



Integrity System Planner User Guide

**Product Version 23.1
September 2023**

© 2023 Cadence Design Systems, Inc.
Printed in the United States of America.

Cadence Design Systems, Inc. (Cadence), 2655 Seely Ave., San Jose, CA 95134, USA.

Open SystemC, Open SystemC Initiative, OSCI, SystemC, and SystemC Initiative are trademarks or registered trademarks of Open SystemC Initiative, Inc. in the United States and other countries and are used with permission.

Trademarks: Trademarks and service marks of Cadence Design Systems, Inc. (Cadence) contained in this document are attributed to Cadence with the appropriate symbol. For queries regarding Cadence's trademarks, contact the corporate legal department at the address shown above or call 800.862.4522.

All other trademarks are the property of their respective holders.

Restricted Permission: This publication is protected by copyright law and international treaties and contains trade secrets and proprietary information owned by Cadence. Unauthorized reproduction or distribution of this publication, or any portion of it, may result in civil and criminal penalties. Except as specified in this permission statement, this publication may not be copied, reproduced, modified, published, uploaded, posted, transmitted, or distributed in any way, without prior written permission from Cadence. Unless otherwise agreed to by Cadence in writing, this statement grants Cadence customers permission to print one (1) hard copy of this publication subject to the following conditions:

1. The publication may be used only in accordance with a written agreement between Cadence and its customer.
2. The publication may not be modified in any way.
3. Any authorized copy of the publication or portion thereof must include all original copyright, trademark, and other proprietary notices and this permission statement.
4. The information contained in this document cannot be used in the development of like products or software, whether for internal or external use, and shall not be used for the benefit of any other party, whether or not for consideration.

Disclaimer: Information in this publication is subject to change without notice and does not represent a commitment on the part of Cadence. Except as may be explicitly set forth in such agreement, Cadence does not make, and expressly disclaims, any representations or warranties as to the completeness, accuracy or usefulness of the information contained in this document. Cadence does not warrant that use of such information will not infringe any third party rights, nor does Cadence assume any liability for damages or costs of any kind that may result from use of such information. Cadence is committed to using respectful language in our code and communications. We are also active in the removal and replacement of inappropriate language from existing content. This product documentation may however contain material that is no longer considered appropriate but still reflects long-standing industry terminology. Such content will be addressed at a time when the related software can be updated without end-user impact.

Restricted Rights: Use, duplication, or disclosure by the Government is subject to restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 et seq. or its successor.

Contents

| | | |
|---|--|----|
| 1 | | 8 |
| Understanding the System Planner Environment | | 8 |
| Related Information | | 9 |
| System Planner Terminology | | 9 |
| System Planner Workspace | | 15 |
| System Planner Toolbar | | 17 |
| System Planner Canvas | | 19 |
| Layers, Objects, and Designs Views | | 22 |
| Flow Manager | | 25 |
| Device Hierarchy | | 27 |
| Attribute Editor | | 31 |
| Selection Filter | | 31 |
| Configuring Tool Preferences | | 33 |
| Report and Log Files | | 34 |
| Scripting | | 34 |
| 2 | | 35 |
| Defining a Substrate Layer Stackup | | 35 |
| Related Information | | 35 |
| Creating Substrates Interactively | | 35 |
| Creating Substrates by Importing a LEF Tech or CSV Layers File | | 37 |
| Using Single Substrate Designs with Allegro X Advanced Package Designer | | 42 |
| Modifying Substrates | | 44 |
| Defining Substrate Grids | | 45 |
| Changing the Substrate Type | | 46 |
| 3 | | 47 |
| Creating Padstacks and Bump Devices | | 47 |
| Related Information | | 48 |
| Creating a Padstack by Importing LEF Pads and LEF Pad Macros | | 48 |
| Creating a Padstack by Using the Edit Padstacks Form | | 50 |
| Importing Padstacks from Allegro X Advanced Package Designer | | 52 |
| Creating a Padstack While Creating Bump Devices Interactively | | 53 |

| | | |
|--|--|-----|
| 4 | | 57 |
| Creating Devices | | 57 |
| Understanding Device Templates | | 57 |
| Creating a Device by Using the Create Device Form | | 60 |
| Creating a Package Device | | 64 |
| Creating a Device by Importing a Die Abstract File | | 67 |
| Creating a Device by Importing LEF/DEF Files | | 69 |
| Creating a Device by Importing a Die CSV File | | 71 |
| Creating a Device by Importing a Pins List CSV File | | 72 |
| Importing a Device from Another System Planner Project | | 74 |
| Creating a Device by Importing a Die Text | | 74 |
| Importing Devices from Allegro X Designs | | 77 |
| Adding a Seal Ring or Scribe Line to a Die Device | | 78 |
| Changing the Origin of a Device | | 79 |
| Changing the Size of a Device | | 81 |
| Changing the Type of a Device | | 81 |
| 5 | | 83 |
| Floorplanning Devices | | 83 |
| Flipping Devices | | 83 |
| Moving Devices | | 84 |
| Copying Devices | | 86 |
| Device Table Editor | | 88 |
| Importing a CSV Devices File | | 89 |
| Ruler Mode to Measure Devices | | 89 |
| Grouping Devices | | 91 |
| Using the Devices View to Selectively Display Devices | | 92 |
| Using Inspect Mode | | 92 |
| 6 | | 93 |
| Defining Netlist Connectivity | | 93 |
| Defining the Device Hierarchy | | 93 |
| Defining Parent Child Connectivity for Allegro X Layout Editors | | 96 |
| Defining Term-based Connectivity for Innovus or Allegro X Layout Editors | | 99 |
| Understanding the Design Requirements | | 99 |
| | | 100 |
| Preparing Term List Files | | 100 |

| | |
|---|-----|
| Importing Term List Files | 101 |
| Exporting a Term List File | 102 |
| Preparing Term Map Files | 102 |
| Importing Term Map Files | 103 |
| Exporting a Term Map File | 104 |
| Creating Terms Using the Netlist Editor | 104 |
| Importing Verilog Netlists | 106 |
| Defining and Exploring Connectivity with the Netlist Editor | 109 |
| Using the Netlist Editor | 109 |
| Bulk Creation of Parent or Child Nets | 112 |
| Bulk Mapping Nets or Terms | 113 |
| Exploring Connectivity | 116 |
| Deleting and Unmapping Nets | 117 |
| Using the Set Selected Pins' Net command | 117 |
| Using the Netlist Editor | 117 |
| Exploring Connectivity with the Net Graph | 119 |
| Modifying and Optimizing Connectivity | 125 |
| Renaming Nets | 125 |
| Bulk Renaming of Nets | 125 |
| Swapping Pins | 125 |
| Defining Net Assignments for Selected Pins | 126 |
| Performing Connection Optimization | 127 |
| 7 | 133 |
| Configuring Nets | 133 |
| Managing Power and Ground Nets | 133 |
| Defining PG Nets During Import | 133 |
| Classifying PG Nets | 133 |
| Defining Net Personality Colors | 134 |
| Exporting Verilog or Net Mapping with or without PG nets | 134 |
| Assigning Power and Ground Patterns | 135 |
| Defining Differential Pairs | 137 |
| Automatically Assign based on Net Names | 137 |
| Importing a CSV Personalities File | 138 |
| Net and Pin Personalities | 139 |
| Net Personalities | 139 |
| Pin Personalities | 139 |

| | |
|--|------------|
| Creating Personalities | 139 |
| Assigning Personalities | 141 |
| Importing CSV Personalities | 141 |
| Exporting CSV Personalities | 142 |
| Using Interfaces | 143 |
| Interactively Classifying Nets as an Interface | 143 |
| Importing/Exporting a Net-Interface Map File | 145 |
| Using the Interface Editor | 145 |
| Using Bundles | 147 |
| Creating a Bundle | 148 |
| Using the Bundle Editor | 151 |
| 8 | 152 |
| Defining Contact Layers and the Design Stack | 152 |
| Creating Contact Layers, Contact Pins, and Contact Devices | 153 |
| Creating Contact Pins | 157 |
| Creating a Contact Device | 157 |
| Managing Contact Layers | 158 |
| Deleting Contact Pins or a Contact Device | 158 |
| Generating Contact Pins or a Contact Device | 158 |
| Running Inter-Substrate Connectivity Checks | 159 |
| Exploring Results | 161 |
| Defining the Design Stackup | 162 |
| 9 | 166 |
| Creating Pin Patterns, Bumps, and TSVs | 166 |
| Creating Pin Patterns for Pins, Bumps, and TSVs | 166 |
| Working with the COVER Pin Groups | 169 |
| Ungrouping Cover Pin Groups | 170 |
| Using the Bumps for All Layer Shapes Command | 170 |
| Creating TSVs Using the Fanout Vias Command | 174 |
| Renumbering Pins | 177 |
| 10 | 178 |
| Interfacing with Allegro X Advanced Package Designer | 178 |
| Importing System Planner Designs into APD | 179 |
| Creating a Die Symbol for APD | 182 |
| Back Annotating APD Changes to System Planner | 190 |

| | |
|--------------------------------------|------------|
| 11 | 192 |
| Exporting Common Design Files | 192 |
| Exporting a CSV Pins File | 192 |
| Exporting a CSV Devices File | 194 |
| Exporting LEF/DEF Files | 195 |
| Exporting Verilog Files | 197 |
| Exporting a Die Text File | 198 |

Understanding the System Planner Environment

System Planner is a flexible tool that enables several different design methodologies used for IC and package design. The unique data structure provides a platform to build and model system-level designs consisting of multiple substrates, devices, and physical design types. Different die devices from varying process technologies can be aggregated together for system-level planning and management.

System Planner manages both the physical and logical relationships between the different devices. Contact layers and contact bumps can be created to establish and manage the physical contact points for each net. Nets can be propagated and mapped between contacted devices to form die-to-die connectivity, as well as connectivity to the top-level system design. The system-level netlist can be assembled by creating or importing netlists or pin-mapping files. Meaningful top-level net names can be defined and managed through the design cycle. These capabilities make the System Planner environment ideal for system design creation, assembly, analysis, and management. System Planner is used to assemble, configure, and simultaneously manage multiple-die design projects.

The standalone System Planner is a GUI driven interactive tool with limited scripting capabilities.

The Integrity 3D-IC Platform includes full Tcl support to enable both interactive GUI commands or Tcl-based operation. See [Tcl Interaction and Commands](#).

Related Information

- Defining a Substrate Layer Stackup
- Creating Padstacks and Bump Devices
- Creating Devices
- Floorplanning Devices
- Defining Netlist Connectivity
- Configuring Nets
- Defining Contact Layers and the Design Stack
- Creating Pin Patterns, Bumps, and TSVs
- Interfacing with Allegro X Advanced Package Designer
- Exporting Common Design Files

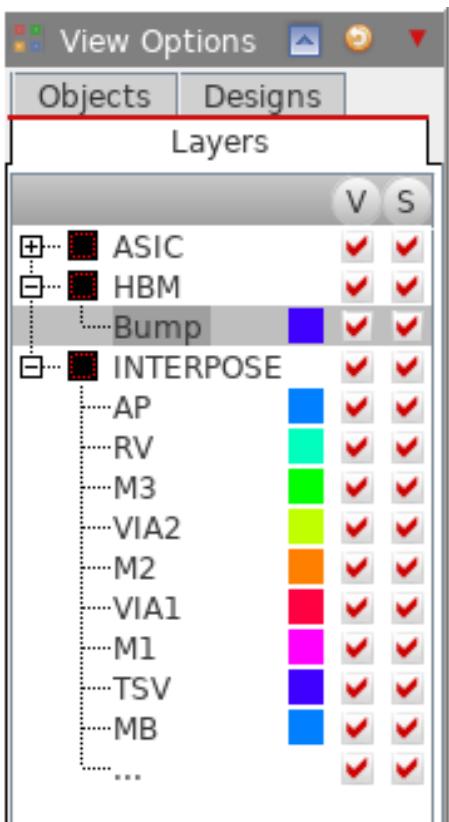
System Planner Terminology

When designing in System Planner, you need to be aware of the various objects, requirements, and terminology involved. This topic describes some of them.

Substrate

A substrate is essentially a layer stackup definition. System Planner enables designs to be assembled from multiple fab technologies, each typically with a unique substrate. Multiple devices can also use the same substrate.

The first step in creating a design in System Planner is to define one or more substrates. A substrate can consist of a single bump pad layer, as shown below under *HBM*. The substrates are displayed and managed in the *Layers* view.



Pin, Pad, Bump, Padstack, and TSV

System Planner supports a variety of pin types and terminologies such as pad, bump, and TSV to enable support for both Innovus or Allegro implementation. To enable the different processes and terminologies, the various dialog boxes will use terminologies to suit the methodology. Innovus designs often use LEF files to define the bumps, while Allegro uses padstacks.

To enable both tools, System Planner provides the following options to define the pin types where applicable:

- Use *Device* type pins when targeting Innovus. A bump is defined as a Bump device in the System Planner data model. When using *Device*-type pins, bump terminology is used. The various dialogs will present the bump devices imported from LEF files or created interactively.
- Use *Pin* type pins when targeting Allegro. When using *Pin*-type pins, padstack terminology is used. A padstack is a pad definition for any given substrate. Pad sizes are defined on the desired layers in the substrate. Padstacks are assigned to pins. Both can be created and managed through a variety of methods.

Through Silicon Vias (TSVs) are defined as pins with a padstack definition.

Device

All design objects are considered devices in System Planner. Devices are displayed and manipulated in the Device Hierarchy view. The device type dictates how the device is handled and displayed in the design. Typical device types include DIE, PACKAGE, MACRO, GROUP, CORE, COVER, BUMP, and BUMPPAD. The device type is displayed and can be modified in the Attribute Editor panel.

Device Template

Each device is created with a unique device template. Devices can be copied, and the copied devices share a common device template. Any changes to one device will affect the other copies as well. The device template is displayed and can be renamed in the Attribute Editor.

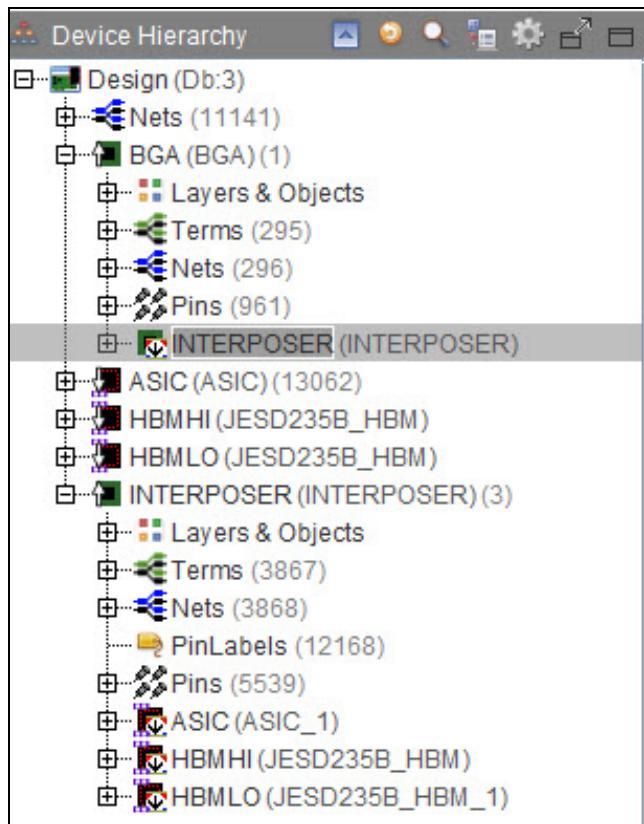
Contact Layer and Contact Pads

System Planner manages the contact point relationship between different devices using contact layers. Use the *Edit – Contact Layer* and *Edit – Design Stack* commands to define or modify contact layers. Typically, when a contact layer is being created, the option to create aligned contact pads on the contacted device is presented.

Creating contact layers also affects the netlist of the design. A contact layer can propagate the nets from the source device to the target device. There are features available to propagate and/or map the nets into the contacted device.

Contact Device

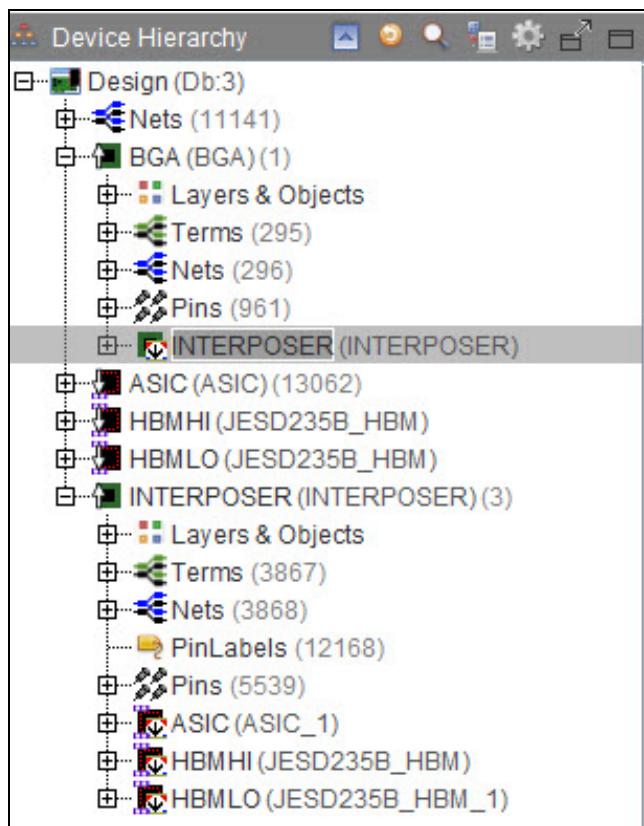
Contact devices are primarily used for Allegro design, where a child device is required in order to create the Allegro device symbol. Creating a contact device creates a copy of the device in the Device Hierarchy, as shown in the image. In the example, the INTERPOSER and BGA designs each have their own hierarchy, while the INTERPOSER device is also represented in the PACKAGE design hierarchy using a contact device.



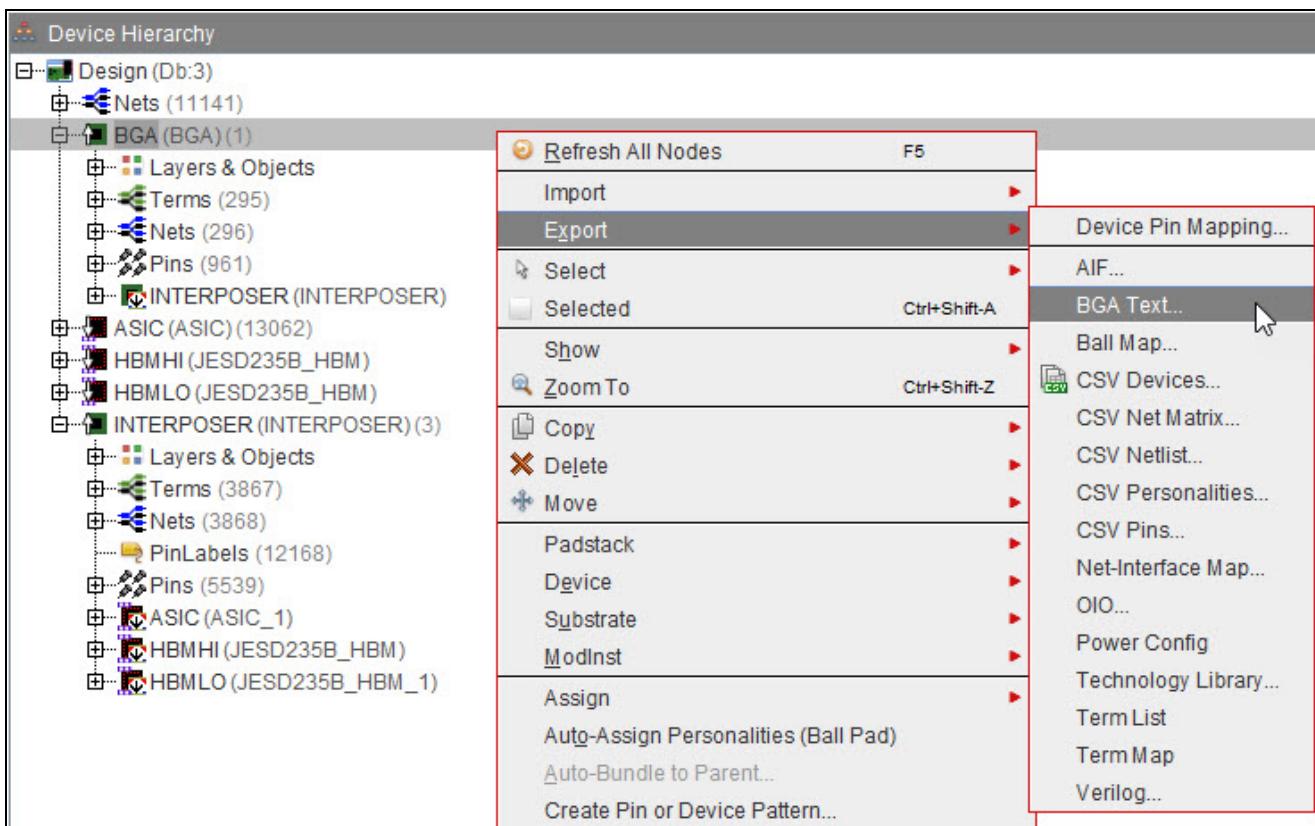
Contact devices are represented with a unique arrow down icon , as shown above for the die device under INTERPOSER.

Device Hierarchy

The Device Hierarchy panel is used frequently during a typical design process. It displays the devices in the design as well as the hierarchical relationship between the devices. It also displays the terms, nets, and pins/bumps that exist for each device, as shown below.



Select any device and right-click to access an object-oriented set of commands to perform on the selected device. These commands include many import and export options to manage the implementation requirements of the device, as shown below.



Netlist

System Planner has a robust set of capabilities for netlist generation, mapping, and management. Each device has a set of named pins that can equate to the net names for the device. Device net names can also be derived from imported sources, such as Verilog or CSV pin lists. They can also be derived by creating levels of device hierarchy and then propagating nets up and down the levels. There are net prefix and mapping options to map devices with differing net names across contacted devices. Contact layers can also be used to propagate and map nets from one device to another.

Integrity-3D-IC designs require a Verilog term-based netlist. This netlist requirement dictates different considerations and formats. For information, see [Defining Netlist Connectivity](#). Users familiar with Verilog should be familiar with the requirements for terms and macros.

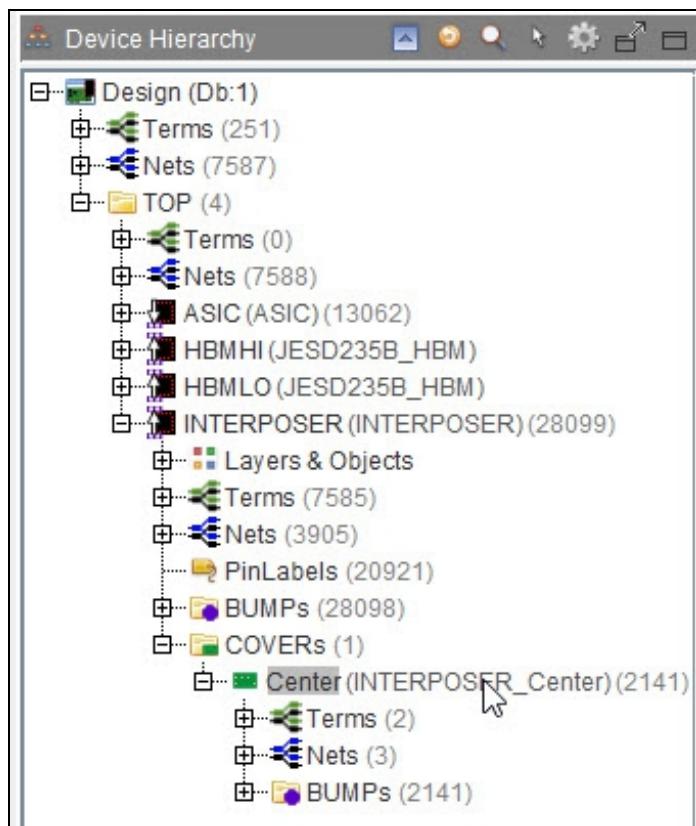
Package designs prepared for Allegro do not use Verilog netlists. A simple child parent net mapping method is used.

Cover

A cover is a group of pins that exist in a device referred to as a cover device.

Using the *Create Pins or Devices Pattern* command will create a *COVERs*-level of hierarchy under the device in the Device Hierarchy panel. Multiple cover patterns can be created and managed under the *COVERs* folder. Cover devices can be manipulated independently of the device.

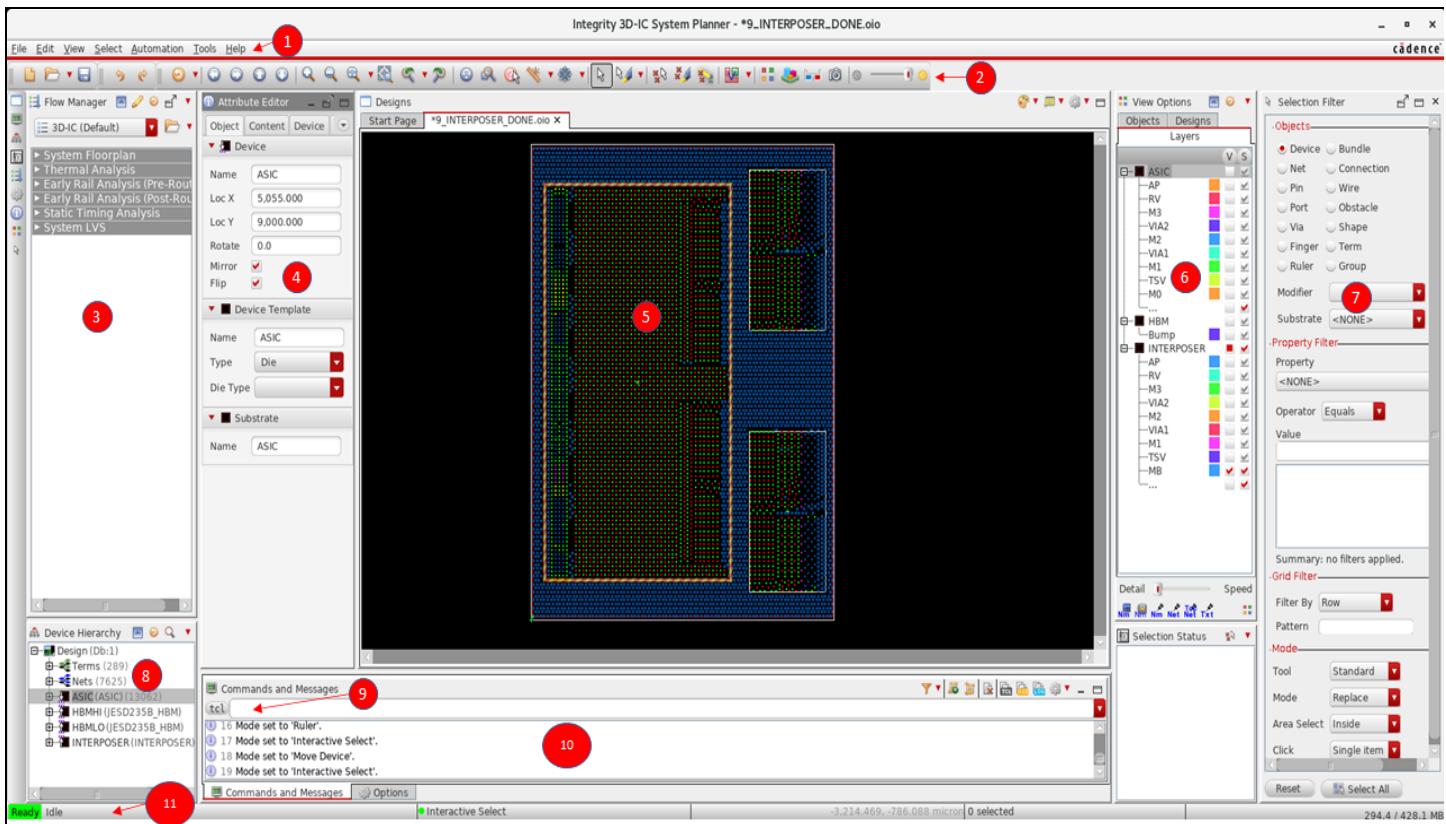
A cover device creates an additional level of netlist hierarchy under the device, as shown below. This may not be desirable, so cover devices can be dissolved into the parent-level netlist. The *Ungroup* command can be used to move the cover pins into the parent-level device netlist hierarchy and dissolve the cover device. Once ungrouped, the pins are no longer contained in or manipulated as a group. They become individual device pins.



System Planner Workspace

The following image shows various System Planner interface elements, which are identified in the number chart shown below.

Integrity System Planner User Guide
 Understanding the System Planner Environment-System Planner Workspace



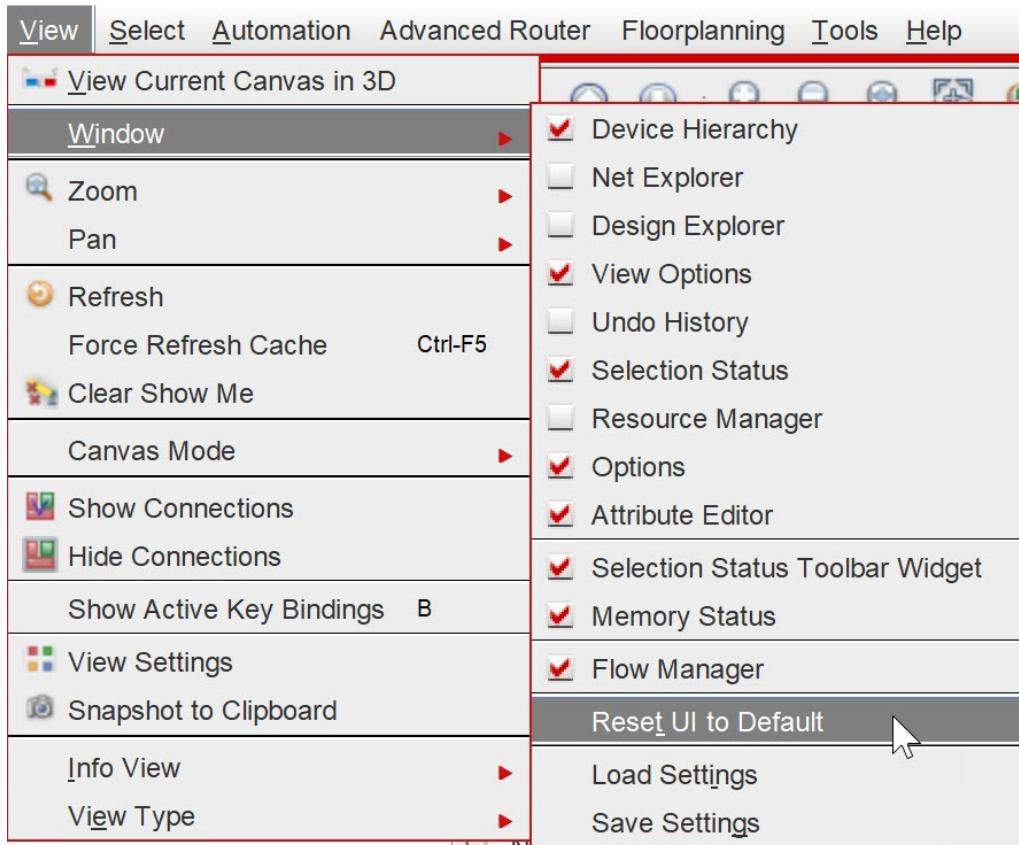
| # | Element |
|----|----------------------------------|
| 1 | Main Menu |
| 2 | Main Toolbar |
| 3 | Design Flows |
| 4 | Attribute Editor |
| 5 | Canvas |
| 6 | Layers and Objects Filter |
| 7 | Selection Filter |
| 8 | Design Hierarchy |
| 9 | Command Prompt |
| 10 | Messages Window |
| 11 | Status Bar |

Many of these elements have common Windows widgets in the upper-right corner to close,

minimize, maximize, or float the element.

The default view configuration is designed to offer the best access to design information and command controls. System Planner stores the last used view configuration for each user and will attempt to open each time with the last used configuration.

The display of the various elements can be controlled by selecting or deselecting the check boxes in the *View – Window* submenu, as shown below. You can choose *View – Window – Reset UI to Default* to return all windows to the default state.



System Planner Toolbar

The System Planner toolbar provides quick access to commonly used commands and view controls. Some of these toolbar icons are described below.



| Theme | Toolbar Icon Description |
|-------------|---|
| File/Design | <ul style="list-style-type: none">• New: Open a new design• Open: Browse to open an existing design; use the pulldown menu to select from previously opened designs• Save: Save active design |
| Action | <ul style="list-style-type: none">• Undo: Undo the last step• Redo: Perform the last step again |
| Pan | <ul style="list-style-type: none">• Refresh: Redraw the canvas• Pan Left: Pan canvas to the left• Pan Right: Pan canvas to the right• Pan Up: Pan canvas up• Pan Down: Pan canvas down |
| Zoom | <ul style="list-style-type: none">• Zoom In: Zoom the canvas in• Zoom Out: Zoom the canvas out• Zoom Fit: Zoom fit the canvas• Zoom Selected: Zoom the canvas to fit the selected objects• Zoom Previous: Zoom the canvas to the previous zoomed area• Zoom Next: Zoom the canvas to the next stored zoomed area |

| | |
|--------|---|
| Modes | <ul style="list-style-type: none">• Navigation Mode: Continuous panning mode• Zoom Region Mode: Continuous zoom mode• Inspect Mode: Continuous click to inspect objects mode• Ruler Mode: Continuous click to draw rulers mode• Interactive Move Mode: Continuous click to move mode• Interactive Select Mode: Continuous click to select mode |
| Select | <ul style="list-style-type: none">• Highlight Selected: Highlight the selected objects• Clear Selection: Deselect all the selected objects• Clear Highlight: Clear all highlighted objects |

System Planner Canvas

The System Planner canvas is the main graphic display area and is used to interact with the physical design.

Mouse Actions in the Canvas

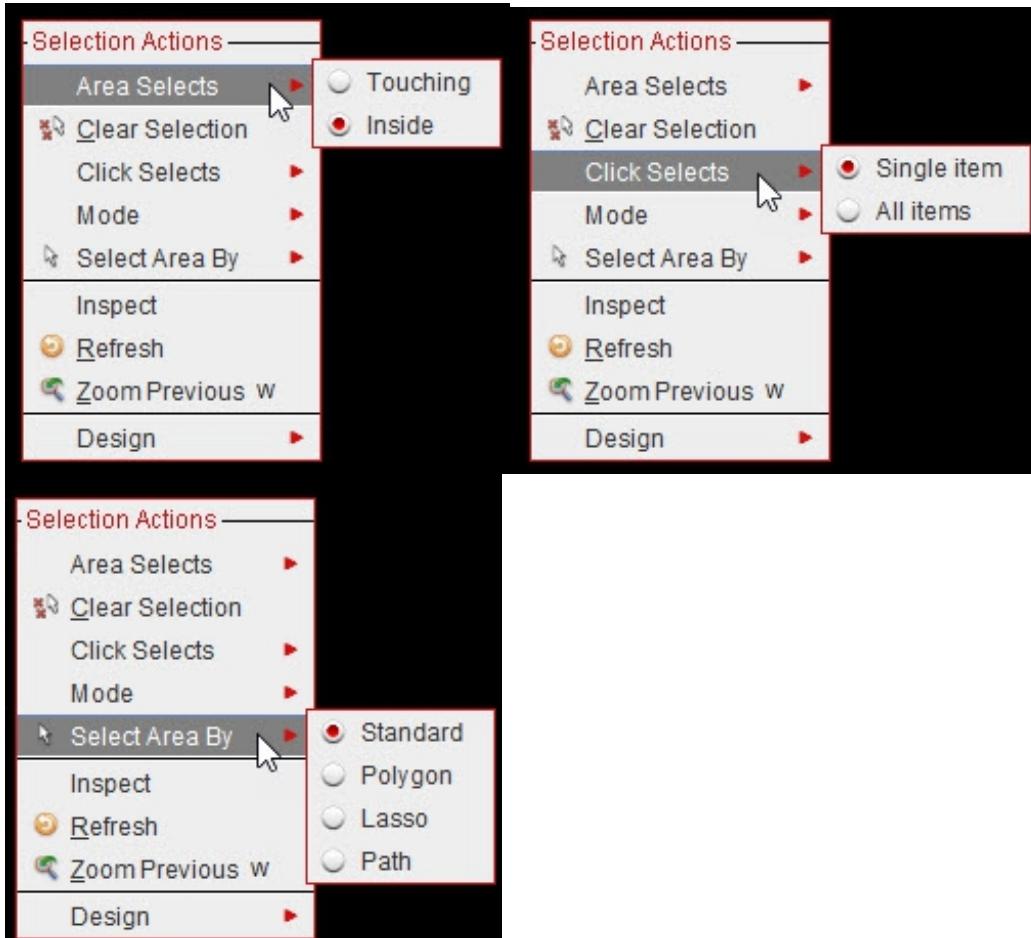
Selecting Objects

Rectangles can be drawn on the canvas to select multiple objects in the range. Use `Shift` and `Ctrl` to select multiple objects.

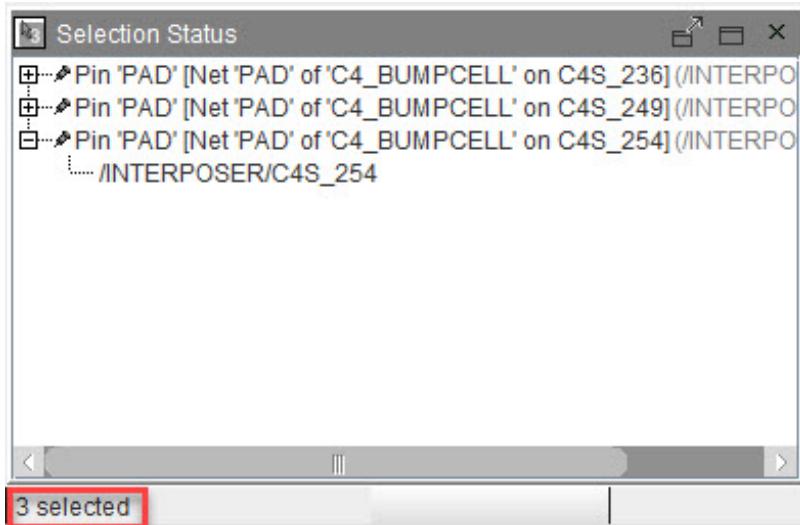
Objects must be visible on the canvas to allow them to be selected. Be sure to adjust the layer

visibility in the *Layers* view accordingly.

There are additional selection controls under the right-click menu on the canvas, as shown below. These options are self-descriptive.



The number of selected objects is displayed in the status bar at the bottom and in the *Selection Status* panel, as shown below.



Right-Click Menus

Right-clicking an object opens a pop-up menu of available commands for that object.

The available commands are sometimes determined by the view in which the right-click was performed. Some of these menus extend with multiple levels of expanded selections.

Zooming and Panning

To zoom in/out, you can do the following:

- Use the mouse wheel to zoom in and out.
- Use the right mouse button (RMB) to draw a rectangle to zoom to that specific area.
- Use the middle mouse button (MMB) or wheel for panning.

The mouse zoom speed can be set on the *User Interface* page of the Preferences form (*Tools – Preferences*).

Setting the Default View Mode

The *Default View Mode* setting controls the default mode to use when the cursor is on the canvas. The modes dictate the usage and display of the cursor on the canvas. Once a mode is selected, it remains active until another mode is selected. If undesired cursor behavior is detected, click the *Interactive Select Mode* icon on the toolbar. The *Interactive Select Mode* is used primarily with typical 3D-IC design activities.

You can change the default view mode on the *Design* page of the Preferences form (*Tools – Preferences*).

The available *Default View Mode* settings are:

- *Inspect* – Provides information for selected objects.
- *Ruler* – Measures the distance between selected points.
- *Navigate* – Pans to browse the design.
- *Zoom Region* – Zooms into and out of areas using zoom rectangles.
- *Interactive Select* (preferred) – Selects objects.

You can switch to any of these modes by selecting the corresponding icon on the toolbar.

Setting the Canvas Background Theme

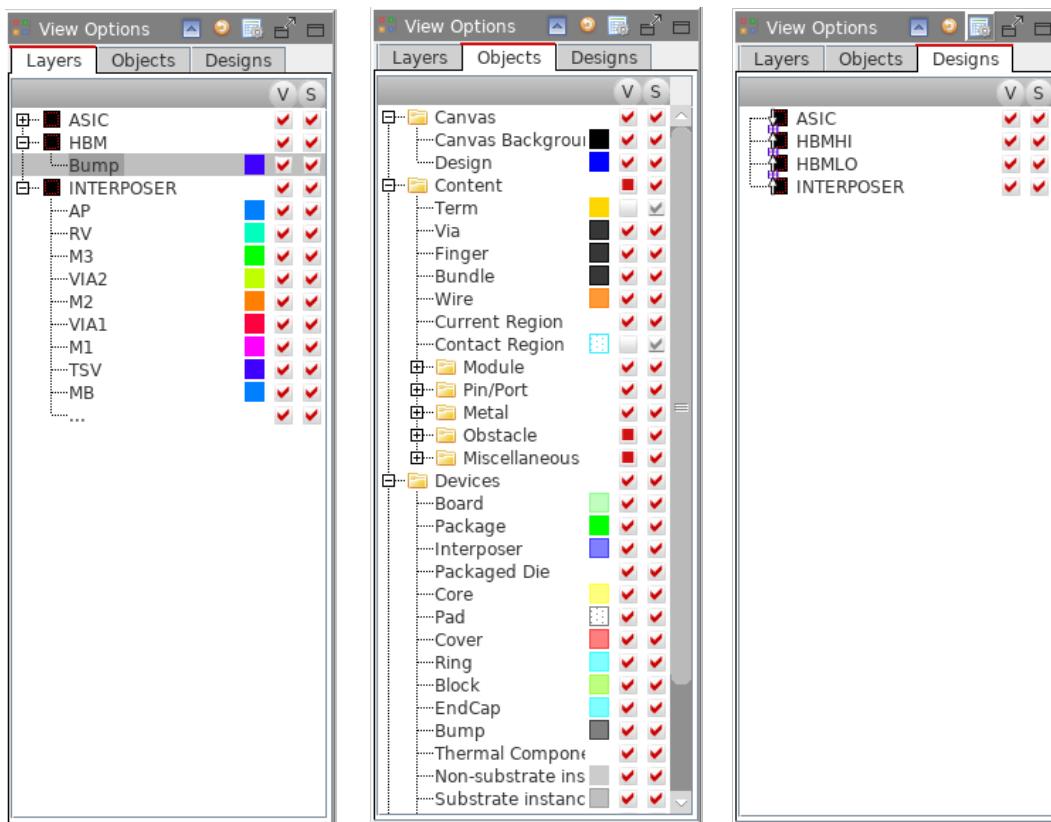
System Planner includes the *Classic Dark* and *Classic Light* canvas background themes. This guide uses the *Classic Dark* background theme in the images.

To change the background to a dark theme, select *Classic Dark* from the drop-down menu of the *Change the canvas theme* icon () at the top of the canvas.

Layers, Objects, and Designs Views

The View Options panel is displayed by default to the right of the System Planner canvas. It has three tabs, as shown below:

- *Layers* - Allows you to control the visibility and selectability of the layers in various substrates.
- *Objects* - Allows you to control the visibility and selectability of different types of objects in the canvas.
- *Designs* - Allows you to control the visibility and selectability of the designs in the canvas.



Selecting the various layer or object toggle switches will turn on or off the display on canvas for that layer or object. In the *Layers* view, next to each substrate, a top-level toggle controls visibility for all the layers in that substrate. Clicking on the colored squares will present a color palette to enable the selection of a new color for that object or layer.

Additional right-click capabilities are provided for each substrate in the *Layers* view. These include quickly toggling on/off the visibility for all layers, substrate renaming, and layer modification using the *Layers Table*. The *Layers Table* can also be opened by clicking on the *Layers Table* icon (in the *Layers* view header.

Additional view options are provided under the *Layers* and *Objects* views, as shown below.



The *Detail-Speed* slider enables configuration of the level of detail displayed in the canvas versus the speed it takes to refresh the screen. Setting it all the way to the left will enable pins to be displayed with decreased zoom levels.

The following additional view toggles are available:

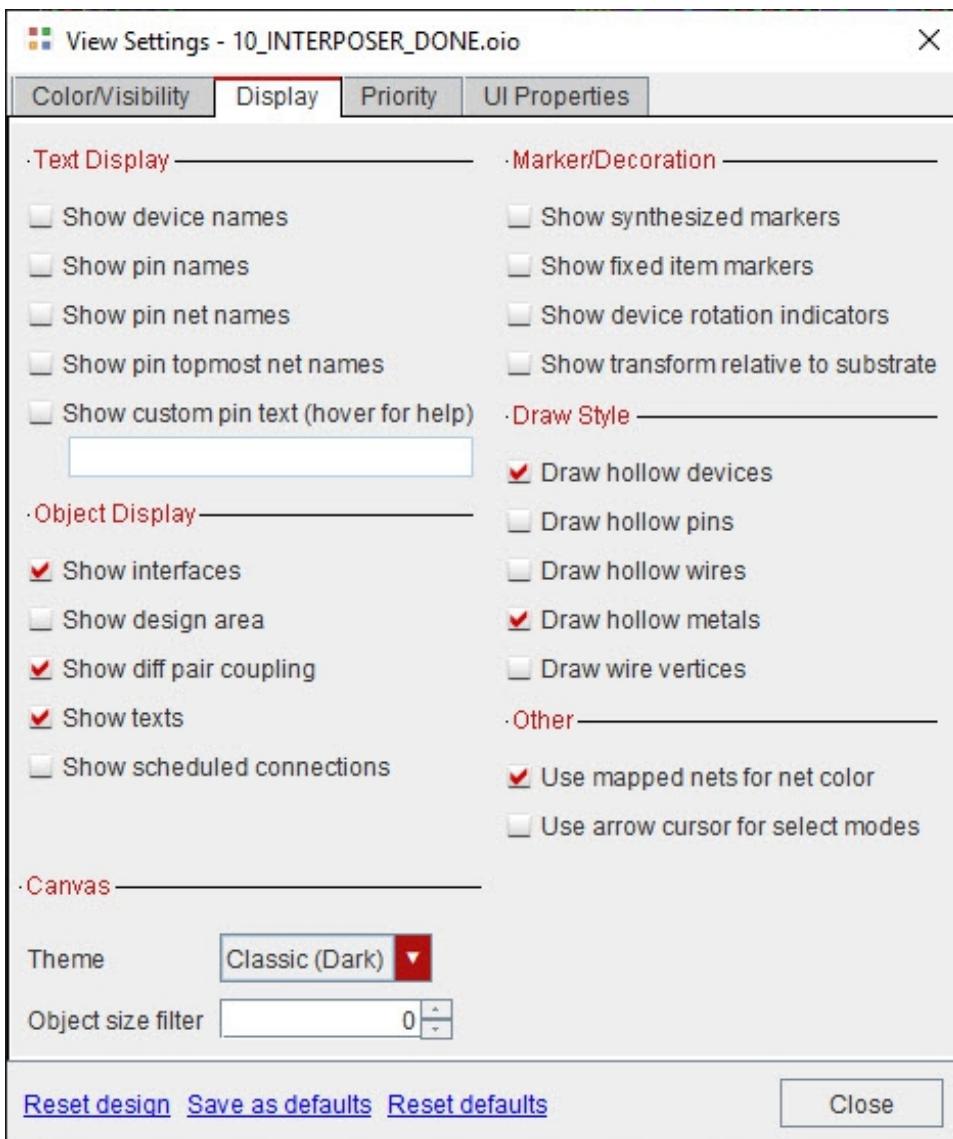
- – Toggles the display of device names.

-  – Toggles the display of device template names.
-  – Toggles the display of pin names.
-  – Toggles the display of net names on the pin.
-  – Toggles the display of the top-level net name.
-  – Toggles the display of the custom pin text.

