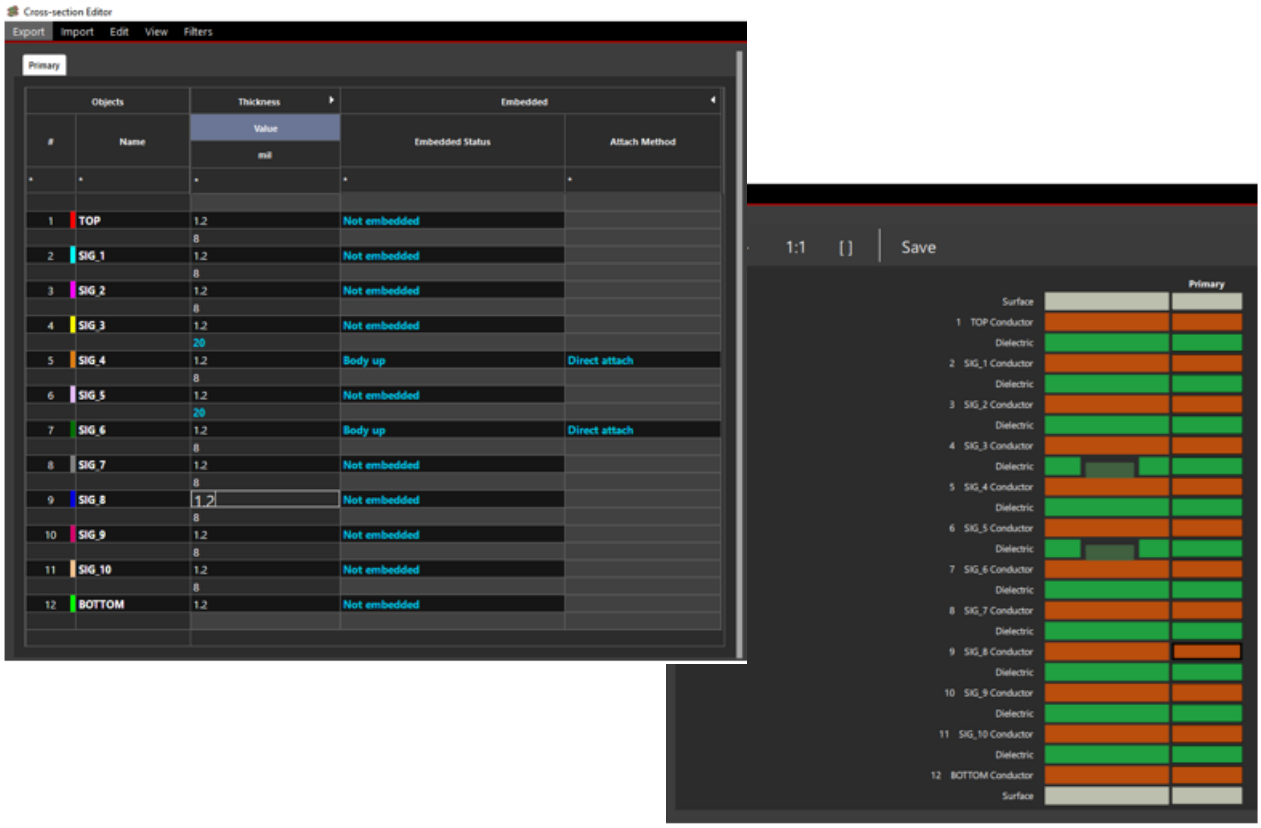
- *Embedded Status* is set to either *Body up* or *Body Down*

■ *Attach Method* is set to *Direct Attach* or *Indirect Attach*



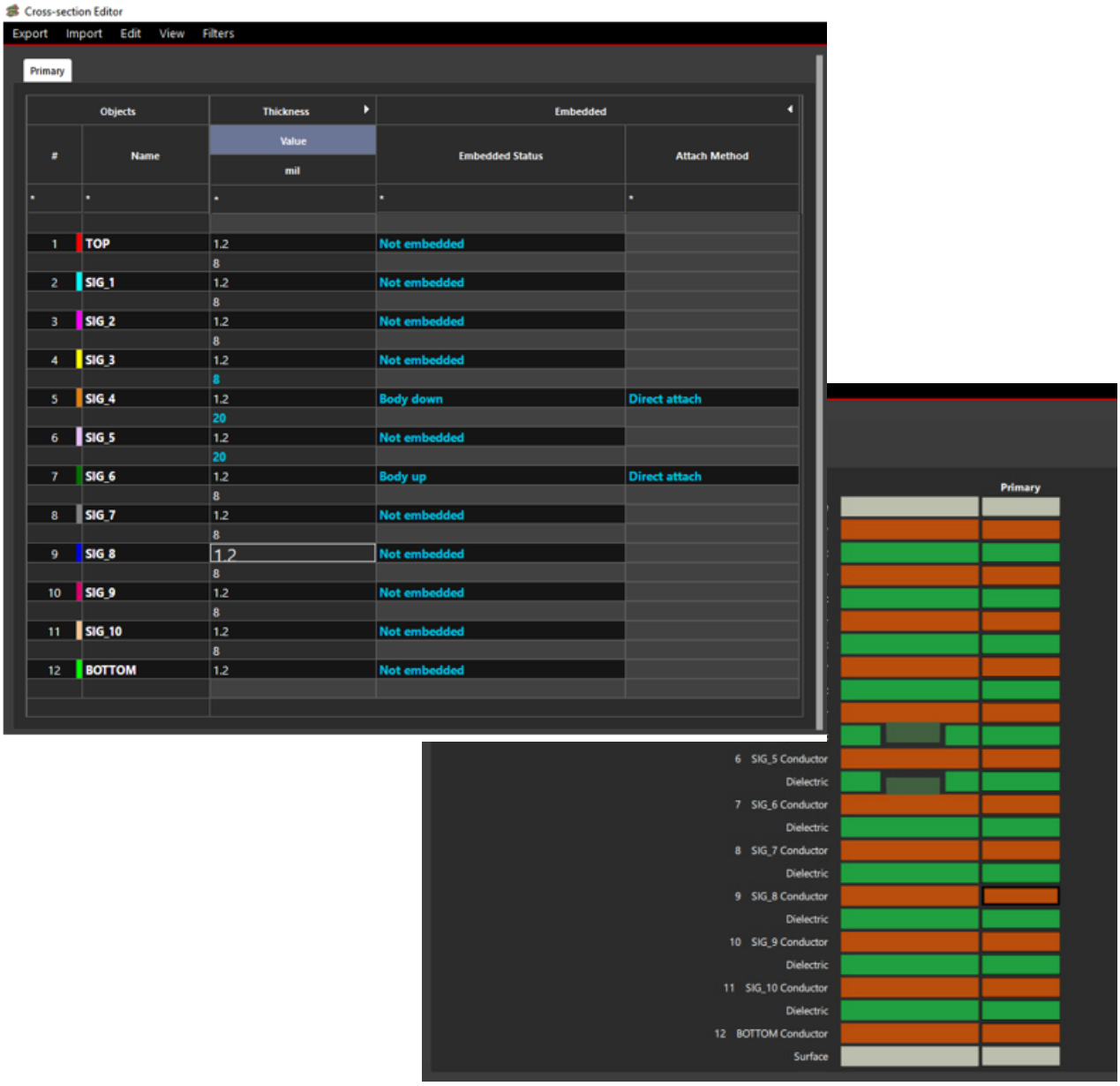
Dual Layer Mounted (Separate Cores)

Components are placed on two internal layers. This process needs to be weighed against utilizing a single core for placement.



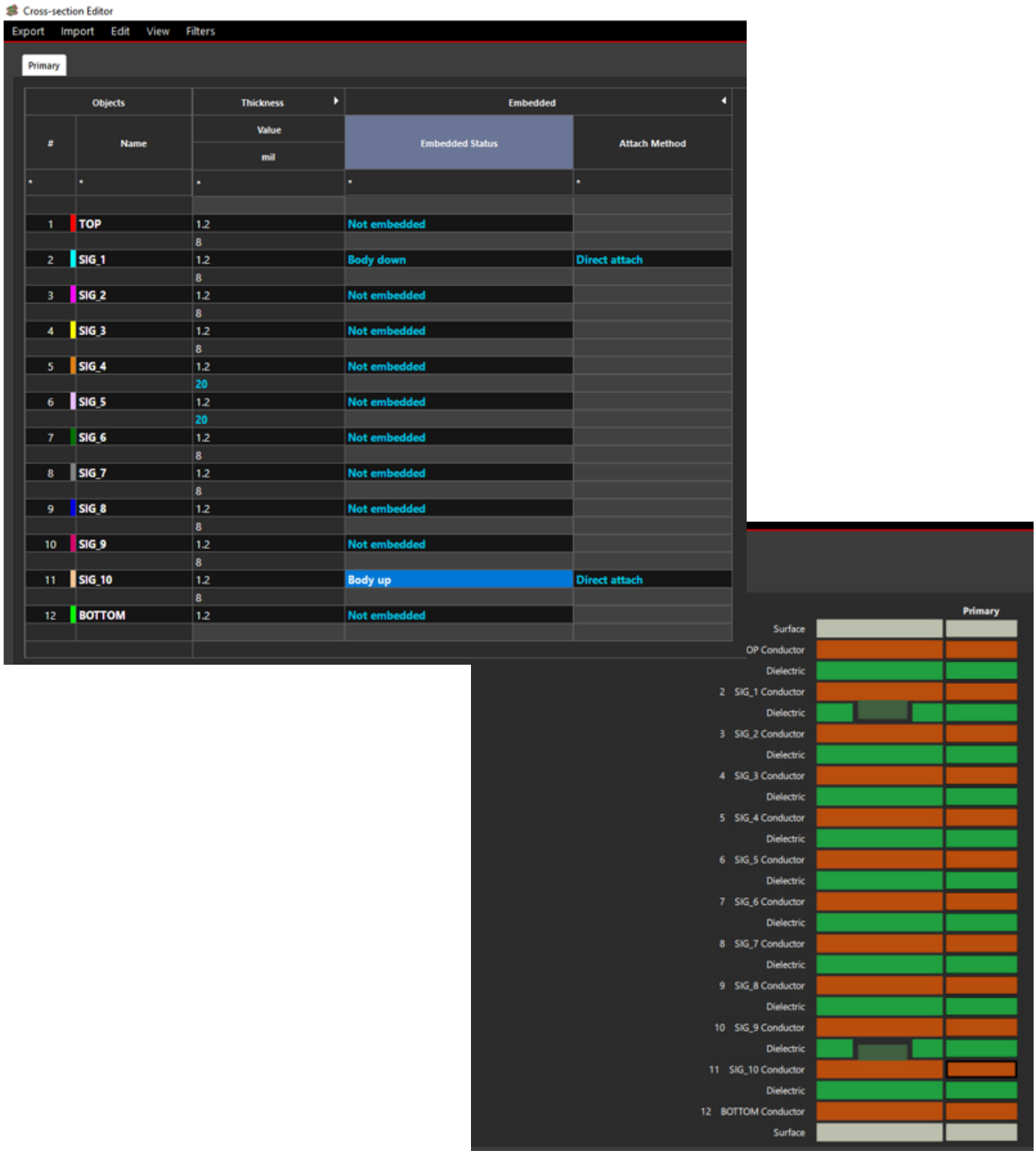
Dual Layer Mounted (Same Core)

Components are placed on two layers comprising the core. Components may need to offset to prevent collisions in the Z direction.



Dual Layer Mounted (Separate Cores But Near Surface Layers)

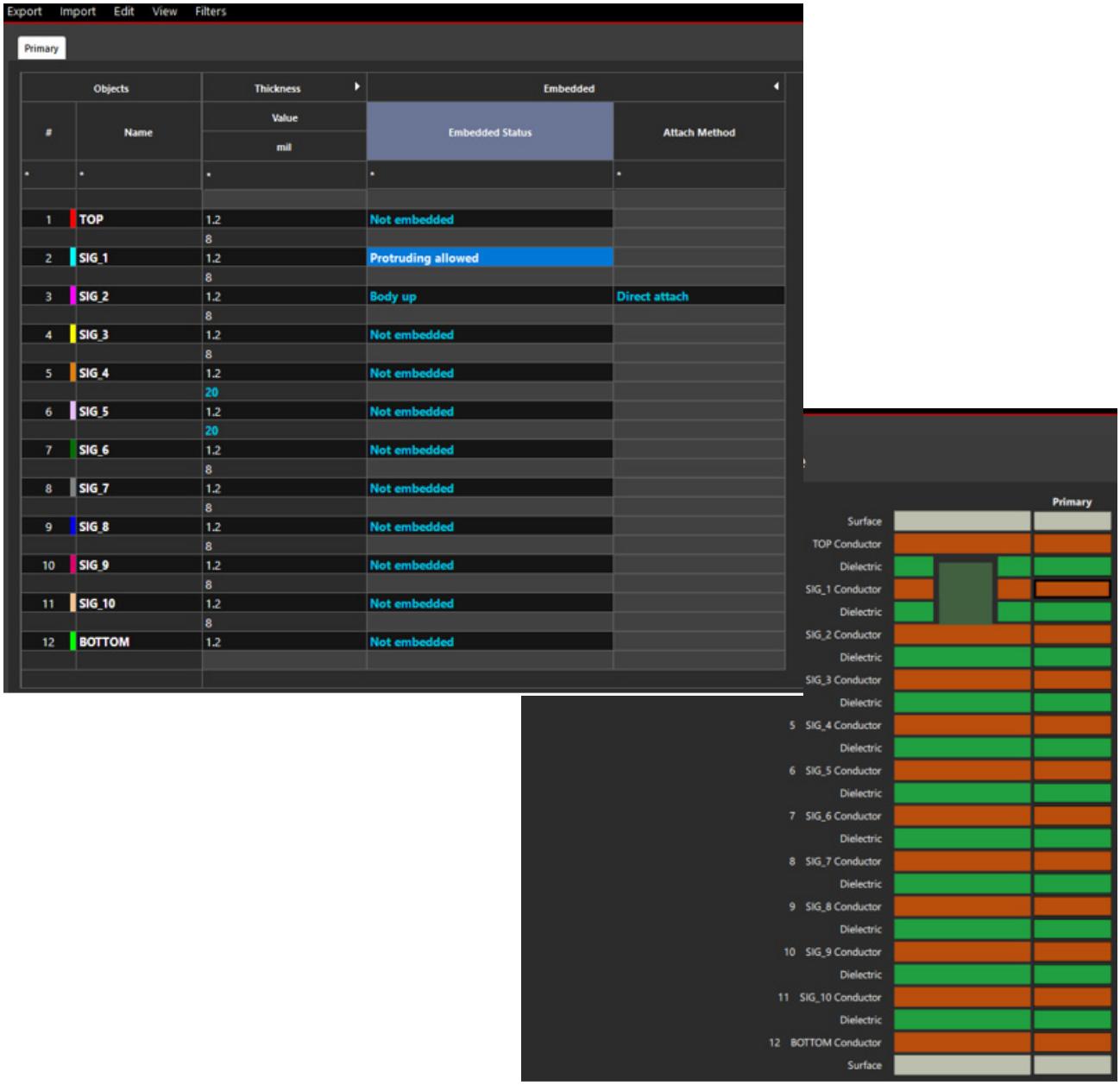
Components are placed on two separate layers near the PCB surface. This reduces the signal length to the component pins.



Single Layer Mounted (Dielectric Thickness Does Not Support Height of Components)

Components are placed on a single layer. Dielectric thickness is too thin to support component height.

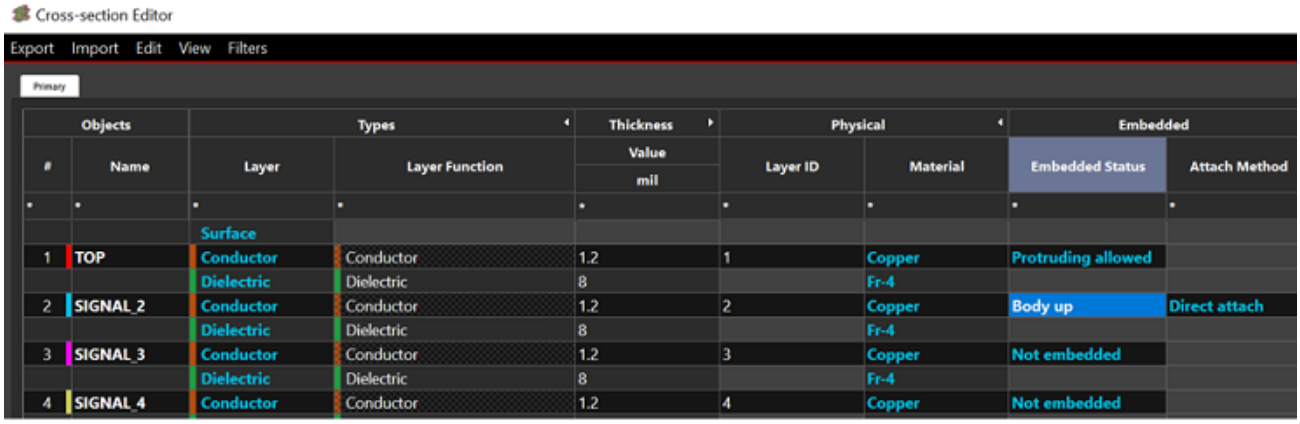
- ❑ Embedded status is set to either *Body up* or *Body Down*"
- ❑ Adjacent signal layer set to *Protruding allowed*. This permits a component to cut across an adjacent signal layer as shown in the bitmap.



Single Layer Mounted (Open Cavity)

Components are placed on a single layer near the surface of the PCB. The cavities about the components are “open” allowing the components to extend beyond the surface layer.

- ❑ Embedded status is set to *Body up*
- ❑ TOP layer is set to *Protruding allowed*. This permits components to cut across the surface layer into air.

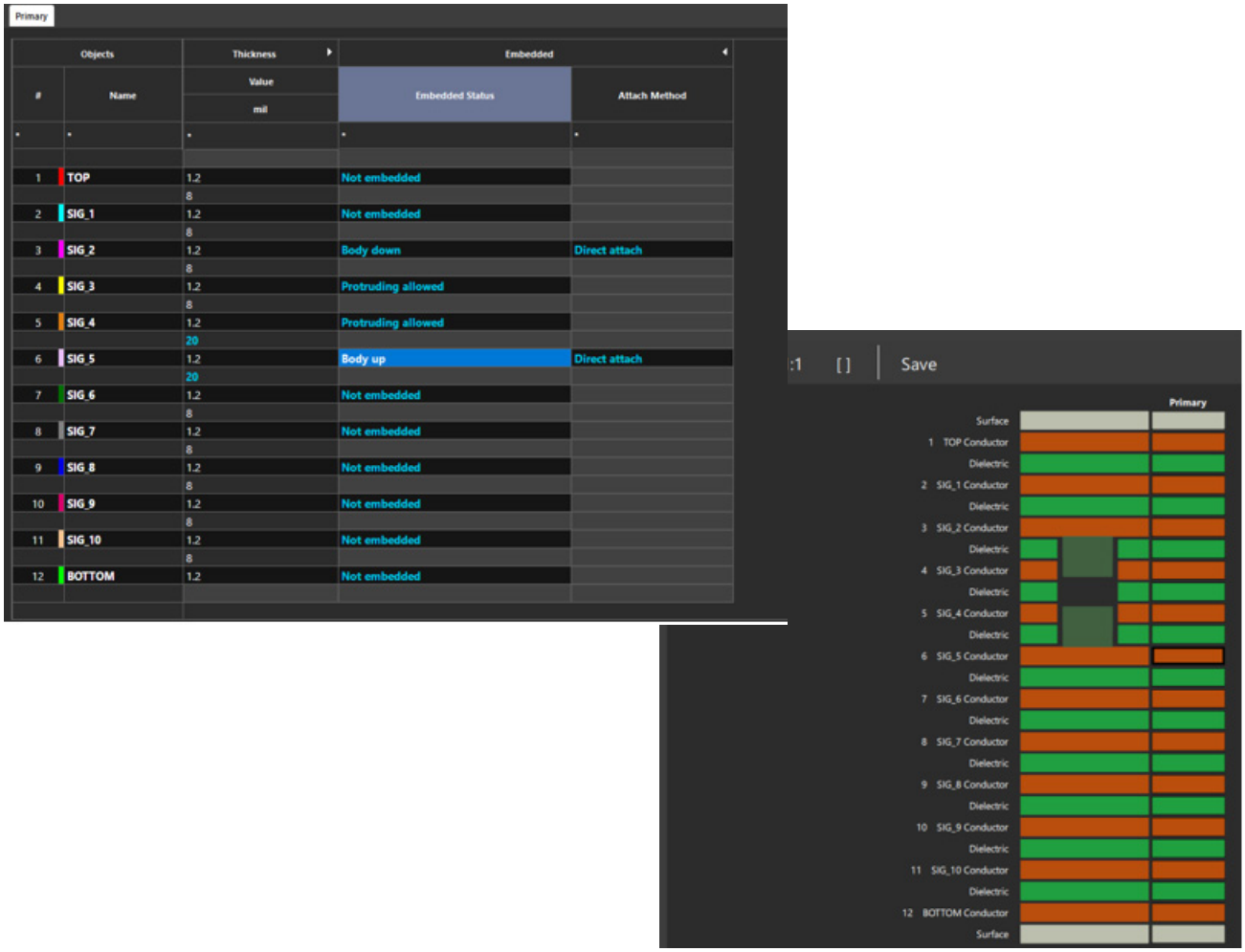


Objects		Types		Thickness	Physical		Embedded	
#	Name	Layer	Layer Function	Value mil	Layer ID	Material	Embedded Status	Attach Method
		Surface						
1	TOP	Conductor	Conductor	1.2	1	Copper	Protruding allowed	
		Dielectric	Dielectric	8		Fr-4		
2	SIGNAL_2	Conductor	Conductor	1.2	2	Copper	Body up	Direct attach
		Dielectric	Dielectric	8		Fr-4		
3	SIGNAL_3	Conductor	Conductor	1.2	3	Copper	Not embedded	
		Dielectric	Dielectric	8		Fr-4		
4	SIGNAL_4	Conductor	Conductor	1.2	4	Copper	Not embedded	

Dual Layer Mounted (Multiple Cores)

Components are placed on two separate thin cores. A two layer primary core separates the thin cores.