Configuring View Settings

Additional canvas visibility control options are available in the View Settings form, which can be accessed by selecting the *View – View Settings* command or the *View Settings* toolbar icon (). The view settings control display behavior defaults, display priorities, and name text display. Many of these are related to other methodologies or tool capabilities. Default settings are recommended for *Priority* and *UI Properties* to ensure proper tool behavior.

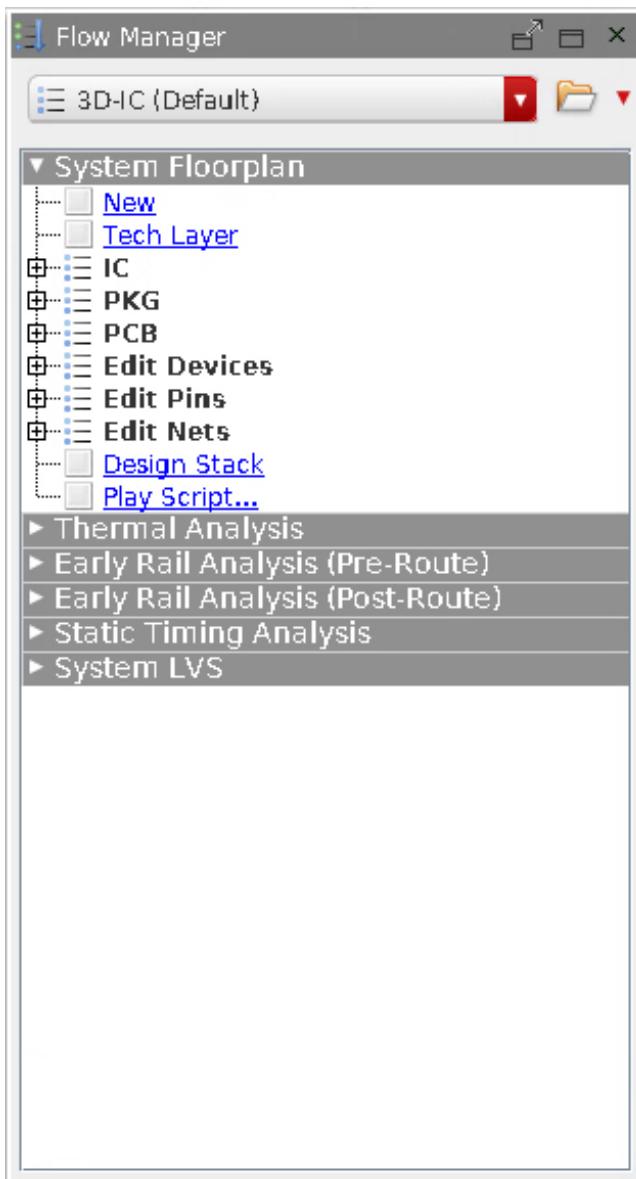
The settings most relevant to 3D-IC designs are under the *Display* tab, as shown below.



Flow Manager

The Flow Manager panel is used to organize and present commands relevant to certain types of design objects or design tasks. These commands are also found in the *Edit* menu. The Flow Manager is used to simplify access to commonly used commands.

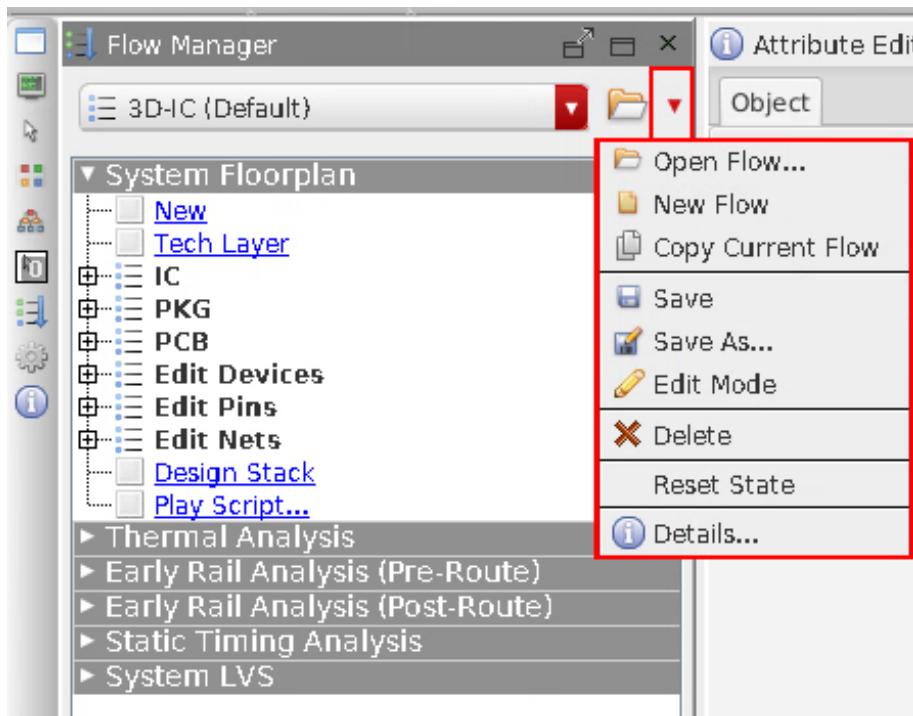
Double-clicking the *Hierarchy Expand* icon (☰) presents the commands relevant to that type of object, as shown below. Double-clicking expanded menus collapses them.



Once a command has been executed from the Flow Manager, a green check mark is displayed in its check box. This is helpful when using design flows with a specific set of steps. The *Reset State* command in the *More* (▼) menu can be used to clear these markers.

Using Custom Design Flows

The *More* (▼) menu in the Flow Manager header provides capabilities to import new designs and manage and modify design flows, as shown below.



Custom design flows are used for different types of design methodologies.

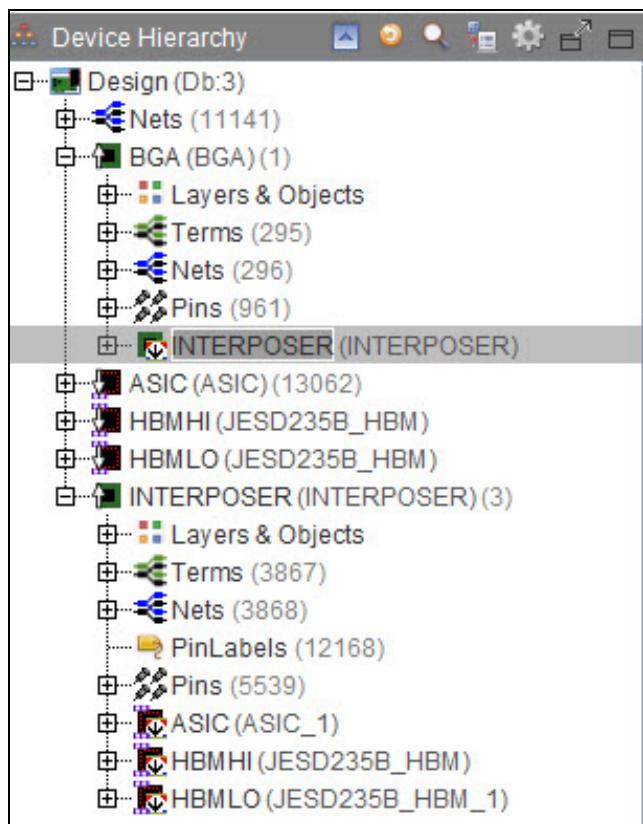
Device Hierarchy

The Device Hierarchy panel is used to display, select, and manipulate objects in the design. It is the most commonly used view when performing typical design activities.

Understanding Devices

All main design elements such as die, interposers, packages, bumps, and boards are considered devices in System Planner and are displayed in the Device Hierarchy.

Actions can be taken on any device, by selecting it and using the right-click menu commands. Data can be imported and exported for each selected device. This enables a system design to be aggregated, while still maintaining each device independently. The terms, nets, pins, bumps, and other aspects of each device are displayed under the device, as shown below.

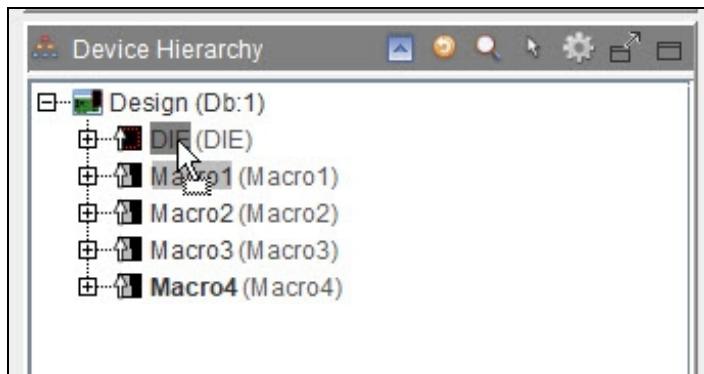


The image above represents a design with an active ASIC design, interposer design, and a package design, all represented in a single, system-level System Planner project. Note the levels of hierarchy – the *Design* (top-level), the *ASIC*, the *INTERPOSER*, and the *BGA*. The devices that are contained in each of these levels are displayed under them, as seen with the *INTERPOSER* with an *ASIC* and two *HBM*s. By expanding the hierarchy, the *Terms*, *Pins*, *Nets*, *Bumps*, and so on for each device are displayed.

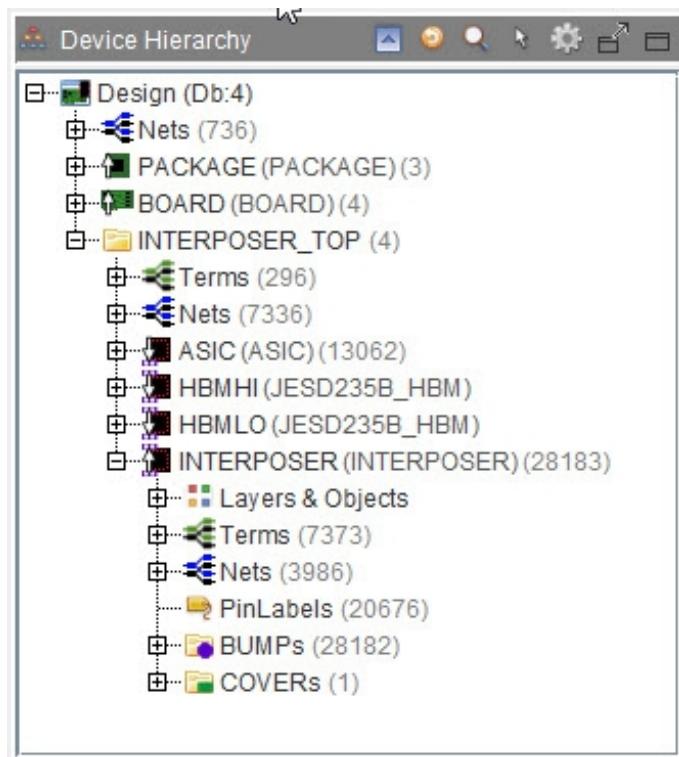
Creating Hierarchy

Relationships can be established between devices by creating levels of hierarchy. Children of a parent die are all associated with the parent die design and are contained in its netlist.

Hierarchy can be created by right-clicking a device and choosing *Move – Device to New Parent* or by dragging the device and placing on top of another device in the Device Hierarchy, as shown below.



A parent-level of the hierarchy can be created for any selected group of devices by using the *Edit – Devices – Group Selected* command. The INTERPOSER_TOP group shown below was created using this method. The group will initially be created with a default name of *UserGroup*. This name can be changed by selecting the group and using the Attribute Editor to enter a new directory and template name.



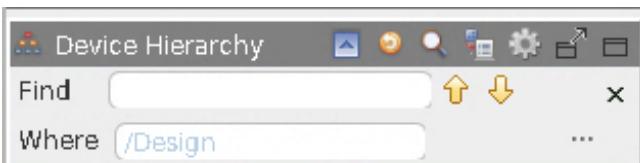
The top-level *Design* can also be used as the top-level netlist, as shown below. Notice the *Nets* and *Terms* directories are populated with top-level nets. The top-level terms define the external nets for the system design, such as those that connect to the package design, for example.



In the above image, this INTERPOSER design project uses *Design* as the top-level netlist. All dies are on the same level as the Interposer in the hierarchy. This must be represented like this in the Verilog netlist. The default top-level *Design* name can be defined when creating a new project.

Device Hierarchy Toolbar

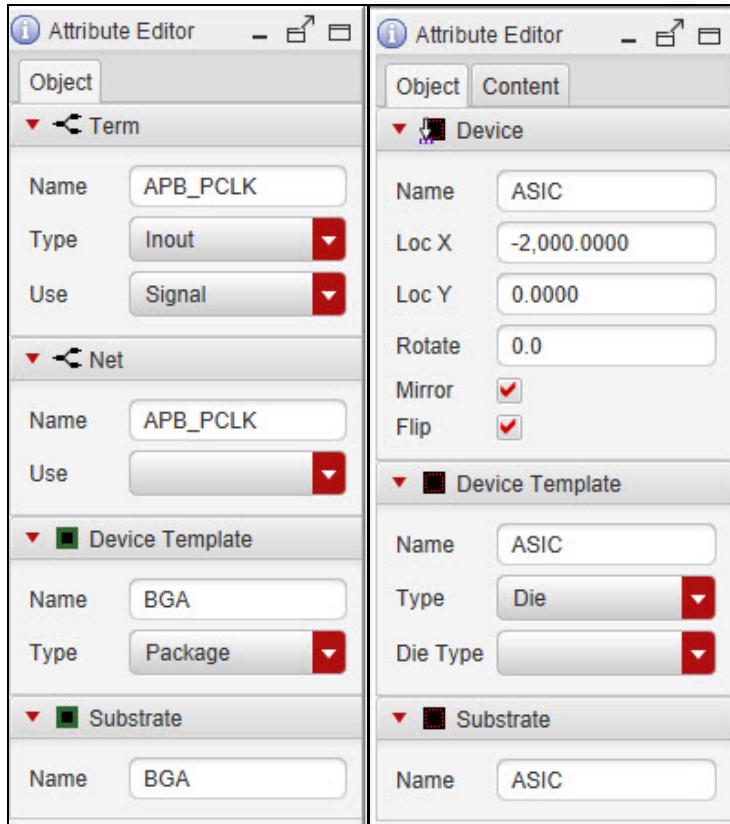
The Device Hierarchy panel contains toolbar icons in the view banner, offering view-specific functionalities.



- *Collapse Nodes* – Collapses all expanded hierarchy and returns the view to a default state.
- *Refresh All Nodes* – Redraws hierarchy to reflect any name changes or device manipulation.
- *Search* – Presents a *Find* field to enter the text string to be searched and a *Where* field to enter the location to be searched.
- *Search/Pin Selected* – Scrolls to display the pre-selected objects.
- *Properties* – Provides configuration options for the view.

Attribute Editor

The Attribute Editor is used to display and modify attributes for any selected object, as shown below.



The attributes are fairly self-explanatory. The object attributes can be modified by typing new values or selecting from the drop-down menus. Ensure the intended object is selected and displayed in the Attribute Editor before making any changes. Click the *Refresh All Nodes* icon on the Device Hierarchy toolbar to see any recent changes.

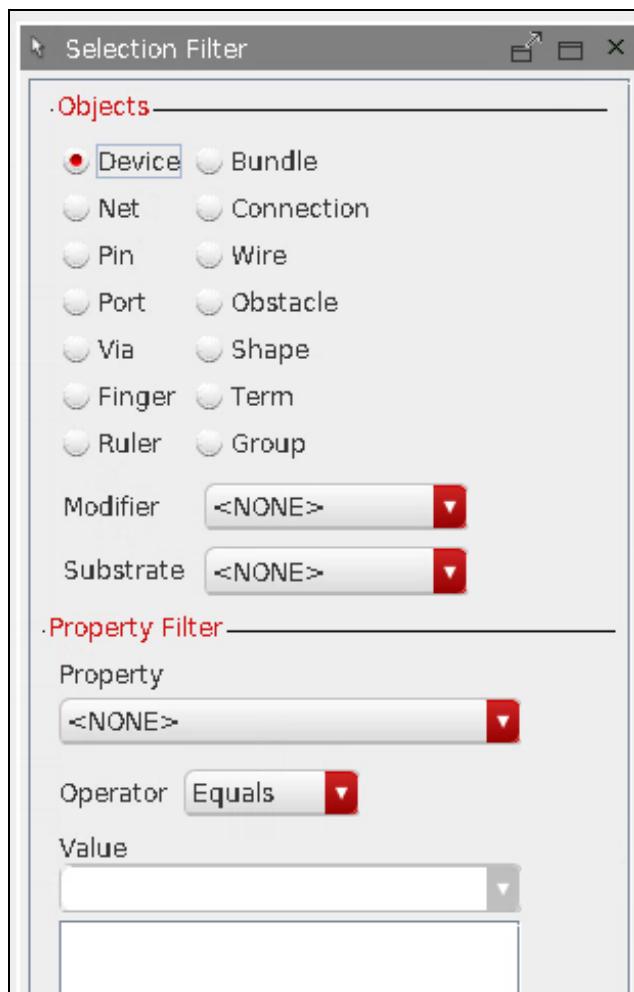
The *Device Template – Type* value can be used to change the type of object for a selected device. Changing a DIE to a PACKAGE is an example of the functionality of this attribute. This will also change the device type icon displayed in the Device Hierarchy. Many of the values available under the *Type* drop-down menu are not used in traditional design.

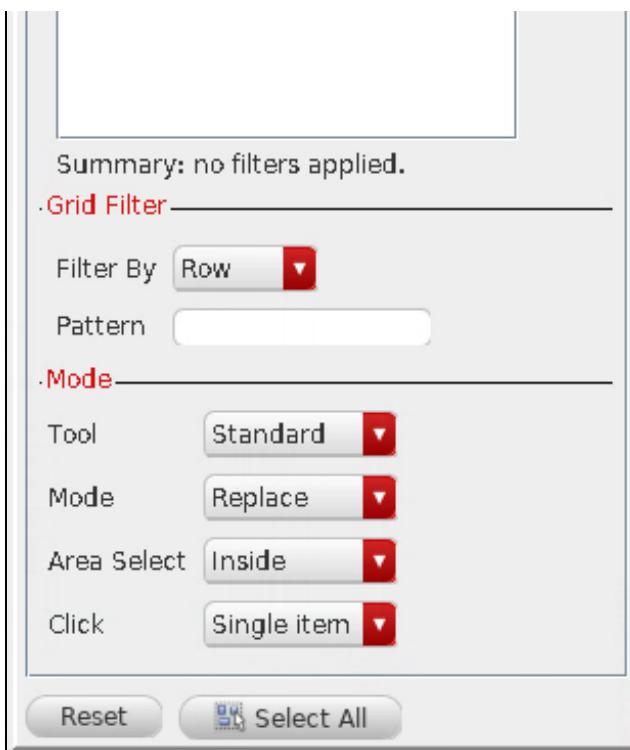
Selection Filter

The Selection Filter in the System Planner window allows specific object types to be selected on the canvas. A variety of filters allow you to make a specific selection of desired objects.

The following selection filters are available:

- *Objects* – Defines the type of objects to be selected.
- *Modifier* – Defines specific selection conditions for the selected object type.
- *Substrate* – Defines the specific substrate from which to select objects. Leaving it set to <NONE> will select objects from all substrates.
- *Property Filter* – Enables filtering based on specific properties and values for the objects to be selected. Many of these values are not used in traditional 3D-IC design. A common property filter used for selecting pins is *Net Class* to filter Signal, Power, and Ground pins.
- *Mode* – Enables various methods for selecting objects. The values are self-explanatory and are also available from the right-click menu on the canvas. The *Intersect* mode is used to enter additional filter modifiers. To use the *Intersect* mode, first select objects using the primary filter settings. Then set the mode to *Intersect*, set secondary filter settings, and select the objects again to apply the second filter settings to the first set of selected objects.





Configuring Tool Preferences

Configuration settings for the System Planner environment are available under *Tools -Preferences*. Many of the settings are self-explanatory. Some are for older legacy methodologies. The settings relevant to system design are covered below.

General

- *Application Configuration Directory* – Specifies the alternate System Planner configuration directory. This controls how System Planner behaves and is intended for custom environment developers. Recommended to leave this as the default value.

The other general settings are self-explanatory.

User Interface

- *GUI Startup Directory* – Defines where to invoke System Planner and to write .log and .cmd files.
- *Mouse wheel zoom speed* – Controls the zoom in/out speed when using the mouse wheel.

Other settings are either self-explanatory or not applicable.

Design

- *Default View Mode* - Recommended to set to *Interactive Select* for interactive system design.

Report and Log Files

System Planner creates the following three files in the GUI startup directory each time it is invoked:

- `integrity_planner.msg.log` - Contains the information displayed in the *Commands and Messages* view within the System Planner session.
- `integrity_planner.cmd.log` - Contains a history of the commands run during the System Planner session.
- `integrity_planner.tcl.log` - Contains a tcl version of the history of the commands run during the System Planner session.

An extra number is appended after the file name if there is a name collision. For example, `integrity_planner.msg.log2`, `integrity_planner.cmd.log2`, and `integrity_planner.tcl.log2`.

Scripting

System Planner scripts can be written in Java. They can also be created interactively within the tool by using the *Tools – Record Script* command. They can be replayed by using the *Tools – Play Script* command. This guide does not contain detailed scripting instructions.

Defining a Substrate Layer Stackup

A substrate is essentially a layer stackup definition. Various devices will likely have different substrates. The first step in creating designs in System Planner is to create a substrate for targeting a device(s). Substrates can be created interactively or by importing a LEF technology file or a CSV file. Default substrates can also be created when using the *Create Device*, *Create Package*, or *Import > Die Text* commands.

The *Import – Die Abstract* and *Import – Allegro BRD, SIP, or MCM* commands will automatically create substrates matching the layer stackups defined in the imported file.

Related Information

- [Creating Substrates Interactively](#)
- [Creating Substrates by Importing a LEF Tech or CSV Layers File](#)
- [Using Single Substrate Designs with Allegro X Advanced Package Designer](#)
- [Modifying Substrates](#)
- [Defining Substrate Grids](#)
- [Changing the Substrate Type](#)

Creating Substrates Interactively

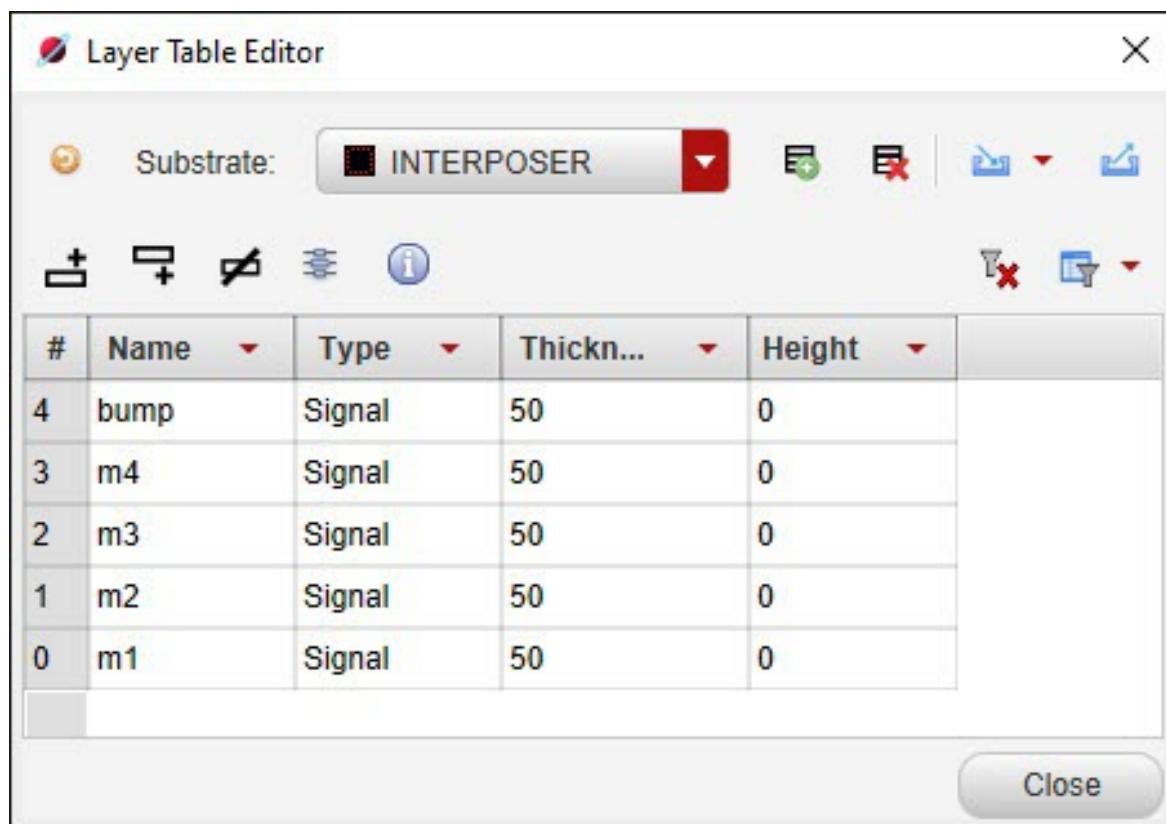
To create a substrate interactively:

1. Choose either the *Edit – Layers* command from the menu bar or the *Layer Stackup* command from the Flow Manager. If using Integrity 3D-IC, select the *Tech layer* command from the Flow Manager. You can also select the substrate in the Layers view and right-click to choose *Layers Table*.
2. A confirmation form appears if no substrates exist in the design. Click *Yes* to create a

substrate. By default, this opens the Create Substrate dialog, as shown below.

3. Update the following in the Create Substrate dialog:
 - a. *Target Type*: Specify the substrate type from the pull-down menu, which includes the DIE, PACKAGE, BOARD, and SPACER options.
 - b. *Substrate Name*: Specify a unique name for the substrate being created.
4. Click *OK* to create the default layer stack for the substrate type and name entered and open the Layer Table Editor, as shown below.

ⓘ A default layer stackup is defined and displayed for each substrate type. The default layer stackup can be modified after the substrate is created.



In the Layer Table Editor, the following are some of the tasks that you can perform:

- Import a substrate layer stackup using a LEF or CSV format file by selecting the *Import Layers* icon ().
- Modify the name, type, thickness, or height of selected layers by either entering or selecting a

value in the layer's *Name*, *Type*, *Thickness*, and *Height* fields, respectively.

- Modify the layer stackup by using the *Insert layer above*, *Insert layer below*, and *Delete layer* (  ) icons to create or delete layers, as required.
- Create new substrates in a selected layer by clicking the *Create new substrate* icon ().
- Delete substrates by clicking the *Delete substrate* icon ().
- Export a CSV substrate file by selecting the *Export Layer CSV* icon ().

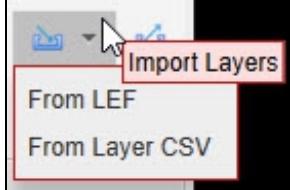
Related Information

- [Defining a Substrate Layer Stackup](#)

Creating Substrates by Importing a LEF Tech or CSV Layers File

Choose either the *Edit – Layers* command from the menu bar or the *Layer Stackup* or *Tech layer* commands from the Flow Manager to open the Layer Table Editor dialog.

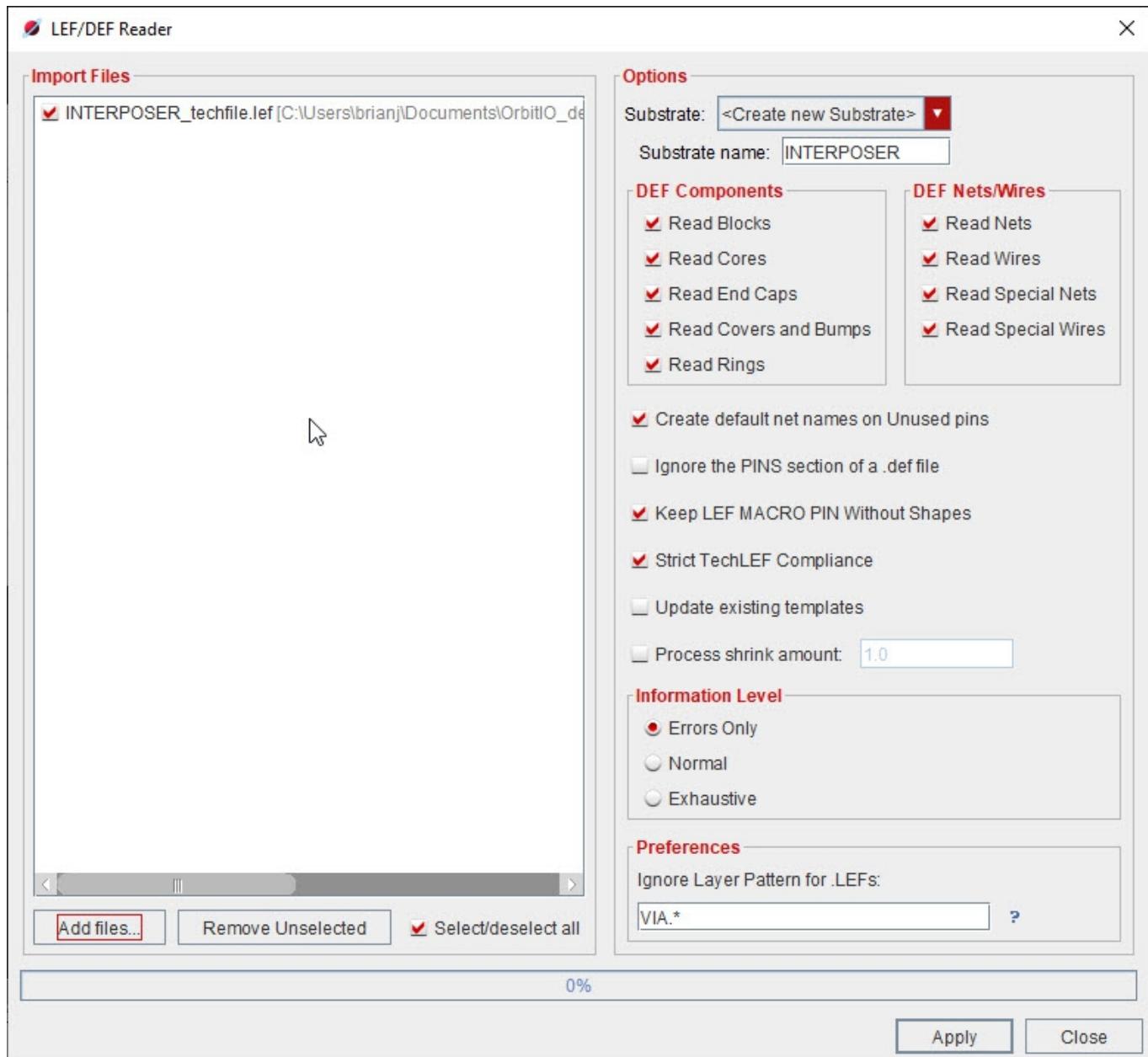
From the *Import Layers* pull-down menu, choose either the *From LEF* or *From Layer CSV* option.



Importing a LEF Tech File

To create a substrate by importing a LEF tech file:

1. Choose *From LEF* from the *Import Layers* pull-down menu. This opens the *LEF/DEF Reader* dialog, as shown below.



2. Enter the name of the new substrate to be created in the *Substrate name* field. This field is available only when *Create new Substrate* is selected from the *Substrate* pull-down menu. LEF and DEF files can be imported for existing substrates by selecting the target substrate from the *Substrate* pull-down menu. When updating or adding LEF files to an existing design, all LEF files for the design should be re-imported along with the new additions.
3. Use *Add files* to browse and select LEF files. Once LEF files are selected, they appear in the *Import Files* list. This dialog can also be used to import LEF pads, macros, and DEF layout information. The LEF tech file should be the first file shown in the *Import Files* list and the DEF

file should be last. Right-click a file to view options to adjust the file import order. Using the same LEF/DEF files that are being used for implementation, ensure matching layer stackups between System Planner and Innovus implementation. The LEF tech file is automatically passed back and forth when System Planner is invoked from Innovus.

4. Click *Apply* to create the new substrate based on the LEF tech file.

Importing a CSV Layers File

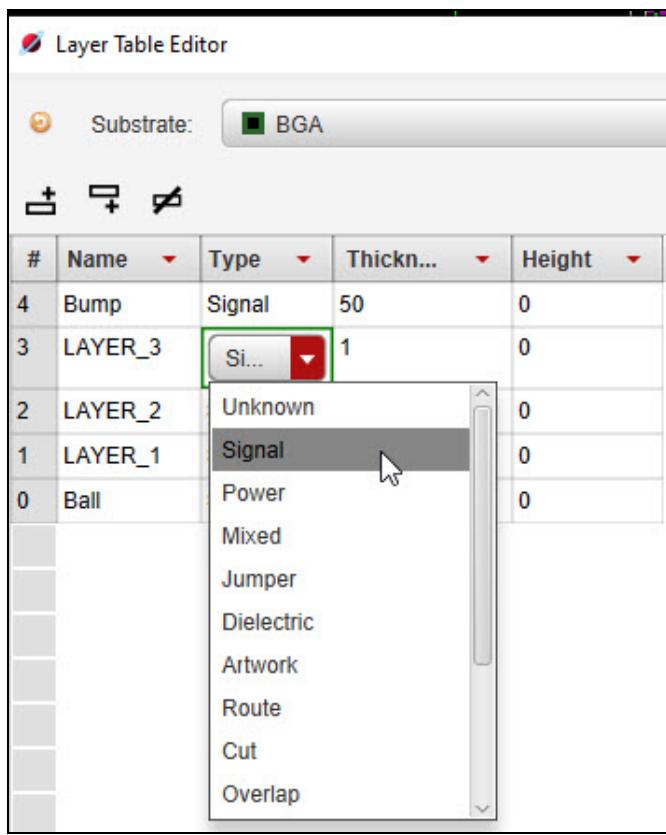
To create a substrate by importing a CSV Layers file, choose *From Layer CSV* from the *Import Layers* pull-down menu and browse to a CSV file. Click *Open* to create and display the new substrates defined in the CSV file in the *Layers* view.

The Layers file CSV format is shown below.

| Order | Substrate | Name | Height | Type |
|-------|------------|-----------|--------|-------------|
| 4 | PACKAGE | Bump | 50 | Signal |
| 3 | PACKAGE | Layer3 | 50 | Signal |
| 2 | PACKAGE | Layer2 | 50 | Signal |
| 1 | PACKAGE | Layer1 | 50 | Signal |
| 0 | PACKAGE | Ball | 50 | Signal |
| 10 | INTERPOSER | AP | 15 | Signal |
| 9 | INTERPOSER | DIELECT3 | 9 | Dielectric |
| 8 | INTERPOSER | M3 | 15 | Signal |
| 7 | INTERPOSER | DIELECT2 | 9 | Dielectric |
| 6 | INTERPOSER | M2 | 15 | Signal |
| 5 | INTERPOSER | DIELECT1 | 9 | Dielectric |
| 4 | INTERPOSER | M1 | 15 | Signal |
| 3 | INTERPOSER | DIELECT0 | 9 | Dielectric |
| 2 | INTERPOSER | MB | 15 | Signal |
| 0 | HBM | Bump | 110 | Signal |
| 11 | ASIC | AP | 11 | Route |
| 10 | ASIC | RV | 11 | Cut |
| 9 | ASIC | M3 | 11 | Route |
| 8 | ASIC | VIA2 | 11 | Cut |
| 7 | ASIC | M2 | 11 | Route |
| 6 | ASIC | VIA1 | 11 | Cut |
| 5 | ASIC | M1 | 11 | Route |
| 4 | ASIC | TSV | 11 | Cut |
| 3 | ASIC | M0 | 11 | Route |
| 2 | ASIC | STZone | 0 | MasterSlice |
| 1 | ASIC | TSV_Block | 0 | MasterSlice |
| 0 | ASIC | OVERLAP | 0 | Overlap |

The *Order* column defines the way the substrates will appear in the *Layers* view from top to bottom. Notice that *HBM* only has a single contact pad layer defined.

The available layer *Types* can be displayed and set by expanding one of the *Type* fields in the *Layer Table Editor*, as shown below.



Typical layer types for Allegro X designs include Signal and Dielectric. Dielectric layers do not have to be added in System Planner. As designs are imported into Allegro, Dielectric layers are automatically added between Signal or Route layers.

For Innovus designs, the layer types are defined in the imported LEF files and should not be modified in System Planner. These files are often supplied by the silicon vendor and are also used for implementation, so they need to remain consistent.

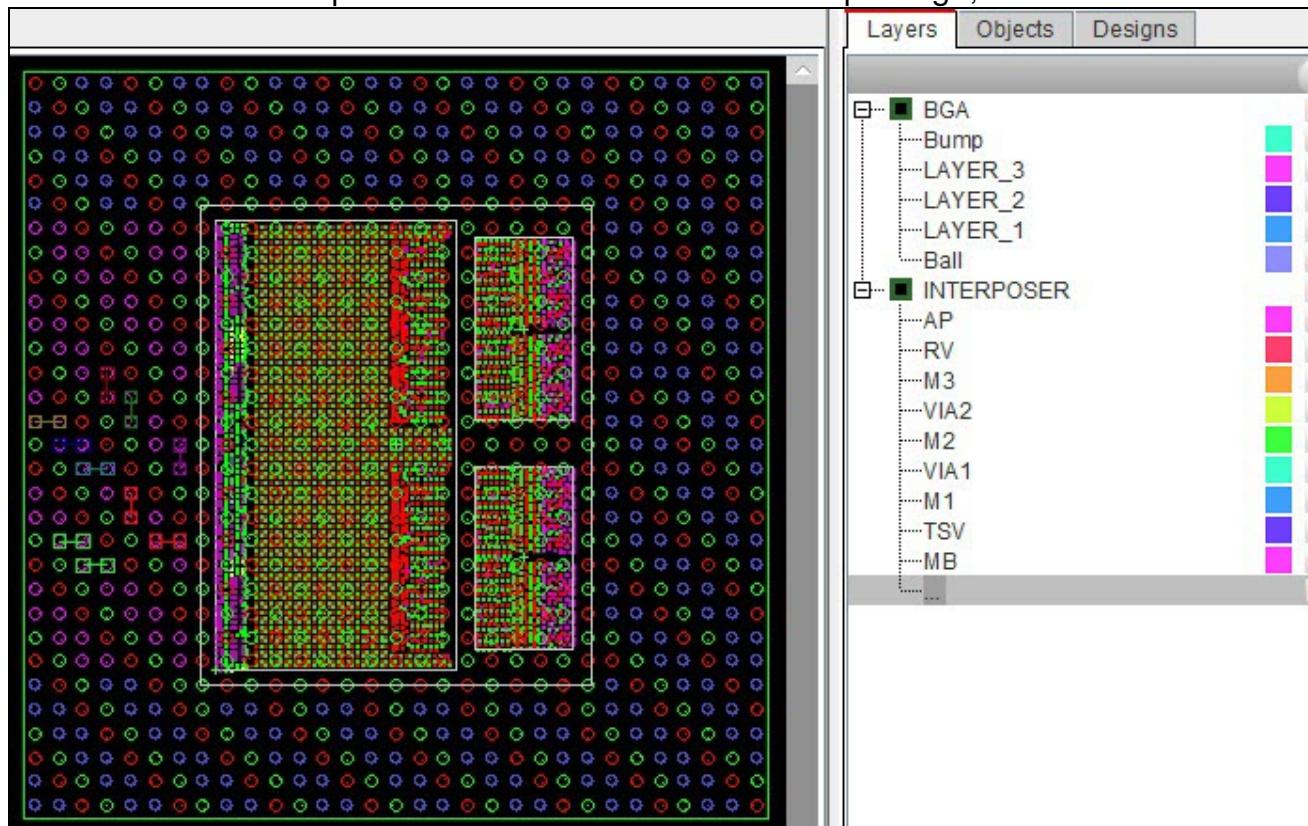
Related Information

- [Defining a Substrate Layer Stackup](#)

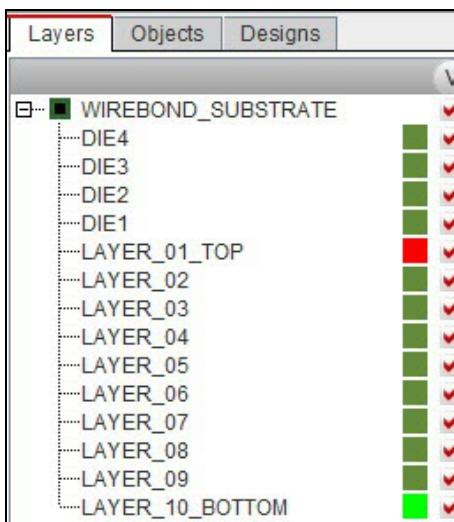
Using Single Substrate Designs with Allegro X Advanced Package Designer

Typically each unique device will target a unique substrate, which is the Cadence recommended methodology. However, for certain types of designs using Allegro X APD a single substrate can be used. Die pads can be imported directly onto to the top layer of the substrate. This eliminates the need to create contact layers and it simplifies the design, as shown below with the die pads displayed on the top INTERPOSER- AP layer.

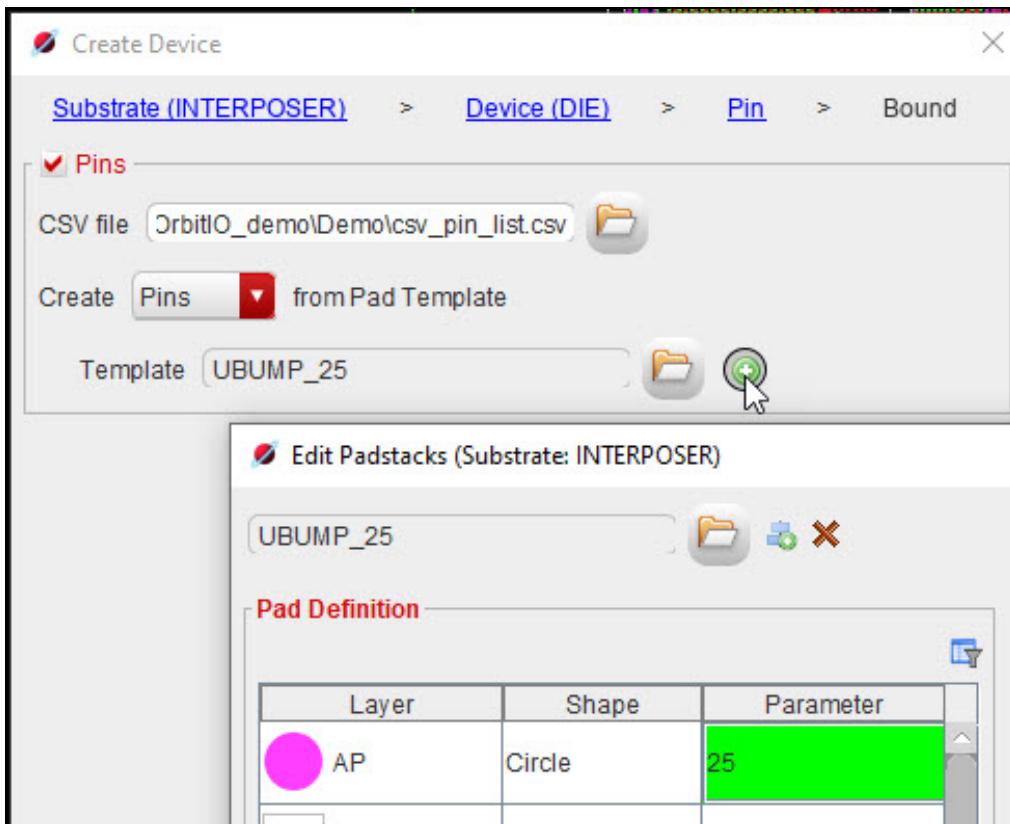
A substrate is defined for each Allegro design. For example with 2.5D designs, a substrate would be defined for the interposer and another substrate for the package, as shown below.



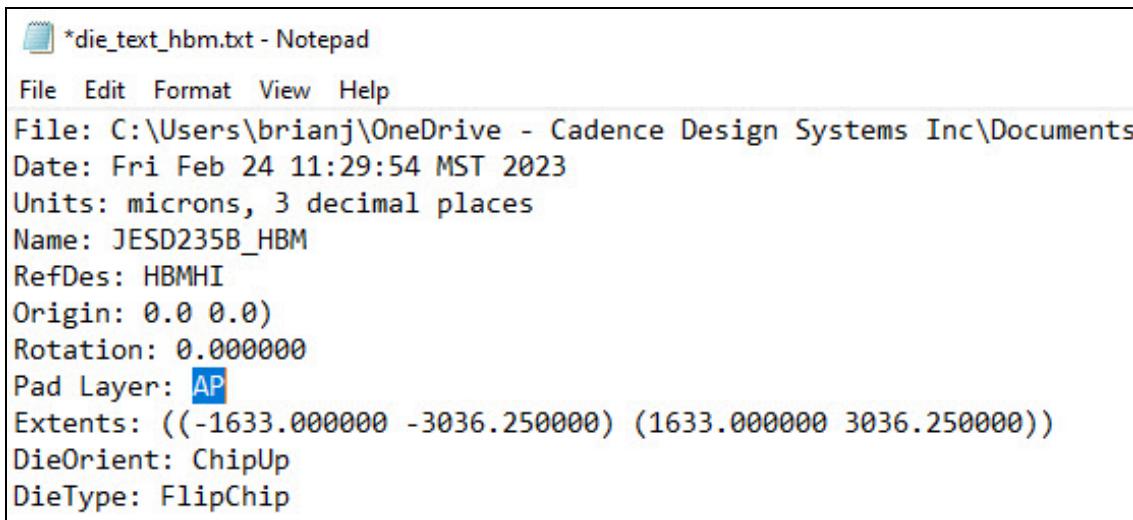
Wirebond designs also typically use a single substrate design structure, where the pad layer for each die is defined on a layer in the combined layer stackup, as shown below.



To import die directly onto a specific substrate layer, the *Create Device* command can be used in conjunction with a CSV pin list file. The Padstack defined in the CSV pin list file should use a pad defined on the top layer, as shown below.



The *File > Import > Die Text* command can also be used. The die text file can be modified to define the top *Pad Layer*.



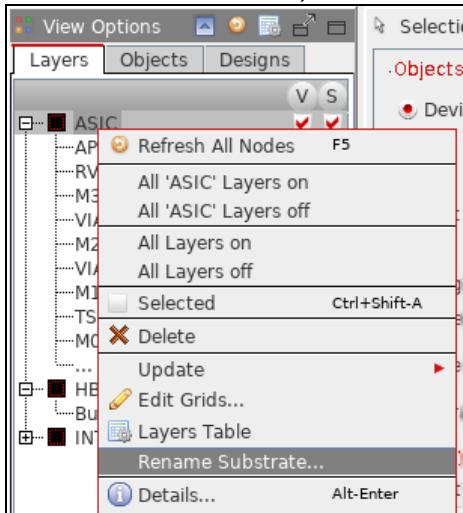
```
*die_text_hbm.txt - Notepad
File Edit Format View Help
File: C:\Users\brianj\OneDrive - Cadence Design Systems Inc\Documents
Date: Fri Feb 24 11:29:54 MST 2023
Units: microns, 3 decimal places
Name: JESD235B_HBM
RefDes: HBMHI
Origin: 0.0 0.0)
Rotation: 0.000000
Pad Layer: AP
Extents: ((-1633.000000 -3036.250000) (1633.000000 3036.250000))
DieOrient: ChipUp
DieType: FlipChip
```

Related Information

- [Defining a Substrate Layer Stackup](#)

Modifying Substrates

Substrates can be modified by selecting them in the Layers view and right-clicking to choose the desired modification, as shown below.



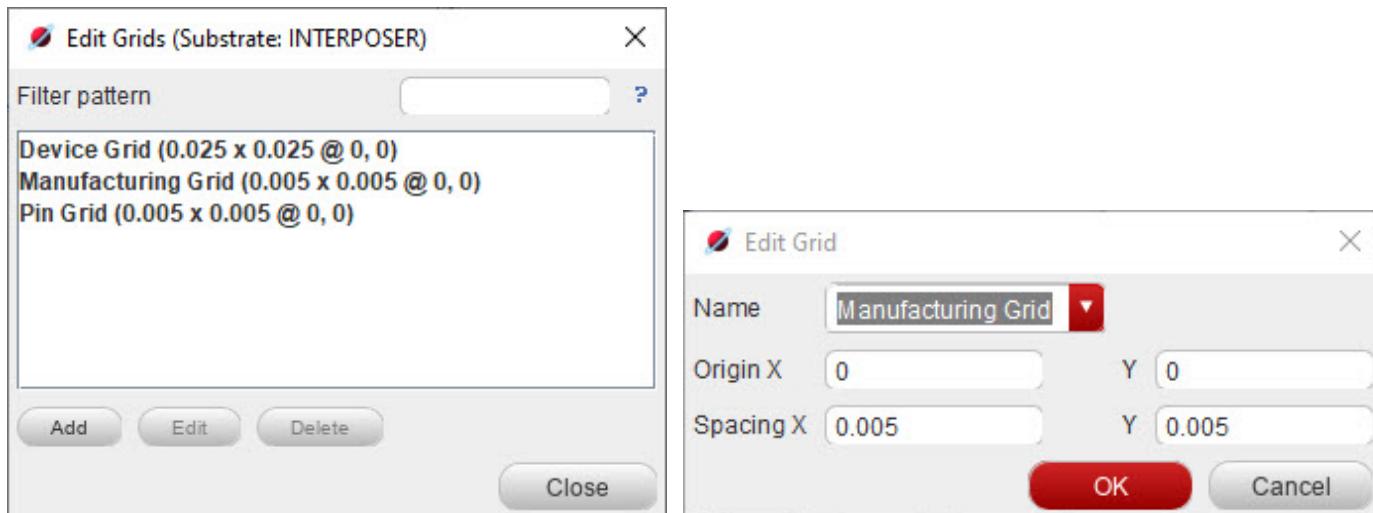
Related Information

- Defining a Substrate Layer Stackup

Defining Substrate Grids

Grids can be defined for each substrate. These include Manufacturing Grid, Pin Grid, and Device Grids.

To define or modify substrate grids, select the substrate in the Layers view and right-click to choose *Edit Grids*.



The Edit Grids dialog allows you to do the following:

- Click *Add* button to define a new Manufacturing, Device, or Pin grid.
- Click *Edit* to modify an existing grid.
- Click *Delete* to remove a grid.
- Enter X and Y Spacing values in the *Spacing X* and *Y* fields.
- Define the grid origin offset values from the project 0,0 coordinates.

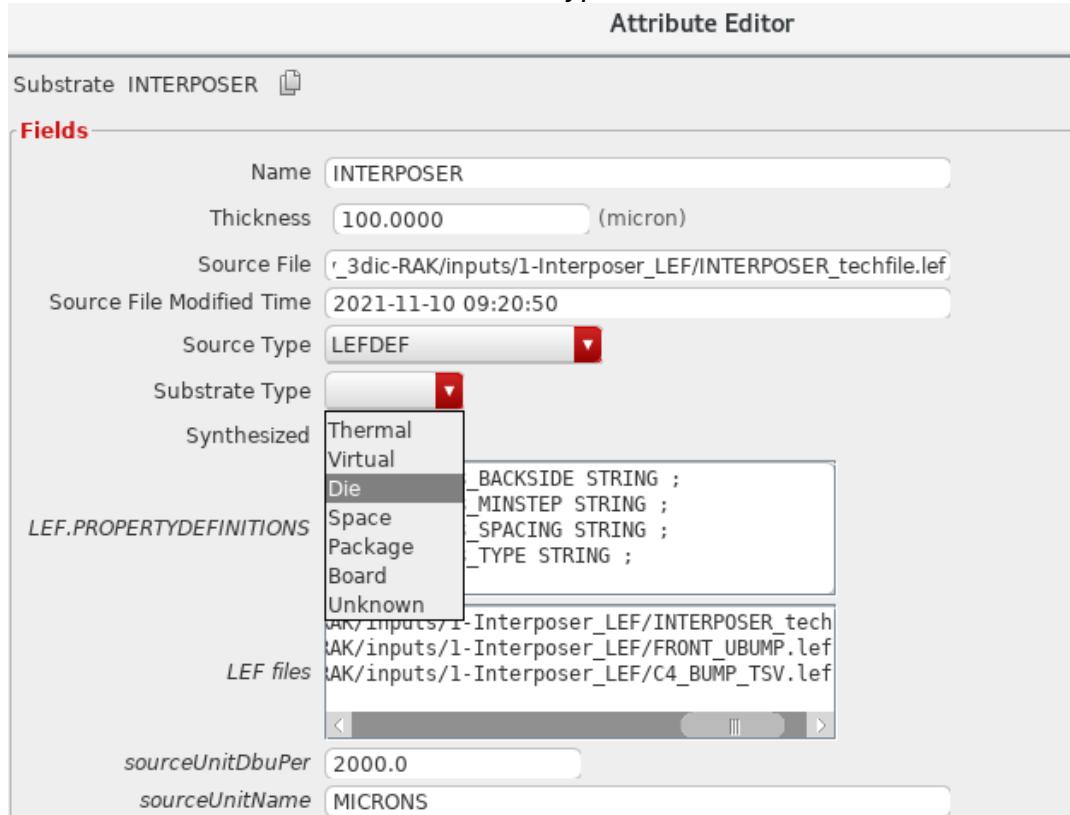
Related Information

- Defining a Substrate Layer Stackup

Changing the Substrate Type

The Substrate Type typically matches the type of device targeting the substrate. It is often set by default as the device is being created. Different icon symbols are used to represent different substrate types.

To change the Substrate type, select the substrate in the Layers view and right-click to choose *Details*. Select the desired *Substrate Type*, as shown below.



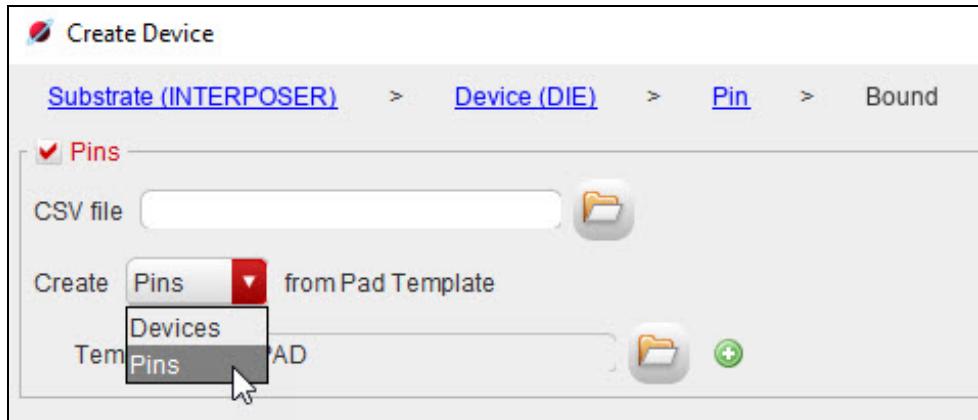
Related Information

- [Defining a Substrate Layer Stackup](#)

Creating Padstacks and Bump Devices

Different methodologies use different terms to describe the pads in the design, such as pads, bumps, pins, micro bumps, and Through Silicon Vias (TSVs). The System Planner attempts to expose the user to the terminology they are most familiar with. There is a fundamental difference in the way pins are defined to support both Innovus and Allegro implementation requirements.

The System Planner supports two different types of Pins in order to support both LEF bump-based designs with Innovus and padstacks with Allegro. When creating devices or pin patterns, you are presented with a pulldown menu to choose either Devices or Pins, as shown below.



For silicon design using Innovus:

- The bumps are typically defined in LEF pad files. The LEF pads are automatically created using *Devices* style pins with unique device templates, where the entire pad definition is contained in the LEF file. These are often supplied by the silicon vendor and should not be modified. Bumps imported from LEF files are not made available for edit in the Padstack Editor for this reason. Changes to LEF pads should be made in the source LEF files.

For Allegro designs:

- Use *Pins* style pins with padstack template defined. A padstack is simply a definition of the various layers, shapes, and sizes for each pad used in the design. You can create padstacks ahead of time and then use them in the devices that you create. Padstacks can also be created interactively as devices are being created by entering the geometries and

parameters. Padstacks can also be imported from Allegro to align with a known padstack library. Same name padstacks can be shared across the System Planner and Allegro. During System Planner design import into Allegro, the default option will use padstacks already defined in the Allegro design, if the name coming from System Planner matches.

Padstacks target a specific substrate and can define pads on the substrate for one or more layers. Once defined, they appear in the appropriate dialogs for selection when creating devices, bottom side bumps, and pin cover groups. A padstack can be used to define contact pads, pads on multiple substrate layers, and TSVs.

Related Information

- [Creating a Padstack by Importing LEF Pads and LEF Pad Macros](#)
- [Creating a Padstack by Using the Edit Padstacks Form](#)
- [Importing Padstacks from Allegro X Advanced Package Designer](#)
- [Creating a Padstack While Creating Bump Devices Interactively](#)

Creating a Padstack by Importing LEF Pads and LEF Pad Macros

The recommended method for creating padstacks in System Planner for 3D-IC designs is to import LEF pad files. LEF pad files enable the layer characteristics used for implementation and analysis. Using the same LEF pads that are being used for implementation ensures matching pads between System Planner and the Innovus implementation.

Using LEF Macros

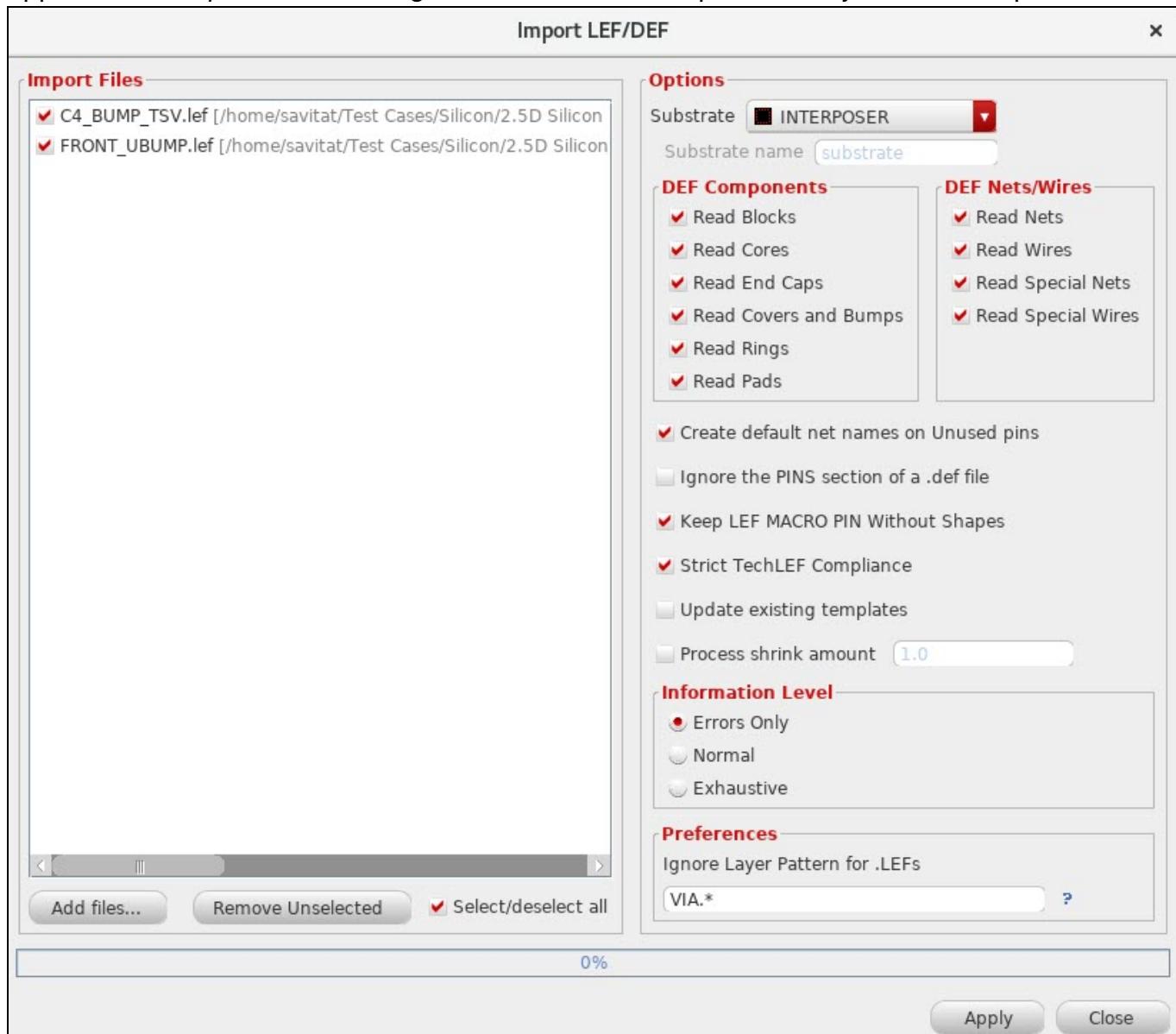
This capability is only available for silicon-based design using Innovus or Integrity 3D-IC. You can import LEF format macros that enable multiple pins and pin/pad configurations to be created as a group of objects acting as a single device. An example may be several top-layer pads and vias funneling down to one larger C4 bump for increased current capacity, shown below.

When LEF macros contain bumps on both the top and bottom substrate layers, two unique logic terms are required in the design for proper Verilog netlist construction. Often term list files are created for each die and a separate file for all bottom side C4 bumps.

You can import LEF pad files while creating the substrate.

You can also import LEF pad files separately by following these steps:

1. Choose either *File – Import – LEF/DEF* from the menu bar or *IC – Import – LEF/DEF* from the Flow Manager. This opens the *Import LEF/DEF* dialog.
2. Select the substrate for which the padstack is to be created in the *Substrate* field.
3. Use *Add files* to browse and select LEF pad files. Once LEF pad files are selected, they appear in the *Import Files* list. Right-click a file to view options to adjust the file import order.



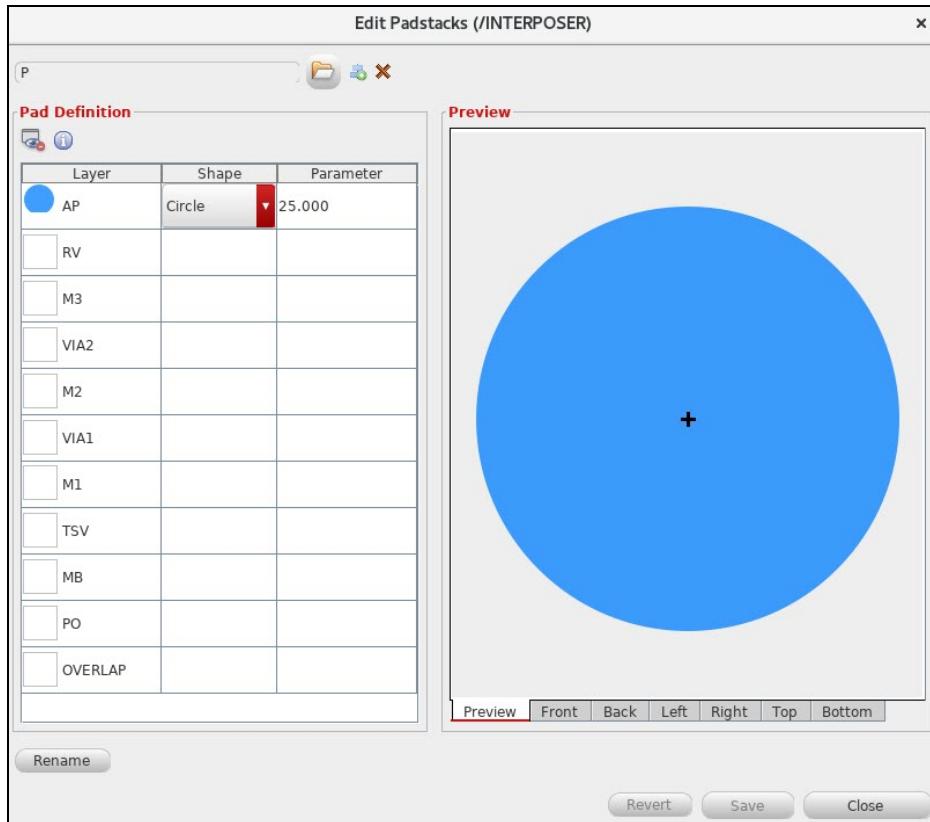
4. Click *Apply* to create padstacks based on the LEF file content.

Related Information

- [Creating Substrates by Importing a LEF Tech or CSV Layers File](#)
- [Defining Term-based Connectivity for Innovus or Allegro X Layout Editors](#)
- [Creating Padstacks and Bump Devices](#)

Creating a Padstack by Using the Edit Padstacks Form

Padstacks can be created and modified by right-clicking any device with the target substrate in the *Device Hierarchy* and choosing *Padstack – Edit Padstacks*. This opens the *Edit Padstacks* dialog, as shown below.



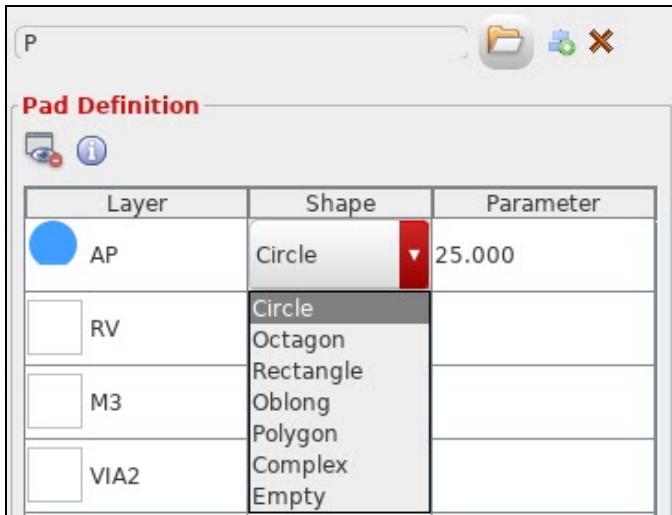
Existing padstacks defined for the substrate appear when clicking the file browser icon at the top of the dialog, with the following exception:

- Padstacks created by importing LEF files will not appear in the Edit Padstack dialog. Such pads can be modified only by editing the LEF files directly, which is preferred so as to keep the LEF files as the source.

To modify an existing padstack, choose a padstack from the pull-down menu, and the pad configuration for each layer on the substrate is displayed.

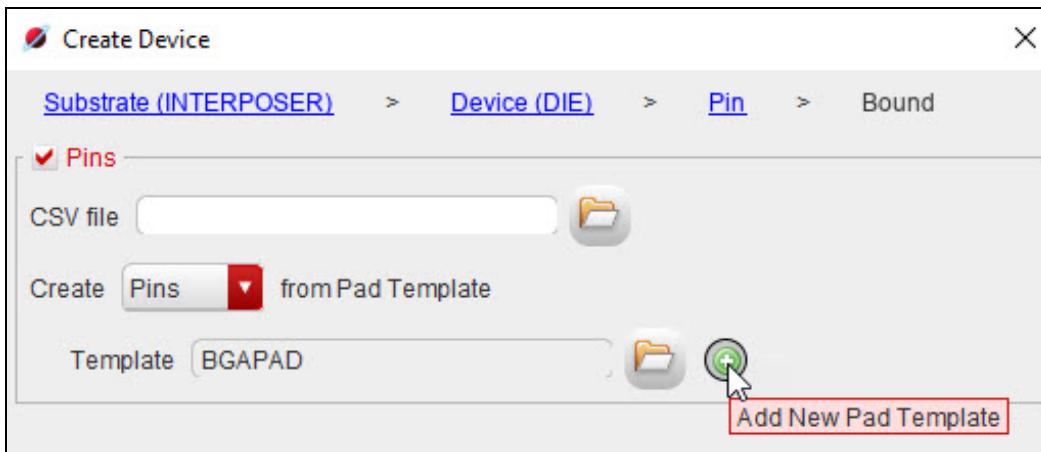
To create a new padstack, click the *Create new padstack* icon () and enter a padstack name when prompted.

Pad shapes can be defined on any layer by clicking in the *Shape* field and choosing a value from the pull-down menu. Pad sizes can be entered by clicking in the *Parameter* field and entering the value. The *Enter* key must be pressed for the change to take effect. Padstacks can be deleted by clicking the *Delete current selected padstack* icon ().



Configuring Padstacks while Creating Devices or Packages

The *Pins* pane of the *Create Device* dialog enables selection of an existing padstack or creation of a new padstack, as shown below.



When creating a device from a CSV pin list file, the file can include a PADSTACK column, where one or more padstack names can be defined. If the CSV pin list file contains PADSTACK entries, the pins will be created with the padstacks defined. If the padstack does not exist, it will be created with default parameters that can be modified later using the *Padstack – Edit Padstack* command.

Defining Padstacks while Importing Device Source Files

When creating devices by importing TXT or CSV format files that do not have padstacks defined in them, the *Edit Padstacks* dialog is presented to enable the pad size to be defined. The device will be created with the padstack parameters defined.

Related Information

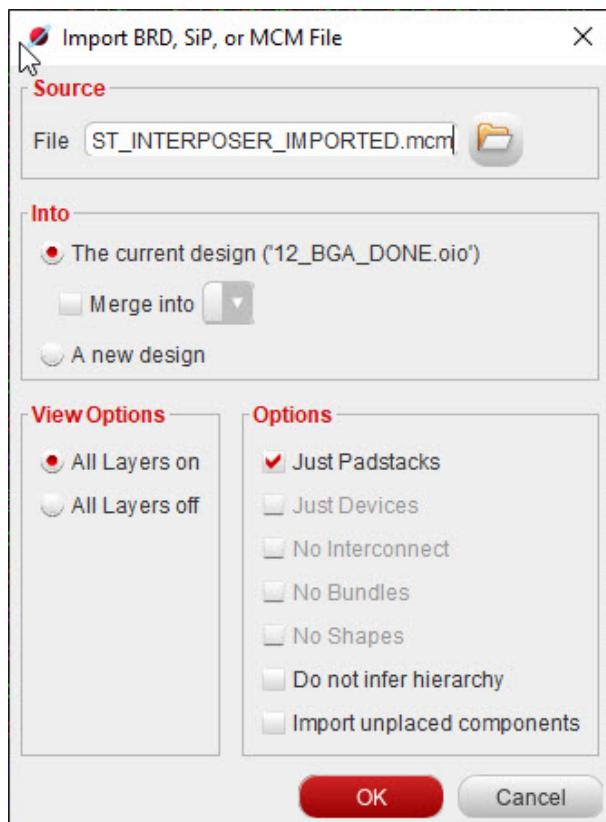
- [Creating Padstacks and Bump Devices](#)

Importing Padstacks from Allegro X Advanced Package Designer

Layer stackups and their matching padstacks can be imported from an existing Allegro X APD .mcm file.

To import padstacks from Allegro perform the following steps:

1. Select *File – Import – Allegro BRD, SiP, or MCM* to invoke the dialog below:



2. Select the *Just Padstacks* option to import the layer stackup and the padstacks from the .mcm file. Notice the substrate appears in the Layers view and the padstacks will be available for new devices.

Related Information

- [Creating Padstacks and Bump Devices](#)

Creating a Padstack While Creating Bump Devices Interactively

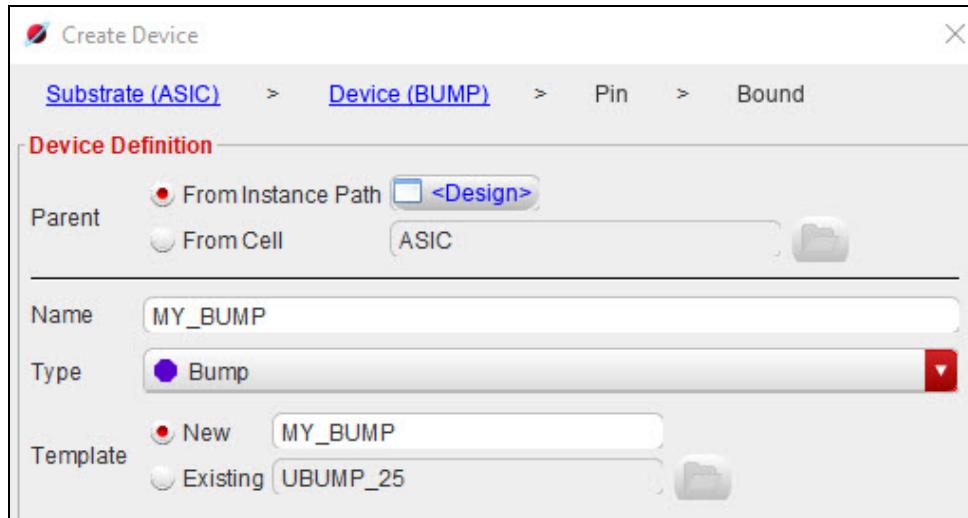
Cadence recommends defining bumps in a LEF file. However, you can create bump devices as a standalone device that can be used for device or pin pattern creation.

To create a pad device, do the following:

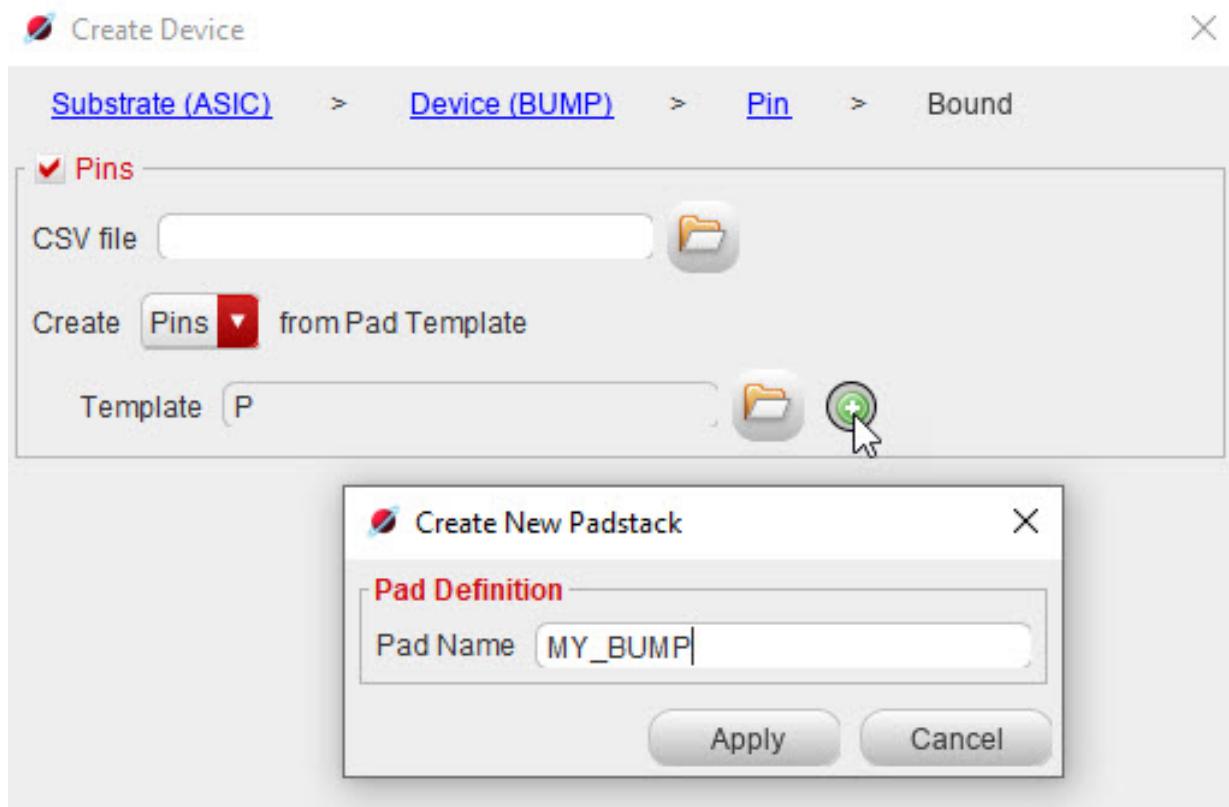
1. Choose *Edit – Devices – Create Device* from the menu bar or *IC – Create Device* from the

Flow Manager.

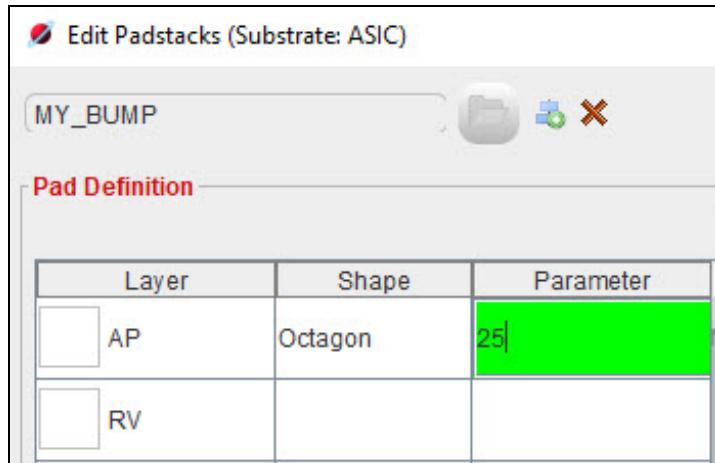
2. Select the target substrate from the *Substrate* pull-down menu and click *Next* to continue.
3. Set the *Type to Bump*.
Type Bump is only available in Die type substrates.
4. Either enter a new, unique device template name, or select an existing device template in the *Template* section.



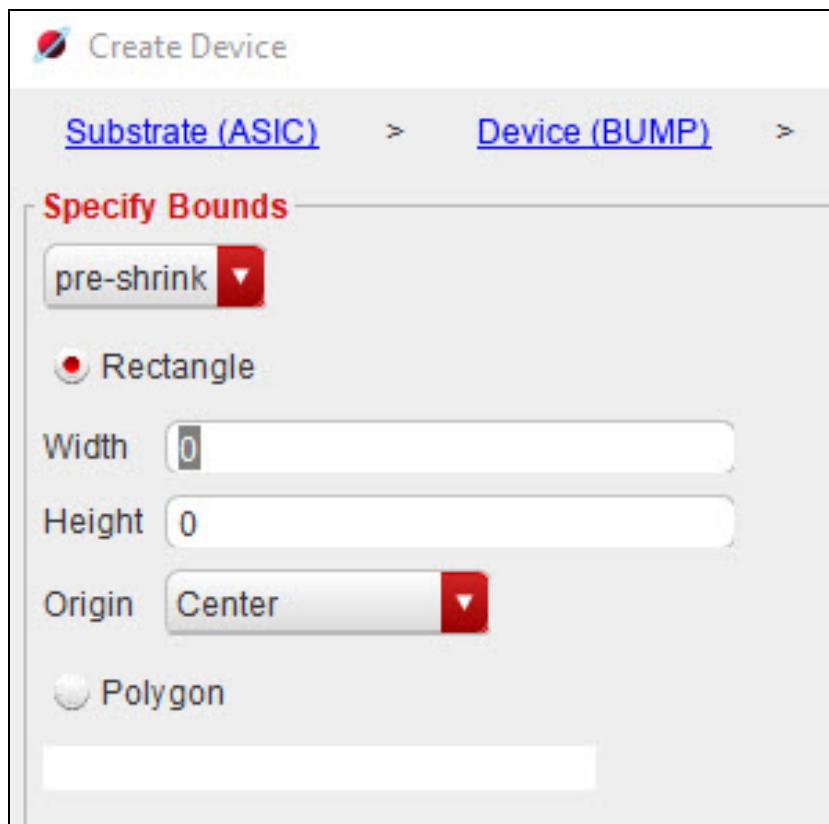
5. Click *Next* to continue.
6. Even though it is not intuitive for bump devices, select the default *Pins* value in the *Create from Pad Template* drop-down list. This option enables the Edit Padstacks dialog to be used to define bump sizes interactively.
7. Click the Add New Pad Template icon . In the resulting Create New Padstack dialog, enter the *Pad Name* you defined in the previous pane and click *Apply*.



8. In the *Edit Padstacks* dialog, specify the pad parameters. The *Enter* field needs to be pressed to initiate the *Size* entries.



9. Click *Save & Close*.
10. Click *Next* to display the *Specify Bounds* pane, as shown below.



11. Click *Create* and *Close* to create the bump devices with the default parameters.

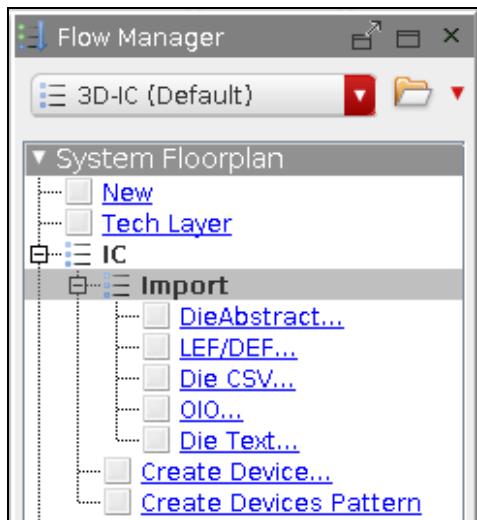
Related Information

- [Creating Padstacks and Bump Devices](#)

Creating Devices

Several methods are available to create devices in System Planner. This section focuses on creating the various types of devices used with typical multi-die package, 2.5D, and 3D-IC designs.

Select *File – Import* from the menu bar or expand the *IC – Import* folder in the *Flow Manager* to view the options, as shown below.

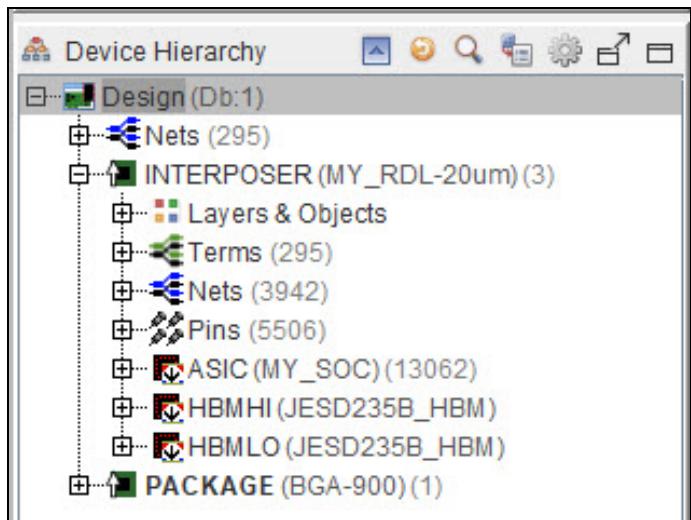


Understanding Device Templates

Each device will have a unique device name and a device template. For duplicated devices, the device templates should be the same.

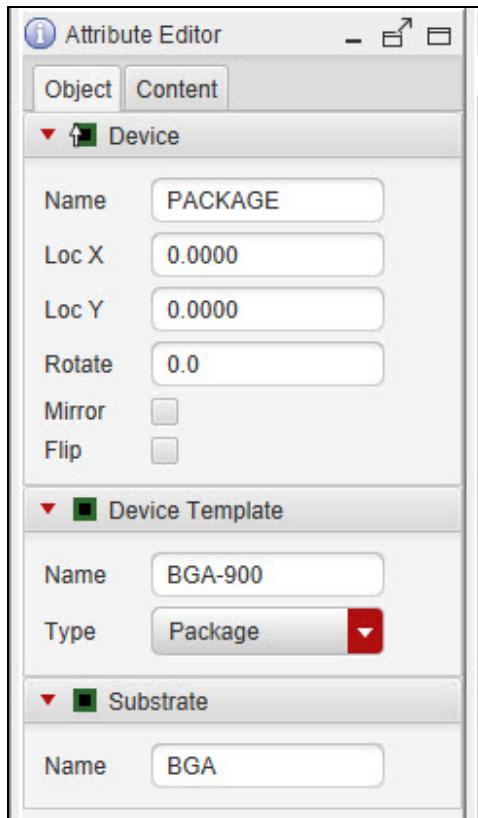
Some selection and replacement commands rely on device template names. Use forethought when defining the device template names in order to easily identify them or to use selection filtering.

The device template is displayed in parenthesis next to the device name in the Device Hierarchy, as shown below.



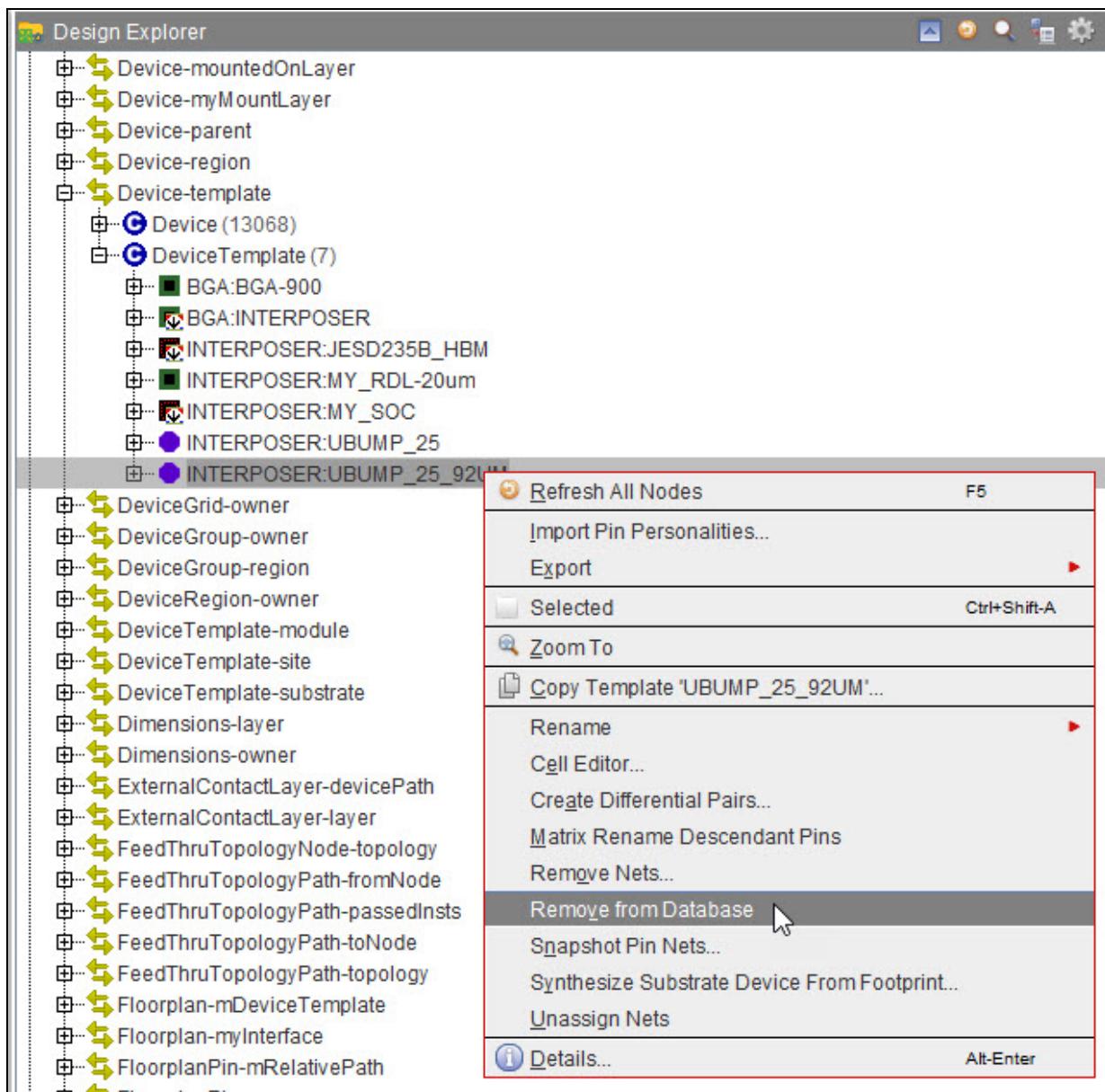
Changing the Device Template Name

The device template name can be modified using the Attribute Editor in the *Device Template - Name* field, as shown below.



Deleting Device Templates

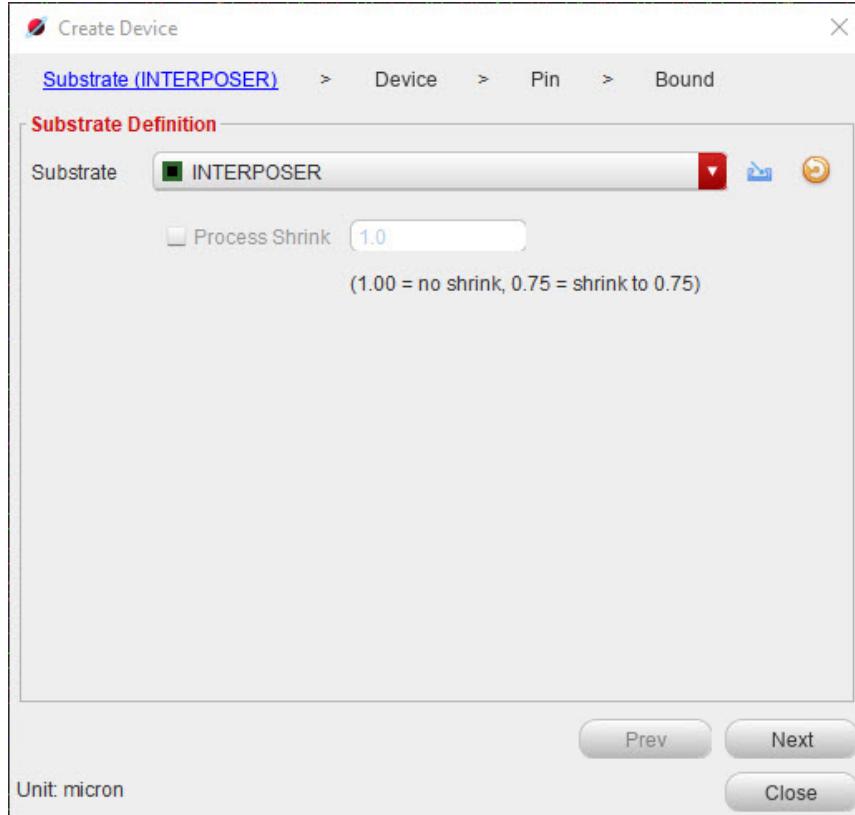
Device templates can be removed using a couple of methods. As devices are deleted, you are prompted to also remove the device template. Device templates can also be removed in the Design Explorer view. Expand *Device-template > DeviceTemplate* to view the templates in the design and right-click to choose *Remove from Database*, as shown below.



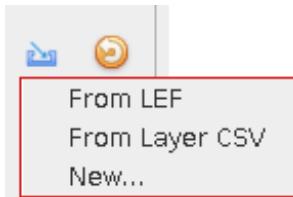
Creating a Device by Using the Create Device Form

The Create Device command can be used to create initially empty devices, such as Interposers or Packages. It can also be used to create devices using a CSV pin list file. Options for pad configurations are presented.

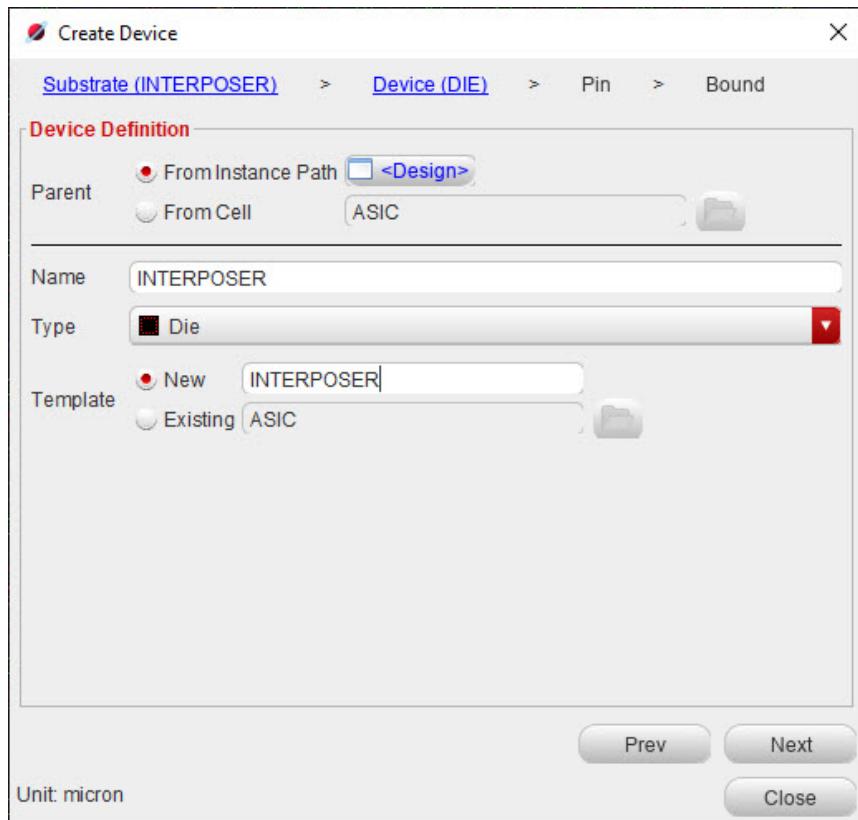
1. Choose either *Edit – Devices – Create Device* from the menu bar or *IC – Create Device* in the *Flow Manager* to invoke the first pane of the *Create Device* dialog, as shown below.



2. Devices need to target a specific substrate.
 - a. Specify one of the following in the dialog:
 - Select an existing target substrate for the device *Substrate*.
 - Create a new substrate by selecting the *Import Layers* icon and either importing a LEF or Layer CSV file or by selecting *New* to invoke the Layer Editor, as shown below.



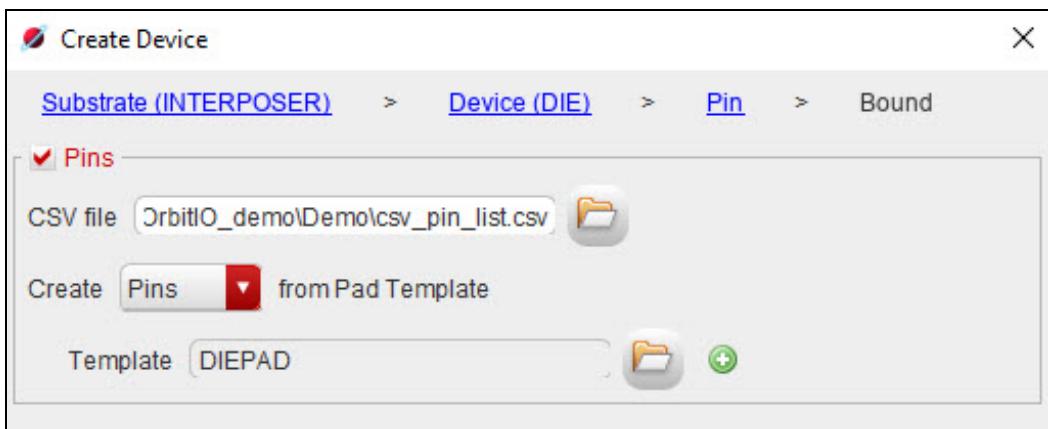
- b. In the *Process Shrink* field, you can specify the intended shrink percentage. The pin coordinates will be scaled appropriately upon import.
 - c. Click *Next* to continue to the second pane of the dialog.
3. In the next pane that is displayed, specify the following:
- a. Choose the parent.
 - a. Use the default *From Instance Path* to create the device at a specific location in the Device Hierarchy and select the desired level of hierarchy level by clicking on *Design* and selecting it in the *Select Root* dialog box.
 - b. Use the *From Cell* option to define a specific device in the Device Hierarchy to create the device under.
 - b. Enter a new, unique device name in the *Name* section.
 - c. Select the type of device template to create using the *Type* pull-down menu.
 - d. Enter a new, unique device template name in the *New* field or select an *Existing* device template in the *Template* section.
 - e. Click *Next* to continue.