- Core layers (SIG_3and SIG_4) are set to *Protruding allowed* to permit components in that space.



Summary

The minimum requirements for embedded component design are:

- Assign the EMBEDDED_PLACEMENT property to components targeted for embedded applications.
- This property can be assigned at the schematic or layout level.
- A drawing level property, EMBEDDED_SOFT can be applied, but caution is advised. When using this methodology, any and all components in the database are candidates for embedded applications.

- Use Package symbol library as is.
- Construct your embedded component design strategy. This includes deciding on:
  - Embedded status of *Body up* / *Body down*
  - Attachment methods of *Direct* or *Indirect*
  - Whether to use or not use *Protruding allowed* to extend cavities into adjacent layers
- Use *Setup – Cross-section* to access the setup form.

With the above requirements completed, you can now place components on the layers tagged as embedded. Parameter and constraint considerations are covered in the following sections.

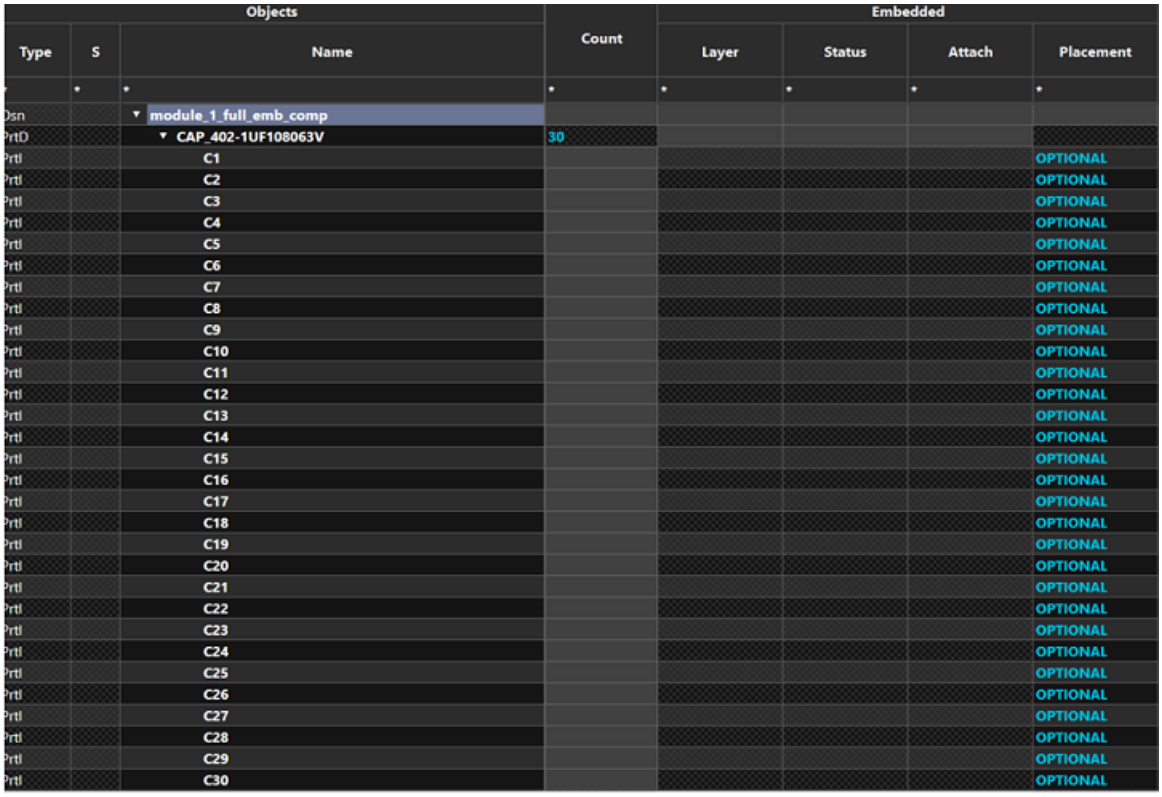
Adding Embedded Properties

Applying properties in Allegro layout editors has historically been done using *Edit – Properties* in combination with the appropriate *Find* filter setting and then selection of elements in the workspace.

Use Allegro Constraint Manager to add the EMBEDDED_PLACEMENT property to the relevant components targeted for embedded applications.

1. Start Allegro Constraint Manager.
2. Open the *Properties* domain.
3. Open the *General* worksheet located in the *Component* workbook section.

4. Apply property value to each component instance as shown in the following image.

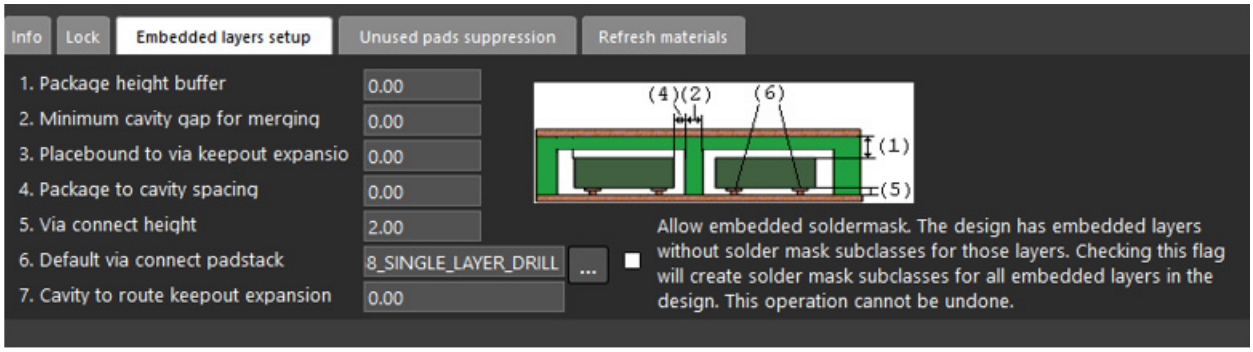


Objects			Count	Embedded			
Type	S	Name		Layer	Status	Attach	Placement
Dsn		module_1_full_emb_comp					
PrtD		CAP_402-1UF108063V	30				
Prtl		C1					OPTIONAL
Prtl		C2					OPTIONAL
Prtl		C3					OPTIONAL
Prtl		C4					OPTIONAL
Prtl		C5					OPTIONAL
Prtl		C6					OPTIONAL
Prtl		C7					OPTIONAL
Prtl		C8					OPTIONAL
Prtl		C9					OPTIONAL
Prtl		C10					OPTIONAL
Prtl		C11					OPTIONAL
Prtl		C12					OPTIONAL
Prtl		C13					OPTIONAL
Prtl		C14					OPTIONAL
Prtl		C15					OPTIONAL
Prtl		C16					OPTIONAL
Prtl		C17					OPTIONAL
Prtl		C18					OPTIONAL
Prtl		C19					OPTIONAL
Prtl		C20					OPTIONAL
Prtl		C21					OPTIONAL
Prtl		C22					OPTIONAL
Prtl		C23					OPTIONAL
Prtl		C24					OPTIONAL
Prtl		C25					OPTIONAL
Prtl		C26					OPTIONAL
Prtl		C27					OPTIONAL
Prtl		C28					OPTIONAL
Prtl		C29					OPTIONAL
Prtl		C30					OPTIONAL

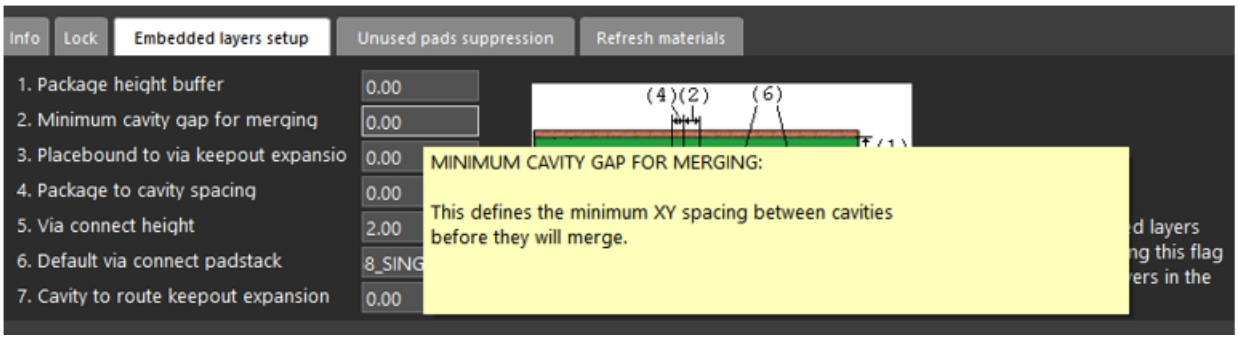
Embedded Component Parameters

Direct Attach Method

Parameters associated with Embedded Component Design are available in the *Cross-section Editor – Embedded Layer Setup* tab.