4. In the next pane, select *Pins* only if a CSV pin list file is being used to import pins for the device.

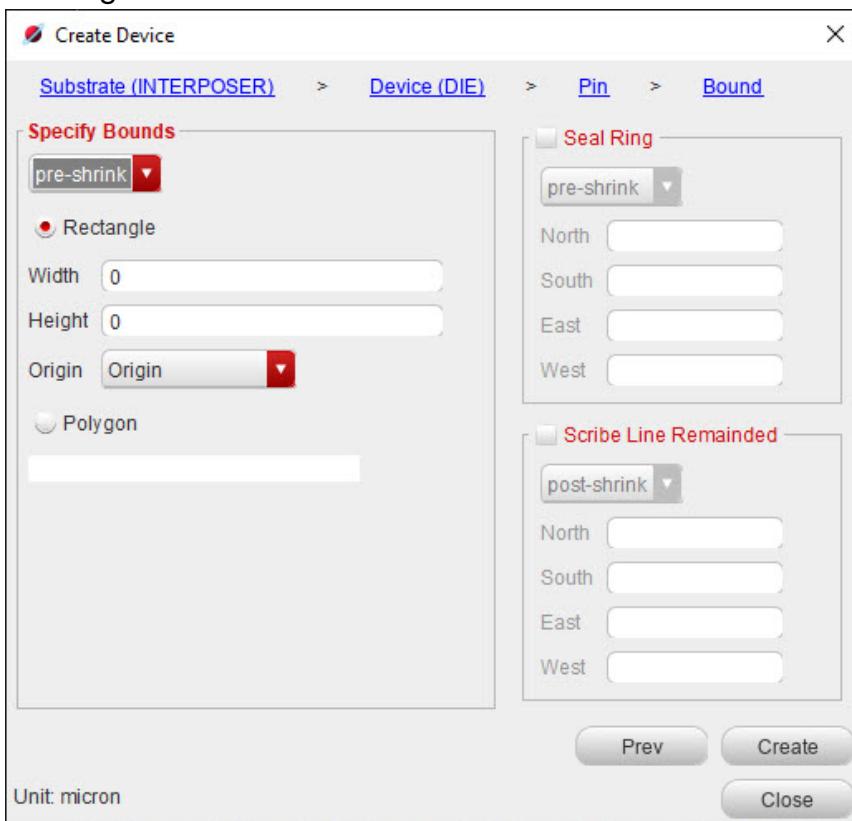
Leave the *Pins* check box unchecked if you do not want to import a pins list or define default pad sizes.

5. If you have opted to select the *Pins* check box, specify the following in the dialog:
- Specify a CSV pins list in the *CSV file* field. The CSV pin list file format is described in [Creating a Device by Importing a Pins List CSV File](#).
 - Select either *Pins* or *Devices* from the *Create* pull-down menu. *Devices* style pads are typically used with LEF bumps, while *Pins* are used for package designs with padstacks.
 - Click *Next* to continue.



6. Specify the following in the Bound pane of the dialog, as shown below.

- Specify the device size by entering values for *Width* and *Height*. Using the default 0 values will create a device size based on the minimum bounding box around all device pins.
- In the *Seal Ring* and *Scribe Line Remained* fields, you can enter the *North*, *South*, *East*, and *West* distance values. The values are incremental distances based from the existing device outline.



7. Click *Create* to complete creating the device based on the defined inputs.

If required, you can first click *Prev* to cycle back through the dialog panes, for example, to modify any settings or to create additional devices using some of the data already defined. Click *Close* to close the *Create Device* dialog without creating the device.

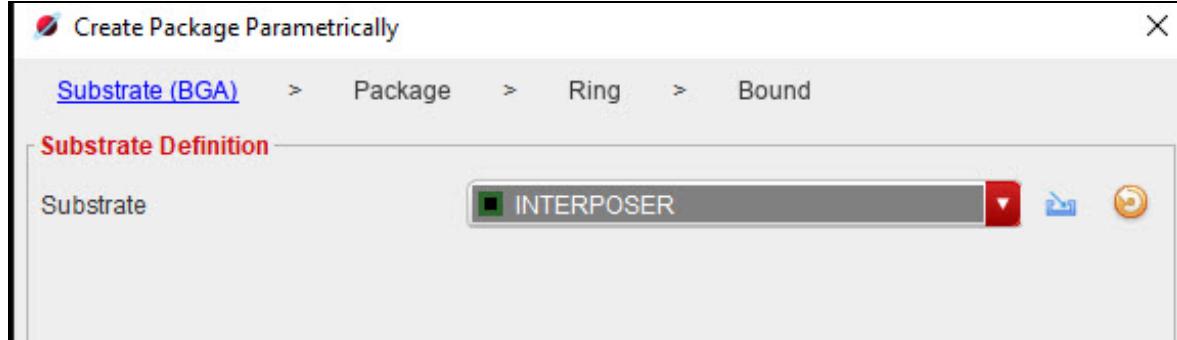
Related Information

[Creating Padstacks and Bump Devices](#)

Creating a Package Device

Package devices can be automatically created with the Ball pad pattern generated by using a variety of parameters. Many configuration options are presented to create a variety of typical package ball patterns.

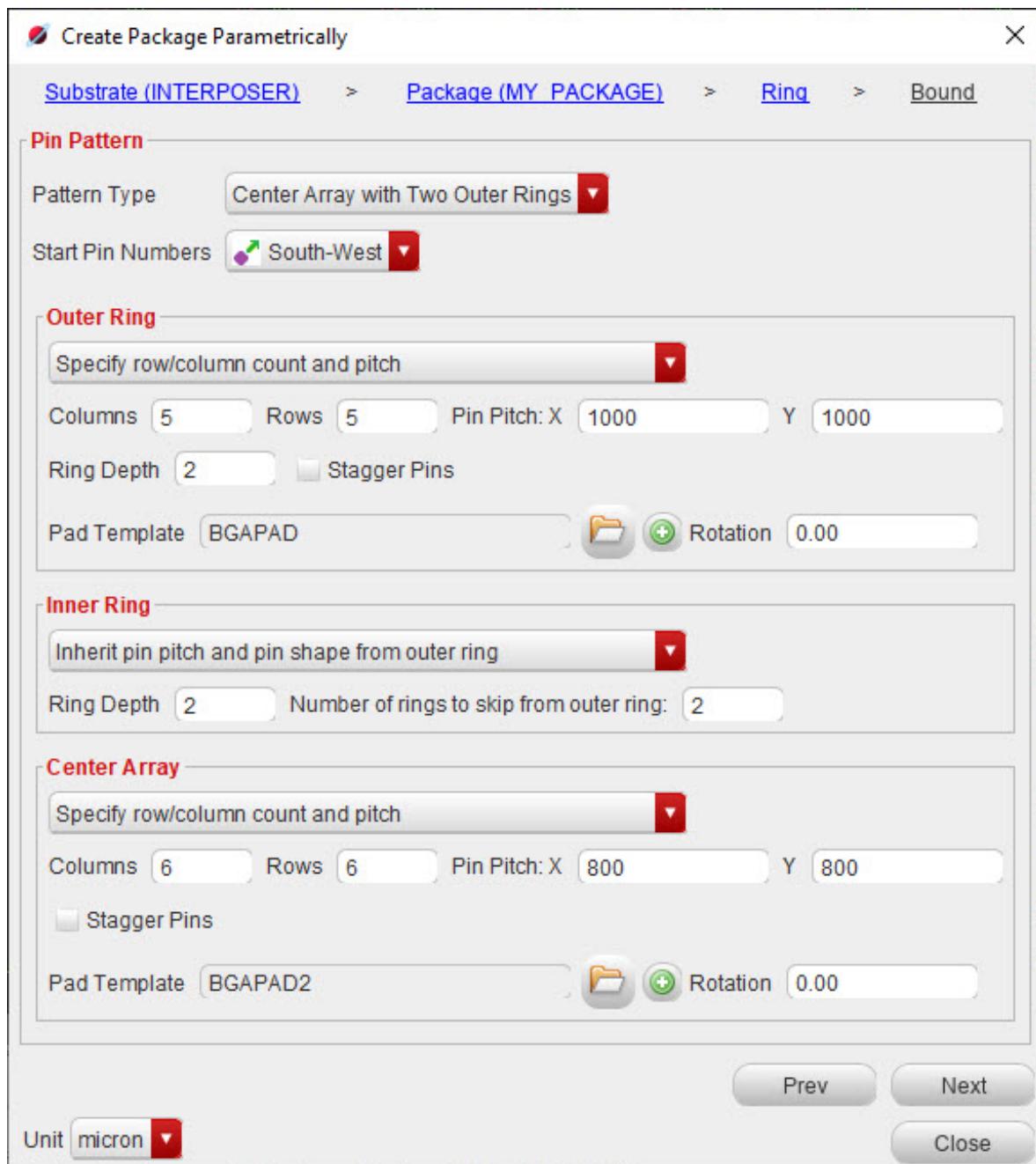
1. To create a package device, select *Edit – Devices – Create Package* or *PKG – Create Package* in the Flow Manager to invoke the Create Package Parametrically dialog.



2. Select an existing *Substrate* to target or select the *Import layers* icon to import or create a new substrate and click *Next*.
3. Enter a *Template* name and device *Name*.
4. Click *Next*.
5. Define the pin pattern parameters in the *Ring* panel.
 - a. Choose a *Pattern Type*. These include Uniform Array, One Ring, Two Rings, and Center

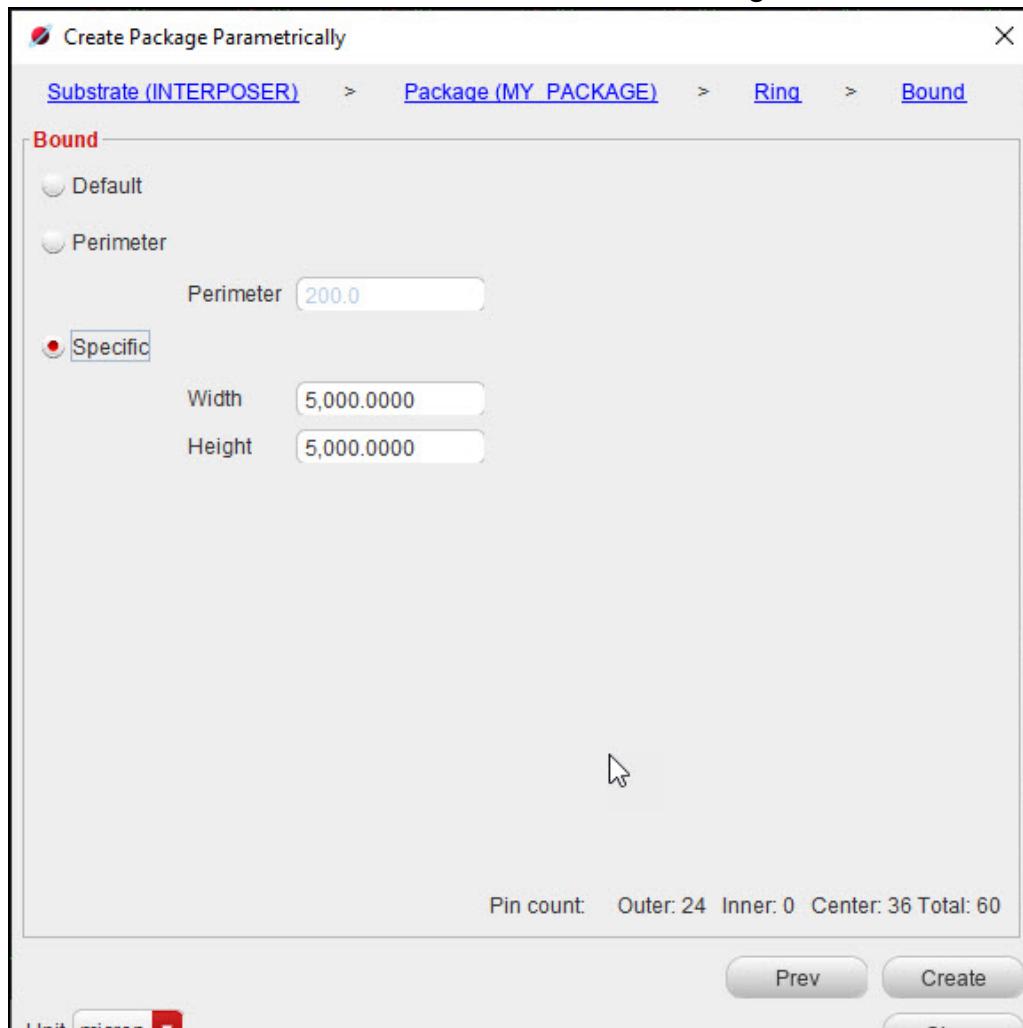
Array with Outer Ring(s). Center Array with Outer Rings patterns can use two pad sizes and spacing values for the array and rings, as shown above.

- b. Select the origin to begin numbering pins from the *Start Pin Numbers* pulldown menu.
- c. Depending on the Pin Pattern selected, define the *Outer Ring*, *Inner Ring* and/or *Center Array* parameters.
- d. Select an existing *Pad Template* or create a new one by clicking the *Add New Pad Template* icon ().
- e. Complete the desired parameters and click *Next*.



6. Define the Package size in the *Bound* pane using one of the following methods.
 - a. Choose *Default* to use a size equaling the minimum bounding box that contains all of the pads.
 - b. Choose *Perimeter* and enter a value to grow the default package size by *Perimeter* value entered.
 - c. Choose *Specific* to enter a *Width* and *Height* values for the package.

Notice the Pin count totals at the bottom of the dialog.



7. Click *Create* to create the package.

Creating a Device by Importing a Die Abstract File

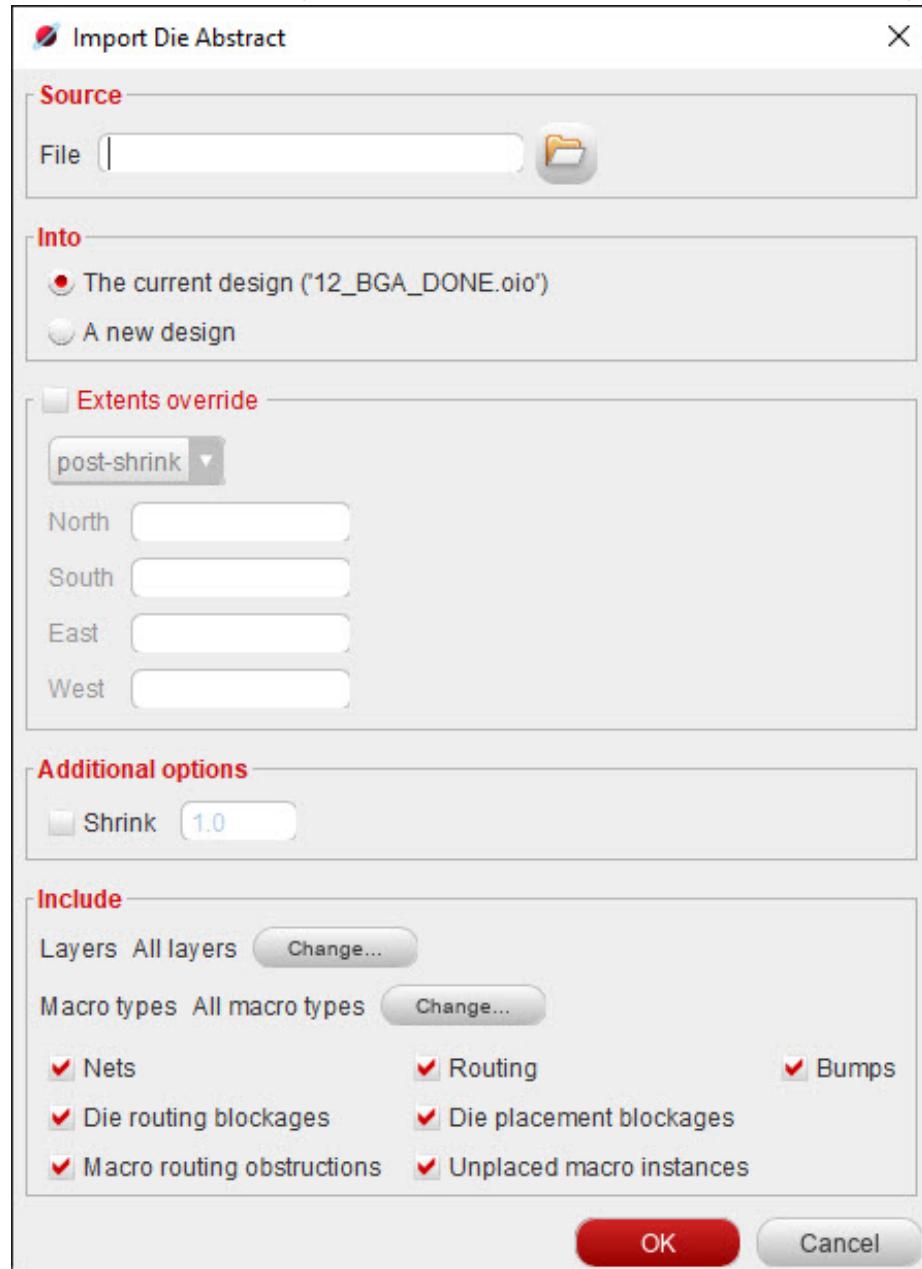
A die abstract file is a Cadence-defined format designed to pass single-pad-layer die information easily between System Planner, Allegro, and Innovus. Either of these tools can also be used to create a die abstract file. The die abstract format is sometimes preferred when extra security is needed when sending files such as LEF/DEF between different design groups or companies.

Die abstract files use the *.xda* extension.

To create a device by importing a die abstract file:

1. Choose either *File – Import – Die Abstract* from the menu bar or *IC – Import – Die Abstract*

from the *Flow Manager* to invoke the *Import Die Abstract* dialog, as shown below.



2. Specify the following in the dialog:
 - a. Specify the die abstract file (.xda) in the *File* field.
 - b. Choose to import the specified die abstract into either the current design project or a new project in the *Into* section.
 - c. Specify the device boundary scribe information in the *Extents override* section.

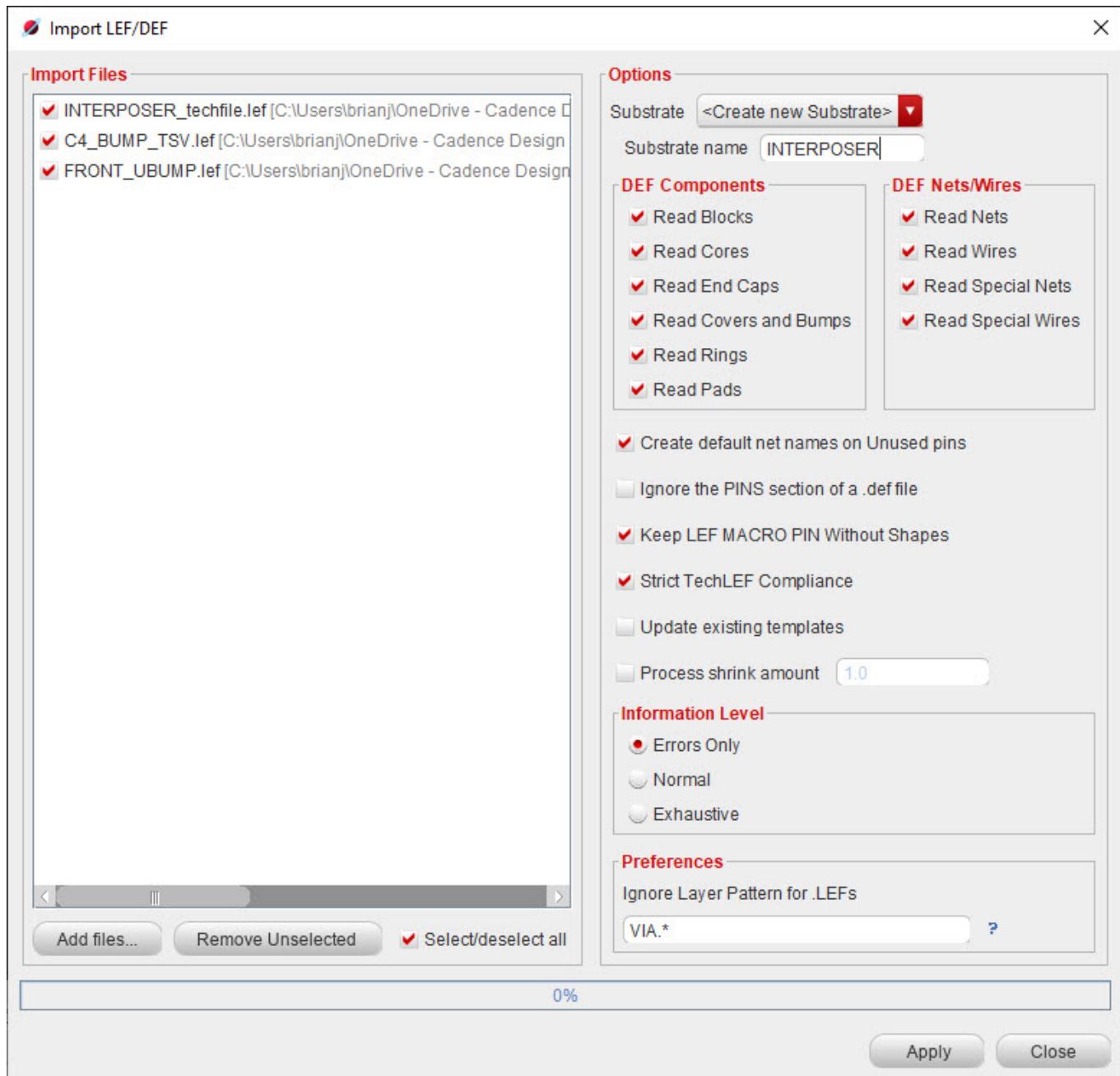
- d. Specify a *Shrink* value in the *Additional options* section. The shrink processing will take place during the import process.
 - e. Specify the data to be imported from the die abstract file in the *Include* section. This relies on the data being included in the file when it was created.
3. Click *OK* to create the device based on the input die abstract file data. A new substrate will be created to support the newly imported device.
- Importing a die abstract device automatically creates a substrate for the device matching the layer stackup used when the die abstract file was created. A die abstract device cannot be imported targeting an existing substrate. To update an existing device with a new die abstract file, use the *Tools > Merge Updated Die Abstract* command.

Creating a Device by Importing LEF/DEF Files

DEF format can be used to import the die bump locations. The DEF file should only contain the abstract bump information and not the die logic or routing.

To create a device by importing a LEF/DEF file:

1. Choose either *File – Import – LEF/DEF* from the menu bar or *IC – Import – LEF/DEF* from the *Flow Manager* to invoke the *LEF/DEF Reader* dialog, as shown below.



2. Specify the following in the dialog:

- Select an existing substrate to target the device or create a new substrate for the device in the *Substrate* pull-down menu.
- Enter a name in the *Substrate name* field if you selected *<Create new substrate>*.
- Click *Add files* to browse and select LEF/DEF source files. Once files are selected, they

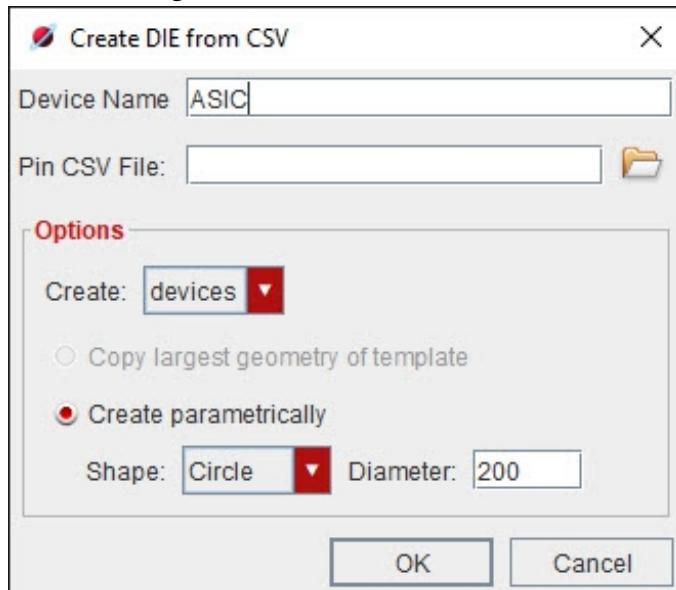
appear in the *Import Files* list. There are right-click options to adjust the file import order. The LEF tech file should be the first file shown in the *Import Files* list, followed by any LEF pad files and then the DEF file. Right-click a file to view options to adjust the file import order.

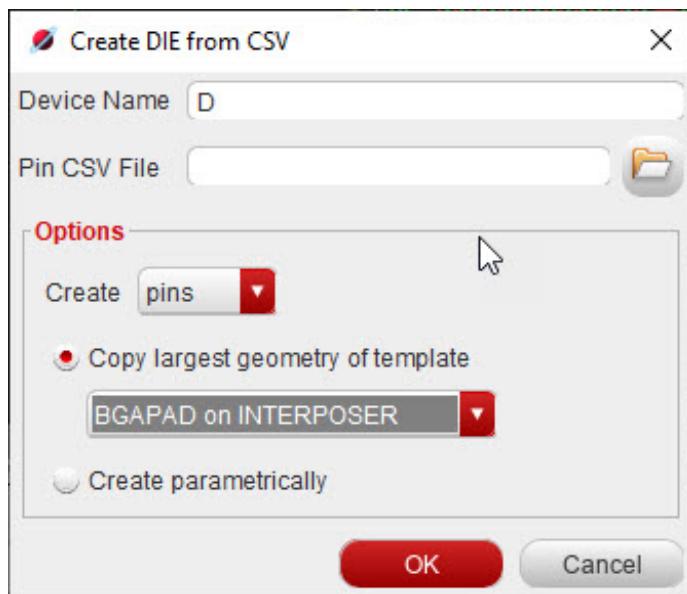
- d. Specify the remaining options in the dialog, as needed.
3. Click *Apply* to create the device with the specified LEF/DEF data.

Creating a Device by Importing a Die CSV File

Devices can be created by importing a die format CSV file:

1. Choose either *File – Import – Die CSV* from the menu bar or *IC – Import – Die CSV* from the *Flow Manager* to invoke the *Create DIE from CSV* dialog, as shown below.





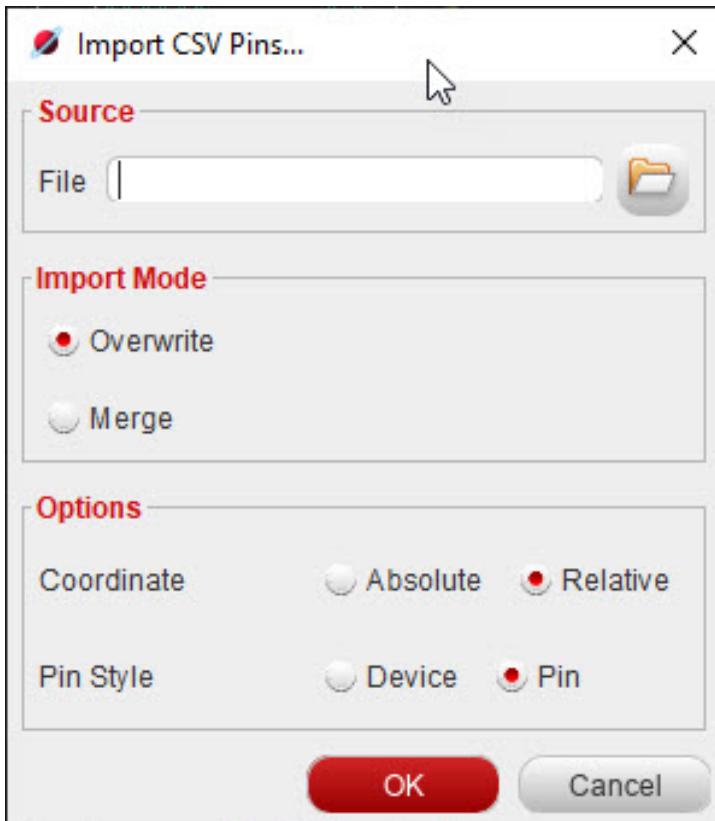
2. Specify the following in the dialog:
 - a. Enter a name for the device in the *Device Name* field.
 - b. Select an input pin format CSV file in the *Pin CSV File* field. To see the format for the Pin CSV file, click [here](#).
 - c. From the *Create* pull-down menu under *Options*, specify the pin style. Select *devices* style pins for IC design or *pins* style pins for package design.
 - d. Define pad parameters in the *Create parametrically* section to create padstacks. Alternatively, to use an existing padstack, select the *Copy largest geometry of template* option. If you select this option, the available padstacks are presented for selection from a pull-down menu.
3. Click *OK* to create the device based on the input file data. A new substrate will be created to support the newly imported device.

Creating a Device by Importing a Pins List CSV File

Devices can be created by importing a pin list format CSV file. The pin list can be specified when using the *Create Device* command. CSV pin lists are used for variety of applications such as to create a new device or to add or modify pins on an existing device. Importing CSV pin lists can be used to overwrite all existing device pins or be merged with them.

Pins list CSV files can be imported for an existing device by right-clicking the device in the *Device Hierarchy* and choosing the *Import – CSV Pins* command from the shortcut menu. The *Import CSV*

Pins dialog is presented, as shown below.



Complete the dialog as follows:

1. In the *File* field, browse to select the CSV pins list file to be imported.
2. In the *Import Mode* section, select *Overwrite* to replace existing device pins or *Merge* to merge the imported pins with the existing device pins. Existing pins will be redefined if included in the import CSV file.
3. In the *Options* section, set the *Coordinate* to base the pin coordinates as *Relative* to the device origin or *Absolute* to use design origin.
4. In the *Options* section, define the *Pins Style* as *Device* for Innovus or *Pin* for Allegro.
5. Click *OK* to import the device pins with the defined options.

The format for the pin list CSV file is displayed below. The required columns are highlighted. The other columns are optional.

| PIN NUMBER | PIN NAME | PIN USE | X | Y | PIN_DIRECTION | PADSTACK | NET_NAME | FIXED | ROTATION | DEVICE_PIN_TERM | DEVICE_PIN_TEMPLATE | PIN_PERSONALITY |
|------------|----------|---------|--------|--------|---------------|--------------|----------|-------|----------|-----------------|---------------------|-----------------|
| A1 | | | -12000 | -12000 | | BGA_BALL_PAD | | FALSE | 0 | | | |
| B1 | DTB2[3] | SIGNAL | -12000 | -11200 | INOUT | C4_DIE_PAD | DTB2[3] | FALSE | 0 | | | |
| C1 | VDD | POWER | -12000 | -10400 | INOUT | BGA_BALL_PAD | VDD | FALSE | 0 | | | |
| D1 | VSS | GROUND | -12000 | -9600 | INOUT | BGA_BALL_PAD | VSS | FALSE | 0 | | | |
| E1 | DTB2[16] | SIGNAL | -12000 | -8800 | INOUT | C4_DIE_PAD | DTB2[16] | FALSE | 0 | | | |
| F1 | DTB2[15] | SIGNAL | -12000 | -8000 | INOUT | C4_DIE_PAD | DTB2[15] | FALSE | 0 | | | |

Related Information

[Creating a Device by Using the Create Device Form](#)

Importing a Device from Another System Planner Project

A device from an existing System Planner project can be used as an input to another project. The input .oio file must first be created by right-clicking the device in the source project and choosing *Export – O/I/O*. The exported file can then be used as the source for importing the device into another project. The OIO file being imported must be prepared this way.

To import a System Planner .oio source file, choose *File – Import – O/I/O* from the menu bar or *IC – Import – O/I/O* from the *Flow Manager*. A file browser is presented to choose the input .oio file. A device is created based on the data in the source .oio file.

The *File – Import – O/I/O* command will not accept .oio files for existing designs.

Cadence does not recommend using this format for die or package creation.

Creating a Device by Importing a Die Text

A device can be created by importing a text format file:

1. Choose *File – Import – Die Text* from the menu bar or *IC – Import – Die Text* from the *Flow Manager* to invoke the *Import Die Txt* dialog, as shown below.



2. Specify the following in the dialog:
 - a. In the *File* field, browse to select the input text format file.
 - b. Set the *Coordinates* as either *Absolute* or *Relative*, depending on how the source text file was created.
 - c. Create a new substrate or select an existing project substrate from the *Substrate* pull-down menu.
 - d. Select the *Pin Style*. Select *Pin* for Allegro design or when LEF bumps are not being used for the die being created. Select *Device* style when the target substrate has LEF pads available for use with this device.
 - e. Click inside the *Parent Device* field to open the Select Root form, where you can select an existing device as parent for the new device in the *Device Hierarchy*. The default is the top-level of the design.
 - f. Define a *Process shrink* value for the shrink processing to take place during import.
3. Click *OK* to complete creating the device based on the text file data.

The format for the TXT file is displayed below. The *Net Name* column is optional.

```

File: C:\Users\brianj\OneDrive - Cadence Design Systems Inc\Documents\System_Planner_Training
Date: Fri Feb 24 11:29:54 MST 2023
Units: microns, 3 decimal places
Name: JESD235B_HBM
RefDes: HBMHI
Origin: 0.0 0.0)
Rotation: 0.00000
Pad Layer: Bump
Extents: ((-1633.00000 -3036.250000) (1633.00000 3036.250000))
DieOrient: ChipUp
DieType: FlipChip

Pin Number Pin Name Padstack X Coord Y Coord Rotation Net Name
1 AERRA UBUMP_25 1080.0 1581.25 0 AERRA
2 ARFUA1 UBUMP_25 1368.0 636.25 0 ARFUA1
3 CH1 UBUMP_25 936.0 -1581.25 0 CH1
    
```

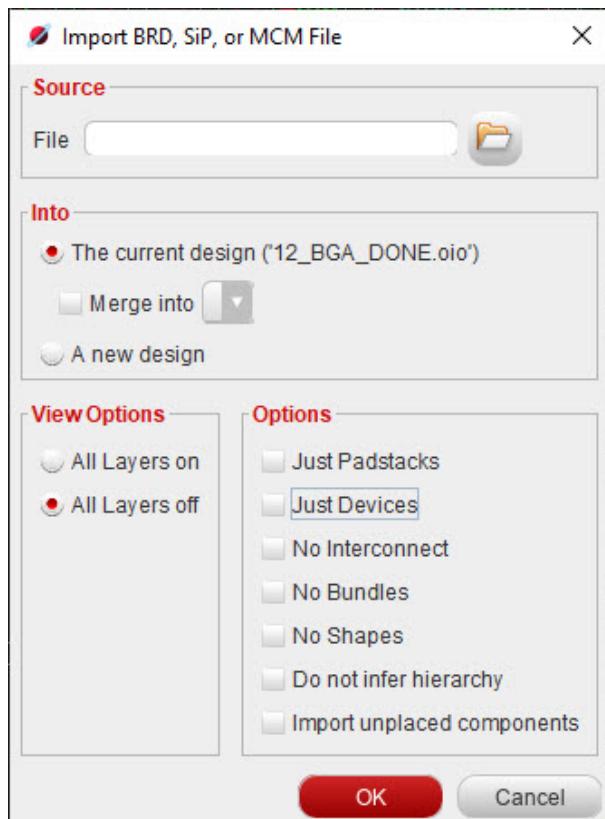
Additional information can be defined in the die text file such as:

| Information | Description |
|-------------|---|
| Units | Project units. Use micron. |
| Name | Device template name. |
| RefDes | Device name. |
| Origin | Origin of the component, often 0.0. |
| Rotation | Import orientation. |
| Pad Layer | Layer to create pads on targeted substrate - can be edited to import devices onto a single substrate Allegro design. (ZZZ Link to Single substrate designs in Defined Substrate Layers section) |
| Extents | Outline extents coordinates form origin. |
| DieOrient | Flip chip orientation. |
| Die Type | FlipChip or WireBond. |

Importing Devices from Allegro X Designs

Devices can be created from existing Allegro X designs. These can be .brd, .sip, or .mcm Allegro designs. After importing devices from a design, they can be either used within that version of the design or exported and reused in other designs. Various formats such as die text, pin CSV, die abstract could be used to reuse the devices in other designs. By default, the devices will be created with their current net assignments, but that can be modified afterward. Die, interposer, package, and discrete devices can be imported for reuse. Discrete components will be imported as IP Components. This device type will ensure a DISCRETE type is defined in Allegro layout or substrate editors when importing the System Planner design.

To import an Allegro design, select *File > Import > Allegro BRD, SiP, or MCM* or select *PKG > Import > Allegro BRD, SiP, or MCM* in the Flow Manager to invoke the Import BRD, SiP, or MCM File dialog, as shown below.



Browse to select the .brd, .sip, or .mcm File.

You can configure the following options:

- *Into* - import the design into *The current design* or *A new design*. The *Merge into* function is not yet available.

- *View Options*- Option to set all of layers of the imported substrate either *All layers on* or *All layers off* in the Layers view.
- *Options*:
 - *Just Padstacks* - Imports only the substrate layer stackup defined in the imported Allegro file.
 - *Just Devices* - (Preferred) - Creates only the logical design with no routing , vias, obstructions, and so on, which is typically all that is desired for system planning.
 - *No Interconnect* - Imports everything except routed metal.
 - *No Bundles* - Excludes all defined bundles.
 - *No Shapes* - Excludes all metal shapes.
 - *Do not infer hierarchy* - Creates a flat design with the die and the package at the same level of the hierarchy. Often, Allegro designs are built this way with a top-level parent design and the package and die as child devices. Otherwise, the design will be built with the package as the parent level with each child die device underneath it.
 - *Import unplaced components* - Imports devices not yet placed in the Allegro design.

Once the design is imported, the desired devices can be selected and exported for reuse. Cadence recommends die text files because they contain the size and layer information and can be easily modified. Die text files can also be exported directly from Allegro and imported into the System Planner to skip this import step all together.

Adding a Seal Ring or Scribe Line to a Die Device

Seal Rings and Scribe lines can be added to the device for use in manufacturing rule checks during implementation. These values are transferred to the proper object layers in Allegro to enable these checks. These values can only be added to Die type devices.

The Create Device command enables a Seal ring or Scribe Line to be added in the last pane of the dialog.

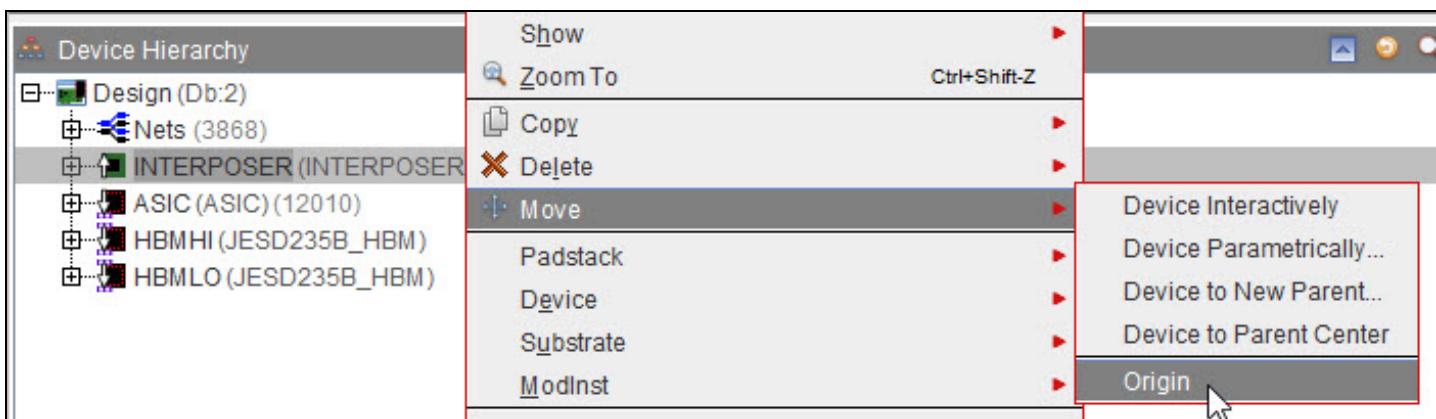
To add or modify a Seal Ring or Scribe Line for an existing device, select the device in the Device Hierarchy and right-click to choose *Device > Edit > Die Extents* to invoke the Die Extents dialog as shown below.



Click the options desired and enter the *North*, *South*, *East* and *West* values. These entries are incremental distances from the existing die outline. Visibility can be controlled in the Objects view under *Devices > Die > Seal Ring or Scribe Line*.

Changing the Origin of a Device

Occasionally, the origin of a device needs to be changed to enable alignment, exact placement, or easier rotation of the device in place when floorplanning. To change the origin of a device, right-click the device in the Device Hierarchy and choose *Move – Origin*, as shown below.



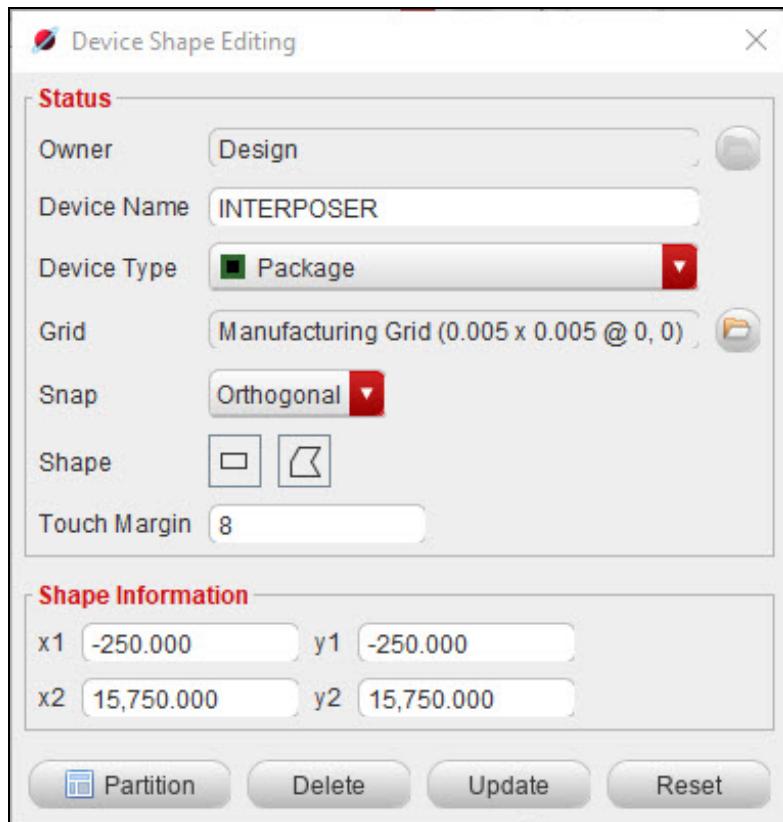
The *Move Origin* dialog is presented to enable setting the following:

- The *Coordinate* pulldown menu can be set to *Relative* for device coordinates or *Absolute* to use the design coordinates.
- The Position pulldown menu can be set to either corner or *Center*.

Setting a Specific Device Origin

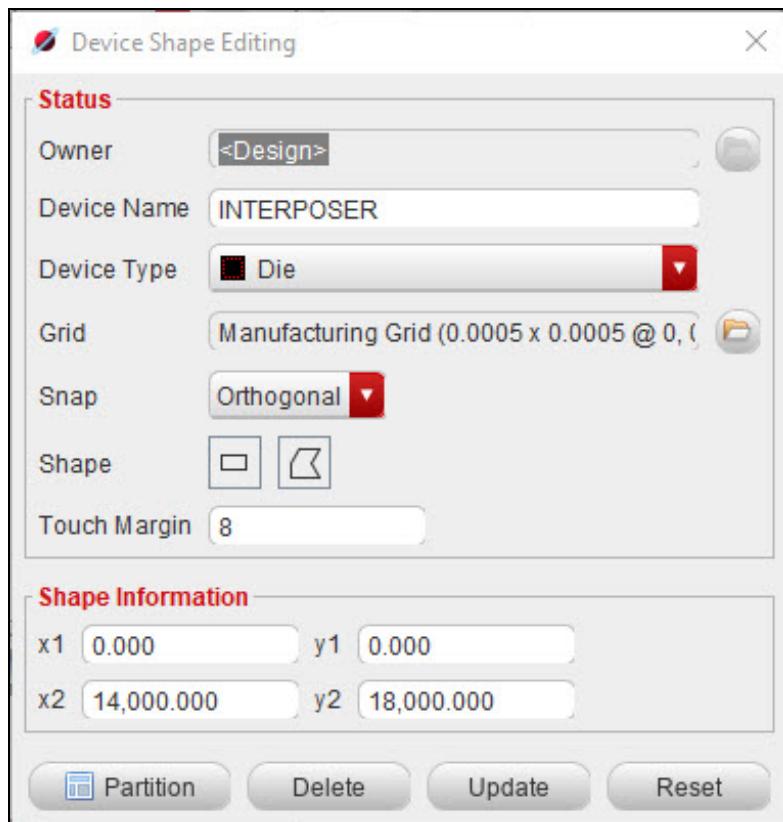
A specific coordinate such as pin 1 can be set as the device origin. To set a specific origin coordinate, select the device and right-click to choose *Device > Edit > Device* to display the Device Shape Editing dialog.

The *Shape Information* coordinates can be set to define an origin with a specific coordinate.



Changing the Size of a Device

The size of an existing device can be modified by selecting the device and right-clicking to choose the *Device – Edit – Device* command. The *Device Shape Edit* dialog is presented, as shown below.



This puts you into Device Shape Editing Mode. Modify the *Shape* and *x1,x2,y1,y2* fields to change the die size. The die size can also be modified interactively by dragging the die outline edges.

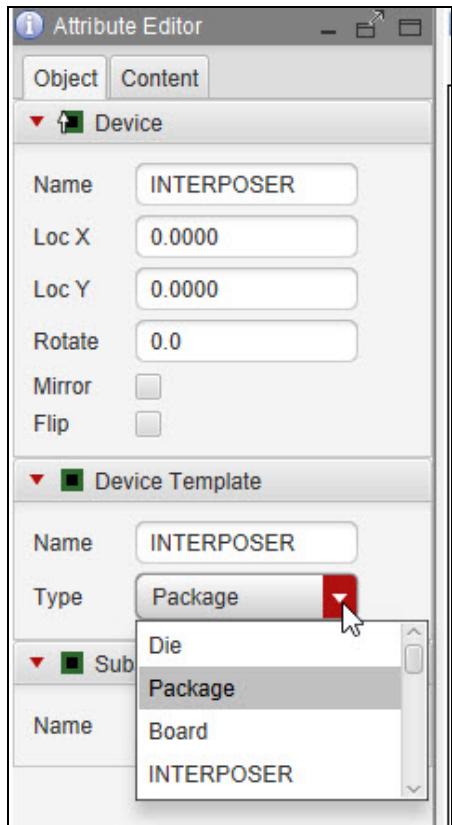
Click the window header X to close the form and select Interactive Select Mode from the toolbar to exit *Device Shape Editing* mode.

Changing the Type of a Device

Occasionally, the device type needs to be changed for a variety of reasons. Certain commands are only available for specific device types. A few examples include *Export – Die Abstract* is only available for Die type devices, *Ungroup* is only available to Group types, and Allegro – Import – Integrity System Planner features are designed to only work with Package type devices.

To change a device type, select the device in the Device Hierarchy. In the Attribute Editor, change

the Type, as shown below.



Each device type will display a different icon in front of the device in the Device Hierarchy. The main device types used are Die, Package, IP Component (DISCRETE), Contact Device, and Group.

Cadence does not recommend using the device type INTERPOSER. Use either Die for silicon interposers or Package for RDL interposers.

Floorplanning Devices

There are several features to manipulate the placement of devices in the design. Ultimately, the connectivity between the devices in the design can help dictate the proper placement, orientation, and alignment.

System Planner provides system-level connectivity display across the design. Several [connectivity optimization features](#) are available to help reduce connection crossings and length.

The following can be used to explore and define device placement in the design.