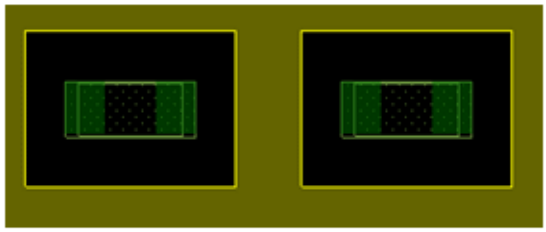
Hovering over the parameter name or value field displays a brief description.



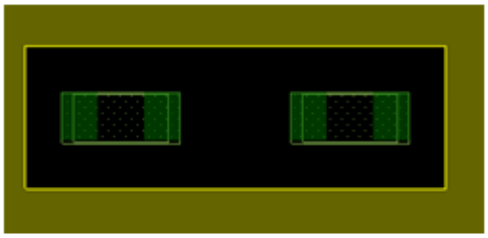
- Package height buffer: (#1) Defines a clearance or buffer when calculating component height violations in a substrate. For example, you may have concerns about placing a `10 mil` tall component in a `12 mil` substrate. The value entered in this field is essentially added to the height of the components placed on the embedded layer.
- Minimum cavity gap for merging (#2): Defines the minimum spacing between the edges of a cavity before a merger takes place.

  Note: Not all vendors support merged or super cavities.It is recommended to verify

fabricator's process before using cavities.

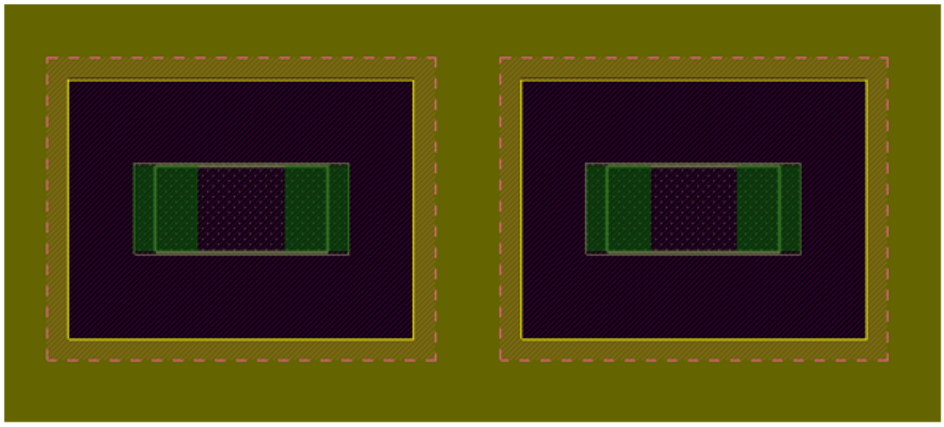


Cavity separation greater than parameter value



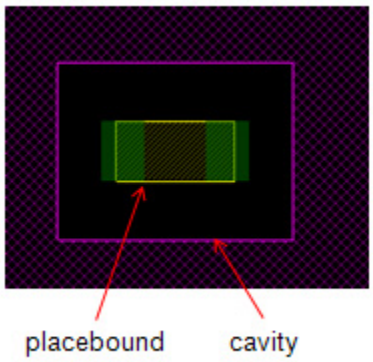
Cavity merger – separation at parameter value or less

- Placebound to via keepout expansion: (#3) Creates an extended via keepout area about the placebound geometry. Follow the guidelines recommended by fabricator.

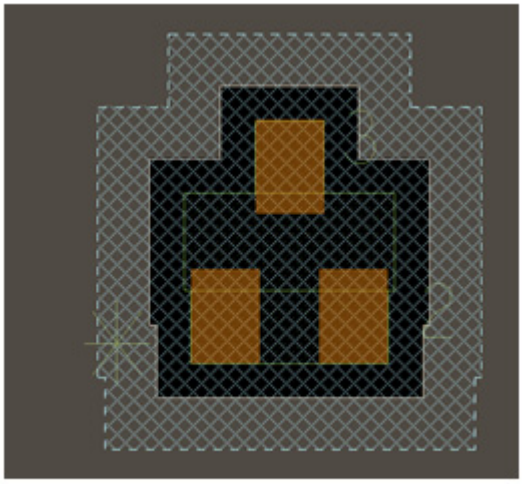


Optional Via Keepout Area extended beyond cavity area

- Package to Cavity:(#4) This parameter is actually a constraint and defines the clearance from the edge of the placebound shape to the cavity outline.



- Cavity to Route Keepout Expansion:(#7) This parameter becomes relevant when a layer is set to *Protruding allowed*. While the cavity has inherent route keepout behavior, it may be desirable to extend the keepout area beyond the cavity profile.

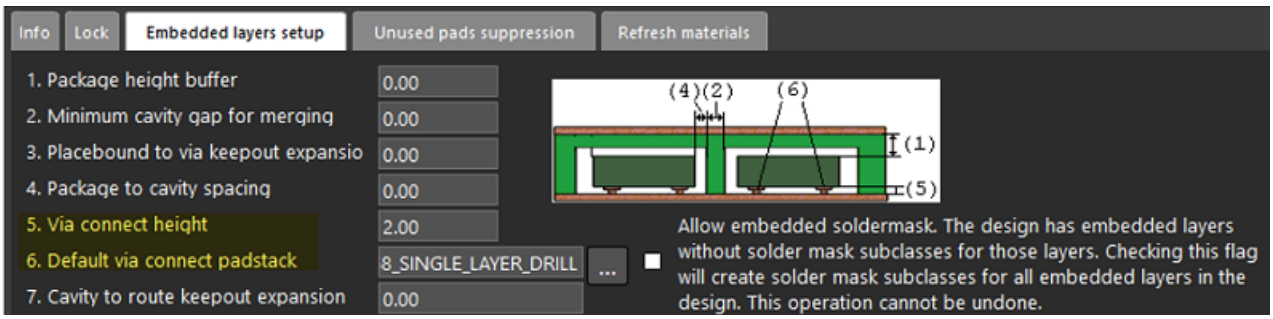


Expanded Route Keepout area on protruding layer(s)

Indirect Attach Method

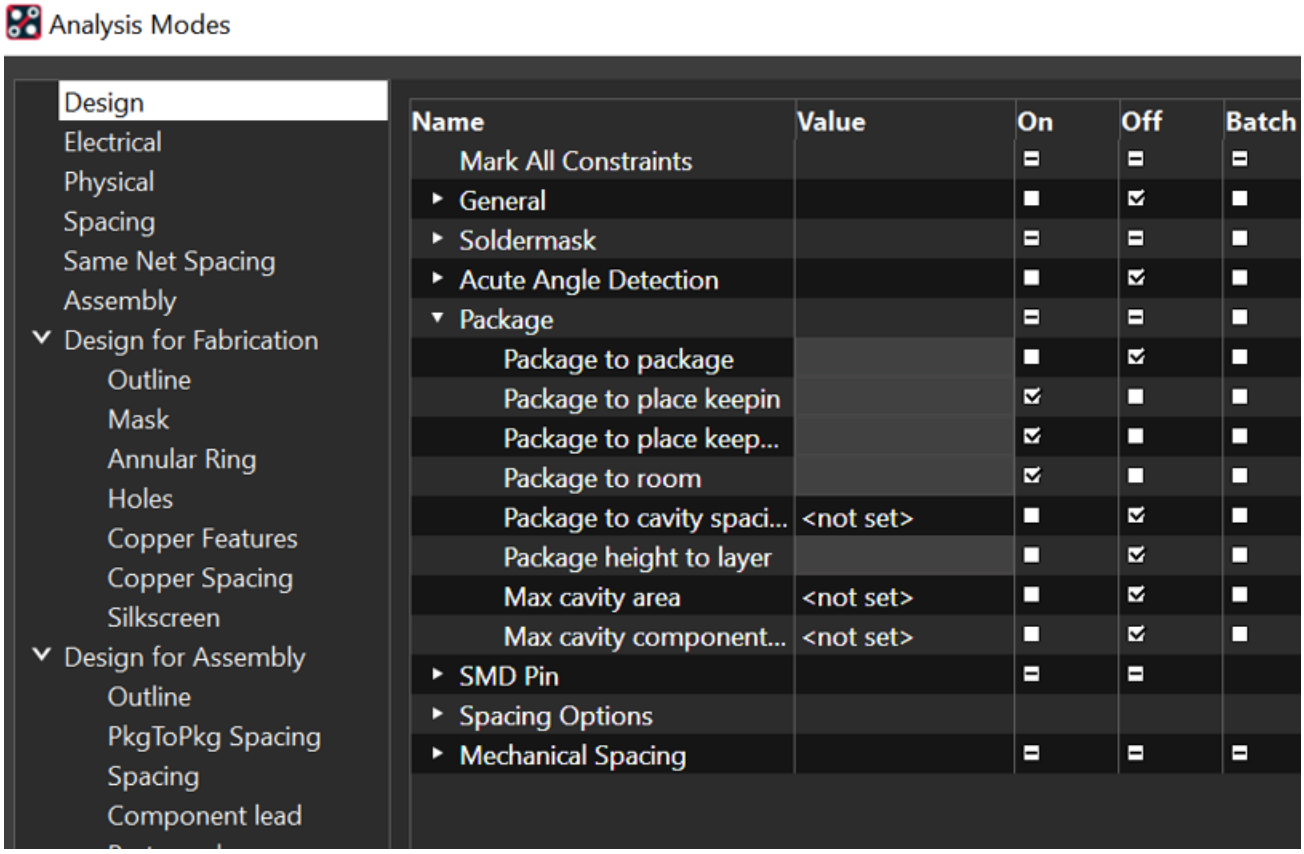
The two remaining parameters 5 and 6 are editable when *Indirect Attach* is selected as the attach method.

- Via connect height (#5): Defines the height of the single layer via used to connect to the component mounted in the substrate. Essentially, this value is the distance the component sits off the foil. This value may be seen as negligible if ample space in the vertical direction is available.
- Default via connect padstack (#6): This padstack (single layer) serves as the ‘connect pin’ on indirect attached embedded components. It can also be applied as a drawing level property EMB_VIA_CONNECT_PADSTACK.



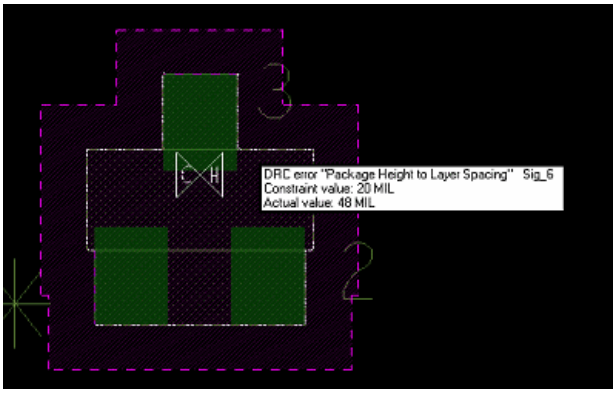
Embedded Component Constraints

Constraints specific to embedded components can be found in *Setup – Constraints – Modes – Design – Package*.