- [Flipping Devices](#)
- [Moving Devices](#)
- [Copying Devices](#)
- [Device Table Editor](#)
- [Importing a CSV Devices File](#)
- [Ruler Mode for Measuring Devices](#)
- [Grouping Devices](#)
- [Using the Devices View to Selectively Display Devices](#)
- [Using Inspect Mode](#)

Flipping Devices

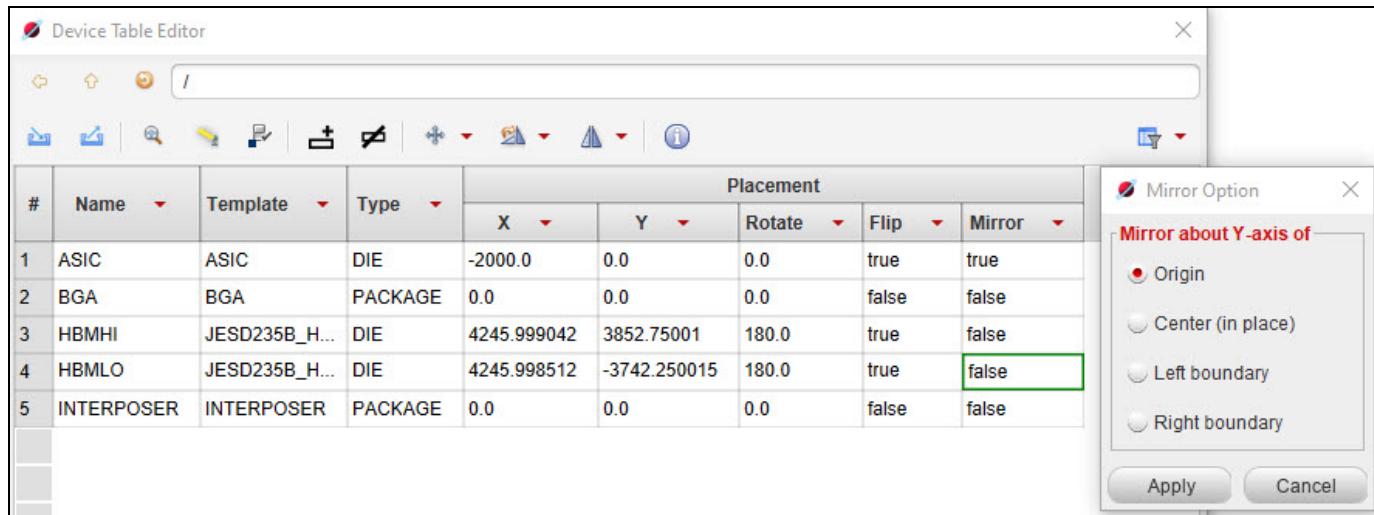
Flip-chip devices are often imported or created with their original orientation and need to be flipped to accurately interface with their contacted device. To flip a device, right-click the device in the Device Hierarchy and choose the *Flip in Place* command. The icon displayed next to the device in the Device Hierarchy panel will now point down rather than up. Flipping can also be performed in the Attribute Editor and in the Device Table Editor.

The icon in the Device Hierarchy will have an arrow indicating the direction the bumps are facing, up or down.

Handling Already Flipped Die Sources

Occasionally, die files are provided that are already flipped in the source data. System Planner assumes that all die will need to be flipped upon import.

To configure already flipped die source files, import the die source files and then select the parent level of hierarchy that contains the die in the Device Hierarchy. Right-click to choose *Devices Table*. Set the *Flip* field for the desired device to *true* and this will also set the *Mirror* field to *true*. Set the *Mirror* field back to false and click *OK* in the *Mirror Option* dialog to accept the Origin option, as shown below.



System Planner will treat the die as flipped without actually flipping it.

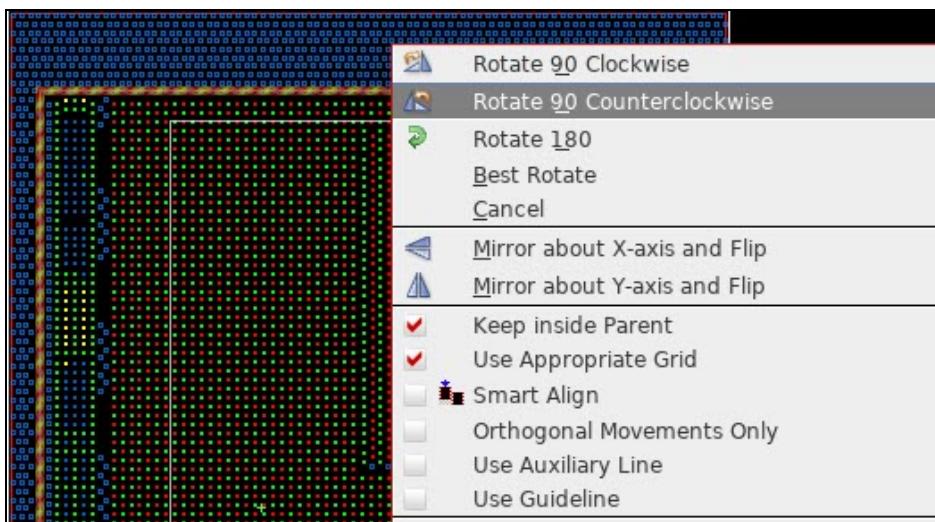
Moving Devices

Several commands and options are available to adjust the placement for devices.

Moving Devices Interactively

Right-click a device in the Device Hierarchy panel and choose *Move – Device Interactively* to begin dragging the device on the canvas.

Right-clicking will show options such as *Rotate* and *Mirror*. Clicking on the canvas will place the devices at that location. One of the default right-click options is *Keep Inside Parent*, which will prevent the device from being dragged outside the parent boundary. Unset this option to freely move the child device. Click *Cancel* to cancel or end the move operation.

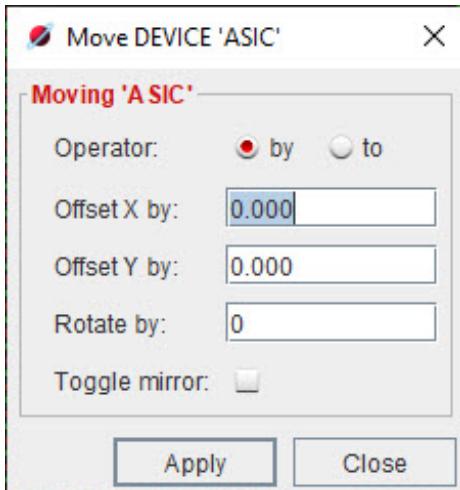


The *Interactive Move Mode* can also be used when moving devices. Select the main toolbar

Interactive Move Mode icon  to enter the mode. Objects can be selected for the move by clicking on them in the canvas. The mode remains active until another mode is selected, so different objects can be continually selected for the move. The *Selection Filter* is not activated while using this mode, so be very careful that you are moving the intended object. For example, a pin can get inadvertently selected when a device was intended for the move.

Moving Devices Parametrically

Right-click a device in the Device Hierarchy and choose *Move – Device Parametrically* to invoke the *Move DEVICE <device>* dialog, as shown below.



The options are self-explanatory. Clicking *Apply* or pressing the *Enter* key will make the move.

The devices remain selected and additional moves can be performed by entering new values in the

dialog and clicking *Apply* or pressing the *Enter* key.

Moving Devices to Parent Center

This command is often used to align center origins for child/parent devices. A good example may be to move an interposer to be centered on a package substrate. The origins of both devices should be Center. Select the child device and right-click to choose *Move > Device to Parent Center*.

Moving Devices Logically

Occasionally, devices need to be moved up or down the Device Hierarchy to properly construct the design. Select the device in the Device Hierarch and drag it on to the new desired level of hierarchy. You may be prompted to create a contact layer when doing this. Ensure the contact layers are correct and enter the desired [contact layer options](#). Moving a device under a new parent will now include that device's nets in the parent device logic hierarchy.

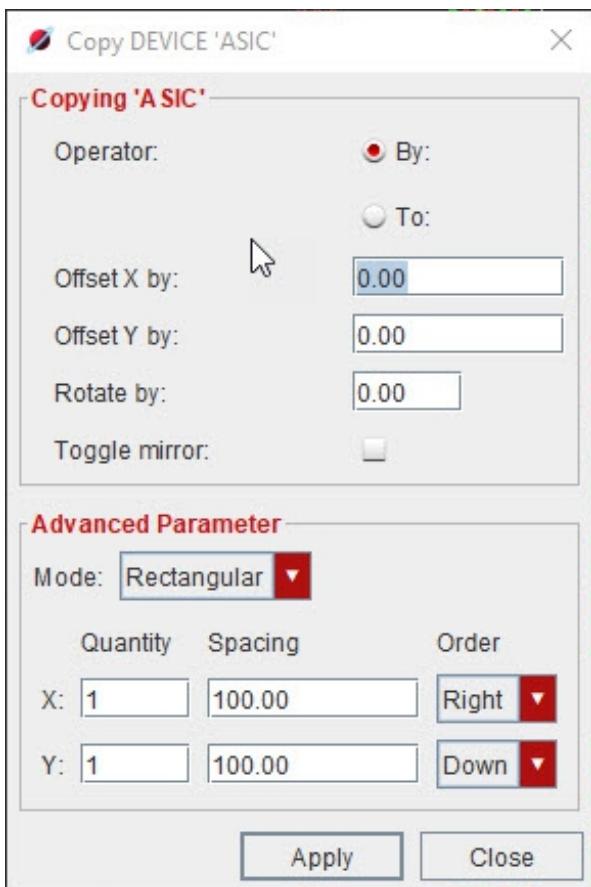
Alternately, you can select the device and right-click to choose *Move > Device to New Parent*. Select the desired parent and click *OK*.

Copying Devices

Devices can be copied in the design. Copied devices share the *Device Template* of the original, meaning that changes to one will affect the others.

Copying Devices to a Location

Right-click a device in the Device Hierarchy and choose *Copy – Device to Location* to invoke the *Copy DEVICE <device>* dialog, as shown below.

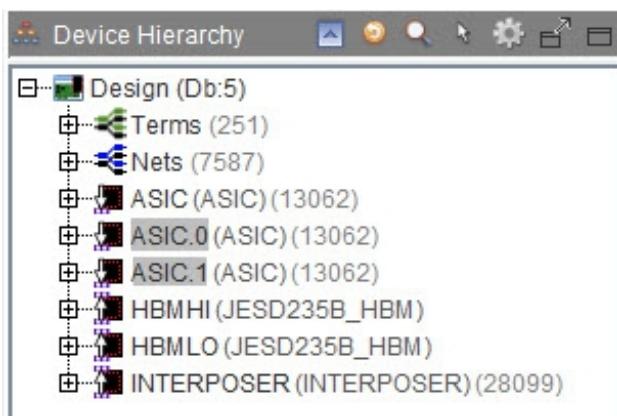


The options in the *Copying* section are self-explanatory.

The options in the *Advanced Parameter* section enable you to make a pattern of multiple copies of the device by entering the *Quantity*, *Spacing*, and *Order* values.

Clicking *Apply* or pressing the *Enter* key will copy the device.

The copies of the device will be displayed at the same level as the original in the Device Hierarchy. The copies receive default names consisting of the original name with *.0*, *.1*, ... extensions, as shown below. The names can be changed by selecting the device and entering a new name in the *Attribute Editor*.



The device(s) remain selected and additional copies can be performed by entering new values in the dialog and clicking *Apply* or pressing the *Enter* key.

Device Table Editor

The Device Table Editor is available at each level of the hierarchy in the Device Hierarchy panel by right-clicking the parent-level device or the top-level of the design and choosing *Devices Table*.

The screenshot shows the 'Device Table Editor' dialog with the following table data:

| # | Name | Template | Type | Placement | | | | |
|---|------------|---------------|---------|-------------|--------------|--------|-------|--------|
| | | | | X | Y | Rotate | Flip | Mirror |
| 1 | ASIC | ASIC | DIE | -2000.0 | 0.0 | 0.0 | true | true |
| 2 | BGA | BGA | PACKAGE | 0.0 | 0.0 | 0.0 | false | false |
| 3 | HBMHI | JESD235B_H... | DIE | 4245.999042 | 3852.75001 | 180.0 | true | false |
| 4 | HBMLO | JESD235B_H... | DIE | 4245.998512 | -3742.250015 | 180.0 | true | true |
| 5 | INTERPOSER | INTERPOSER | PACKAGE | 0.0 | 0.0 | 0.0 | false | false |

The top field enables navigation within the Device Hierarchy.

The icons enable importing and exporting of a CSV placement file; canvas zooming, highlighting, selection, changing the origin, or orientation of the desired device on the canvas; and the creation, removal, or movement of devices.

The *Placement* location and orientation can be modified by selecting the appropriate field and entering a new value. The *Enter* key must be pressed to make the change.

Importing a CSV Devices File

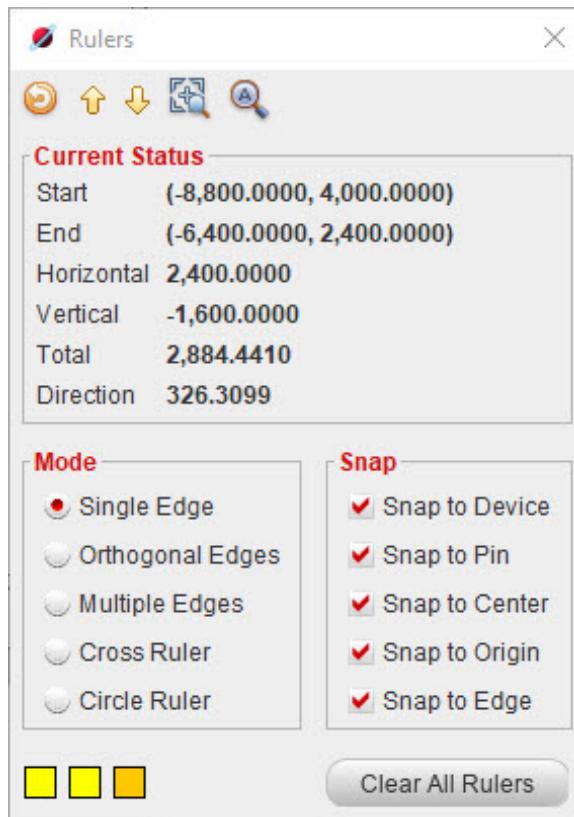
Placement can also be imported using a CSV devices placement file. Right-click any parent level of hierarchy in the Device Hierarchy, choose *Import – CSV Devices*, and select an input CSV placement file. The design placement will be adjusted per the data in the placement file.

The CSV Devices file format is shown below.

| Device | Substrate | Device_Template | X | Y | Flip | Mirror | Rotate |
|------------|------------|-----------------|-------|----------|-------|--------|--------|
| ASIC | ASIC | ASIC | 4550 | 8000 | TRUE | TRUE | 0 |
| INTERPOSER | INTERPOSER | INTERPOSER | 0 | 0 | FALSE | FALSE | 0 |
| HBMHI | HBM | JESD235B_HBM | 10944 | 11825.25 | FALSE | FALSE | 180 |
| HBMLO | HBM | JESD235B_HBM | 10944 | 4230.25 | FALSE | FALSE | 180 |

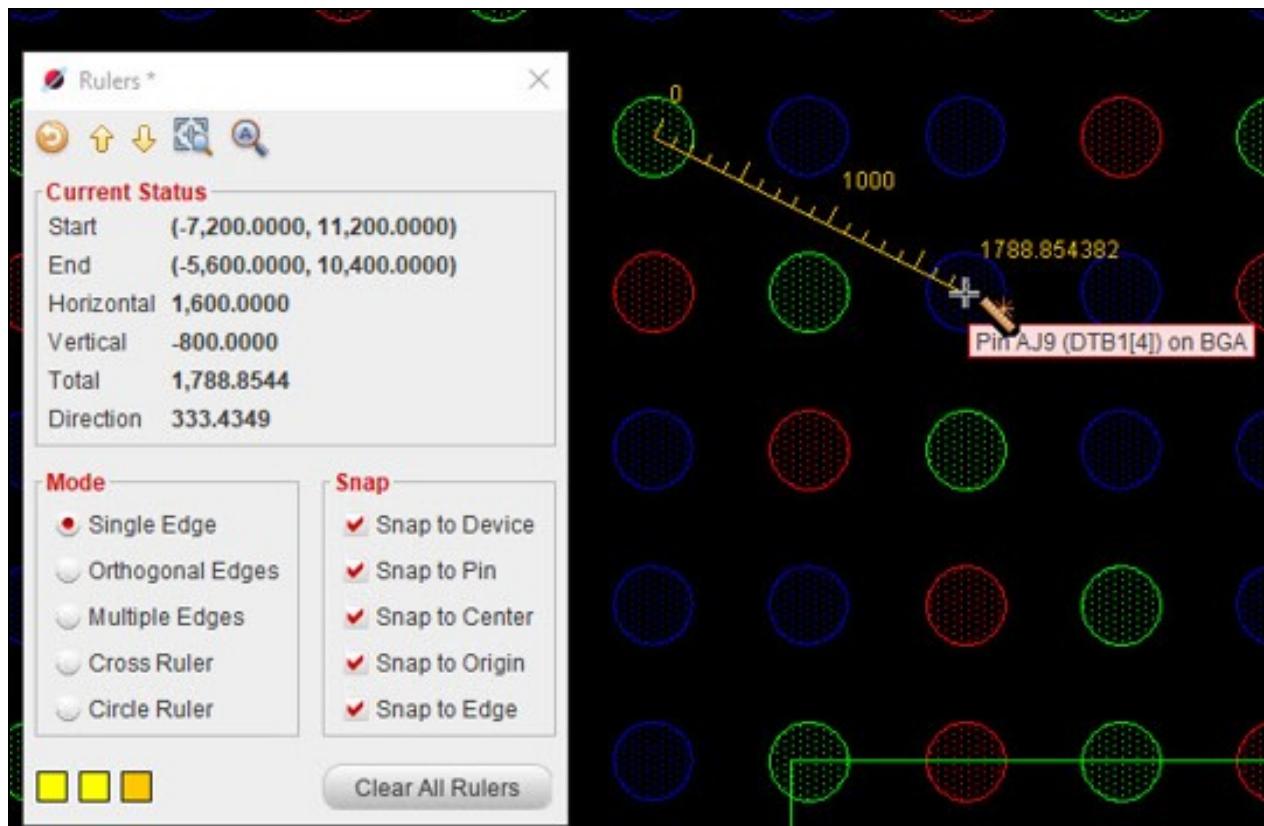
Ruler Mode to Measure Devices

The *Ruler Mode* can be used to measure and align devices exactly. Select the *Ruler Mode* icon () on the main toolbar to invoke the *Rulers* dialog, as shown below.



Rulers are created by clicking in the canvas to measure between selected objects. As they are created, the *Current Status* values are updated with the appropriate distances, as shown above.

Multiple rulers can be created and displayed on the canvas at once. They can be cycled through by using the Up and Down arrow icons, as shown below.

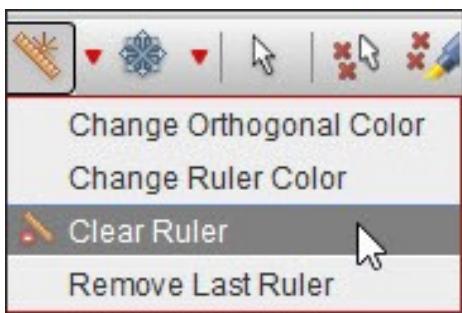


The *Mode* section provides options for the type of ruler desired.

The *Snap* section provides options for object snapping. Ensure that you are zoomed in close enough to snap to and select the desired objects. This is particularly important when snapping to pin centers.

Ruler colors can be defined by clicking on the color palette squares.

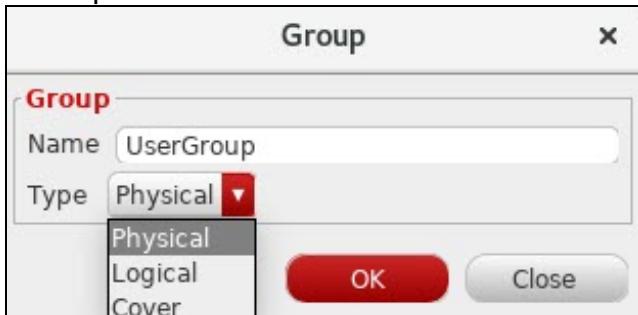
The *Clear All Rulers* button or toolbar pulldown menu command will erase all current rulers, as shown below.



You can exit from the *Ruler Mode* by selecting any other mode from the main toolbar.

Grouping Devices

Devices can be grouped to enable group manipulation such as move, copy, rotate. To group devices, select them in the Device Hierarchy and right-click to choose *Group*. In the Group dialog, enter the desired group *Name* and select the group type as *Physical*, *Logical*, or *Cover*. Selecting *Physical* will add a level of physical hierarchy in the netlist, while *Logical* will group the devices, but keep the logical hierarchy intact. Grouping devices physically may require the nets to be remapped to the parent level nets. The Netlist Editor can be used to remap the group nets.



Once grouped, the devices can be selected and manipulated as a group.

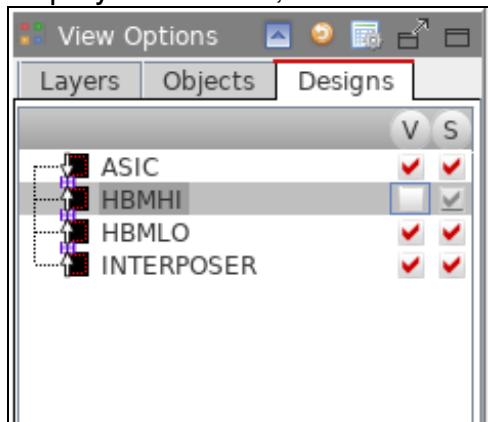
The Group command can also be used to group Bump Devices to group and manipulate pins in the design.

Ungrouping Devices

To remove a group, select it and right-click to choose *Ungroup*. The devices will be absorbed into the parent level of hierarchy.

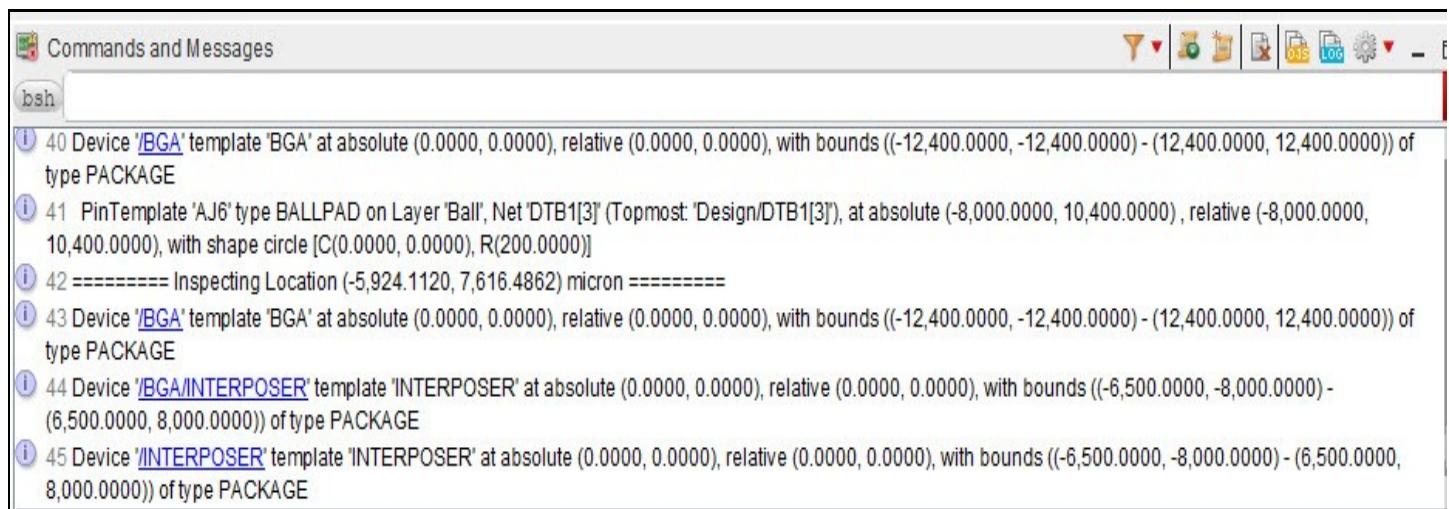
Using the Devices View to Selectively Display Devices

Devices can be filtered from display in the design by using the Designs view. Select the devices to display in the view, as shown below.



Using Inspect Mode

Information can be displayed about selected objects by using Inspect Mode. Click the *Inspect Mode* icon (🔍) in the toolbar to enter Inspect Mode. Select objects to see information about them in the Commands and Messages view, as shown below.



Defining Netlist Connectivity

System Planner supports multiple methodologies and design tools with differing netlist requirements. Connectivity can be defined using a variety of methods. The basic premise across all methodologies utilize a parent/child relationship for net mapping. Child-level nets are mapped to parent-level nets. The parent-level could either be a device such as an interposer or package with child die, or a top-level system netlist with multiple devices such as die, interposer, package, etc. The physical hierarchy defined in the Device Hierarchy dictates the logical hierarchy.

For Allegro, System Planner simply needs pads with matching net names to establish connectivity.

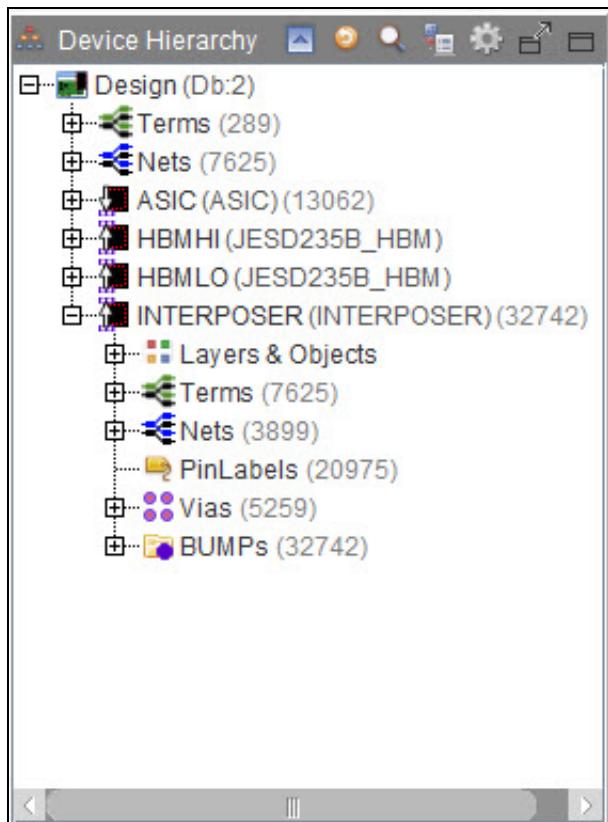
For Integrity 3D-IC using Innovus, a top-level Verilog term-based netlist is required for implementation. This netlist is also usable for Allegro implementation. Connectivity can be established by importing one or more Verilog netlists or by importing CSV term list and term map files. The Verilog format must also match the Device Hierarchy format exactly. Module and port names must be identical to the devices in the design.

Defining the Device Hierarchy

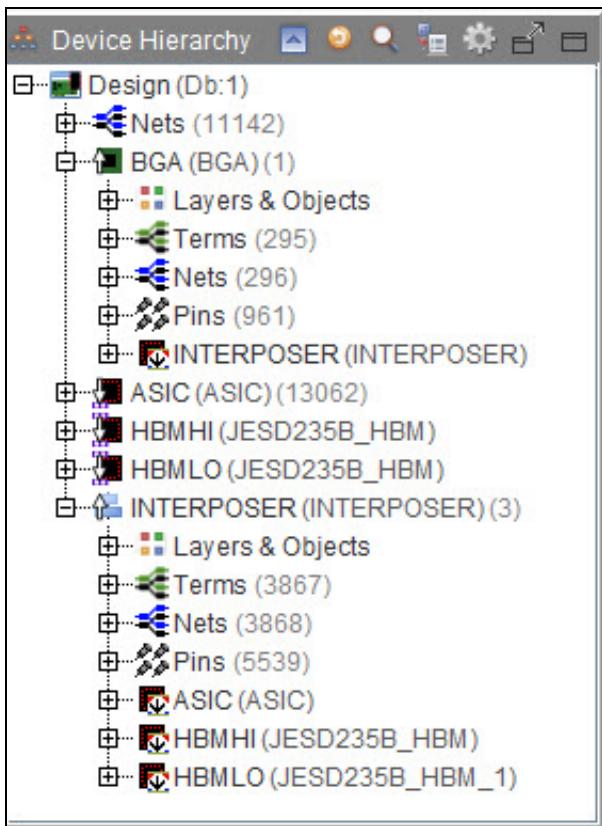
The Device Hierarchy is used to define the logic hierarchy for the design. The structure of the design in the Device Hierarchy needs to match the structure of the logical netlist. Any exported netlist files will follow the format of the Device Hierarchy.

For Integrity 3D-IC and Allegro

The Cadence recommended design structure is to have all devices flat at the top level of the design, as shown below. Each unique device should have a unique substrate defined. A top-level netlist is created to map connectivity across all of the devices. The top-level netlist does not get implemented. This structure is a requirement for Integrity 3DIC and it offers additional flexibility when using Allegro. However, there are a number of Allegro design methodologies such as InFo or wire bond designs that require traditional parent-child hierarchy.



Allegro requires that contact devices be created under the parent design, as shown below. These are used to create the symbol for the parent Allegro design, as shown below.



Contact device connectivity also needs to be defined and managed along with the source device. Connectivity is passed to the contact device as it gets created. It should stay in sync with the source device. However, there are circumstances where the contact device may need to be refreshed. Attention should be paid during design updates to ensure the contact device is in sync. Refer to [Running Inter-Substrate Connectivity Checks](#) to ensure the net mapping is in sync. Refer to [Creating Contact Layers, Contact Pins, and Contact Devices](#) on how to refresh contact devices.

For Certain Allegro Methodologies

Allegro designs require a parent and child structure where the parent is the device being implemented and each of the components in the design are child devices. In most cases, the child devices will be contact devices created from top-level devices, as shown above.

Alternately, devices can be moved directly under the parent device, as shown on the left below. This structure is typically used for single substrate designs only such as wire bond or Info. A single substrate design has devices created directly onto the specific layers of the design substrate, as shown below. The ASIC and HBM pads are on the AP layer, not in a substrate of their own. This type, of design is typically only done using standalone System Planner, as Integrity 3DIC requires top-level devices. Most designs imported from Allegro will be built with this type of design hierarchy.



Child nets are mapped or propagated to parent device nets to produce the connectivity. Net mapping can be done interactively or with a simple parent child CSV mapping file.

Designs imported from Allegro will typically be created using this traditional style hierarchy when the default import option to infer hierarchy is used.

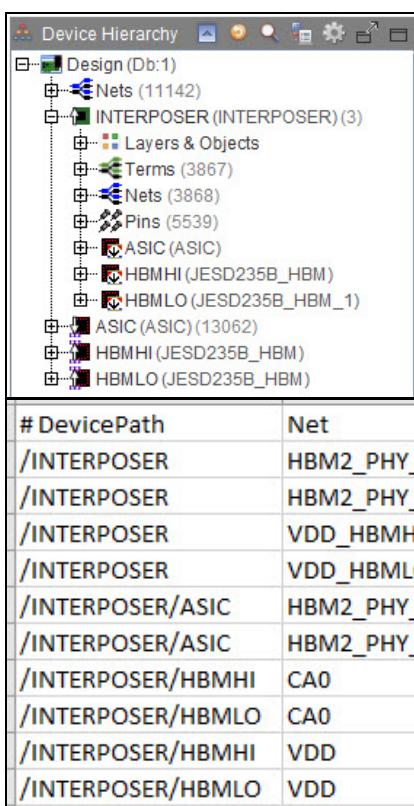
Cadence recommends using top-level devices with contact devices for multi-substrate designs in order to define proper pad configuration between the two substrates.

Defining Parent Child Connectivity for Allegro X Layout Editors

When preparing designs for Allegro X implementation, a simple parent/child mapping approach is used. Pin names on the source child die devices are mapped to parent device or to the top-level system netlist. Nets can be mapped interactively in the Netlist Editor or by importing a CSV net mapping file.

Importing a CSV Net Mapping File

Connectivity can be defined by creating and importing a CSV format net mapping file(s). Net mapping can be defined using a simple child to parent format, as shown below.



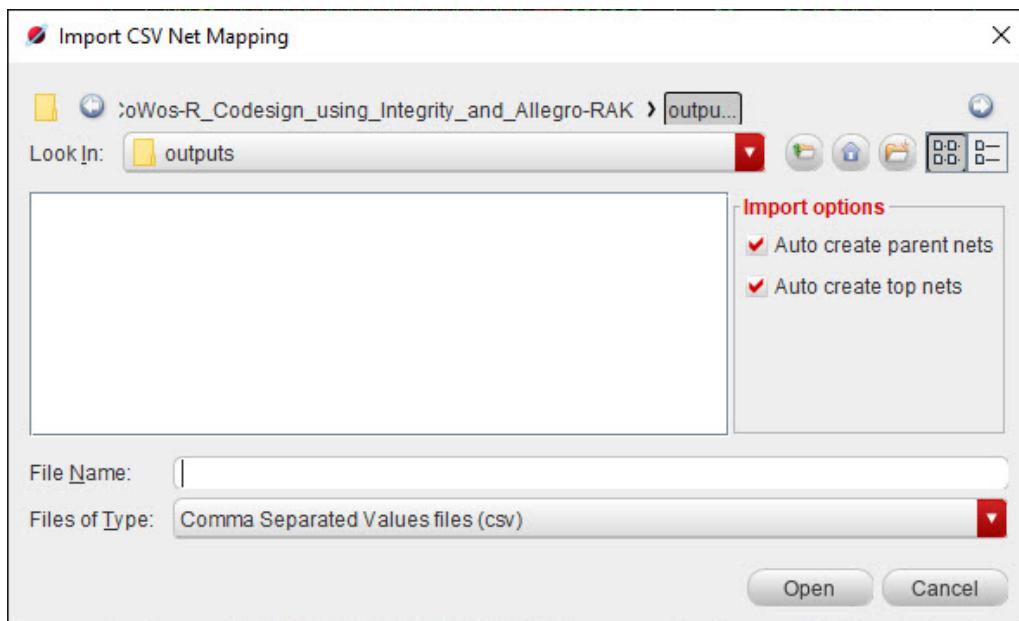
| #DevicePath | Net | ParentNet |
|-------------|-------------------|-------------------|
| /INTERPOSER | HBM2_PHY_HI_CA[0] | HBM2_PHY_HI_CA[0] |
| /INTERPOSER | HBM2_PHY_LO_CA[0] | HBM2_PHY_LO_CA[0] |
| /INTERPOSER | VDD_HBMHI | VDD_HBMHI |
| /INTERPOSER | VDD_HBMLO | VDD_HBMLO |
| /ASIC | HBM2_PHY_HI_CA[0] | HBM2_PHY_HI_CA[0] |
| /ASIC | HBM2_PHY_LO_CA[0] | HBM2_PHY_LO_CA[0] |
| /HBMHI | CA0 | HBM2_PHY_HI_CA[0] |
| /HBMLO | CA0 | HBM2_PHY_LO_CA[0] |
| /HBMHI | VDD | VDD_HBMHI |
| /HBMLO | VDD | VDD_HBMLO |

| #DevicePath | Net | ParentNet |
|-------------------|-------------------|-------------------|
| /INTERPOSER | HBM2_PHY_HI_CA[0] | HBM2_PHY_HI_CA[0] |
| /INTERPOSER | HBM2_PHY_LO_CA[0] | HBM2_PHY_LO_CA[0] |
| /INTERPOSER | VDD_HBMHI | VDD_HBMHI |
| /INTERPOSER | VDD_HBMLO | VDD_HBMLO |
| /INTERPOSER/ASIC | HBM2_PHY_HI_CA[0] | HBM2_PHY_HI_CA[0] |
| /INTERPOSER/ASIC | HBM2_PHY_LO_CA[0] | HBM2_PHY_LO_CA[0] |
| /INTERPOSER/HBMHI | CA0 | HBM2_PHY_HI_CA[0] |
| /INTERPOSER/HBMLO | CA0 | HBM2_PHY_LO_CA[0] |
| /INTERPOSER/HBMHI | VDD | VDD_HBMHI |
| /INTERPOSER/HBMLO | VDD | VDD_HBMLO |

The entries in the *#DevicePath* column can be structured to map either the top-level source die (middle image above) or the contact devices (right image above), or both. If mapping top-level source devices, net mapping should be done prior to creating the contact devices. If mapping to contact devices, net mapping should be done after contact devices are created.

To import a CSV net mapping file select the top level of the project in the Device Hierarchy and right-click to choose *Import > CSV Net Mapping*. The Import CSV Net Mapping dialog is displayed, as shown below.

NOTE: CSV net mapping files can only be imported and exported by selecting the top level of the design project in the Device Hierarchy.



Browse to select the CSV net mapping file and click *Open* to import the file.

The *Auto create parent nets* and *Auto create top nets* options will create default net names for levels of hierarchy not specified in the net mapping file. These options are rarely needed, as most Cadence recommended methodologies only use one or two levels of hierarchy.

Exporting a CSV Mapping File

It is a good practice to export a CSV net mapping file if net connectivity has been interactively modified in the System Planner in order to keep the source net mapping file and the design synchronized. To export a CSV net mapping file, select the top level of the project in the Device Hierarchy and right-click to choose *Export > CSV Net Mapping*.

Interactively Mapping Nets

Net connectivity can be defined interactively using either the Netlist Editor or the Net Manager.

Related Information

- [Defining and Exploring Connectivity with the Netlist Editor](#)

Defining Term-based Connectivity for Innovus or Allegro X Layout Editors

If a top-level Verilog netlist is not available, connectivity can be defined by preparing and importing term list and term map files. Terms can be defined by importing term list files. The terms are then mapped top-level nets using the term map files.

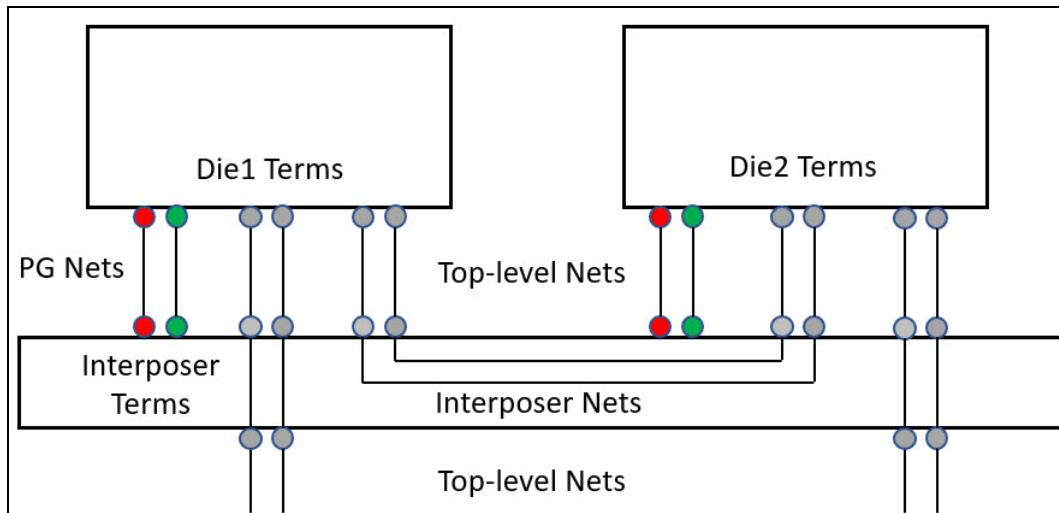
Terms are created by importing a term list CSV file for each device. Each term name in the design must be unique. Net and term prefix names can be used to make the names unique or to better identify the nets at the top-level.

This process is described and used in the *2.5D Silicon Interposer Design with the Integrity 3D-IC Platform RAK*.

You can also create terms easily for most devices by using the *Netlist Editor*. See [Defining and Exploring Connectivity with the Netlist Editor](#).

Understanding the Design Requirements

For Integrity 3D-IC, the Device Hierarchy needs to be structured such that each device is instantiated at the top-level of the design. Terms must be defined for the device being designed, such as the Interposer in the image below. All die and interposer terms are then mapped to top-level nets.



The top-level netlist needs to contain terms for each top-level external net, such as those connecting to the package substrate, as shown at the bottom in the above image. The top-level netlist is represented with the largest rectangle shown above encompassing the entire design. Each device

term requires mapping to top-level nets.

To facilitate easy selection of these external signals for C4 bump generation, they should be named with a unique net name prefix (for example, C4_).

Once the terms are defined in the logical netlist, they must then be mapped to physical pads/bumps in the design. This is done automatically as the pads are created using various commands or the *assign_terms* Tcl command can be used to assign remaining floating terms.

Preparing Term List Files

Term list files are used to create terms on the device and to map them to device-level nets. This is how die-to-die connectivity is defined by mapping a die terms to a different die net names. You can use these term list files to map differing die net names to a single net name in the device-level netlist. This is the method used to connect die with differing net names. If the net names you wish to connect differ between dies, you must first decide which net name you would like to use in the netlist.

It is often easier to prepare a separate term list file for each die in the design. A separate term list is also typically used to define the bottom side "C4" bumps terms on the device.

To prepare a starter term list file, right-click the device in the Device Hierarchy and select the *Export – Term List* command.

Examples of a term list files are shown below that map the same signals for two devices to derive connectivity between them

| Term | Net | Type | Use |
|--------------|--------------------|-------|--------|
| HBMHI_ARFUH2 | HBMHI_ARFUH2 | INOUT | SIGNAL |
| HBMHI_ARFUH3 | HBM2_PHY_HI_ARFUH3 | INOUT | SIGNAL |
| HBMHI_CA0 | HBM2_PHY_HI_CA[0] | INOUT | SIGNAL |
| HBMHI_CA1 | HBM2_PHY_HI_CA[1] | INOUT | SIGNAL |
| HBMHI_CA2 | HBM2_PHY_HI_CA[2] | INOUT | SIGNAL |

| Term | Net | Type | Use |
|-------------------------|--------------------|-------|--------|
| ASIC_HBM2_PHY_HI_ARFUH1 | HBM2_PHY_HI_ARFUH1 | INOUT | SIGNAL |
| ASIC_HBM2_PHY_HI_ARFUH3 | HBM2_PHY_HI_ARFUH3 | INOUT | SIGNAL |
| ASIC_HBM2_PHY_HI_CA[0] | HBM2_PHY_HI_CA[0] | INOUT | SIGNAL |
| ASIC_HBM2_PHY_HI_CA[1] | HBM2_PHY_HI_CA[1] | INOUT | SIGNAL |
| ASIC_HBM2_PHY_HI_CA[2] | HBM2_PHY_HI_CA[2] | INOUT | SIGNAL |

Here:

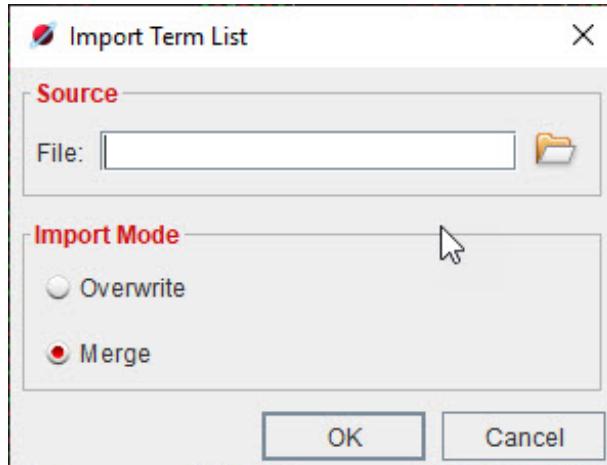
- The *Term* column defines the name of the term to be defined for the device. Unique term names are required for each term. In this example, the *HBMHI_* prefix was added to the term names and the device net names to uniquify the HBM.
- The *Net* column defines the device-level net with which to connect the term. Notice in the example above, the bottom three lines map the HBM CA0-3 terms to the ASIC net names. This is how connectivity is established between die with different net and term names.
- The *Type* column defines the net and term type. The available values are IN, OUT, INOUT, and FEEDTHRU.
- The *Use* column defines the type of term. The values include ANALOG, CLOCK, GROUND, POWER, RESET, SCAN, SIGNAL, TIEOFF, NC, and UNKNOWN. Typically, SIGNAL, POWER, and GROUND are the only values used for 3D-IC designs.

Importing Term List Files

Term list files are imported relative to the device for which the terms are being created.

To import a term list file for a device:

1. Right-click the device in the Device Hierarchy and choose *Import – Term List* to invoke the Import Term List dialog, as shown below.



2. Select one or more term list files for the device in *File*.
3. In the *Import Mode* section, specify whether you want to *Merge* with or *Overwrite* existing terms.
4. Click *OK* to import the term list files.

Review the messages displayed in the *Command and Messages* view to ensure a proper import.

Exporting a Term List File

Term list files can be exported for several purposes. During design creation, they can be used as a starter template to create the term map files. They can also be used to produce current term list files in case changes were made interactively in the design. They can also be used to edit and change connectivity.

To export a term list file:

1. Right-click the device desired for term export and choose *Export – Term List* to invoke the Export Term List dialog.
2. Browse to define *Save In* directory and *File Name*.

Preparing Term Map Files

A term map file maps the device terms to top-level nets. All device terms are propagated to the top-level netlist as nets. Importing the term map file will automatically create the top-level nets. The top-level netlist is not implemented, it facilitates connections between top-level devices in the Device Hierarchy. In the example images, connections between the INTERPOSER, ASIC, and HBMHI are contained in the top-level netlist. Only the nets inside the INTERPOSER device are implemented. Interposer die-to-die connections are defined in the Interposer term list files.

It is often easier to create a separate term map file for each device in the design. Die specific term map files should contain mapping for both the die terms and the connected device terms "interposer", unless it is another die with its own map file.

The example term map files below show a few lines from an ASIC and HBM map files mapping their connections to the Interposer. These nets match the examples shown in [Preparing Term List Files](#).

| Child Device | Child Term | Parent Net |
|--------------|-------------------------|-------------------------|
| ASIC | HBM2_PHY_HI_ARFUH1 | ASIC_HBM2_PHY_HI_ARFUH1 |
| ASIC | HBM2_PHY_HI_ARFUH3 | ASIC_HBM2_PHY_HI_ARFUH3 |
| ASIC | HBM2_PHY_HI_CA[0] | ASIC_HBM2_PHY_HI_CA[0] |
| ASIC | HBM2_PHY_HI_CA[1] | ASIC_HBM2_PHY_HI_CA[1] |
| ASIC | HBM2_PHY_HI_CA[2] | ASIC_HBM2_PHY_HI_CA[2] |
| INTERPOSER | ASIC_HBM2_PHY_HI_ARFUH1 | ASIC_HBM2_PHY_HI_ARFUH1 |
| INTERPOSER | ASIC_HBM2_PHY_HI_ARFUH3 | ASIC_HBM2_PHY_HI_ARFUH3 |
| INTERPOSER | ASIC_HBM2_PHY_HI_CA[0] | ASIC_HBM2_PHY_HI_CA[0] |
| INTERPOSER | ASIC_HBM2_PHY_HI_CA[1] | ASIC_HBM2_PHY_HI_CA[1] |
| INTERPOSER | ASIC_HBM2_PHY_HI_CA[2] | ASIC_HBM2_PHY_HI_CA[2] |

| Child Device | Child Term | Parent Net |
|--------------|------------|--------------|
| HBMHI | ARFUH1 | HBMHI_ARFUH1 |
| HBMHI | ARFUH3 | HBMHI_ARFUH3 |
| HBMHI | CA0 | HBMHI_CA0 |
| HBMHI | CA1 | HBMHI_CA1 |
| HBMHI | CA2 | HBMHI_CA2 |
| INTERPOSER | HBMHI_CA0 | HBMHI_CA0 |
| INTERPOSER | HBMHI_CA1 | HBMHI_CA1 |
| INTERPOSER | HBMHI_CA2 | HBMHI_CA2 |

The columns in the term map file include:

- *Child Device* defines the child device from which the term is being mapped.
- *Child Term* defines the term name in the child device to map.
- *Parent Net* defines the top-level net to which the child term is to be mapped.

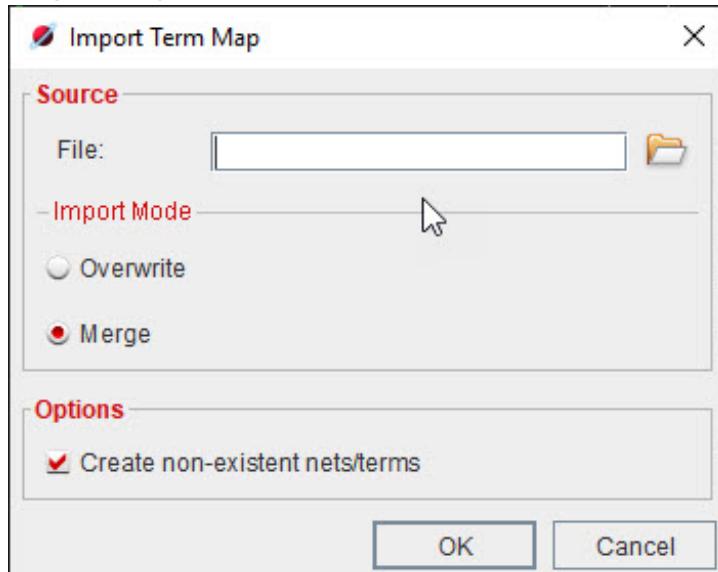
Importing Term Map Files

Term map files are imported at the top-level of the Design Hierarchy since the terms are being mapped to the top-level net names.