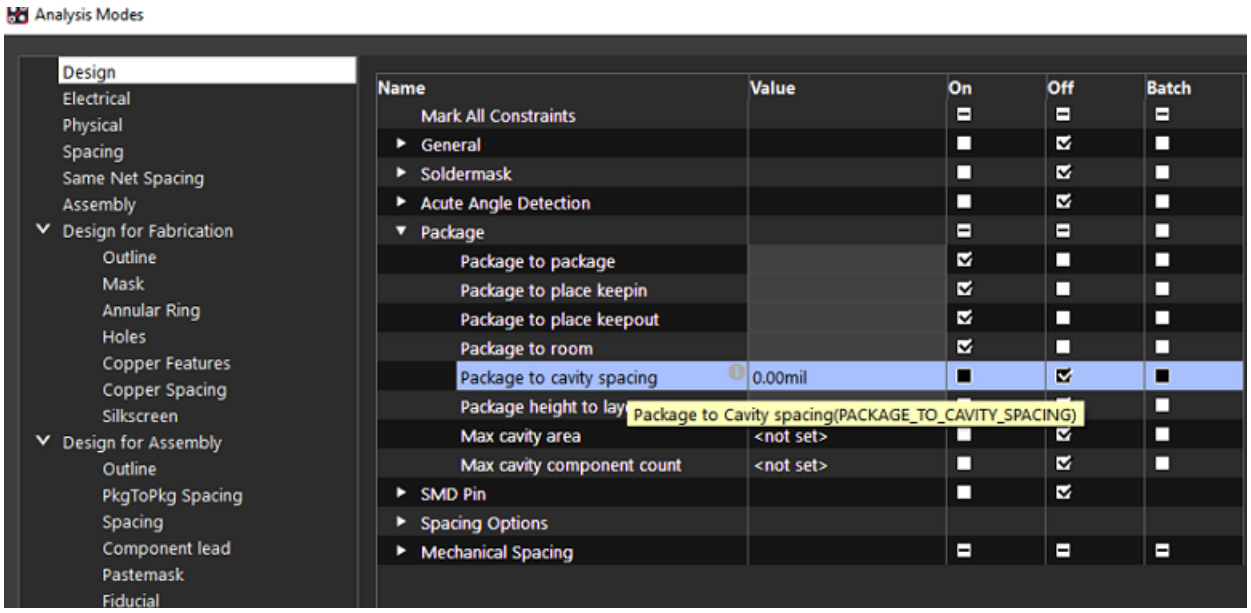


- Package Height to Adjacent layer: Verifies that the package height is less than the thickness of the layer(s) that the cavity spans. The existing package height property is used to store the height value. The thickness of successive layers is used when validating a package place in a multi-layer cavity. For the case of *Indirect attach*, the global parameter value for *Via connect height* will be added to the package height. The *Package Height to Layer* check addresses the possibility that there will be two packages on consecutive embedded layers that have overlapping x-y boundaries. A

DRC violation will be created if the packages intersect in the z-direction. The DRC error code is C-H.



- Edge of Package to Edge of Cavity clearance: Verifies the separation between the edges of a package placebound to edge of cavity outline. The distance can be entered in the Analysis Modes form. If no value is entered, `0` is assumed. The DRC error code is C-S.

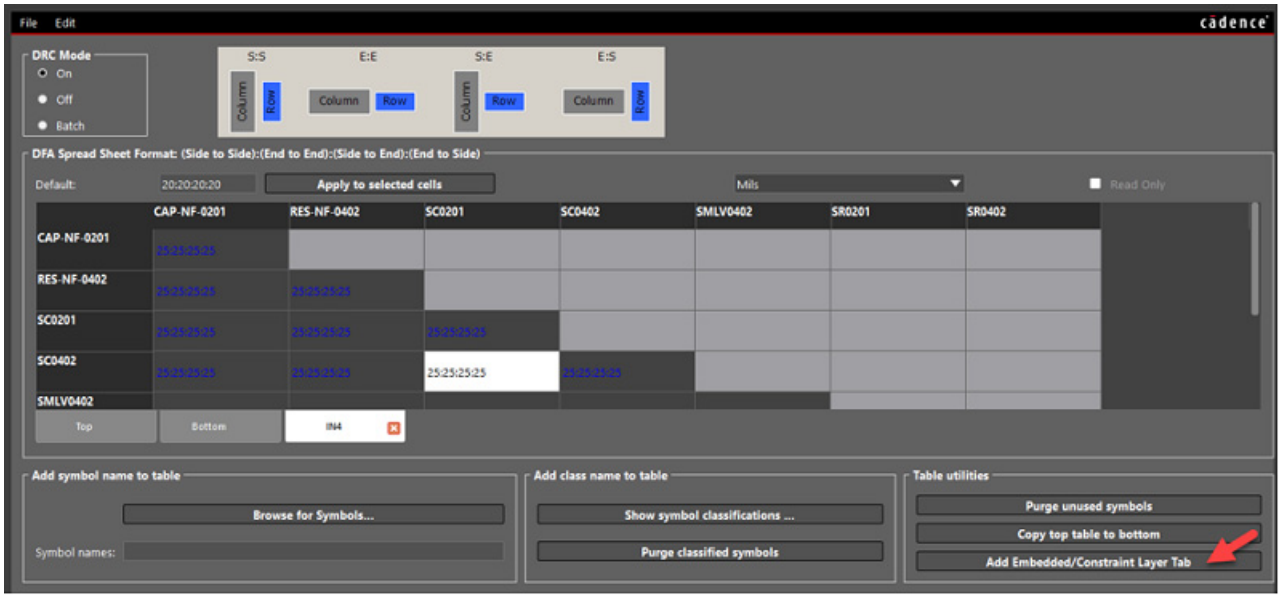


- Illegally Placed Embedded Components: This check runs as part of the legacy package to package DRC check. For example, it flags components with the value of the

EMBEDDED_PLACEMENT property set to `Required` that are placed on the external layers. The DRC error code is C-P.

Component with "Required" property placed on surface layer

- DFA Clearance: By default, the DFA constraints used for embedded component design are those specified for the *Top* layer of the DFA spacing table. To control the spacing of embedded component uniquely, the DFA table supports *Add Embedded Constraint Layer function*. You can constrain all embedded layers the same or at the layer instance level.



- Placebound to Placebound conflicts: Symbol placebound conflicts are handled by existing functionality. The *Package to Package* constraint compares the separation of the respective placebound shapes where an overlap produces a DRC. The check uses placebound shapes that are appropriate for embedded layers.

Summary

Understanding parameters and constraints associated with embedded component design.

- All parameters are located in the bottom section of the Cross-section Editor form. Choose *Embedded Layer Setup* tab to display options.
- Parameter numbers 5 and 6 are reserved for *Indirect Attach*.
- Constraints are used to check for height violations within the substrate, cavity gap from placebound and illegal placement of components.
- Constraint settings are located in *Setup – Constraints – Modes – Design – Package.*
- DFA clearance values are sourced from the *Top* side of the DFA Table by default, but you have the option to control by each embedded layer.

Embedded Geometry Subclasses

The Embedded Geometry supports the following subclasses:

- ASSEMBLY_EMBEDDED_LAYER
- DFA_BOUND_EMBEDDED_LAYER
- DISPLAY_EMBEDDED_LAYER
- PASTEMASK_EMBEDDED_LAYER
- PLACE_BOUND_EMBEDDED_LAYER

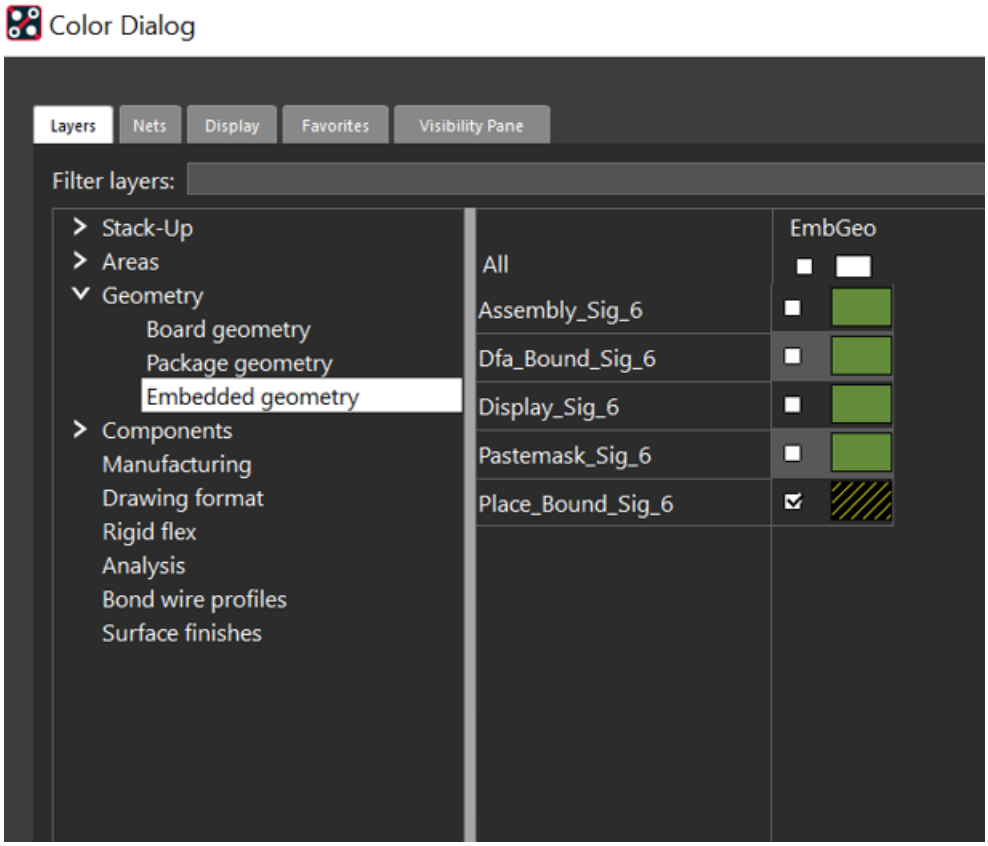
The following existing classes support new fixed subclasses:

Existing Class	New Fixed Subclasses
Component Value	Assembly_Embedded, Display_Embedded
Device Type	Assembly_Embedded, Display_Embedded
Reference Designator	Assembly_Embedded, Display_Embedded
Tolerance	Assembly_Embedded, Display_Embedded
User Part Number	Assembly_Embedded, Display_Embedded

Enabling inner signal layers as embedded *Body Up* or *Body Down* triggers the creation of the necessary subclass support required for placement.

In the following example, layer SIG_6 is enabled for *Body Up/Direct attach* embedded methodology.

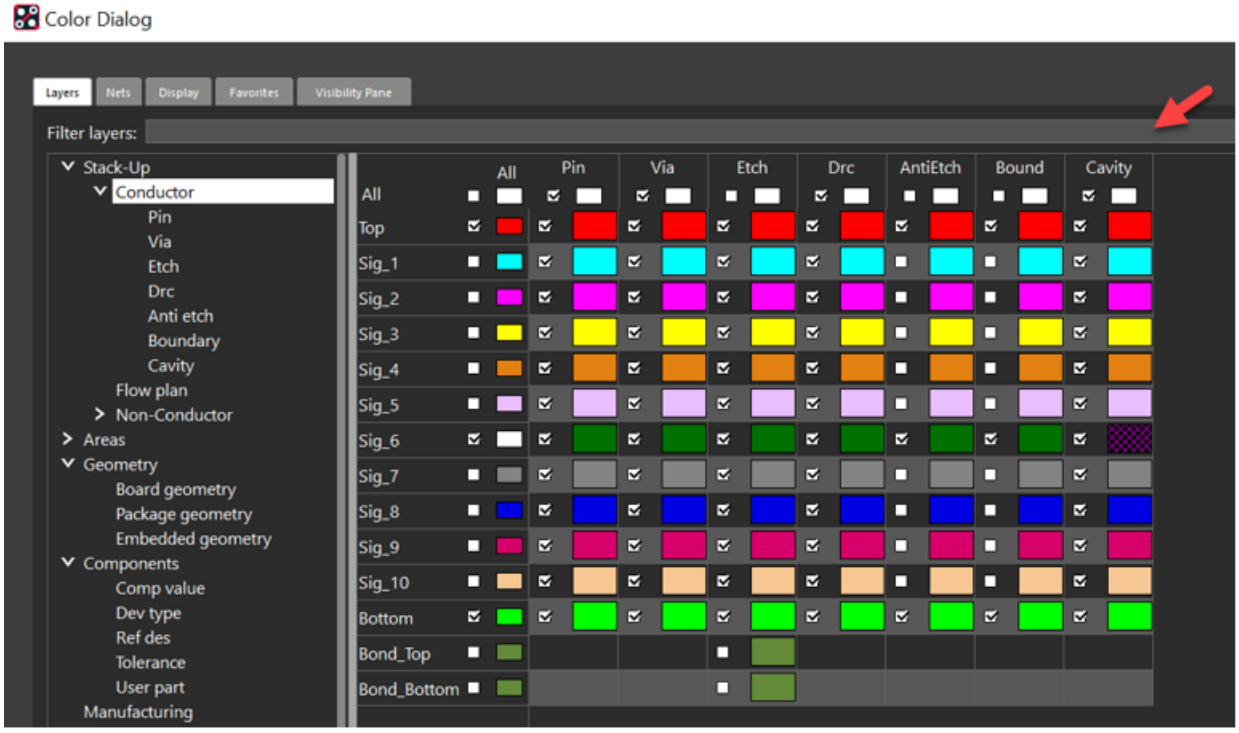
This triggers the creation of 6 subclasses, each with a Signal_6 suffix.



Assembly_Signal_6 and Display_iSgnal_6 subclasses are added to the *Components* folder.

The Cavity subclass is essentially a non-conductive dynamic shape. Voids in the shape are considered cavity objects in the database. When a component is placed on an embedded layer, then auto-generated cavity will be assumed to span the thickness of the dielectric

space. The *Protruding_allowed* option assigned to adjacent layers is designed to extend the cavity height through multiple layers, or to an external surface.



Placement Applications

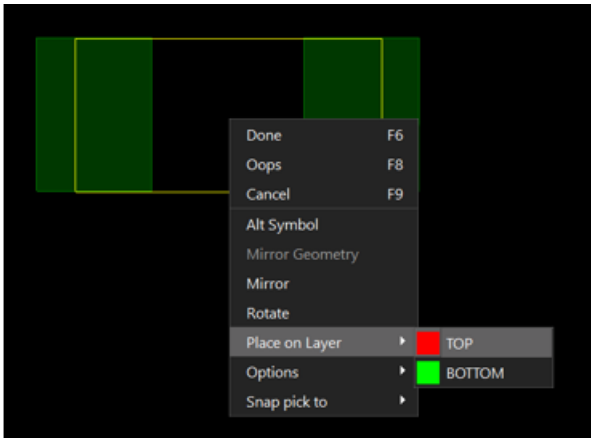
This section focuses on interactive placement using the `place manual` or `move` commands, `quickplace` and context-sensitive use models.

Interactive Placement

The context-sensitive menu of the `place manual` and `move` commands supports the *Place on Layer* function. This function is the primary interactive method to exchange components from the Top or Bottom side of the PCB to internal layers. It can also be used in reverse, to

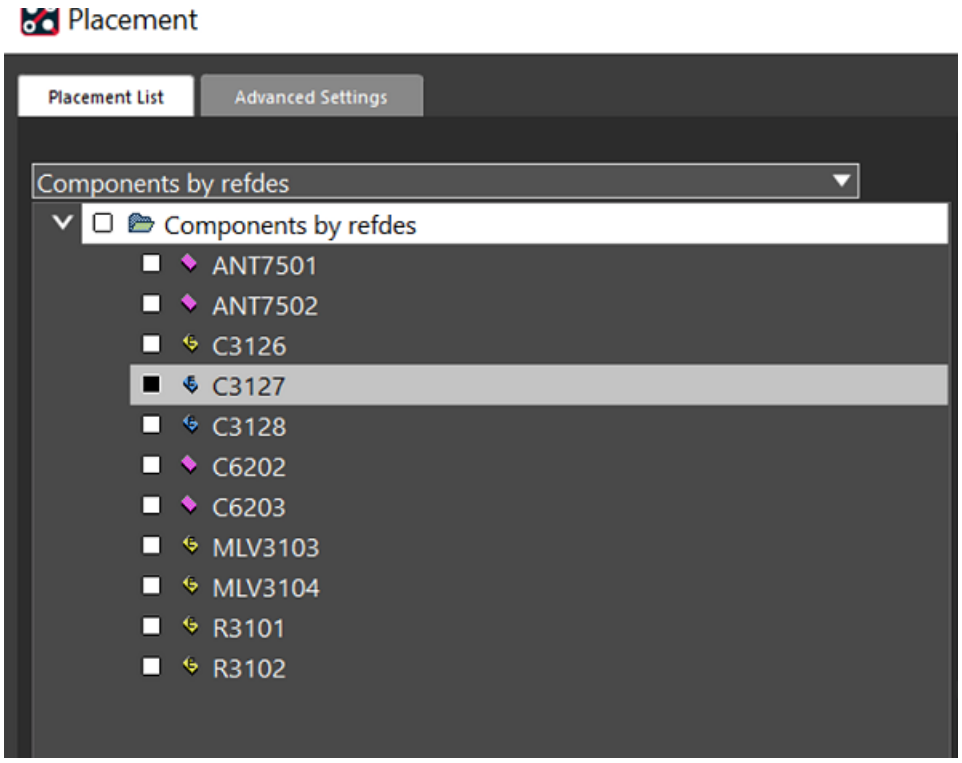
move components from internal to surface layers provided the property value associated with the embedded component is set to `Optional`.

Placing component on Internal Layer



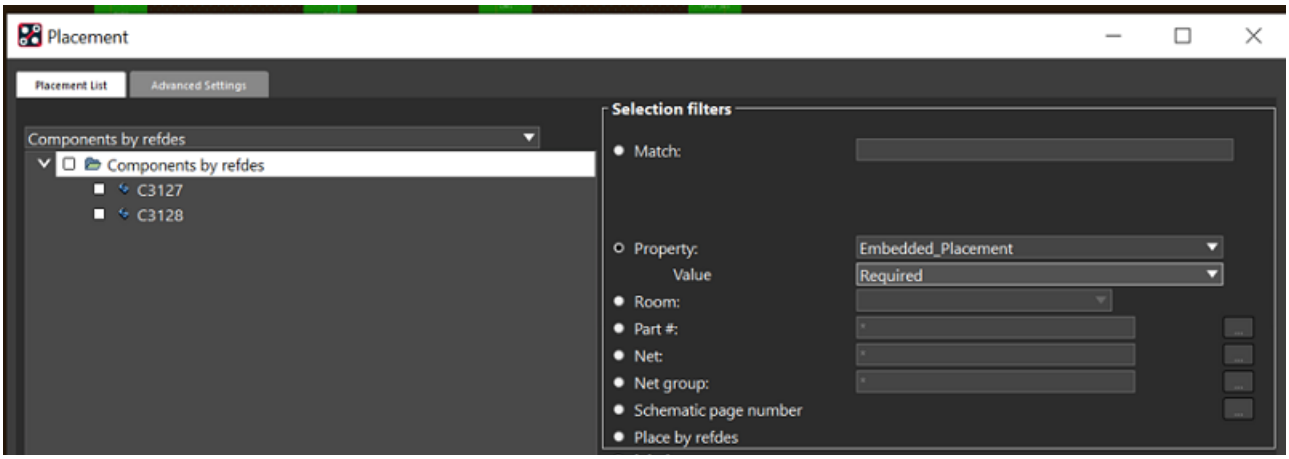
Option to reset embedded component

Embedded components are represented uniquely in the Placement dialog. The letter “E” is used to represent a component that has the embedded placement property assigned to it. The color swatch adds additional differentiation where yellow indicates `Embedded Optional`; blue `Embedded Required`.



Placement form – Embedded attributes

The Placement dialog supports filters to narrow down the list of components for a particular work session. Use the *Property* field in combination with *Value* to filter the component list to embedded components.