Ensure that you import term map files with the top-level of the design hierarchy.

To import a term-map file:

1. Right-click the top-level *Design* and choose *Import – Term Map* to invoke the Import Term Map dialog, as shown below.



2. Select one or more term map files for the design in *File*.
3. In the *Import Mode* section, specify whether you want to merge with or overwrite existing mapping.
4. Select *Create non-existent nets/terms* to create terms for objects in the map file that do not exist in the design.
5. Click *OK* to import the term map files.

Review the messages echoed to the *Command and Messages* panel to ensure a proper import.

Exporting a Term Map File

Term map files can be exported for several purposes. During design creation, they can be used as a starter template to create the term map files. They can also be used to produce current term map files in case changes were made interactively in the design. They can also be used to edit and change connectivity.

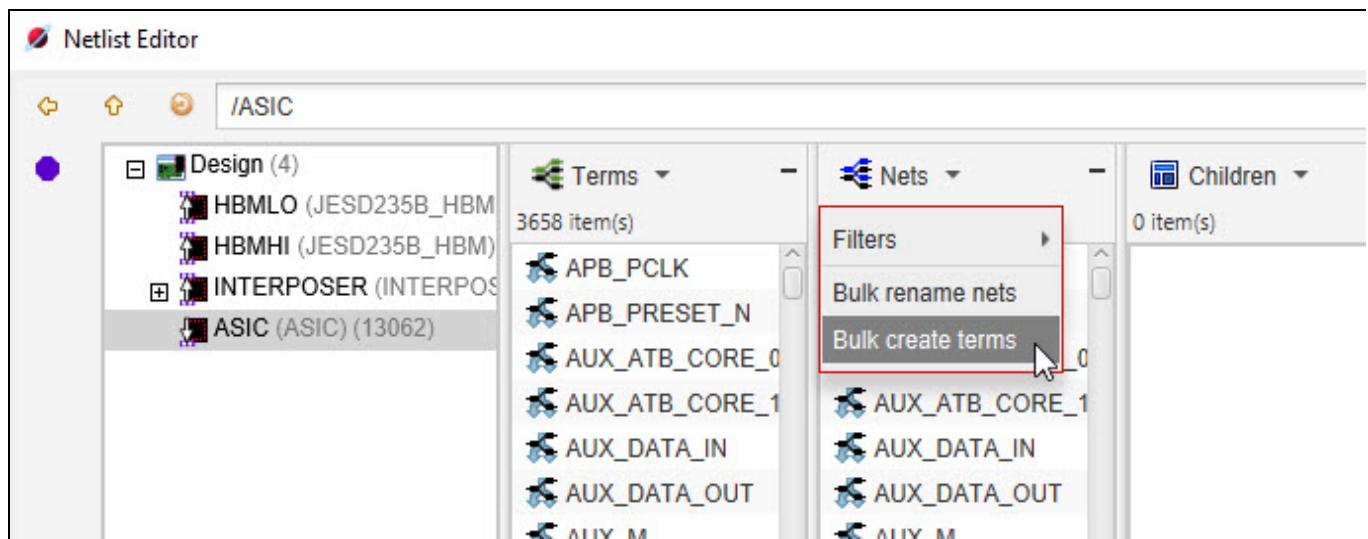
To export a term-map file:

1. Right-click the top-level *Design* and choose *Export – Term Map*.
2. Browse to define *Save In* directory and *File Name*.

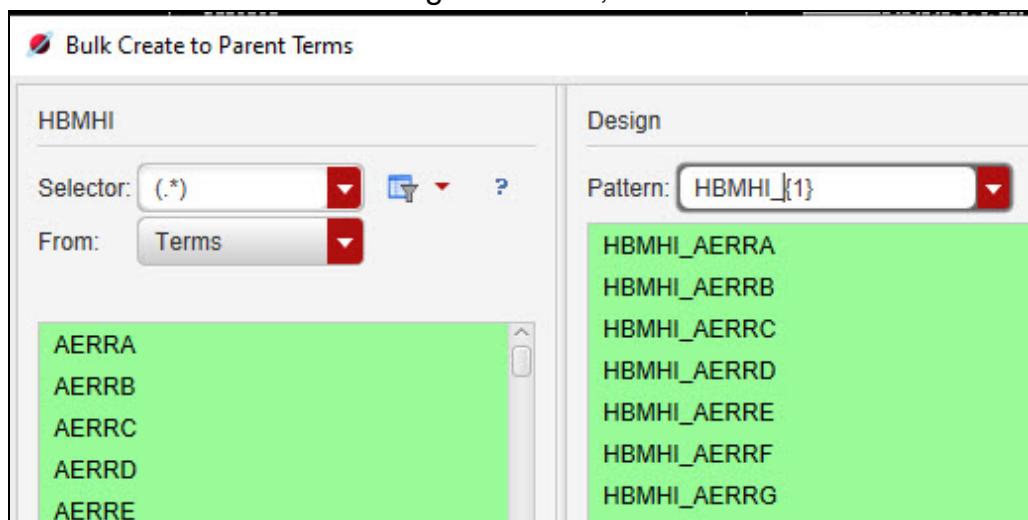
Creating Terms Using the Netlist Editor

Terms can also be created easily for most devices automatically using the Netlist Editor:

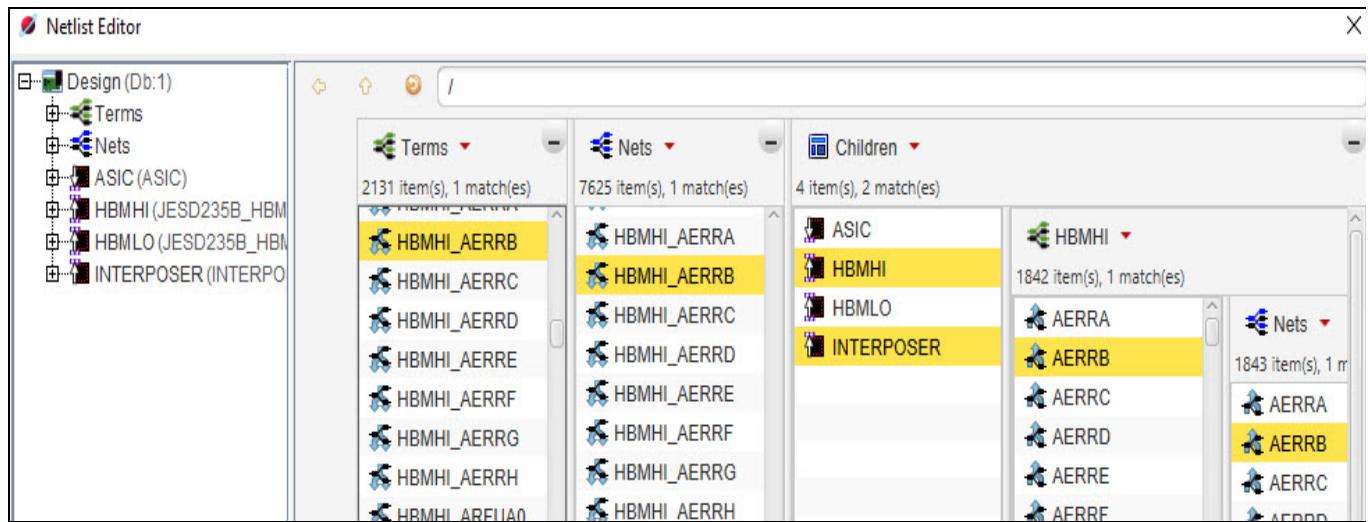
1. Right-click the device in the Device Hierarchy and choose *Netlist* to invoke the Netlist Editor, as shown below.



2. You can display the netlist and terms at any level of the Device Hierarchy in the Netlist Editor by clicking on the desired level in the left-side device browser, as shown above with the ASIC netlist displayed. The nets and terms for the selected level are displayed, and pull-down menu commands are provided to create terms and nets and to map nets interactively.
3. To create terms, choose *Bulk create terms* from the *Nets* pull-down menu, as shown above. The Bulk Create Terms dialog is invoked, as shown below.



4. Nets can be selected using regular expressions for term creation. Enter a term-name prefix, such as *HBMHI_*, as shown above.
5. Click *OK* to create the terms, as shown below.



Note that the terms were created with the designated *HBMHI_* name prefix.

Importing Verilog Netlists

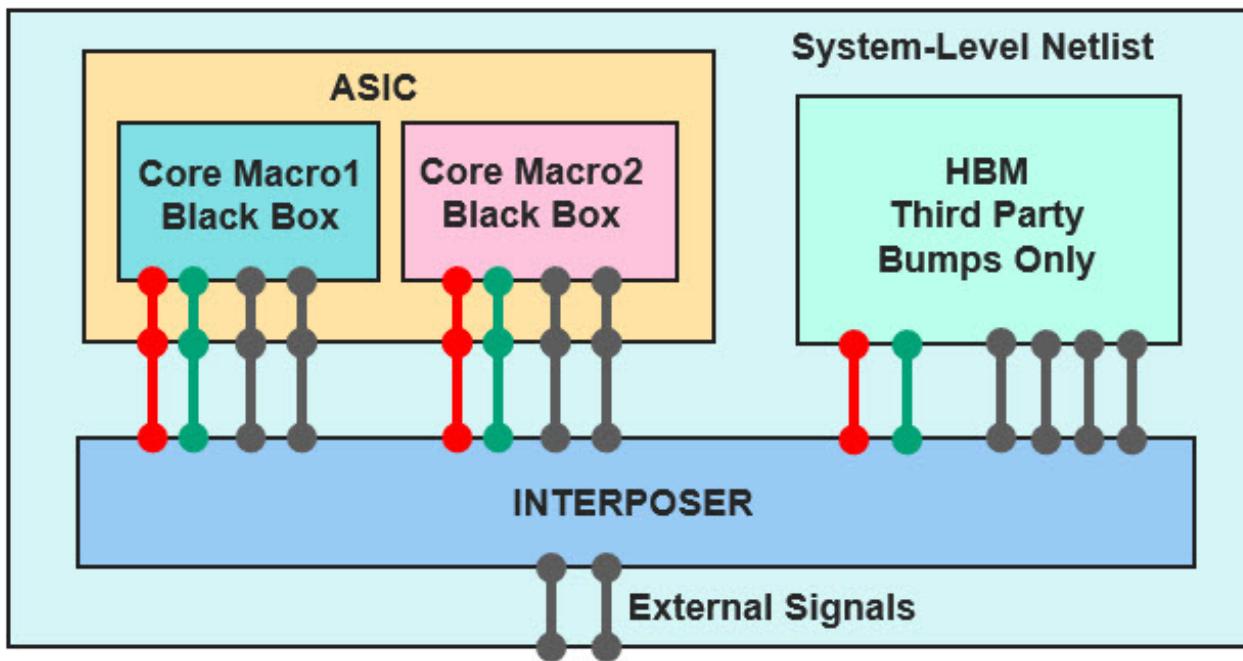
Verilog netlist can be imported to define system level connectivity in the design

Working at an Abstract Level using Black Box Die Macros

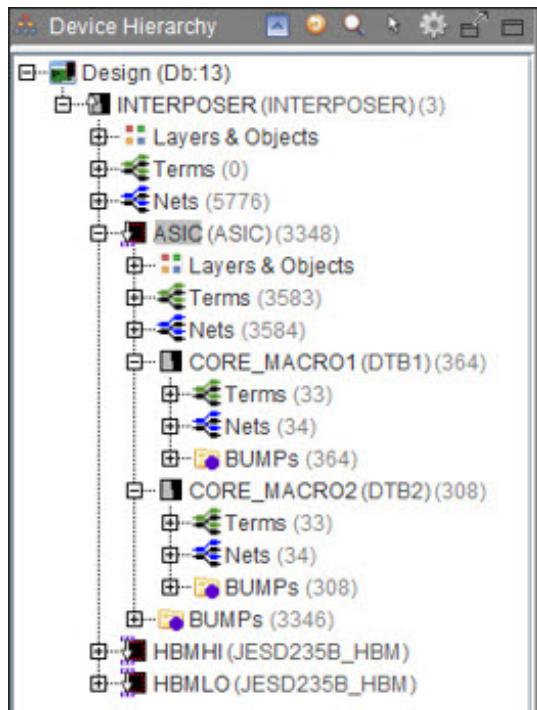
The imported Verilog netlists for each die should only contain the structural part of the design, including external signals and macro-level connections.

System Planner has no use for internal macro connectivity. Including internal macro connectivity in the imported Verilog netlists may result in overly large project files and reduced tool performance.

System Planner is best used for 3D-IC design assembly with a structural netlist that contains black boxes for internal die core macros, as shown below. This reduces the database size, improves performance, and also enables die designers to work in parallel with the top-level design planning efforts. This also allows for optimization of die bumps based on die-to-die or external connections. The die core macro structure will match the Verilog netlist structure exactly.



Work with Structural Netlist Only



Importing a Verilog Netlist

As Verilog files are imported, System Planner will populate existing devices with matching names with the net connectivity from the Verilog module. If devices do not exist, System Planner will create them automatically.

The recommended sequence is to create devices in System Planner before importing the Verilog netlists.

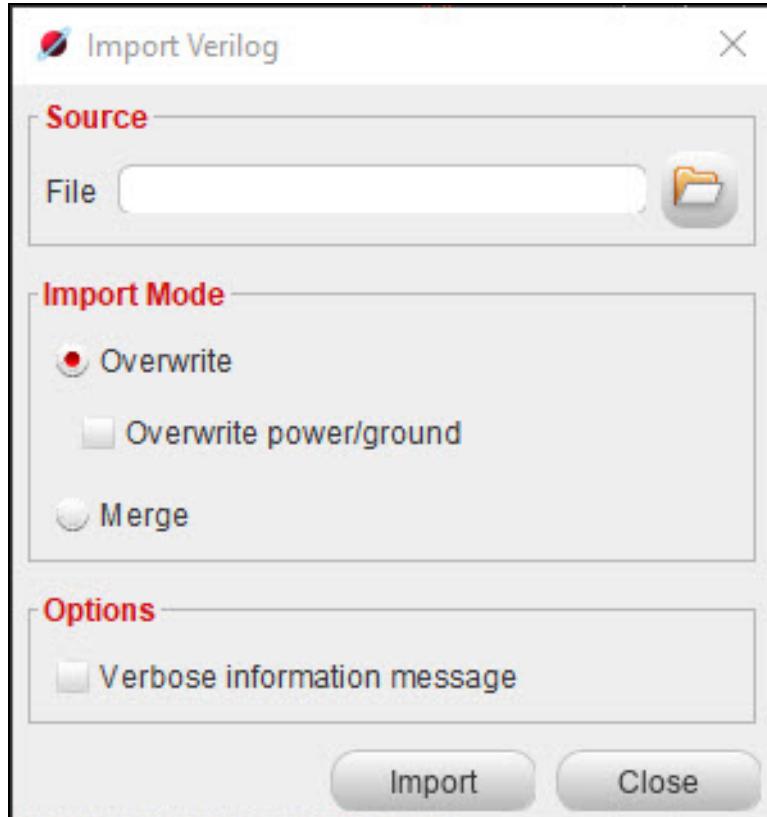
The module and port names in the Verilog netlist being imported must exactly match the device names and pin/bump names in the System Planner design.

Connectivity can be established by importing one or more Verilog netlists.

If macros exist in the device Verilog files, they are automatically created under the COREs directory in the Device Hierarchy.

To import a top-level or device-level Verilog netlist:

1. Right-click the project top-level and choose *Import – Verilog*.
The *Import Verilog* dialog is displayed.



2. Specify the Verilog file to be imported in the *File* field.

3. In the *Import Mode* section, select *Merge* to merge the imported netlist with the existing netlist in the project. Alternatively, you can select *Overwrite* to replace the existing netlist and also overwrite the power/ground nets in the project.
4. Select *Verbose information message* to include additional message details.
5. Click *Import*.
6. Check the *Commands and Messages* window for any errors or warnings.

Nets and terms will created in the design at the appropriate Design Hierarchy levels.

Defining and Exploring Connectivity with the Netlist Editor

The Netlist Editor enables display, map, and creation of nets and terms. The netlist can be displayed or modified at each level of the Device Hierarchy. The Editor provides various methods to propagate and map nets up and down the hierarchy. The nets and term displayed can be filtered based on various criteria. Regular expression can be used to selectively create or map nets and terms.

For Innovus implementation, interactive mapping requires the parent nets be mapped to device terms. The Netlist Editor displays terms and has a lot of term-based functionality.

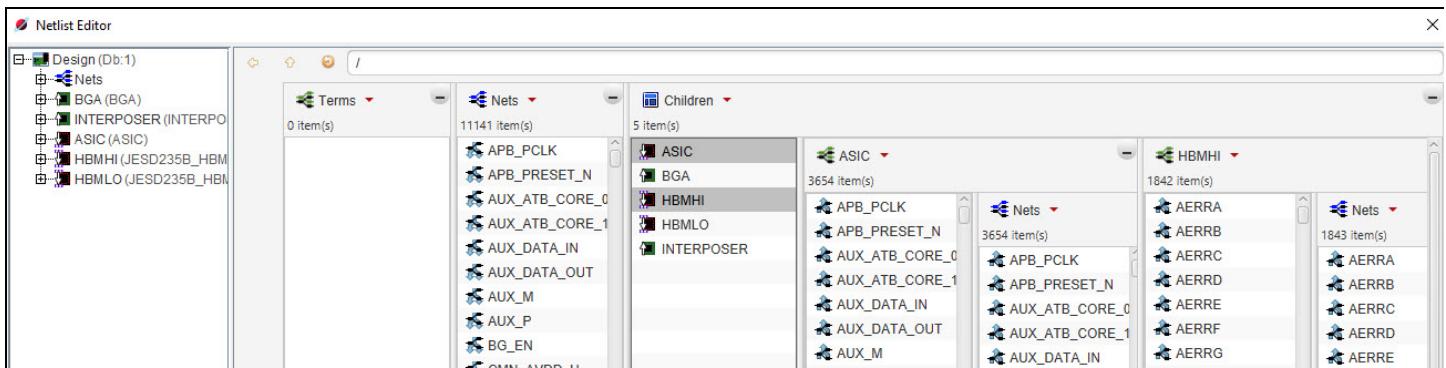
For Allegro, simple net-to-net mapping is all that is required. Allegro users are more inclined to interactively map nets. therefore the images below describe net-to-net mapping. For Innovus, use the map terms commands instead of the map nets commands.

Using the Netlist Editor

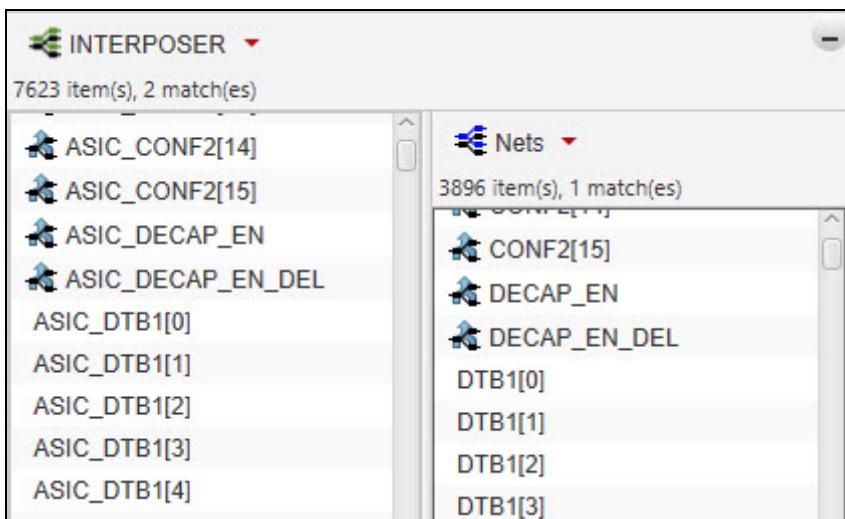
Invoke the Netlist Editor at any device or hierarchy level by right-clicking it and selecting the *Netlist* command.

The Design Hierarchy is shown on the left. The netlist for any level of the design can be displayed by double-clicking it here. The Netlist Editor displays the *Nets*, *Terms*, and *Children* for the level of device hierarchy selected.

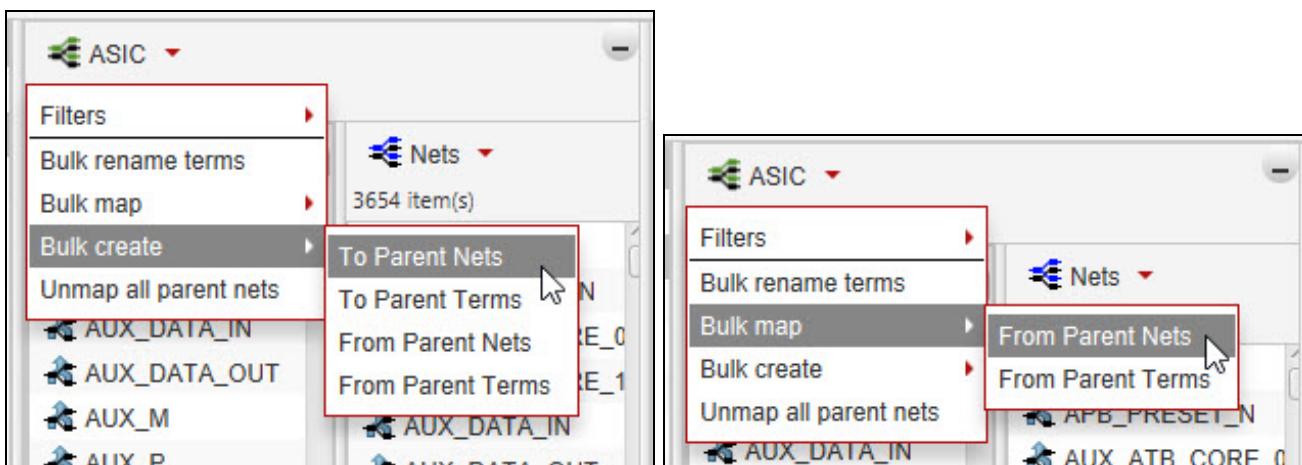
Double-click on an object in the *Children* list to display child devices, as shown below. This typically involves stretching the Netlist Editor window to view more devices. Devices can be collapsed by clicking the minimize icon in upper-right corner of the device column (-).



Nets that are mapped receive the blue icon to the left of the name. Unmapped nets and terms do not, as shown below.

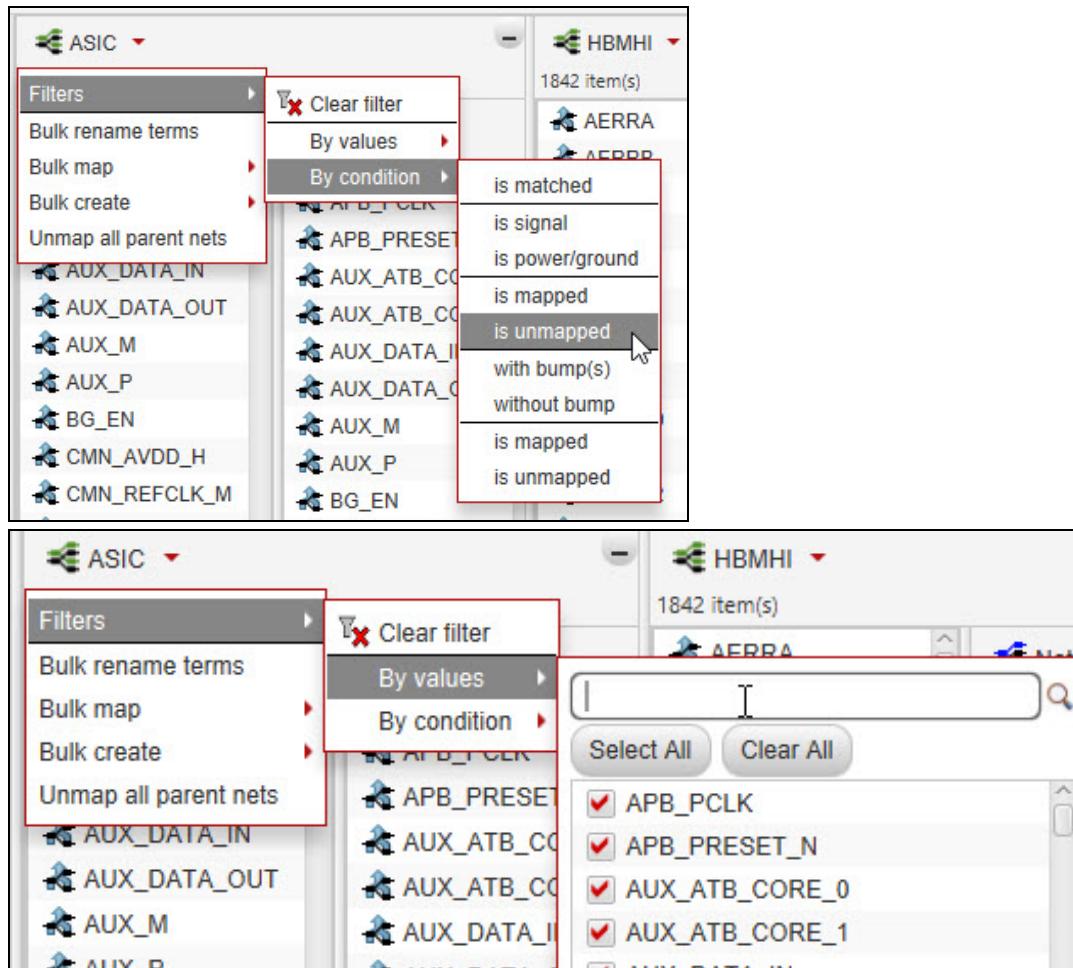


Bulk net creation and mapping commands can be performed for each device by using the device pulldown menu, as shown below.

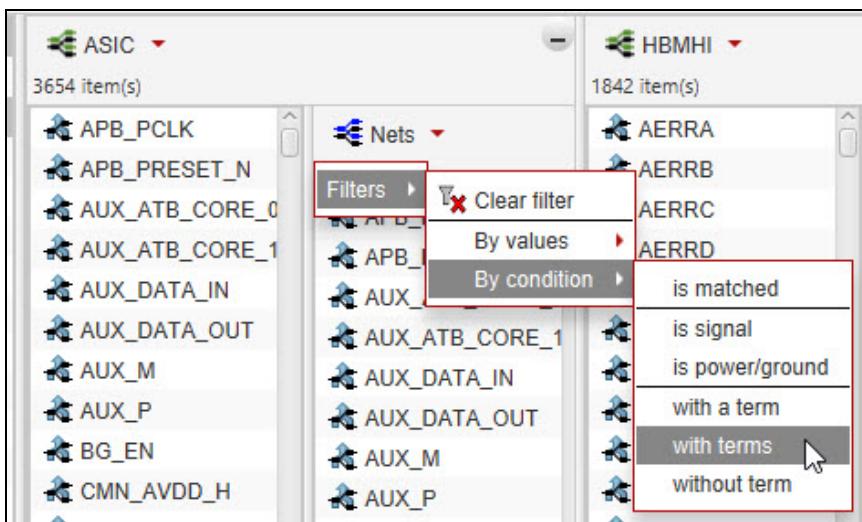


Using Filters to Customize the Nets and Terms Displayed

Display filters can be configured for each device by using the pulldown menu, as shown below.



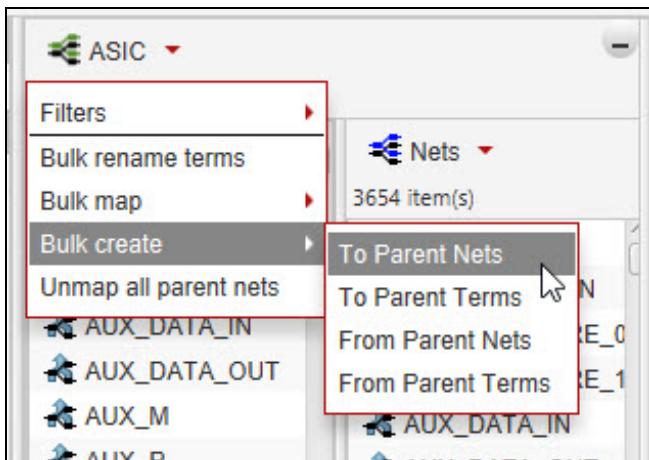
Additional types of filtering are available using the Nets pulldown menu, as shown below.



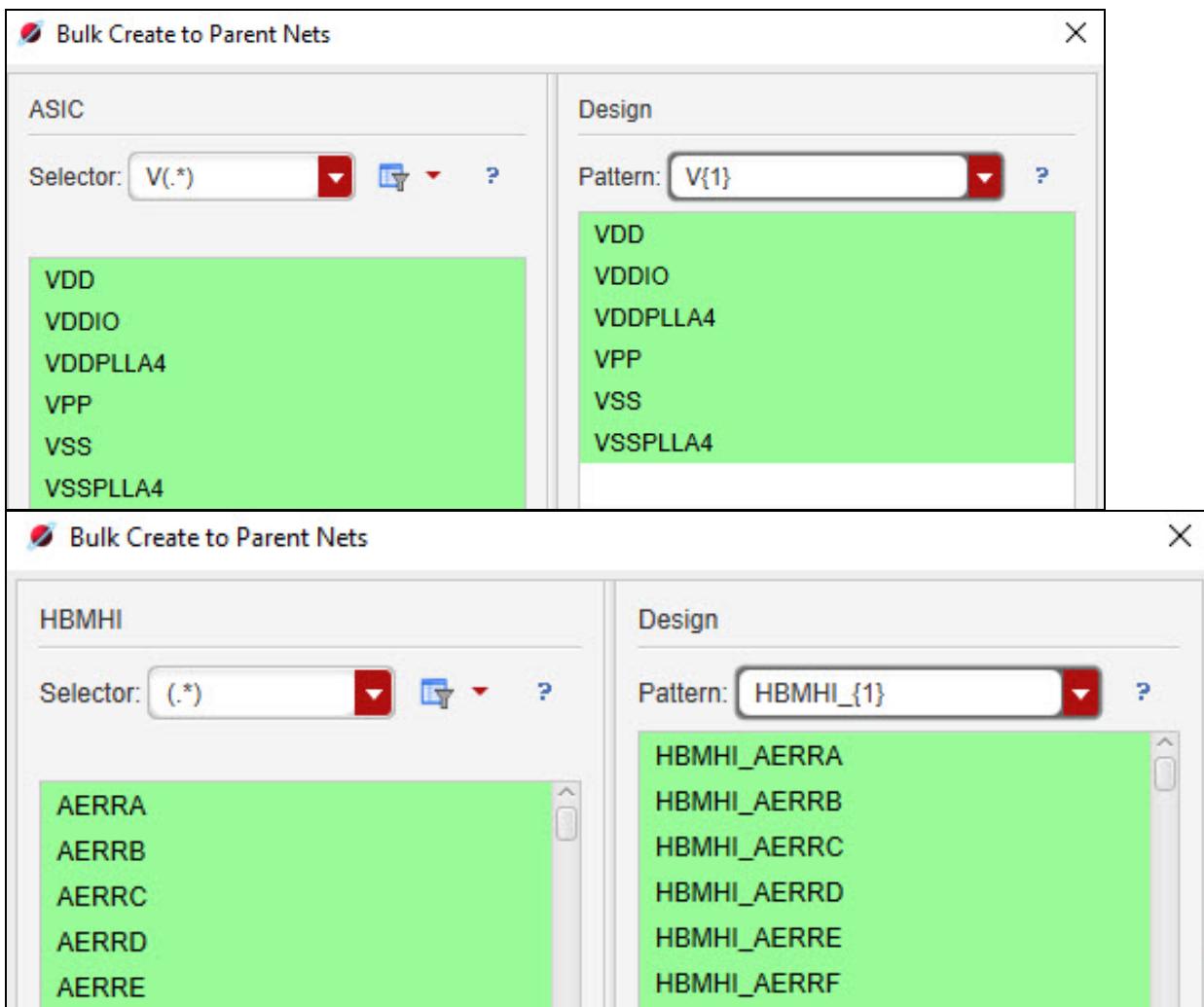
Use *Clear filter* to remove all filters and display all nets and terms.

Bulk Creation of Parent or Child Nets

Nets can be pushed up and down the hierarchy by using the Bulk Create To/From Parent/Child commands under the device pulldown menu. The commands are self explanatory in how they operate.



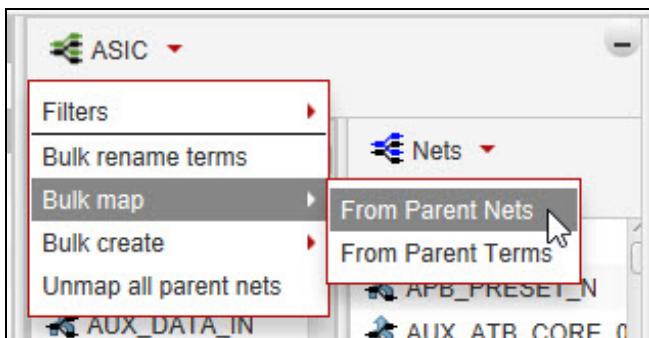
Nets can be created selectively by using the regular expression fields. Net prefixes and name configurations can be added, as shown below.



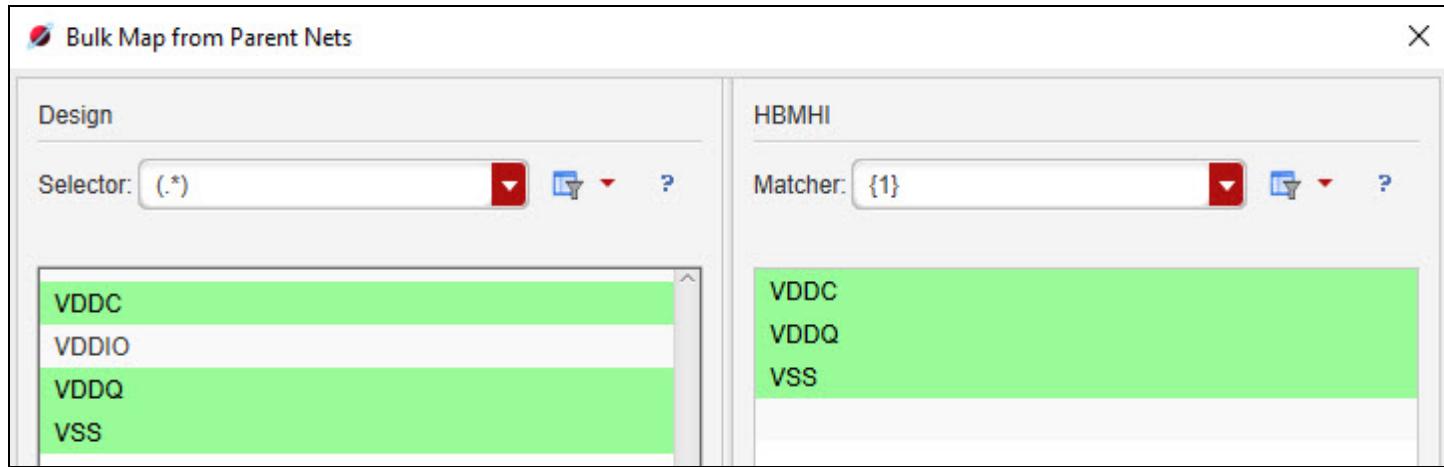
Clicking *Create* will create the desired nets.

Bulk Mapping Nets or Terms

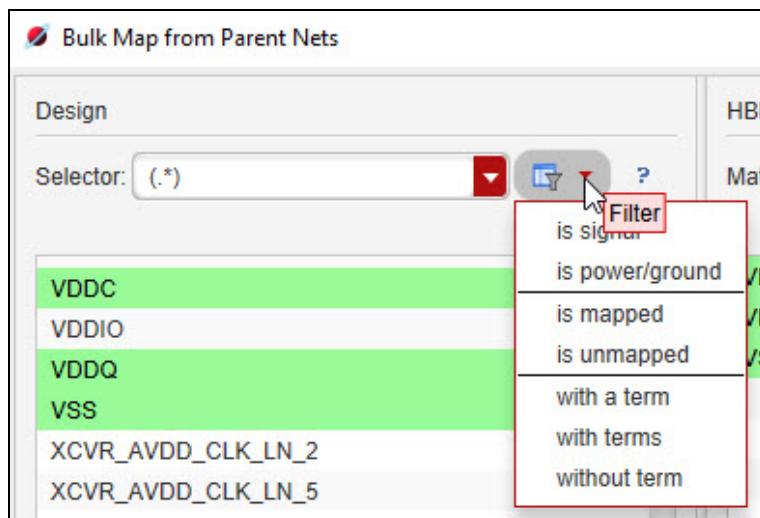
Nets can be mapped from the parent to the child devices only. Bulk mapping commands are only available under the Device pulldown menus and not at the top level.



Selecting *Bulk Map > From Parent Nets* will invoke the Bulk Map from Parent Nets dialog, as shown below.



The Nets or Terms displayed can be filtered based on type or condition by selecting the *Filter* icon, as shown below.



The regular expression *Selector:* field can be used to filter the set of nets to map from the parent. The regular expression *Matcher:* field can be used to match the child nets, as shown below.

The image displays two identical dialog boxes for "Bulk Map from Parent Nets". Each dialog has two main sections: "Design" on the left and "HBMHI" on the right.

Design Section:

- Label: "Selector:"
- Value: "HBM2_PHY_HI_(.*)_[(.*)_]"
- Buttons: A dropdown arrow, a red delete button, and a question mark icon.

HBMHI Section:

- Label: "Matcher:"
- Value: "{1}{2}"
- Buttons: A dropdown arrow, a red delete button, and a question mark icon.

Design List (Left):

- HBM2_PHY_HI_CA[0]
- HBM2_PHY_HI_CA[1]
- HBM2_PHY_HI_CA[2]
- HBM2_PHY_HI_CA[3]
- HBM2_PHY_HI_CA[4]

HBMHI List (Right):

- CA0
- CA1
- CA2
- CA3
- CA4

Design Section (Bottom Dialog):

- Label: "Selector:"
- Value: "HBM2_PHY_HI_(.*)_C[(.*)_]"
- Buttons: A dropdown arrow, a red delete button, and a question mark icon.

HBMHI Section (Bottom Dialog):

- Label: "Matcher:"
- Value: "{1}{2}_C"
- Buttons: A dropdown arrow, a red delete button, and a question mark icon.

Design List (Bottom Left):

- HBM2_PHY_HI_RDQSA_C[0]
- HBM2_PHY_HI_RDQSA_C[1]
- HBM2_PHY_HI_RDQSA_C[2]
- HBM2_PHY_HI_RDQSA_C[3]

HBMHI List (Bottom Right):

- RDQSA0_C
- RDQSA1_C
- RDQSA2_C
- RDQSA3_C

Clicking *Map* will map nets according to the criteria set. The command can be run multiple times to map different types of naming matches.

The previous expressions used are available for reuse or editing under the pulldown menu as a part of the *Selector:* and *Matcher:* fields.

Exploring Connectivity

To explore connectivity, double-click any *Net* or *Term* name in the list columns. The nets that are mapped are centered and highlighted in the list, as shown below. This method can be used to explore the net mapping within the design.

| Nets | Children | ASIC | Nets | HBMHI |
|--|------------------------|--|--|---|
| 11141 item(s), 1 match(es) | 5 item(s), 3 match(es) | 3654 item(s), 1 match(es) | 3654 item(s), 1 match(es) | 1842 item(s), 1 match(es) |
| HBM2_PHY_HI_ARFUH1 HBM2_PHY_HI_ARFUH3 HBM2_PHY_HI_CA[0] HBM2_PHY_HI_CA[1] HBM2_PHY_HI_CA[2] HBM2_PHY_HI_CA[3] HBM2_PHY_HI_CA[4] HBM2_PHY_HI_CA[5] HBM2_PHY_HI_CA[6] HBM2_PHY_HI_CA[7] HRM2_PHY_HI_CAPTUREWR | | ASIC BGA HBMHI HBML0 INTERPOSER | HBM2_PHY_HI_CA[1] HBM2_PHY_HI_CA[2] HBM2_PHY_HI_CA[3] HBM2_PHY_HI_CA[4] HBM2_PHY_HI_CA[5] HBM2_PHY_HI_CA[6] HBM2_PHY_HI_CA[7] HBM2_PHY_HI_CAPTUI | CA1 CA2 CA3 CA4 CA5 CA6 CA7 CAPTUREWR |

To view the multiple terms that are mapped to a net, select *Filters – By condition – is matched* under the *Terms* pull-down menu. Double-click any *Nets* in the column and notice the mapped *Terms* are displayed and highlighted, as shown below.

| INTERPOSER | Nets |
|---|--|
| 2 item(s), 2 match(es) | 3896 item(s), 1 match(es) |
| ASIC_HBM2_PHY_HI_CCI[4] HBMHI_CC4 | HBM2_PHY_HI_CC[1] HBM2_PHY_HI_CC[2] HBM2_PHY_HI_CC[3] HBM2_PHY_HI_CC[4] HBM2_PHY_HI_CC[5] HBM2_PHY_HI_CC[6] HBM2_PHY_HI_CC[7] |

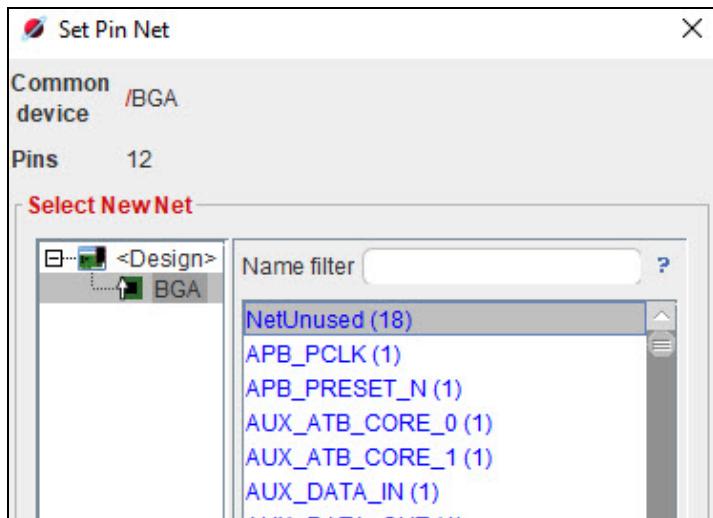
Deleting and Unmapping Nets

Occasionally, nets need to be re-assigned to change the logic or removed entirely to prepare for a netlist update. There are several methods to remove some or all of the net assignments in the design.

The Design Hierarchy Pulldown menu for selected nets enables *Delete* to remove the nets.

Using the Set Selected Pins' Net command

To remove or reassign net assignments for one or more pins, pre-select the pins and right-click to choose *Set Selected Pins' Net*. Select *NetUnused* from the list of nets, as shown below. Ensure the proper level of netlist hierarchy is selected on the left, in order to select *NetUnused*.



Click *OK* to reassign the nets.

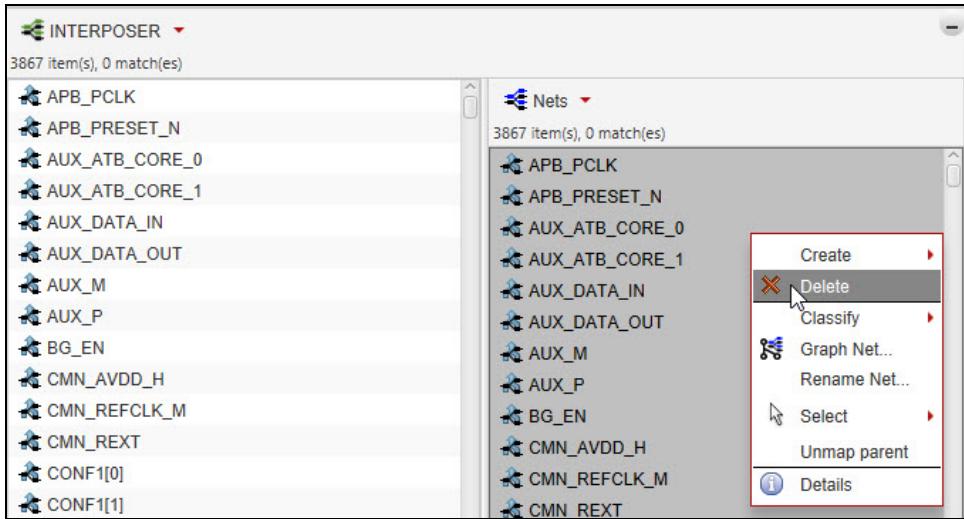
Using the Netlist Editor

Removing all Device Nets

As either CSV net mapping or term files are imported, they are merged with the existing net mapping in the design. It can sometimes be useful to start from scratch with a fresh import of the net mapping.

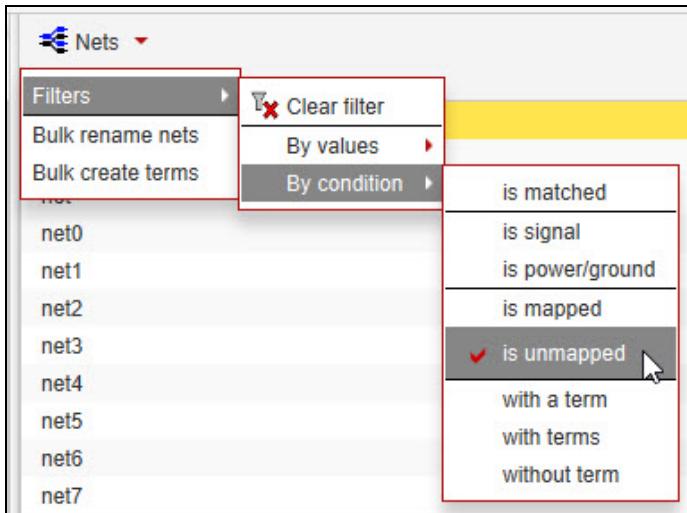
To remove all of the net mapping for a specific device or for the entire design, select all of the device nets in the Netlist Editor and right-click to choose *Delete*, as shown below. *Ctrl-A* can be used to

select all of the nets.

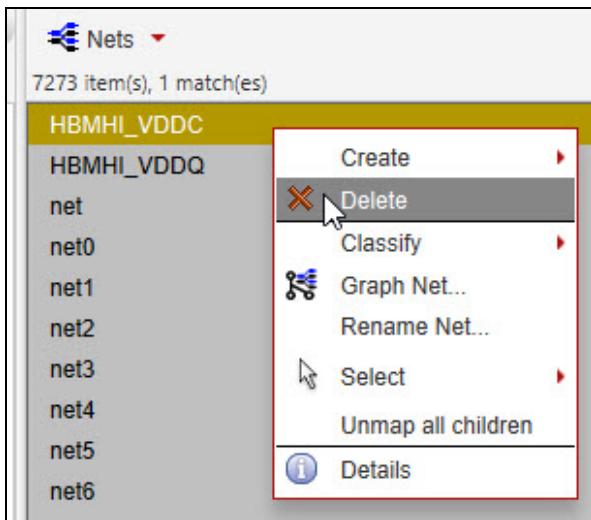


Removing only Unmapped Nets

As nets are interactively mapped, the original nets often become unmapped but still exist in the design. There are also other processes that may result in unmapped nets. It is good practice to filter the nets displayed for each device to only show unmapped nets, as shown below.

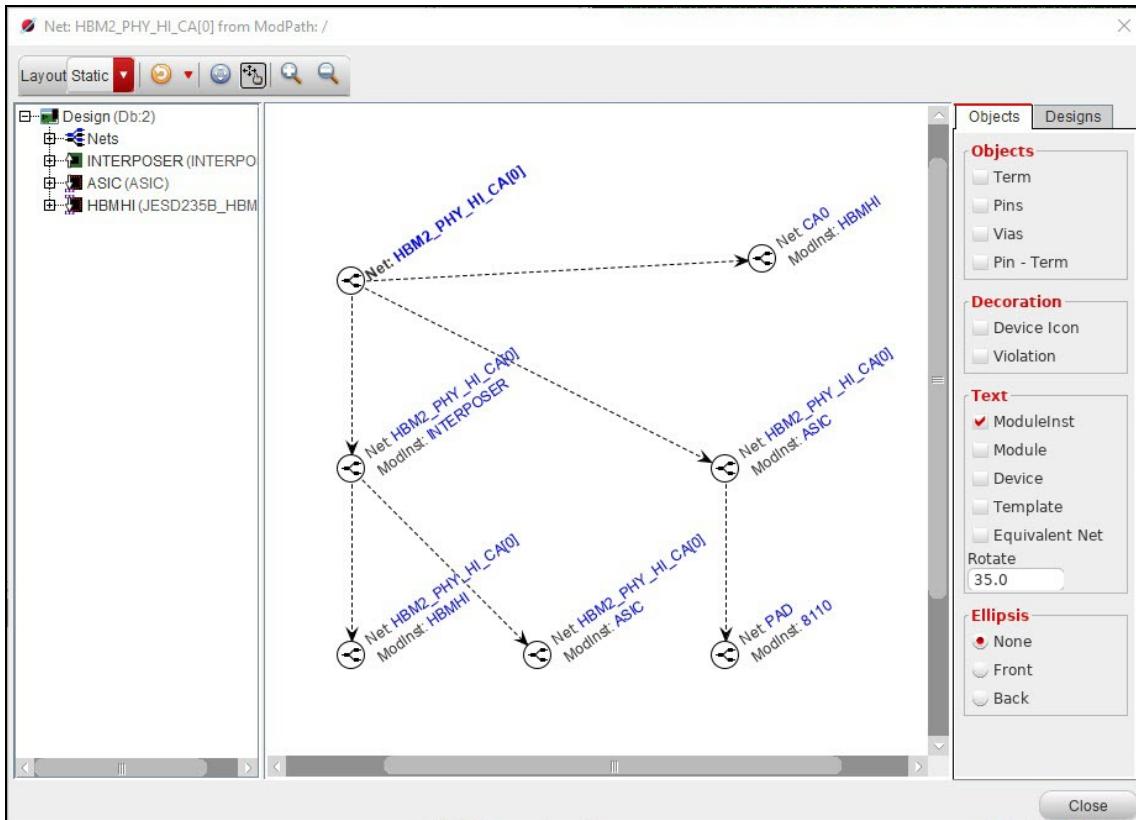


If they exist on the source die, that may be an issue to investigate. If they exist at the top-level or design level, validate that they can be removed. If so, press **Ctrl-A** to select them all and right-click to choose *Delete*, as shown below.



Exploring Connectivity with the Net Graph

Net connectivity across a design can be visualized using the net or term graph. A graph of the connectivity for any net can be displayed by right-clicking the net or term and choosing *Graph Net* or *Graph Term*. The *Net* graph is invoked, as displayed below.



The Net Graph displays the names and types for all the objects on the net. The various objects on the net can be dragged and moved to make the graph more readable.

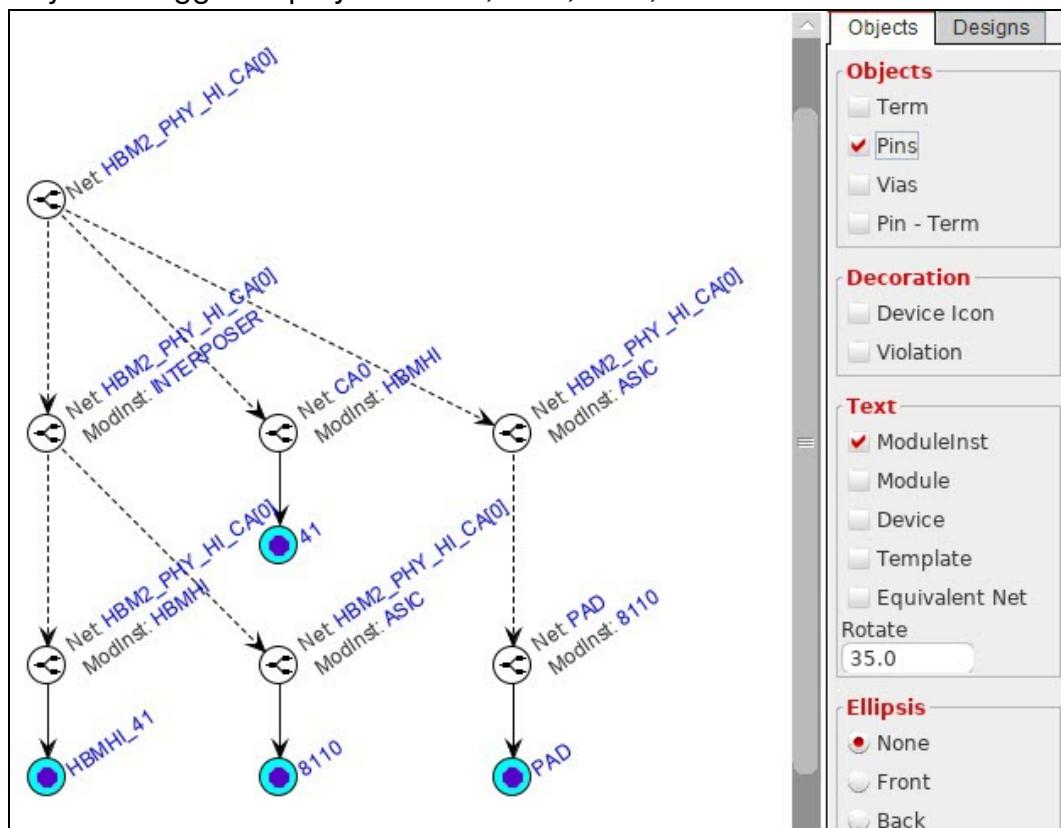
The design hierarchy that contains the net is displayed on the left. There are icons on the view banner to refresh, pan, and zoom the graph canvas. The cursor can also be used to pan and zoom the graph canvas.

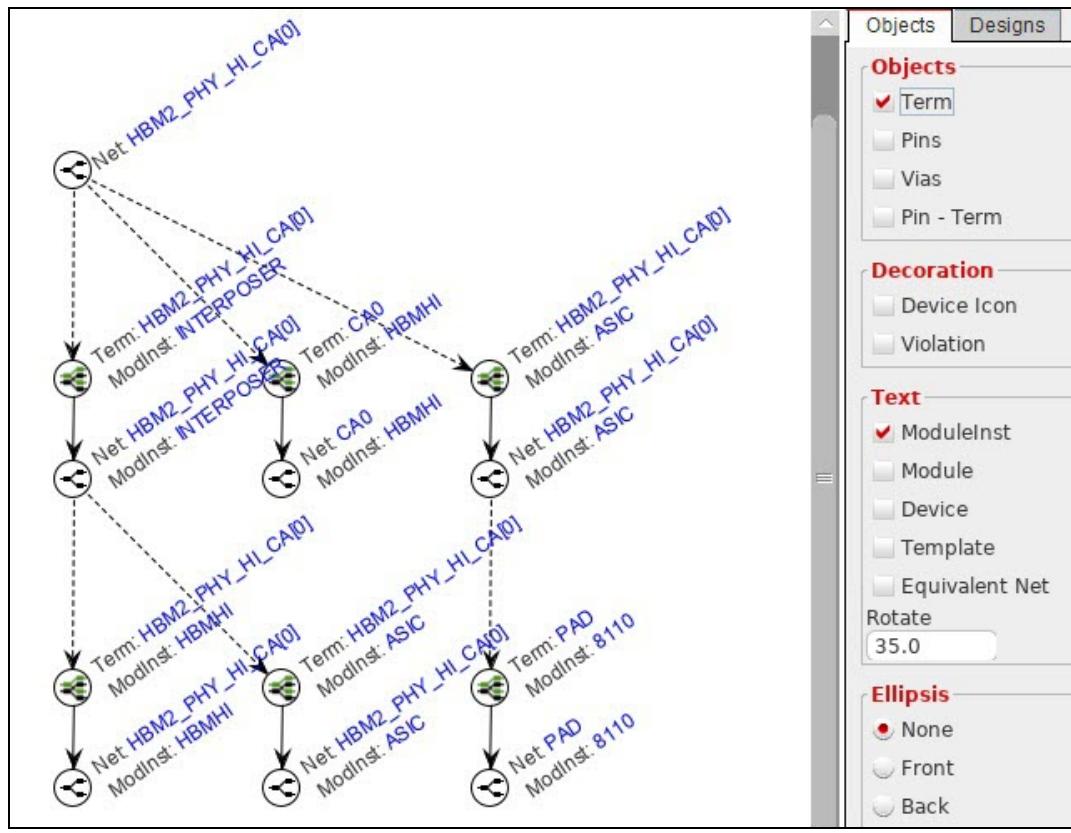
The *Designs* tab can be used to turn on/off devices to display in the graph, as shown below.



The *Objects* tab has the following options for how the graph is presented.

- *Objects*: Toggle display of *Terms*, *Pins*, *Vias*, and *Pin - Term*.

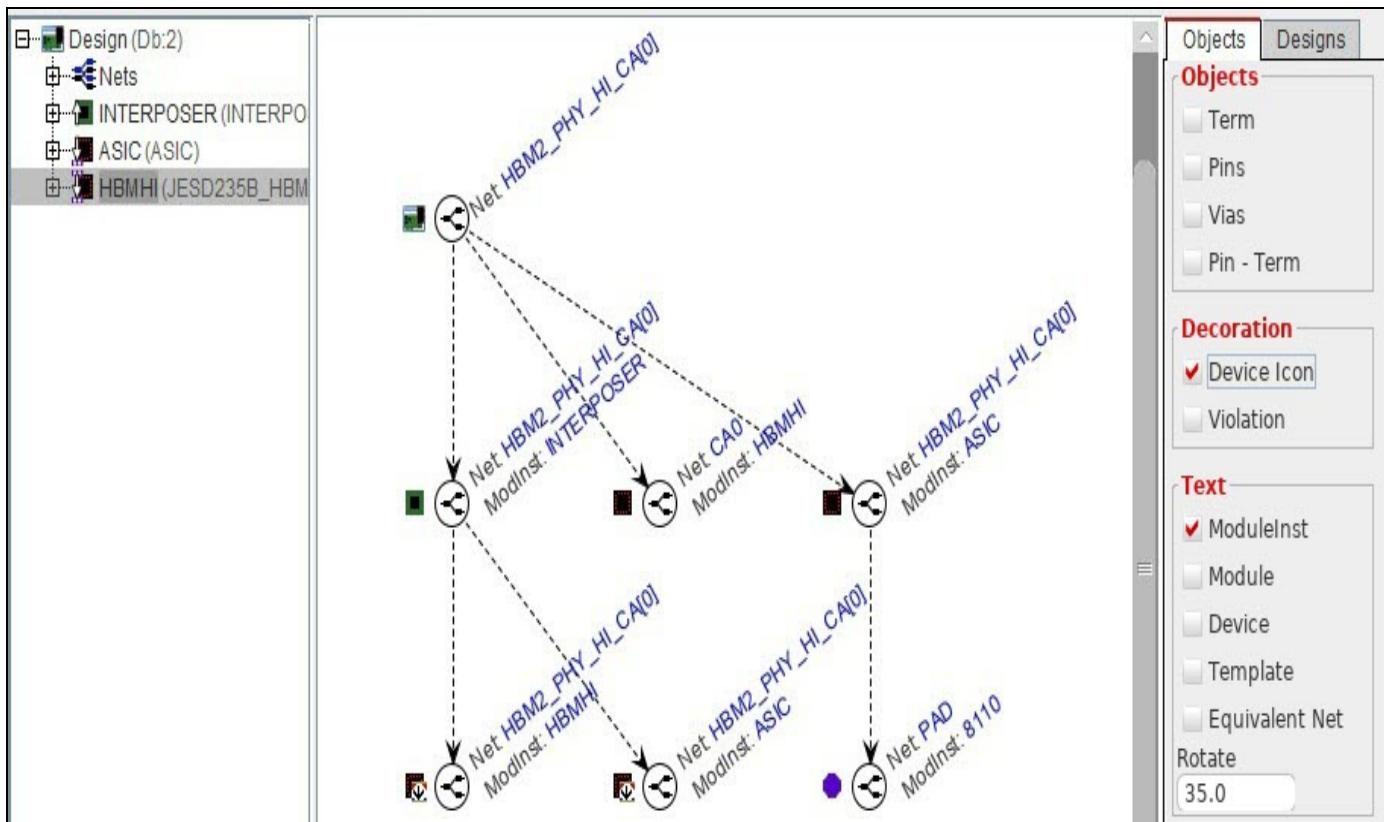




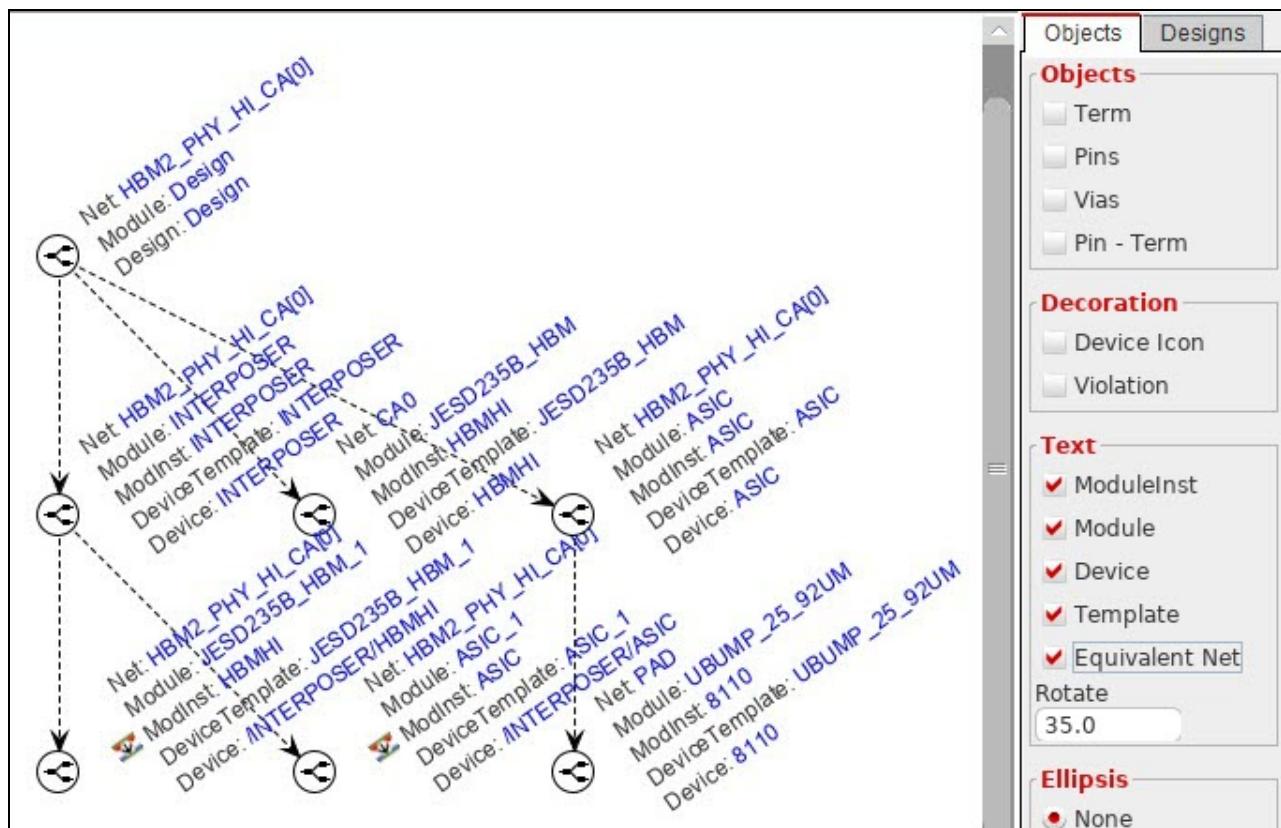
- *Decoration:*

Device Icon: Displays small icon next to each *ModInst* showing the device type such as Die, Package, and Contact Device, as shown below.

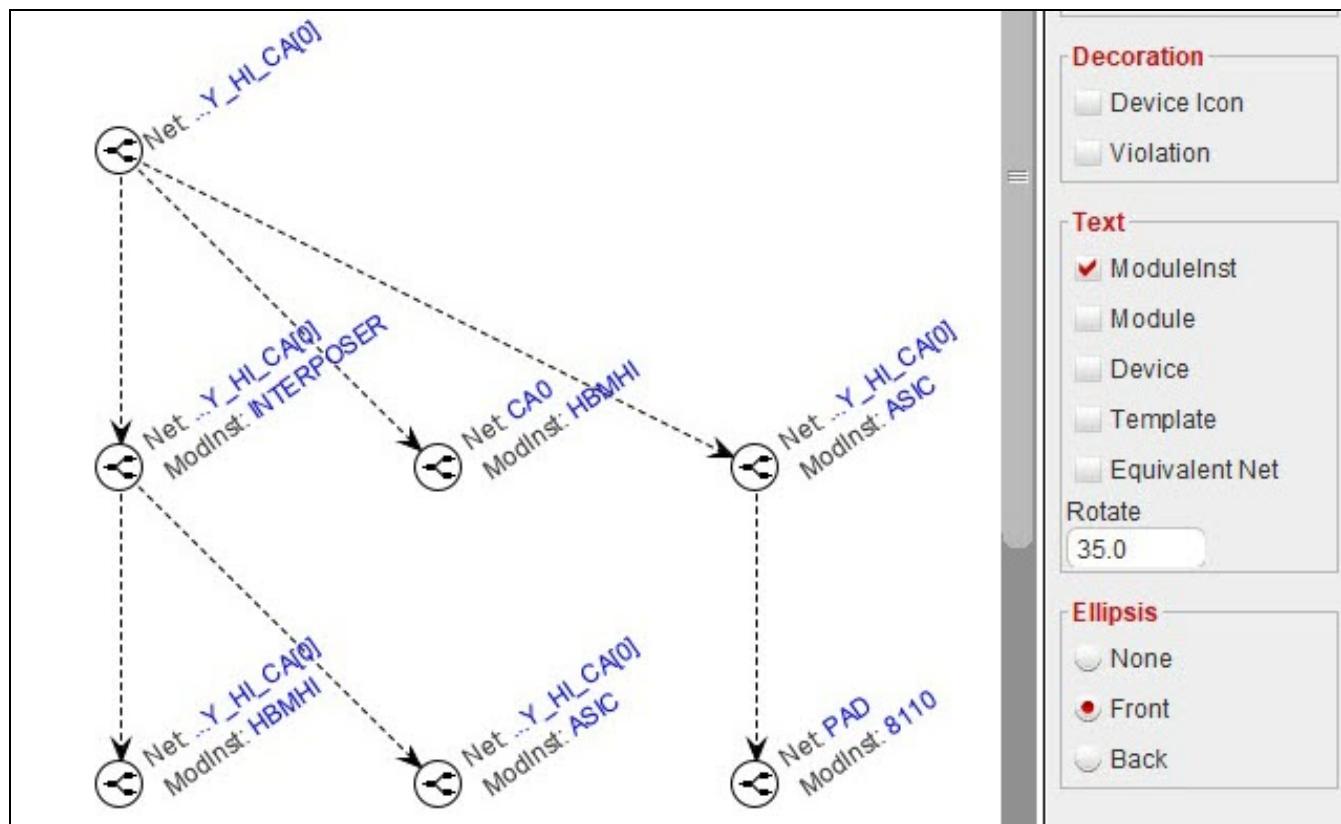
Violation: Displays connectivity violations.



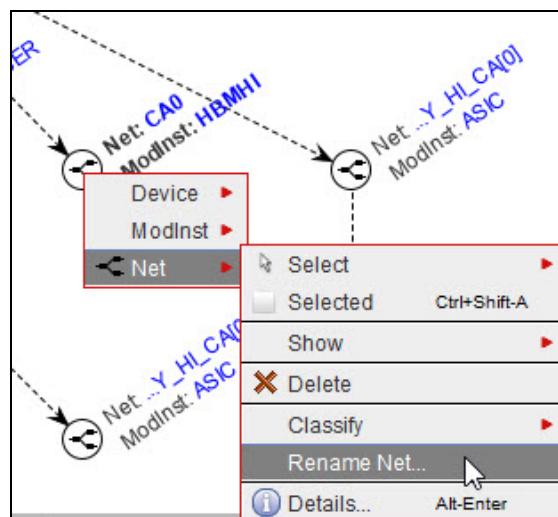
- **Text:** Displays text for the various types of device information, as shown below.



- **Ellipsis**: Controls the length of the text strings displayed. Enabling *Front* or *Back* Ellipsis will truncate the names and display ellipsis (...) in front of or after the first or last ten characters in the names, as shown below. Setting it to *None* will display the full names.



Options to modify nets or devices are available by selecting an object and right-clicking, as shown below.



Modifying and Optimizing Connectivity

There are numerous methods to define, refine, and optimize bump net assignments in order to optimize the routability of the nets. Many of those methods are described in this section.

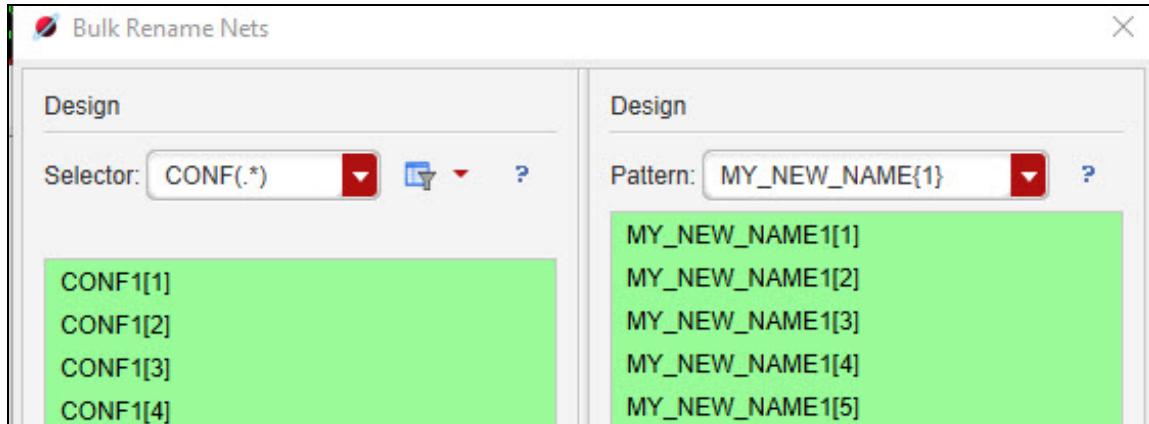
Renaming Nets

Nets can be renamed using the following methods:

- Select nets from the *Nets* folder in the Device Hierarchy and right-click to choose *Rename Net*.
- Select nets in the *Netlist Editor* and right-click to choose *Rename Net*.

Bulk Renaming of Nets

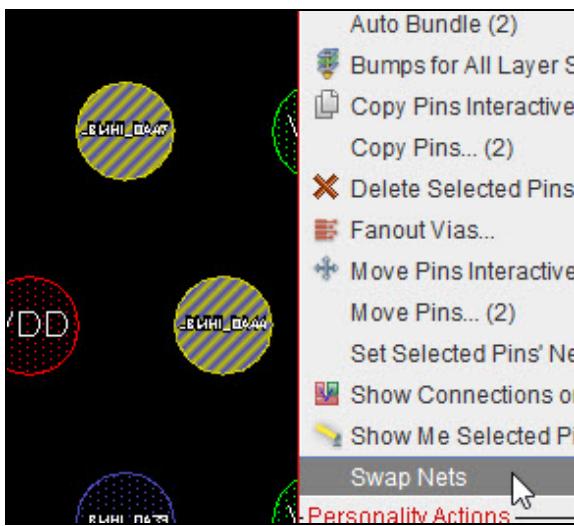
Multiple nets can be renamed using the *Bulk Rename Nets* capability in the Netlist Editor, as shown below.



Regular expression fields are used in the *Selector* field to define the nets to rename, and the *Pattern* field to define the new name, as shown above.

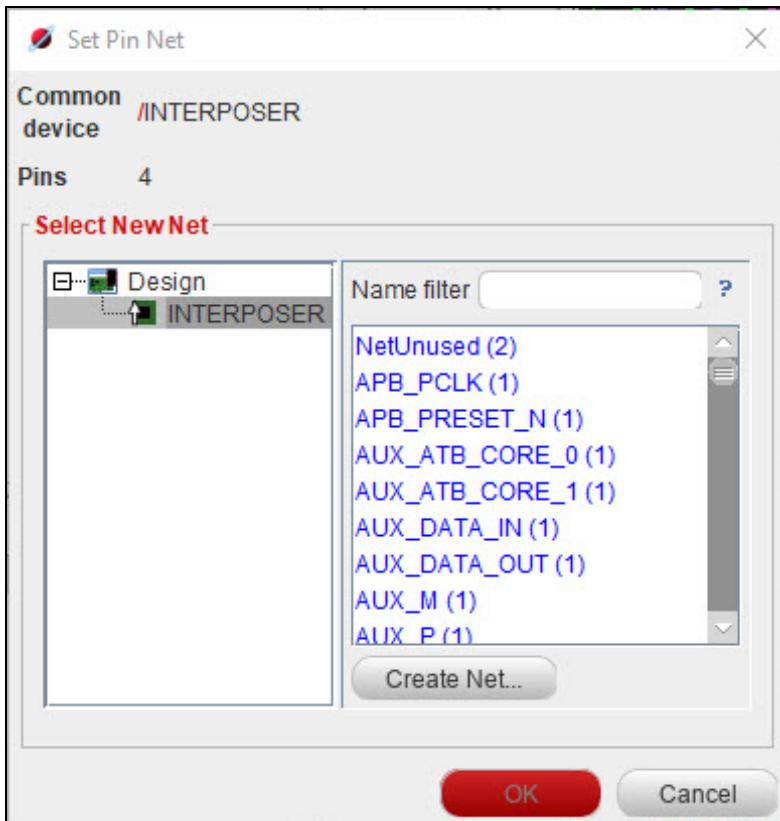
Swapping Pins

The net assignments for any two pins can be swapped by selecting them and right-clicking to choose *Swap Nets*, as shown below. The *Selection Filter* should be set to *Pin* and only two pins can be selected.



Defining Net Assignments for Selected Pins

The net assignments for one or more pins can be defined by first selecting the pins and right-clicking to choose *Set Selected Pins' Net* to involve the *Set Pin Net* dialog, as shown below.



The Device Hierarchy on the left can be used to select parent-level net names. Double-click any

level to display the nets at that level.

Any existing net can be selected to apply to the selected pins. The *Name filter* field can be used to filter the nets displayed.

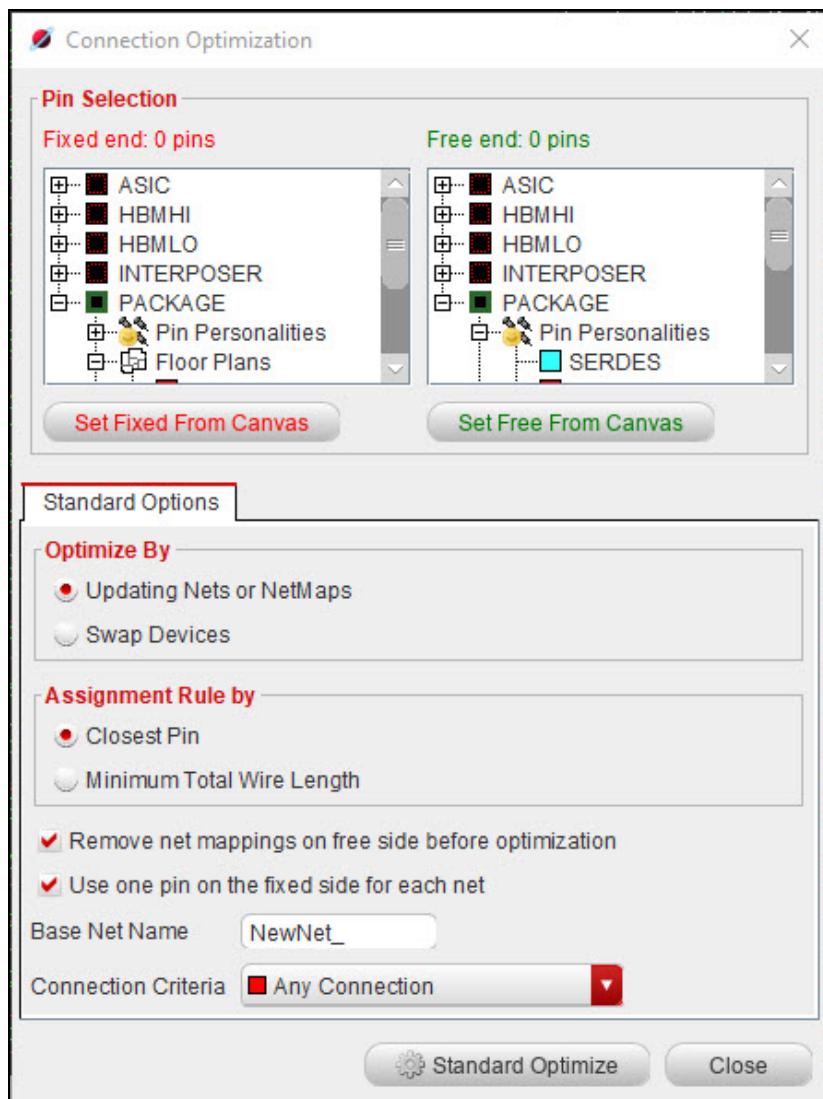
Use the *Create Net* button to create a new net ion the design and assign them to the selected pins.

Performing Connection Optimization

The *Connection Optimization* command is used to perform the optimized pin assignment for simple two pin-to-pin connections. Pins on either end of the connections are selected and automatically assigned nets to reduce net crossing between them. This feature is used for a variety of purposes, including ASIC bump assignment within a die, die-to-die bump alignment for optimized routing, interposer bottom layer C4 bump assignment, and package ball assignment. The command will adhere to differential pair rules and assign the pairs to adjacent pins where possible.

To perform the optimized assignment:

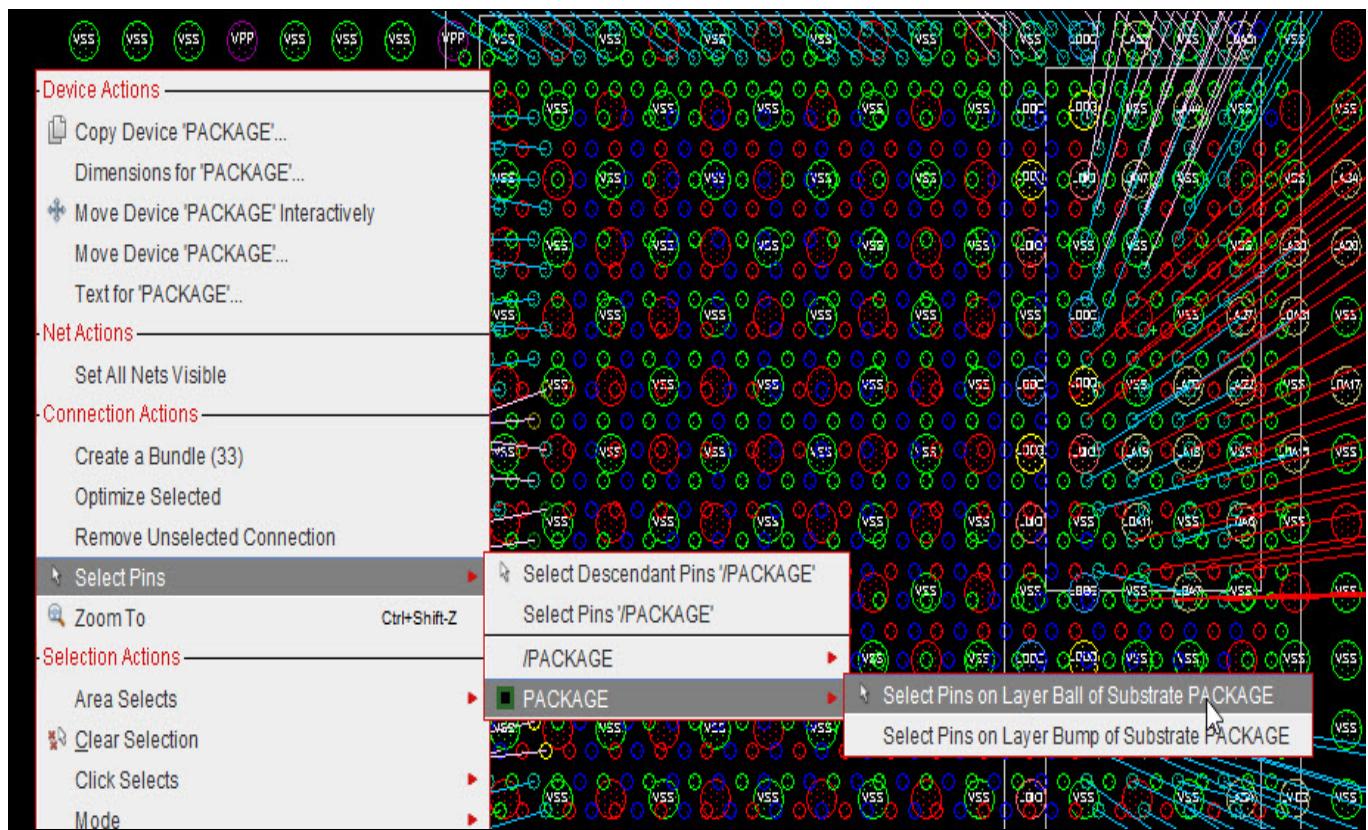
1. Select the *Automation – Connection Optimization* command from the menu bar to invoke the Connection Optimization dialog, as shown below.



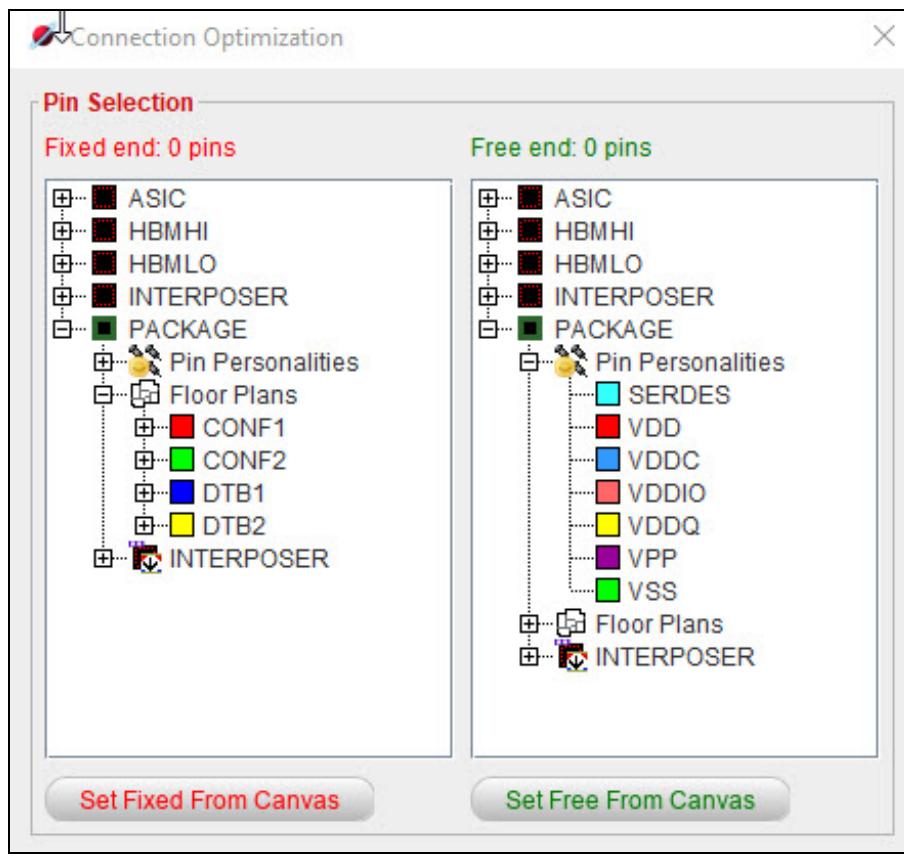
2. The command works on the premise of a *Fixed* set of pins on one end of the connections that do not change, and a *Free* set of pins on the other end that are assigned or reassigned to reduce net crossings. The Fixed end pins are displayed in the left column and the Free end pins on the right.

The fixed and free pins must first be selected. This can be done by using the *Selection Filter* to select the desired *Pins* and then pressing the *Set Fixed from Canvas* or *Set Free From Canvas* buttons to populate the pin counts for each end.

The *Connections* option in the *Selection Filter* can also be used to easily select the pins on either end of the connections. To do so, draw a rectangle on the canvas intersecting the desired connections and right-click to choose *Select Pins* > and choose which device and pin layers to select, as shown below. The connections must be deselected and then reselected in order to set both the fixed and free pin ends.



Pins with Pin Personalities or Interfaces assigned can also be selected using the design hierarchy within the form, as shown below. Interfaces are listed under *Floor Plans*.



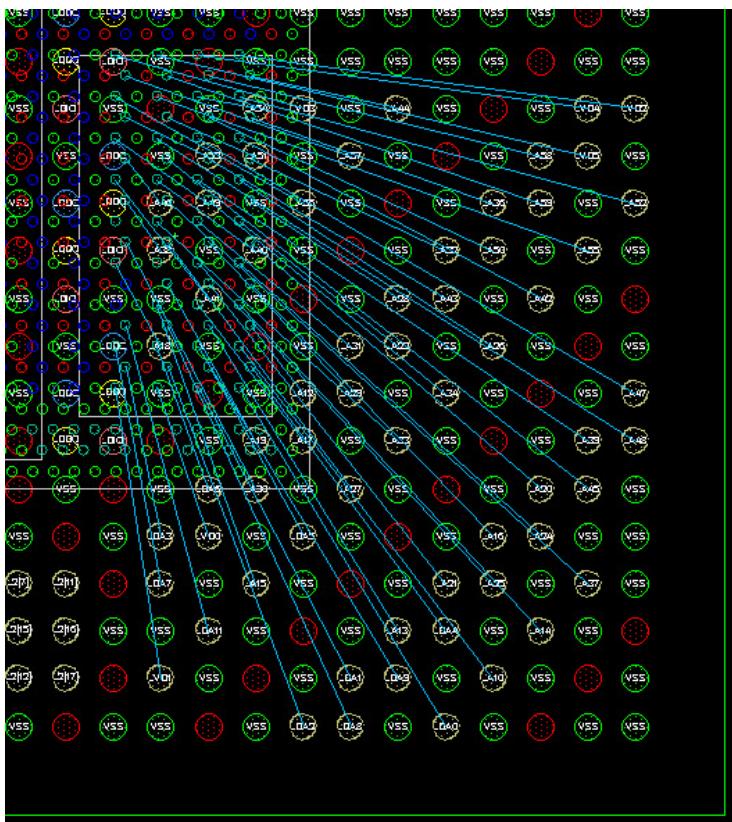
3. After pins are selected, click the *Set Fixed From Canvas* or *Set Free from Canvas* button to load the form with the selected pins. The count of pins selected appears in the dialog next to *Fixed end* and *Free end* at the top of the dialog.
4. Confirm or update the following *Standard Options* in the dialog, as required:

| Option | Description |
|----------------------------------|--|
| <i>Updating Nets or NetMaps</i> | Ensures that newly assigned nets are assigned to the <i>Free end</i> pins and that any existing mapped nets are updated accordingly. |
| <i>Swap Devices</i> | Allows devices to be swapped to achieve optimized pin assignments. |
| <i>Closest Pin</i> | Optimizes based on reducing net crossings. |
| <i>Minimum Total Wire Length</i> | Optimizes based reducing total net length. |

| | |
|---|---|
| <i>Remove Net Mappings on Free Side Before Optimization</i> | Removes preexisting assignments and reassigns the nets. |
| <i>Use One Pin On the Fixed Side For Each Net</i> | <p>Enables you to control net assignment in cases where multiple selected pins have the same net name. Select this check box to assign only one free-end pin for any number of selected fixed-end pins with the same net name. Deselect this check box to assign one free-end pin for every fixed-end pin selected, regardless of whether the selected pins have the same net name.</p> |
| <i>Base Net Name</i> | Defines a net name prefix for newly assigned pins. |
| <i>Connection Criteria</i> | <p>Defines the connection criteria to be used for assignment. The <i>Any Connection</i> option is the default and is the recommended choice for assigning pins. The other options, including <i>Match Personality Names</i> and <i>Match Auto Bundle Instructions</i> support existing personality groups and bundles in the design.</p> |

5. Select the *Standard Optimize* button to assign pins using the options defined in the dialog. An example of typical results is shown below.

Integrity System Planner User Guide



Configuring Nets

The System Planner enables several methods to configure nets in the design. This configuration includes definition of net type and grouping of nets for easier optimization and management.

The following types of net configuration are available in the System Planner:

- Managing Power and Ground Nets
- Defining Differential Pair Nets
- Using Net and Pin Personalities
- Using Interfaces
- Using Bundles

Managing Power and Ground Nets

Power and Ground (PG) nets are handled differently than signal nets in the following ways:

- PG nets are not displayed when connectivity is displayed by selecting *Display Connection* () from the main toolbar. They are filtered out to better show the signal nets. The *Settings* option in the *Display Connection* pulldown enables PG net connectivity to be displayed. Nets with over a 100 connections are automatically treated as PG nets, and their net connections are not displayed when using *Display Connection*.
- The PG nets are displayed under a separate *Nets – Power/Ground* folder in the Device Hierarchy.
- PG nets are often colored by default. A net personality can be created to define the color for PG nets.

Defining PG Nets During Import

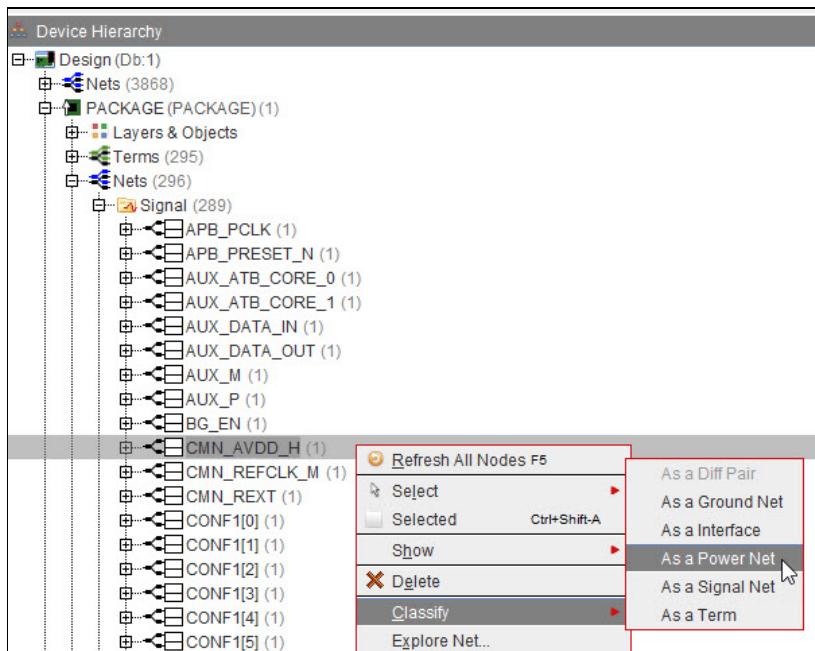
PG nets are automatically identified as Power or Ground during the import of a Verilog netlist, Allegro BRD, SiP, and MCM design, or a Die Abstract (.xda) file.

PG nets can be identified when using the *Import – CSV Pins* command by using the *PIN_USE* column with *POWER* or *GROUND* field entries.

Classifying PG Nets

Any net can be interactively defined as a Power or Ground net by right-clicking the net in the Device Hierarchy and choosing *Classify – As a Power/Ground Net*, as shown below.

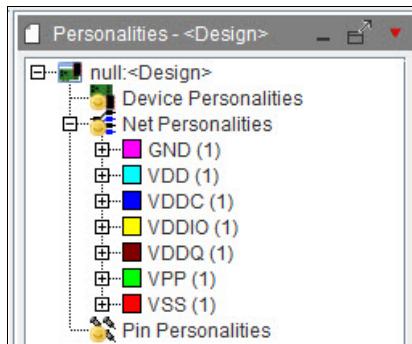
PG nets defined using the *Classify* command will result in the net being displayed under the *Power/Ground* folder.



Defining Net Personality Colors

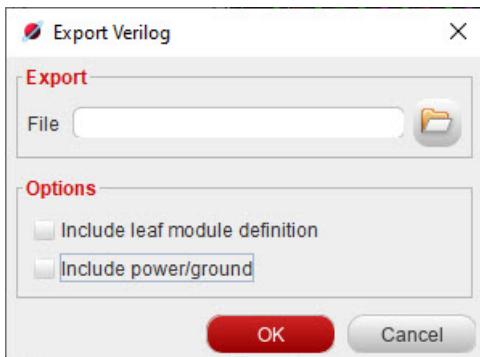
Nets classified as PG nets automatically get a *Net Personality* assigned, allowing them to be colored for easier identification, as shown below.

Colors can be changed in the *Edit > Personalities* Editor. Selecting the device to expand it and select the Net Personality, right-click to choose *Edit*. Select the color box and choose a new color from the color palette presented.

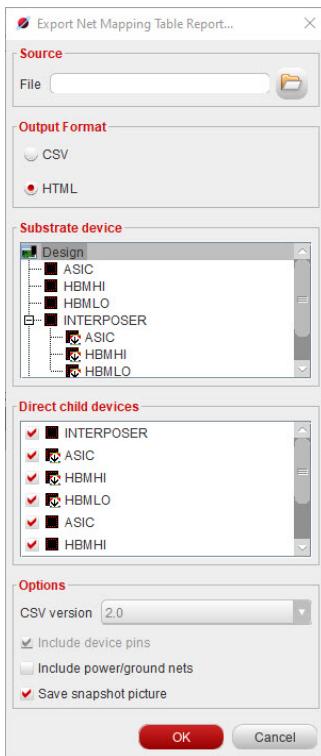


Exporting Verilog or Net Mapping with or without PG nets

The *Export Verilog* command allows PG net ports to be filtered out using the *Include power/ground* option, as shown below.



The *Export Net Mapping Table Report* command also enables PG nets to be excluded to avoid the CSV fields from exceeding character limits due to too many pins, as shown below.

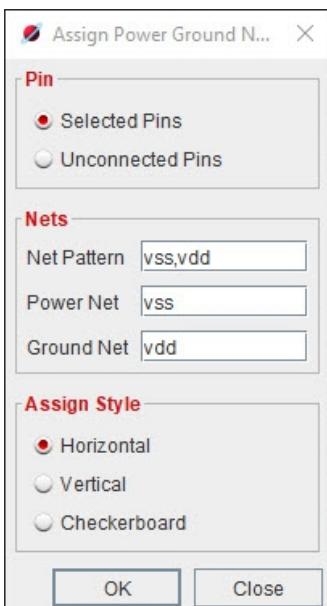


Assigning Power and Ground Patterns

Power and ground (PG) nets can be assigned automatically to a selected group of pins. Options are provided for net assignment patterns, including allowing NC signal pins, to be left within the pattern.

The *Assign Power Ground Nets* command can assign PG nets on a selected set of pins or to the remaining NC pins in the design using the following steps:

1. Select the desired pins using any selection method, including the *Selection Filter*. To select all the pins in an existing *COVERs* pin group, right-click the group and choose *Select – All Pins*.
2. Next, right-click and choose the *Assign Power Ground Nets* command. The *Assign Power Ground Nets* dialog is presented, as shown below.