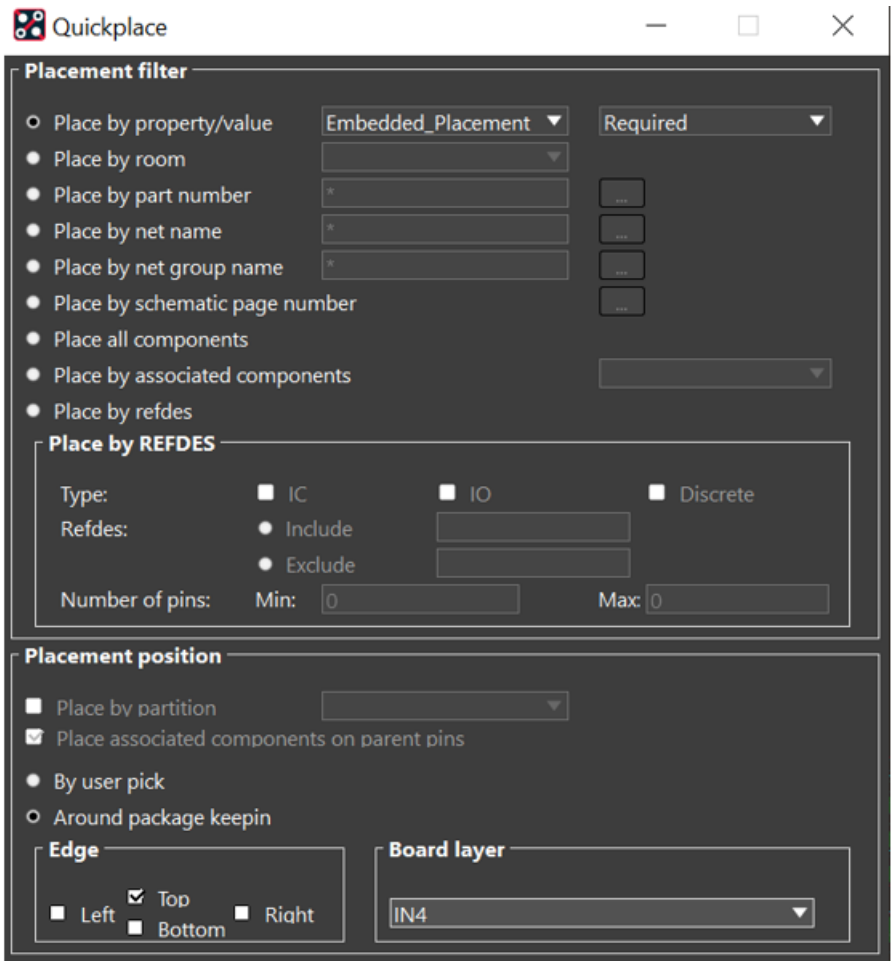
Quickplace

Quickplace is used to quickly disperse components from the placement buffer to user-defined locations; typically around the board perimeter. For embedded design, use this application to quickly place all embedded components to an internal layer. Using quickplace is faster as it does not require right-click to select the *Place on Layer* command.

Tips to enable quickplace for placing embedded components:

- Use *Place by property/value* option, with the property EMBEDDED_PLACEMENT and its value set to `Required` or `Optional`.

- From the *Board Layer* drop-down list, select the embedded layer on which components are to be placed.



Placement Edit Application Mode

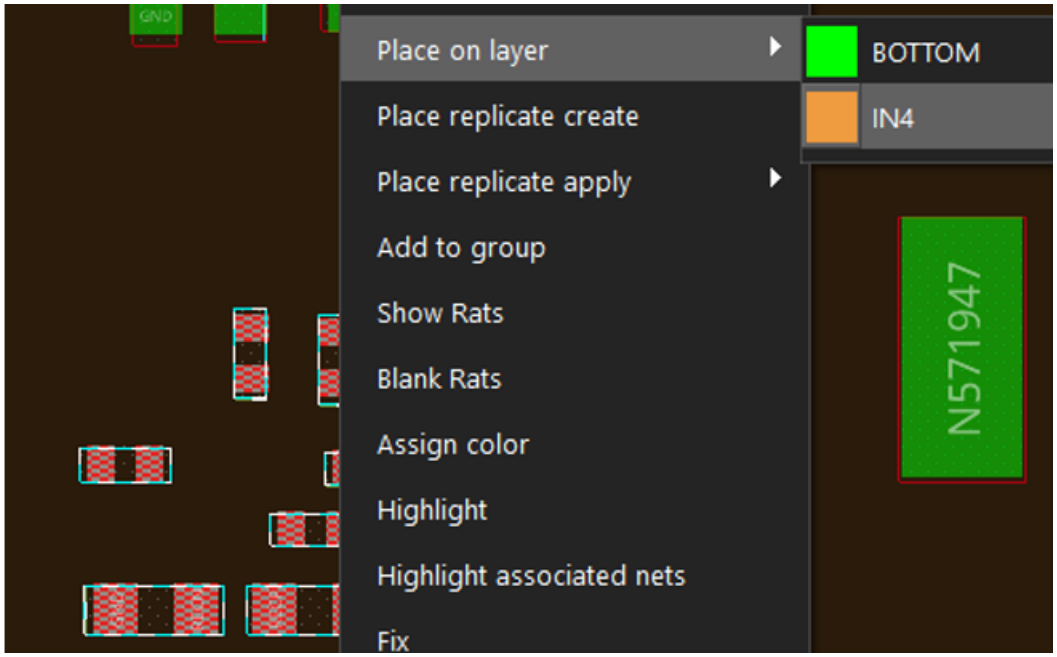
One of the benefits of working in the Placement Edit application mode is the ability to preselect components and then perform an action on them. This mode may become handy when you are embedding components that are already placed on the surface layers of the board. Perform the following steps to pre-select components targeted for an internal layer:

1. Set Application mode to Placement Edit.
2. Set the Super Filter to *Symbol*.
3. Use `Control` key and click to select each component targeted for an internal layer.

   Selected components are highlighted in the design canvas.

4. Right-click and choose *Place on Layer* from the pop-up menu.
5. Select target internal layer from the pull-down list.

   This action moves all selected components to the internal layer maintaining their x,y location.



Pre-selected components targeted for internal layer IN4

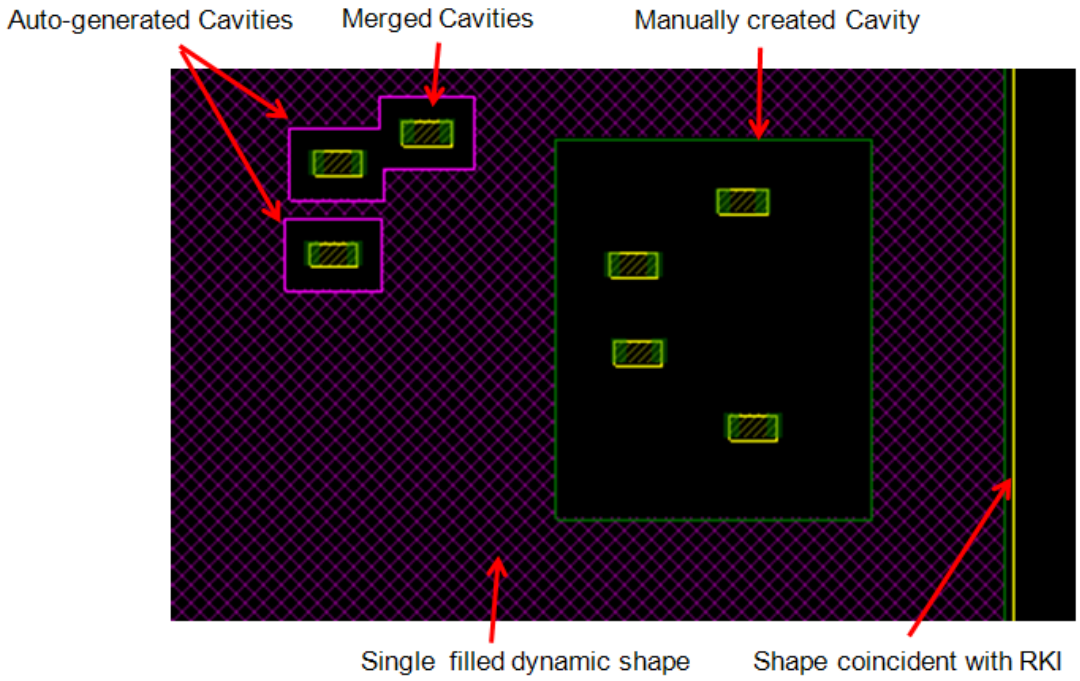
Cavities

A cavity represents the dielectric space between that Etch layer and the Etch layer immediately above (towards TOP). For example, given a four layer board with layers TOP/L1/L2/BOTTOM, the cavity associated with L1 layer represents the dielectric space between L1 and TOP.

The floor and ceiling of the cavity are coincident with the bounding layers. In a traditional design, these bounding layers would be adjacent etch layers. When a component is placed on an embedded layer, the auto-generated cavity is assumed to span the thickness of the dielectric space. If the symbol does not fit in a single dielectric you may designate adjacent layer(s) as protruding allowed. Cavities are created for all layers marked protruding along with route keep out representing a hole in the ETCH layer.

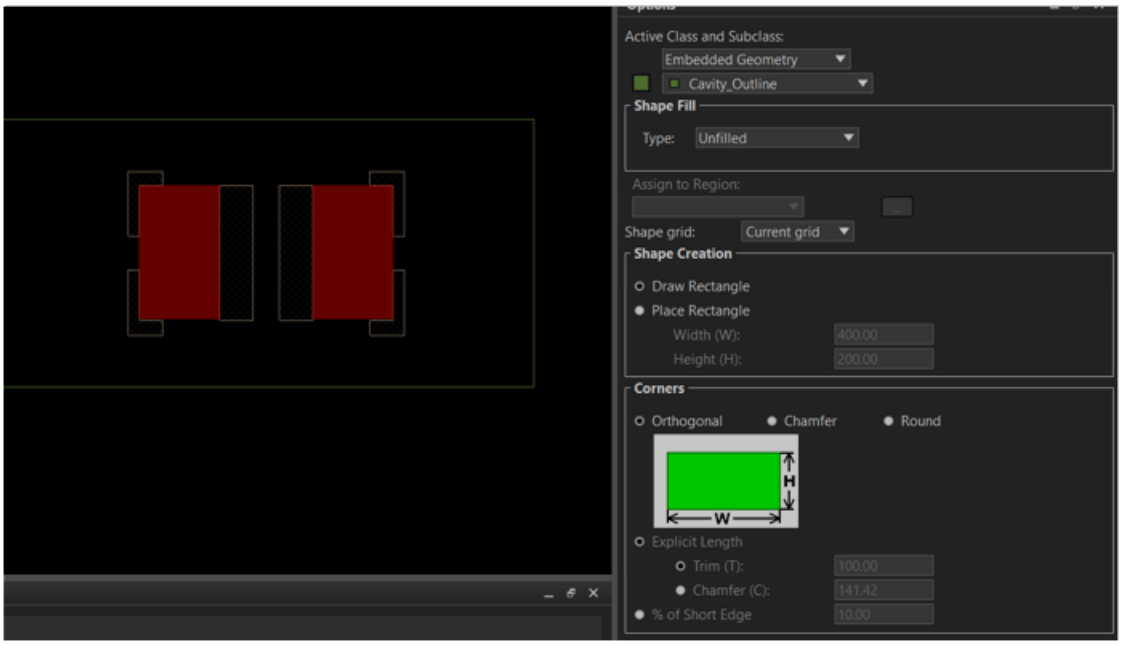
The physical representation of a Cavity subclass is a filled dynamic shape. The boundary of the shape is coincident with the route keep in. There is a one shape limitation per Cavity

subclass. Cavities that are created (either by placing an embedded component or by manual editing) are represented as voids in the dynamic shape. Automatically generated cavities that overlap or are within the minimum cavity spacing value are automatically merged into one void. Manually-edited cavities are not automatically merged. The behavior is similar to that of dynamic etch shapes. No islands are permitted; any islands generated by voiding is automatically removed.

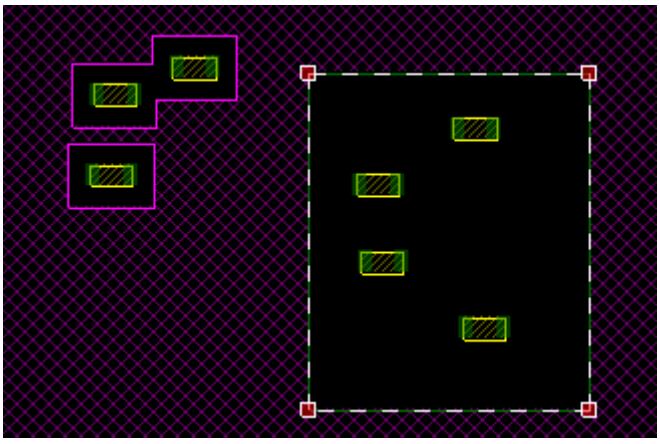


The package symbol definition supports a fixed subclass called Embedded Geometry / Cavity Outline. When the package symbol is placed on an embedded layer, the geometry from the Cavity Outline subclass is copied to the corresponding layer of the Cavity class, thereby generating a void in the dynamic shape. If no geometry exists on the Embedded Geometry / Cavity Outline subclass, then the Embedded Geometry / Place_Bound geometry is used to

generate the void. The board's minimum Package-to-Cavity spacing constraint gets added to the Place_Bound dimensions to create the dimension of the cavity.

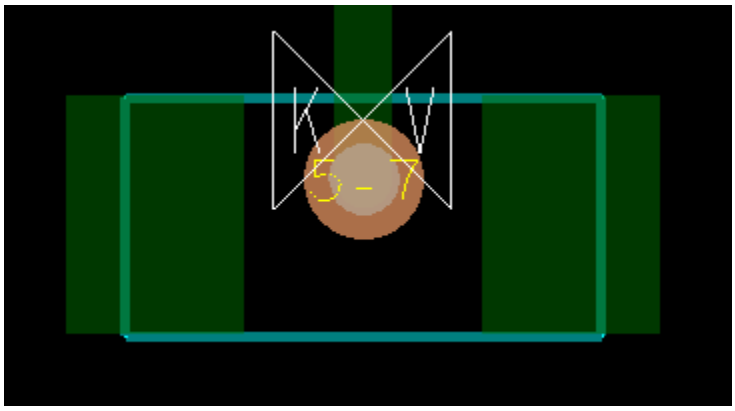


Cavity editing is possible only if the cavity is manually drawn. Auto-generated cavities follow the model used in dynamic shapes (voids are not editable). Use the *Shape – Select Shape or Void/Cavity* command to edit a cavity outline.



Manual cavity

Keep outs are automatically generated for the cavity based on the components height. Via keep outs are automatically created on the top or bottom of the cavity depending if the embedded layer is designated as *Body up* or *Body down*.



A route keep out gets created for multilayer cavities on each layer that the symbol passes through that is labeled *Protruding allowed*. A route keep out gets created on the external surface (top or bottom) if the cavity is open to the external surface. You can specify an oversize value (relative to the cavity outline) to apply to the creation of these keep outs.

Note: When *Body down* is selected as an embedded methodology, the cavity outline does not appear on the layer you would assume, that being the layer the component is mounted on. The cavity outline appears on the next adjacent layer towards the bottom. For example, SIG_3 is an embedded layer with a *Body down* methodology. The cavity graphics appears on layer SIG_4. Note that a cavity is not a single layer entity but rather an opening in material between 2 layers.

Reports

Existing reports that include components placement (for example, Placed Components Report) have been updated to properly report the status of all embedded and external components. Two new reports are also available:

- Embedded Placement Report
- Embedded Cavity Report

The Design Summary Report will be enhanced to include data regarding the embedded components.

Embedded Component Report

_LAYER	REFDES	COMP_DEVICE_TYPE	COMP_VALUE	COMP_TOL	COMP_PACKAGE	SYM_X	SYM_Y	SYM_ROTATE	EMBEDDED_STATUS	EMBEDDED_ATTACH
	C1	CAP_402-1UF108063V	.1UF	10%	R_0402	1084.00	1662.00	0.000	BODY_UP	DIRECT_ATTACH
	C2	CAP_402-1UF108063V	.1UF	10%	R_0402	1081.00	1586.00	0.000	BODY_UP	DIRECT_ATTACH
	C3	CAP_402-1UF108063V	.1UF	10%	R_0402	1159.00	1687.00	0.000	BODY_UP	DIRECT_ATTACH
	C4	CAP_402-1UF108063V	.1UF	10%	R_0402	1332.00	1577.00	0.000	BODY_UP	DIRECT_ATTACH
	C5	CAP_402-1UF108063V	.1UF	10%	R_0402	1336.00	1504.00	0.000	BODY_UP	DIRECT_ATTACH
	C6	CAP_402-1UF108063V	.1UF	10%	R_0402	1431.00	1442.00	0.000	BODY_UP	DIRECT_ATTACH
	C7	CAP_402-1UF108063V	.1UF	10%	R_0402	1423.00	1630.00	0.000	BODY_UP	DIRECT_ATTACH

Cavity Report

Cavity Layer	Location (x,y)	Area (sq in)	Placed Components	Protruding Components
SIG_6	(1124.02 1631.98)	0.00943	C3 C1	
	(1121.02 1555.98)	0.00481	C2	
	(1267.00 1682.00)	0.0834	C7 C6 C5 C4	

Manufacturing Output

Artwork

When creating new artwork film records, use the subclasses marked with the green arrow in the following image as they contain information related to potential embedded output data.

NC Drill

Supports single layer drill syntax used for *Indirect* attachment.

NC Drill Legend

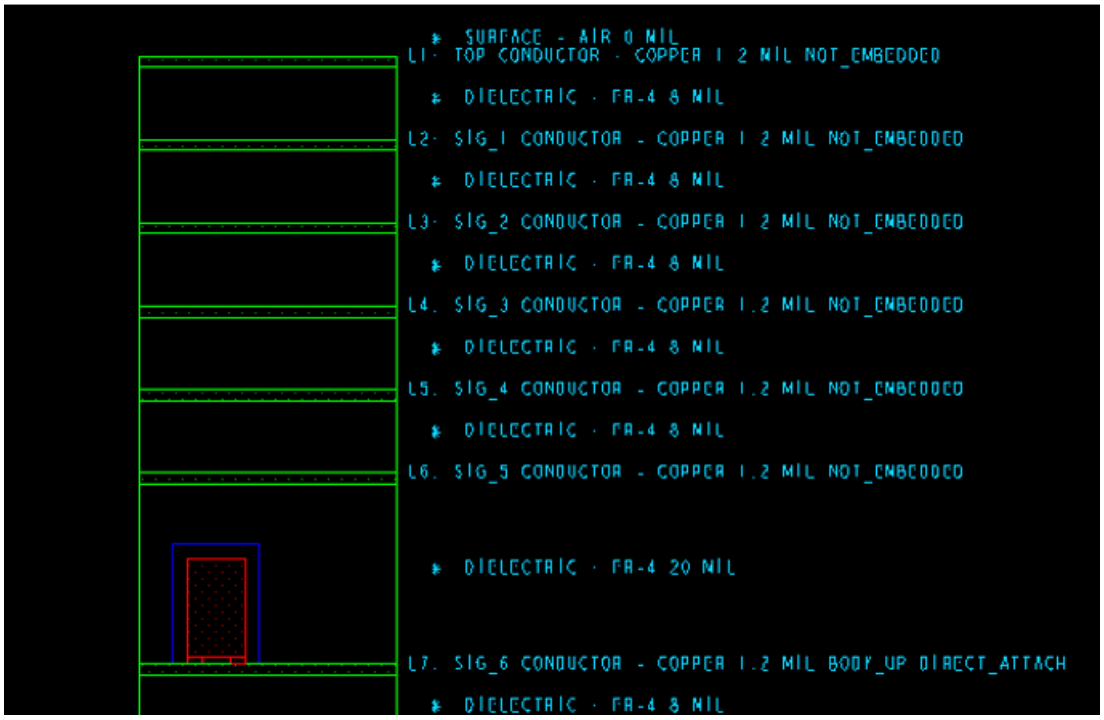
Supports new parameter option *include cavity*. You can include this legend in cavity film records.

CAVITY LEGEND			
U=body_up D=body_down P=protruding_allowed			
CAVITY LAYER	START LAYER	STOP LAYER	COMPS
SIG_6	SIG_5	SIG_6 (U)	2

NOTES:
- U: BODY_UP
- D: BODY_DOWN
- P: PROTRUDING_ALLOWED

Cross-section Chart

A detailed cross-section chart showing B/B vias and embedded methodology can be output similar to the NC Legend function. Choose *Manufacturing – Cross Section Chart* menu option. You can include this in Fab drawing film record.



IPC-356

No changes made; embedded components are ignored.

ODB++

Used 9.1 or higher

DXF

The layer mapping user interface supports the new embedded subclasses.

SKILL

All SKILL interfaces that retrieve placement information should work properly with embedded components. This includes SKILL interfaces that manipulate the placement of embedded components

`axlDBCreateSymbol` supports internal placement and the dbid of the symbol must reflect its placement.