The dialog options are as follows:

| Option | Description |
|-------------------------|--|
| <i>Selected Pins</i> | Assigns PG nets to the selected set of pins. |
| <i>Unconnected Pins</i> | Assigns PG nets to the remaining NC pins in the design. |
| <i>Net Pattern</i> | Defines the pattern of net names to auto-assign to the pins. If net names are entered that do not exist in the design, new nets will be created and added to the device's <i>Nets > Power/Ground</i> folder. If NC signal pins are desired in the pattern, enter a net pattern as shown in the following example: <i>VSS, NetUnused, NetUnused, VDD, VSS, ...</i> The <i>NetUnused</i> net name will leave as many NC pins as defined in the assignment matrix. |
| <i>Power Net</i> | Designates any new design net name entered as a <i>Powertype</i> net. |
| <i>Ground Net</i> | Designates any new design net name entered as a <i>Ground-type</i> net. |
| <i>Horizontal</i> | Assigns the net names in a horizontal style. |
| <i>Vertical</i> | Assigns the net names in a vertical style. |
| <i>Checkerboard</i> | Assigns the net names in a staggered checkerboard style. |

- Click *OK* to assign the nets based on the specified options.

Defining Differential Pairs

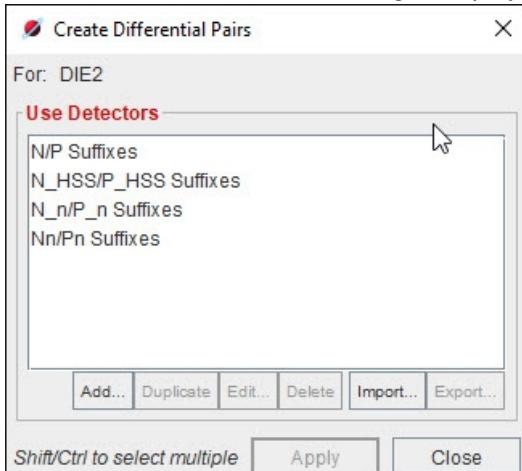
Differential pair (diff pair) signals can be identified in the following ways in System Planner.

- Regular expressions can be used to grep differential pair net naming strategies and automatically classify any nets with matching name patterns as diff pairs. For example, all matching nets with `_N` and `_P` suffixes can be defined easily as diff pair groups.
- Diff pair groups can also be imported using a CSV Personalities file by right-clicking a device in the Device Hierarchy and choosing *Import – Personalities*.

Automatically Assign based on Net Names

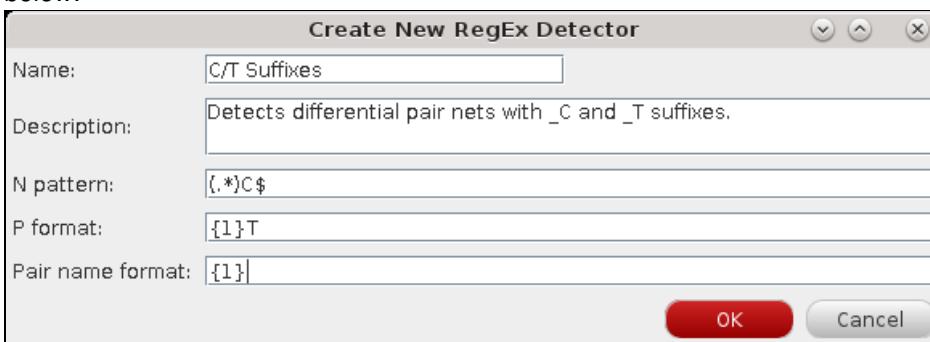
To define diff pair nets based on the net naming strategies:

1. Right-click any level of hierarchy in the Device Hierarchy and choose the *Device – Create Differential Pairs* command.
2. The Create Differential Pairs dialog is displayed, as shown below.



Four common diff pair naming filters, referred as *Use Detectors*, are supplied by Cadence. The Cadence supplied use detectors cannot be edited, but they can be copied and modified.

- New use detectors can be created using the *Add* button and entering the required information. An example is shown below.



This example will create a new use detector recognizing net names with `_C` and `_T` suffixes as diff pairs. The regular expression filters can be used to recognize a wide range of naming strategies for diff pairs.

- You can copy the supplied use detectors by using the *Duplicate* button and edit the copies to see how to define use

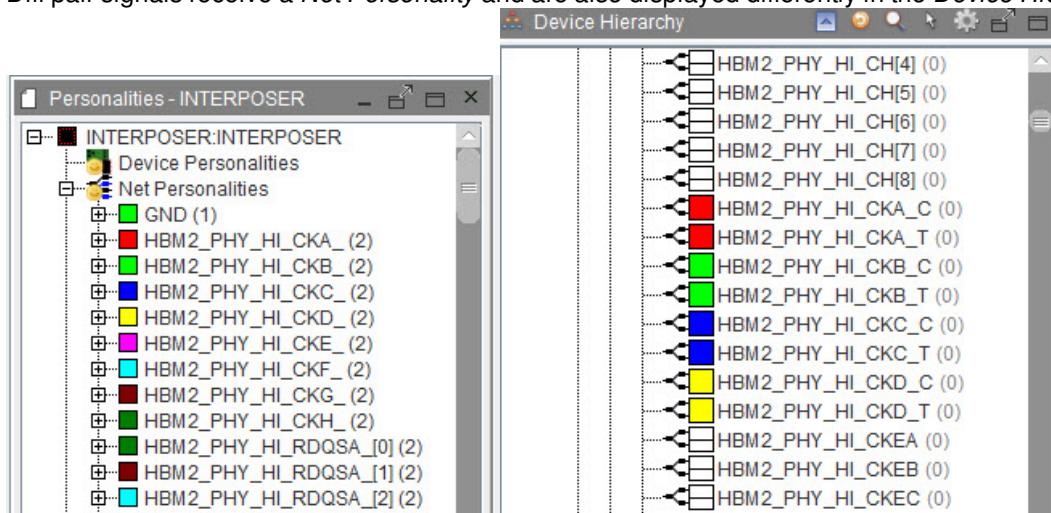
detector regular expressions.

- Previously created use detectors can be modified by using the *Edit* button. Although Cadence supplied use detectors cannot be edited, you can copy them and then modify the copies.
- Use detectors can be saved and reused in other designs. An `xml` format file is written with the *Export* command, and the use detectors can be used again by using the *Import* command.

3. Select the required use detectors from the list and click the *Apply* button. This will scan the selected device for nets matching the selected use detectors and classify those nets as diff pairs. Check the Commands and Messages window to view the assignments made.

The net name minus the use detector characters are used as the diff pair name. For example, the *C* and *T* are removed to form the diff pair net group name, as shown below.

Diff pair signals receive a *Net Personality* and are also displayed differently in the *Device Hierarchy*, as shown below.



Importing a CSV Personalities File

Diff pair net groups can also be imported in a net personality CSV file. Right-click the desired device in the Device Hierarchy and choose *Import – CSV Personalities* to classify the defined net groups as diff pairs automatically.

The CSV Personalities file format for differential pairs includes the three columns below.

| Net | Diff Pair | Diff Pair Color |
|-------------------|------------------|-----------------|
| HBM2_PHY_HI_CKA_C | HBM2_PHY_HI_CKA_ | YELLOW |
| HBM2_PHY_HI_CKA_T | HBM2_PHY_HI_CKA_ | |
| HBM2_PHY_HI_CKB_C | HBM2_PHY_HI_CKB_ | BLUE |
| HBM2_PHY_HI_CKB_T | HBM2_PHY_HI_CKB_ | |

Related Information

[Net and Pin Personalities](#)

Net and Pin Personalities

System Planner uses the term Personality to define a group of nets or pins to easily identify and select them. As an example, power and ground nets receive a net personality to enable unique coloring for easy identification. The *Assign Power Ground Nets* command will automatically create a pin and net personality for each of the voltages entered.

The Personality Editor is used to create and manage personalities. Click *Edit - Personalities* to invoke the Personality Editor.

Net Personalities

Nets can be assigned to net personality groups to enable easier selection and to enable coloring the connections. They are primarily used to control the display color for Power and Ground (PG) nets.

Differential pair signals are also identified as net personality groups. For more information on defining differential pair net personalities, see [Defining Differential Pairs](#).

They can also be used to select pins in the *Connection Optimization* and *Create Bundle* dialogs.

Net personalities can be defined and exported for use in other designs to ensure PG nets are consistent.

Pin Personalities

Pin personalities are used to specify a group of pins for floorplanning and visualization. For example, a section of pins can be assigned a pin personality to reserve them for a specific interface or to color them for identification.

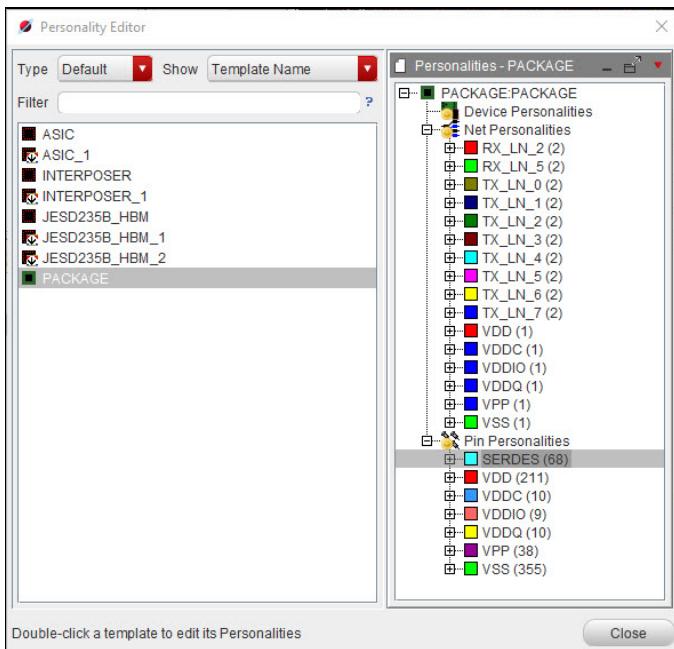
Pin Personalities can be used to create large and unique patterns for PG pin assignment. A small section can be assigned manually and then replicated by right-clicking and choosing *Copy Selected Pins Personalities* to drag the pattern anywhere. The patterns can also be rotated and mirrored. Once the pins have been assigned to a pin personality, they then need to be assigned to a net. For example, assigning a pattern of pins to the VSS Pin personality does not assign them to the VSS net. To assign the net, use *Edit - Personality Editor* and expand to select the desired Pin Personality and right-click to choose *Select Pins*. Then right-click on the canvas to choose *Set Selected Pins' Net* to choose the net to assign the pins.

Creating Personalities

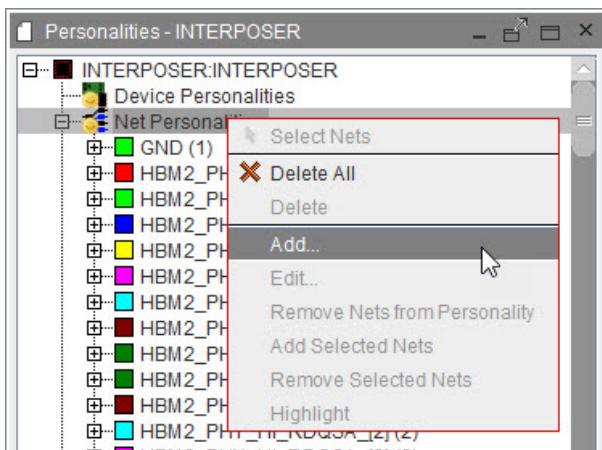
Personalities can be created interactively and assigned to design objects.

To create a personality interactively:

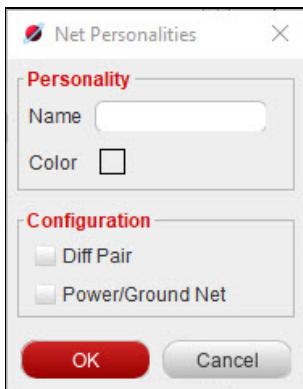
1. Select *Edit – Personalities* from the menu bar to invoke the Personalities Editor, as shown below.



2. Double-click a device in the list on the left pane to populate its existing device personalities on the right. Nets classified as power or ground, as well as differential pairs, are automatically assigned a net personality group and are displayed, as shown above.
3. To create a new personality group, right-click the *Net Personalities* or *Pin Personalities* folder and select *Add*, as shown below.



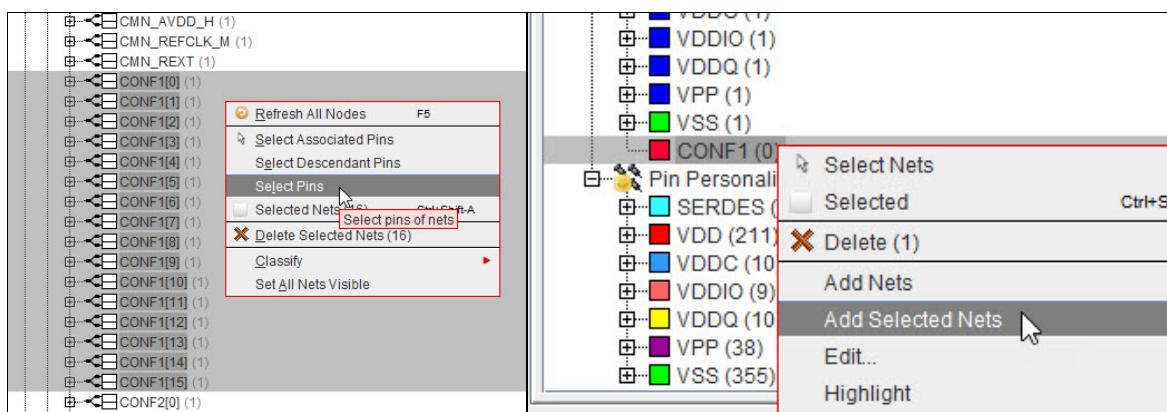
4. The *Net Personalities* or *Pin Personalities* dialog is presented. Specify the relevant details and click *OK* to complete creating the new personality.



Assigning Personalities

Assigning Net Personalities

To assign nets to a personality, first select the nets in the Device Hierarchy, as shown below. Next, right-click the nets and choose *Selected Nets*. Then right-click the personality group in the Personality Editor and right-click to choose *Add Selected Nets*, as shown below.



Assigning Pin Personalities

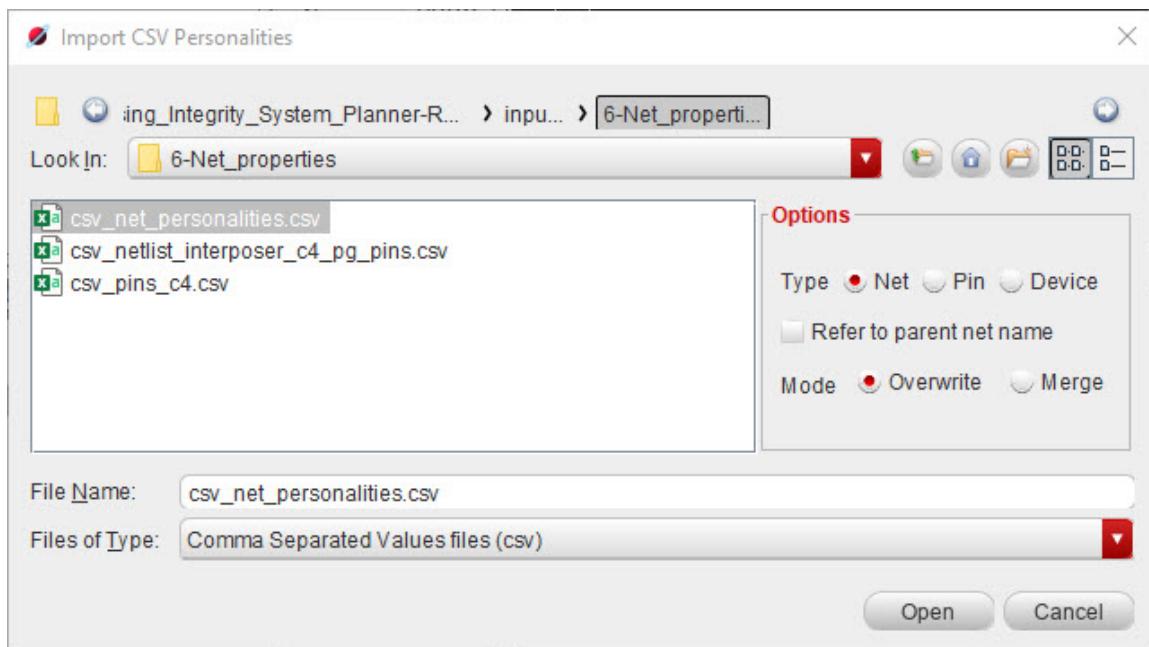
To assign pins to a pin personality, select the pins and right-click to choose *Add Selected Pins to Personality*.

Importing CSV Personalities

Personality groups can be imported in a CSV personalities file.

To import personality groups:

1. Right-click the device in the Device Hierarchy and choose *Import – CSV Personalities*. The Import CSV Personalities dialog is presented, as shown below.



2. Browse and select the CSV personalities file in *Look in*.
3. In the *Personality* section, select the type of personality to import.
4. Confirm or modify the settings in the *Settings* section:
 - *Refer to parent net name* – Uses the parent net name to identify the net names in the design.
 - *Overwrite* – Removes all current personalities defined in the design and replaces them with the imported groups.
 - *Merge* - Merges the imported personalities with those currently defined in the design.
5. Click *Open* to import and define the personality groups to the objects in the design.

An example of a CSV personality file is shown below.

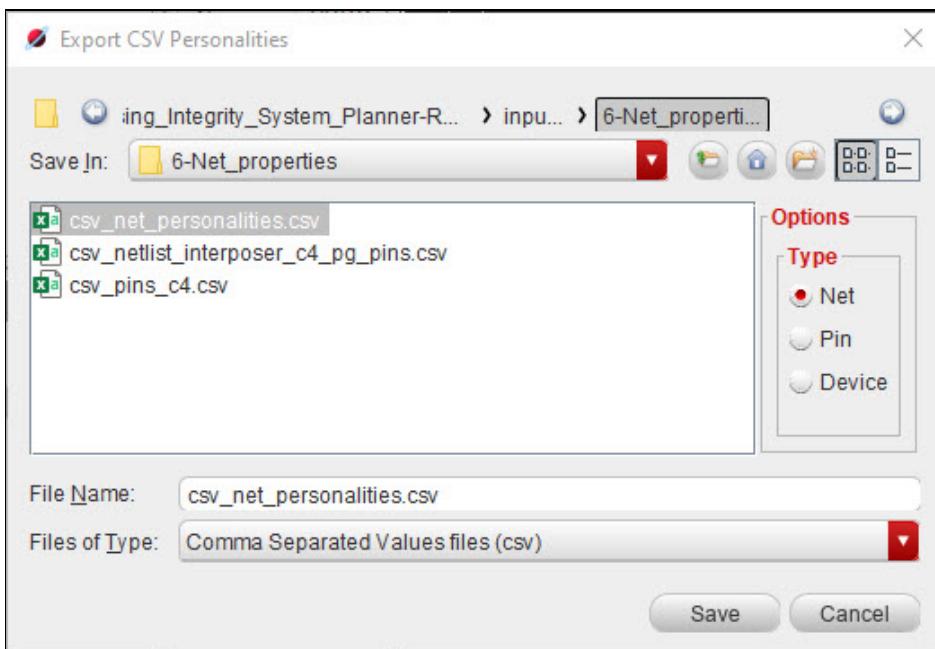
| | A | B | C | D | E | F | G | H |
|----|-------------------|-------|-----------|------------------|---------|-------|-----------------|-----------------|
| 1 | Net | Group | Power Net | Diff Pair | Voltage | Color | Power Net Color | Diff Pair Color |
| 2 | ASIC_VDDIO | | VDDIO | | 1.2 | | YELLOW | |
| 3 | ASIC_VSS | | VSS | | 0 | | GREEN | |
| 4 | GND | | GND | | 0 | | GREEN | |
| 5 | HBM2_PHY_HI_CKA_C | | | HBM2_PHY_HI_CKA_ | | | | RED |
| 6 | HBM2_PHY_HI_CKA_T | | | HBM2_PHY_HI_CKA_ | | | | |
| 7 | HBM2_PHY_HI_CKB_C | | | HBM2_PHY_HI_CKB_ | | | | GREEN |
| 8 | HBM2_PHY_HI_CKB_T | | | HBM2_PHY_HI_CKB_ | | | | |
| 9 | HBM2_PHY_HI_CKC_C | | | HBM2_PHY_HI_CKC_ | | | | BLUE |
| 10 | HBM2_PHY_HI_CKC_T | | | HBM2_PHY_HI_CKC_ | | | | |

Exporting CSV Personalities

Personality groups can be exported to a CSV personalities file for use in other designs.

To export personality groups:

1. Right-click the device in the Device Hierarchy and choose *Export – CSV Personalities*. The Export CSV Personalities dialog is presented, as shown below.



2. Browse and select the export folder in *Save in*.
3. Enter a *File Name*.
4. Select the personality *Type* to export
5. Select *Save* to export the file.

Using Interfaces

Interfaces are used to group nets for easier identification and selection. Nets assigned to Interfaces will receive Net Group properties in Allegro with the Interface name. Interfaces are designed to track and mark nets throughout the system design. As Interfaces are defined, they pins are assigned across the entire system design.

NOTE: Pins within contact devices are not supported for automatic inclusion into the Interfaces as the contact devices are created and managed. They must be manually added if intended for uses such as for selecting pins during *Connection Optimization*.

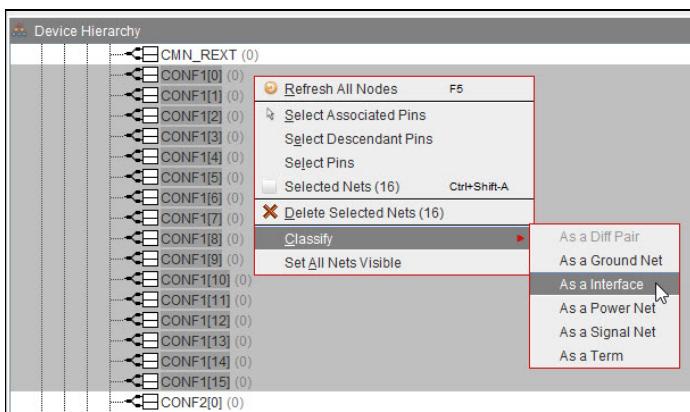
Interfaces can be defined interactively or by importing a Net-Interface Map file.

Interactively Classifying Nets as an Interface

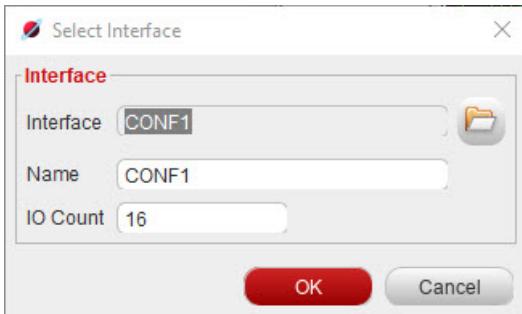
Nets can be interactively added to new or existing Interfaces.

To create a new Interface or add nets to an existing one:

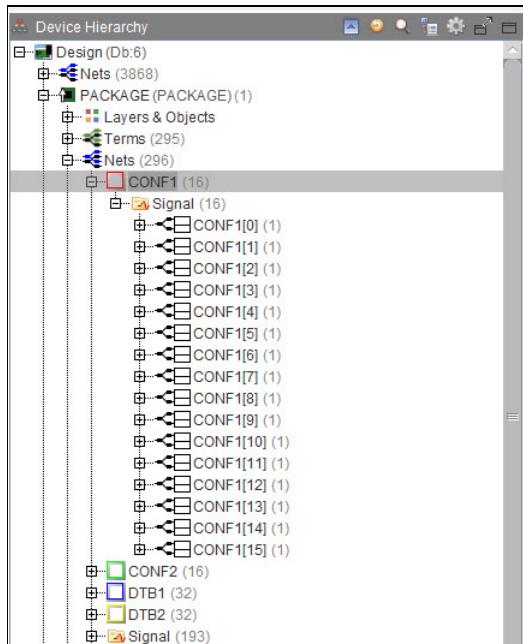
1. Select the desired nets either in the Device Hierarchy or Netlist Editor and right-click to choose *Classify > As a Interface*, as shown below.



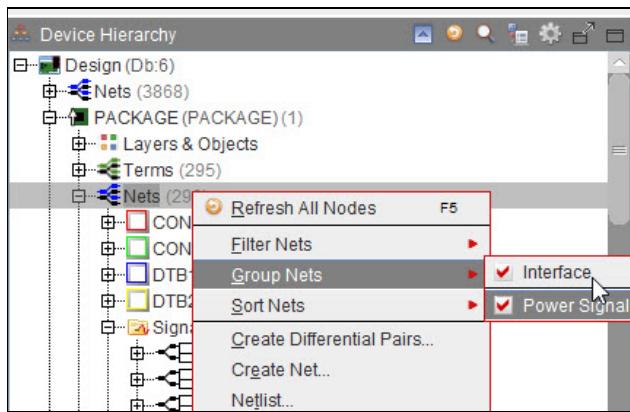
2. Enter an Interface *Name* or browse to select an existing *Interface* to add the nets and click *OK*, as shown below.



Once Interfaces are created, they are displayed in the Device Hierarchy, as shown below.



Ensure that the *Nets - Group - Interfaces* option is enabled in the Device Hierarchy to view the Interfaces, as shown below.



Importing/Exporting a Net-Interface Map File

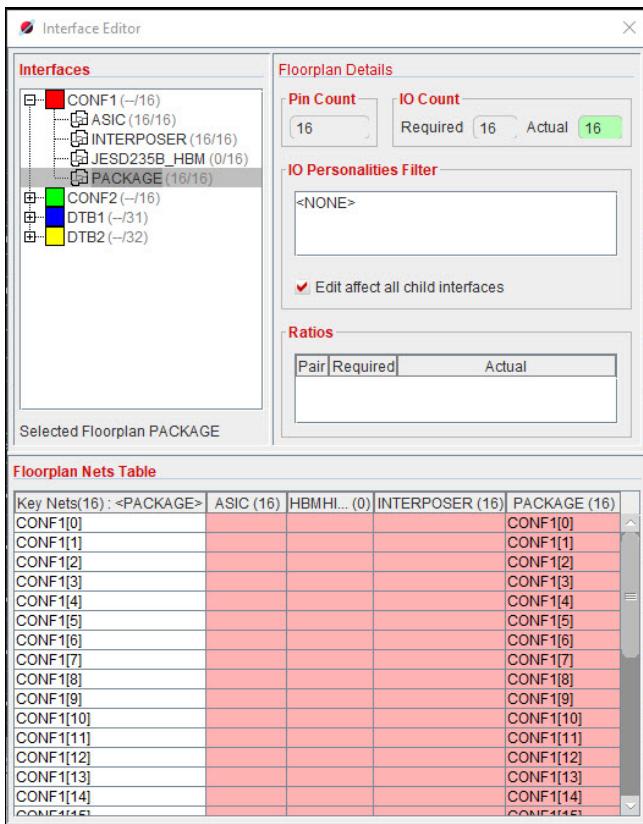
Interfaces can be defined using a CSV format Net-Interface Map file, as shown below. The file can be imported for any selected device by using *Import > Net-Interface Map* or exported using *Export > Net-Interface Map*.

| Net | Interface |
|----------|-----------|
| CONF1[0] | CONF1 |
| CONF1[1] | CONF1 |
| CONF1[2] | CONF1 |
| CONF1[3] | CONF1 |
| CONF1[4] | CONF1 |
| CONF1[5] | CONF1 |

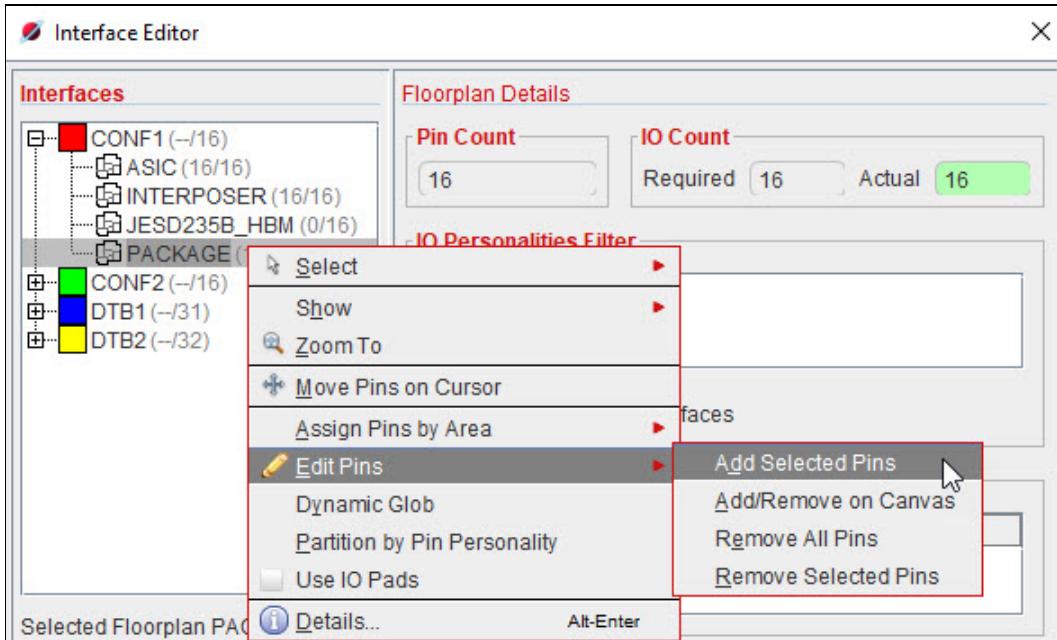
Using the Interface Editor

The Interface Editor is used to manage the Interfaces in the design.

To invoke the Interface Editor, select *Edit - Interfaces*, as shown below.



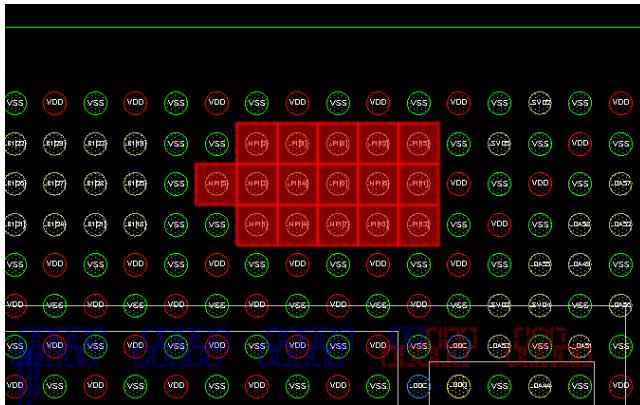
The *Interfaces* hierarchy on the left enables expansion and selection of the interfaces within a specific device. Commands can be executed at each of these levels, as shown below.



Pins can be added or removed from the Interface, as shown above. This example shows how selected pins can be added to the Package interface.

The Dynamic Glob feature enables dragging of the interface across the available pins of a device. For example, the package ball assignment can be experimented with by defining interfaces and dragging them onto unassigned package balls to visualize placement of the pins. Pins will receive the net assignments from the interface once the glob is placed.

Once pins within the Interface are assigned, they will appear with a colored graphic box around them, as shown below.

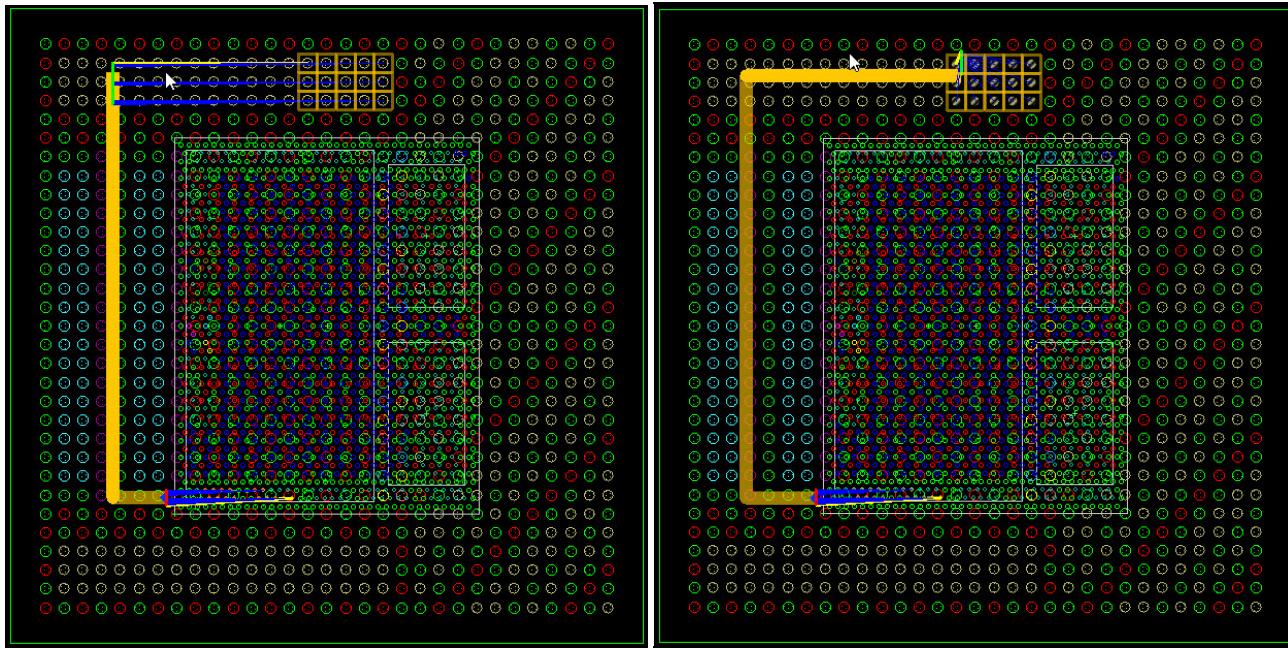


Colors can be defined in the Interface Editor by clicking on the colored square and selecting a new color.

The graphic box display can be toggled off within the *View Settings* toolbar icon - *Display - Show Interfaces*.

Using Bundles

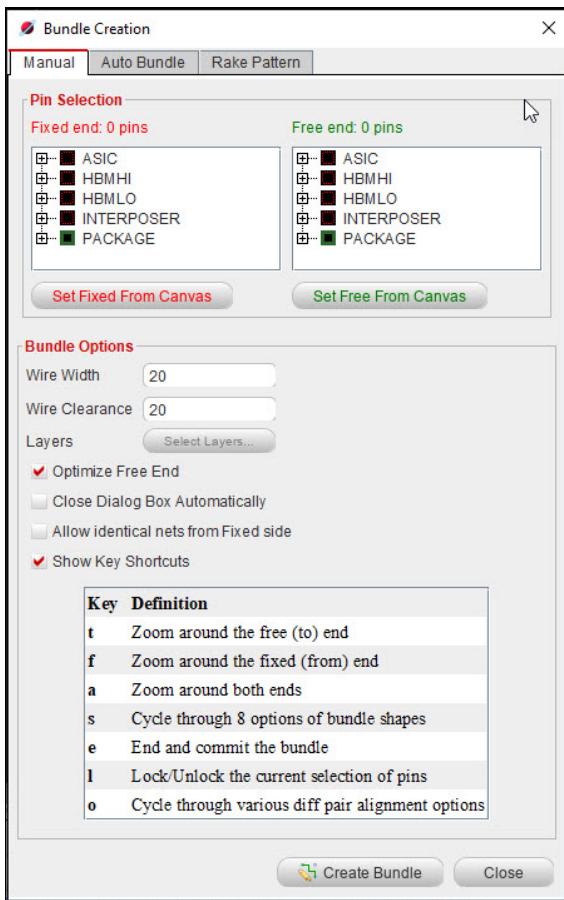
Bundles can be used to communicate routing topology and intended layers to the Allegro package designer. Bundles are used for Allegro implementation and not Innovus. The System Planner *Create Bundle* command works on the same premise as the Connectivity Optimization command where two groups of pins are selected. The Fixed End pins that do not change and the Free end pins. The Free Pins can be assigned or reassigned to reduce route crossings based on the topology of the bundle, as shown below



Creating a Bundle

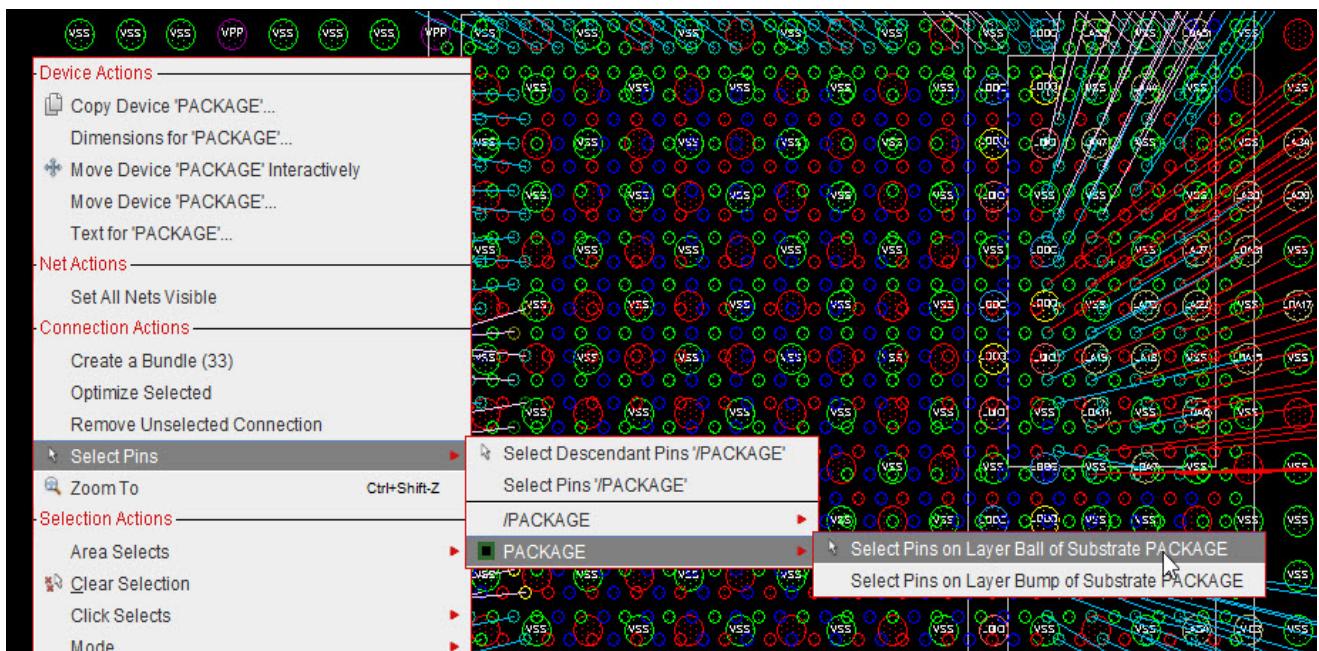
To create a bundle:

1. Select the *Automation – Create Bundle* command from the menu bar to invoke the Create Bundle dialog, as shown below.



The command works on the premise of a *Fixed* set of pins on one end of the connections that do not change, and a *Free* set of pins on the other end that are assigned or reassigned to reduce net crossings. The Fixed end pins are displayed in the left column and the Free end pins on the right. The fixed and free pins must first be selected.

2. Select the Free end and Fixed end pins using one of the following methods:
 - a. Expand and select the device *Nets* in the Device Hierarchy and right-click to choose *Select Descendant Pins*.
 - b. Use the *Pins* option in the *Selection Filter* to interactively select the desired pins.
 - c. Use the *Connections* option in the *Selection Filter* to select the pins on either end of the connections. To do so, draw a rectangle on the canvas intersecting the desired connections and right-click to choose *Select Pins*, and choose which device and pin layers to select, as shown below. The connections must be deselected and then reselected in order to set both the fixed and free pin ends.



d. Pins with Pin Personalities or Interfaces assigned can also be selected using the design hierarchy within the form, as shown below. Interfaces are listed under *Floor Plans*.

3. After pins are selected, click the *Set Fixed From Canvas* or *Set Free from Canvas* button to load the form with the selected pins. The count of pins selected appears in the dialog next to *Fixed end* and *Free end* at the top of the dialog. Repeat the process for both Free and Fixed end pins.
4. Confirm or update the following *Bundle Options* in the dialog, as required:

| Option | Description |
|---|--|
| <i>Wire Width</i> | Defines the width of routed traces. The visible width of the bundle being dragged will reflect the width, spacing, layers, and signal count of the bundle. |
| <i>Wire Clearance</i> | Defines the spacing between routed traces. The visible width of the bundle being dragged will reflect the width, spacing, layers, and signal count of the bundle. |
| <i>Layers</i> | Click to define the routing layers for the bundle. The visible width of the bundle being dragged will reflect the width, spacing, layers, and signal count of the bundle. |
| <i>Optimize Free End</i> | Optimizes the free end pin assignments to reduce routing crossings. |
| <i>Close Dialog box Automatically</i> | Closes the Create Bundle dialog after bundle creation is completed. |
| <i>Allow identical nets from Fixed side</i> | Enables you to control net assignment in cases where multiple selected pins have the same net name. Select this check box to assign only one free-end pin for any number of selected fixed-end pins with the same net name. Deselect this check box to assign one free-end pin for every fixed-end pin selected, regardless of whether the selected pins have the same net name. |

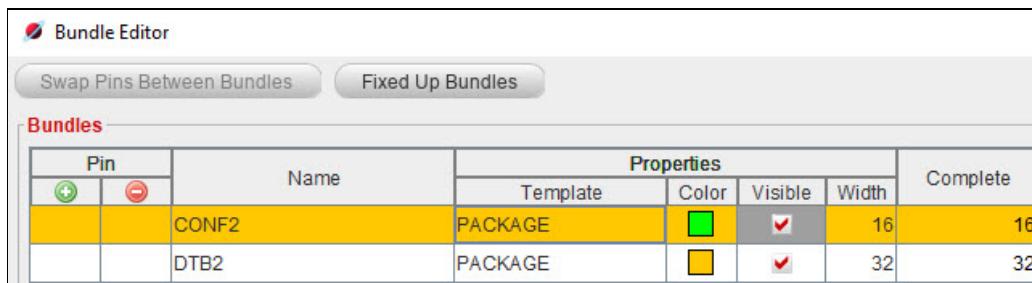
| | |
|---------------------------|--|
| <i>Show Key Shortcuts</i> | Displays the Create Bundle shortcuts. You can use the keystroke shortcuts during interactive bundle creation to control the behavior of the Bundle. |
|---------------------------|--|

5. Click *Create Bundle* and drag and draw the bundle at the desired topology. Use the right-click options to cancel or control the bundle definition. Use the keystroke shortcuts to control visibility and bundle definition.
6. Right-click to choose *End and Commit Bundle*.

Using the Bundle Editor

The bundle editor can be used to manage the bundles in the design.

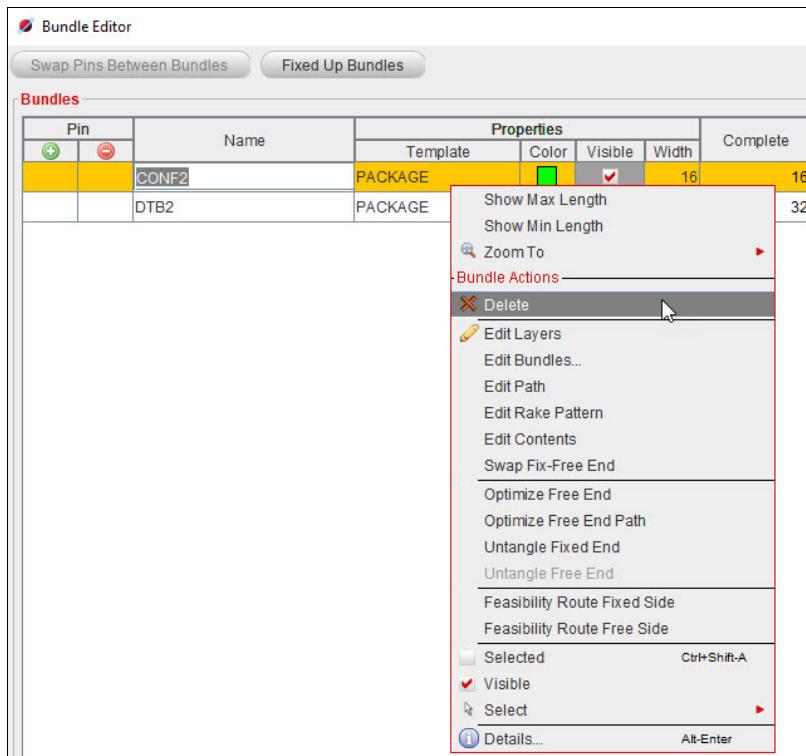
Select *Edit - Bundles* to invoke the Bundle Editor, as shown below.



The screenshot shows the 'Bundle Editor' window with a table titled 'Bundles'. The table has columns for 'Pin', 'Name', 'Properties' (Template, Color, Visible, Width), and 'Complete'. There are two rows: one for 'CONF2' (PACKAGE, green, checked, 16, 16) and one for 'DTB2' (PACKAGE, yellow, checked, 32, 32). Buttons for 'Swap Pins Between Bundles' and 'Fixed Up Bundles' are visible at the top.

| Pin | Name | Properties | | | | Complete |
|-----|-------|------------|---|-------------------------------------|-------|----------|
| | | Template | Color | Visible | Width | |
| | CONF2 | PACKAGE | | <input checked="" type="checkbox"/> | 16 | 16 |
| | DTB2 | PACKAGE | | <input checked="" type="checkbox"/> | 32 | 32 |

Use the right-click options to modify the bundles, as shown below.



The screenshot shows the 'Bundle Editor' window with a context menu open over the 'CONF2' row. The menu includes options like 'Show Max Length', 'Show Min Length', 'Zoom To', 'Delete' (highlighted with a red border), and other bundle actions. The 'DTB2' row is also visible in the background.

Defining Contact Layers and the Design Stack

Contact layers define the relationship between the devices in the design that contact each other. They can be defined between any two devices. When creating a contact layer, the source layer and contacted layers are selected. They can be any layer in the substrate and do not have to be the top and bottom layers. This enables embedded devices within the substrate.

As contact layers are created, contact pads or a contact device can also be created. Contact pads are used for Innovus implementation while Contact Devices are used for Allegro implementation. The Allegro symbol will be created from the child contact device.

By default, all the pins in the source device can be used to create contact pads. A pre-selected set of pins can also be used to create contact pads. This is helpful when multiple pad sizes are used in the source device. Each pad size pad template can be selected using *Selection Filter* and the corresponding contact pads can be created.

Creating a contact layer also establishes a logical relationship between the two devices in the design. When creating a contact layer with contact pads or a contact device, the child device nets automatically appear in the parent-level netlist. The net names can be propagated with net prefixes or mapped to existing top-level nets after the initial contact layer is created. A lot of options are available for defining the netlist relationship and mapping of the two devices.

The Design Stack is used to visualize the assembled design. The design stack is used to configure some of the analysis tools in Integrity 3D-IC. For Allegro implementation, the design stack diagram that is displayed is merely a visual aid and it does not define relationships between the devices.

Creating Contact Layers, Contact Pins, and Contact Devices

Contact layers are created and managed within the Contact Layer Table Editor. The Contact Layer Table Editor can be invoked using either the *Edit – Contact Layer* or *Edit – Design Stack* commands. A contact layer is also sometimes defined automatically when a child device is created in the Device Hierarchy.

To create a contact layer:

Select the *Edit – Contact Layer* command to invoke the ContactLayer Table Editor.

Alternatively, you can use the *Edit – Design Stack* command. In this case, the Contact Layer table is displayed in the lower pane of the Design Stack Editor.

As contact layers are created, they appear in the contact layer chart, as shown below.

| # | Check | Top | | Bottom | | Contact Pin | Thickness | Name | |
|---|-------|-------------|-------|---------------------|-------|-------------|-----------|------|--|
| | | Device Path | Layer | Device Path | Layer | | | | |
| 1 | ✓ | ■ /ASIC | AP | ■ /INTERPOSER/ASIC | AP | 0.0 | | | |
| 2 | ✓ | ■ /HBMHI | Bump | ■ /INTERPOSER/HBMHI | AP | 0.0 | | | |
| 3 | ✓ | ■ /HBMLO | Bump | ■ /INTERPOSER/HBML0 | AP | 0.0 | | | |

Close

Notice in this example, the *Top* and *Bottom* columns indicating the two *Device Paths* and *Layer*. The *Top* device shows the *ASIC AP* layer and the two *HBM Bump* layers contacting the *Bottom* device - *INTERPOSER AP* layer.

The toolbar icons provide the following functionality:

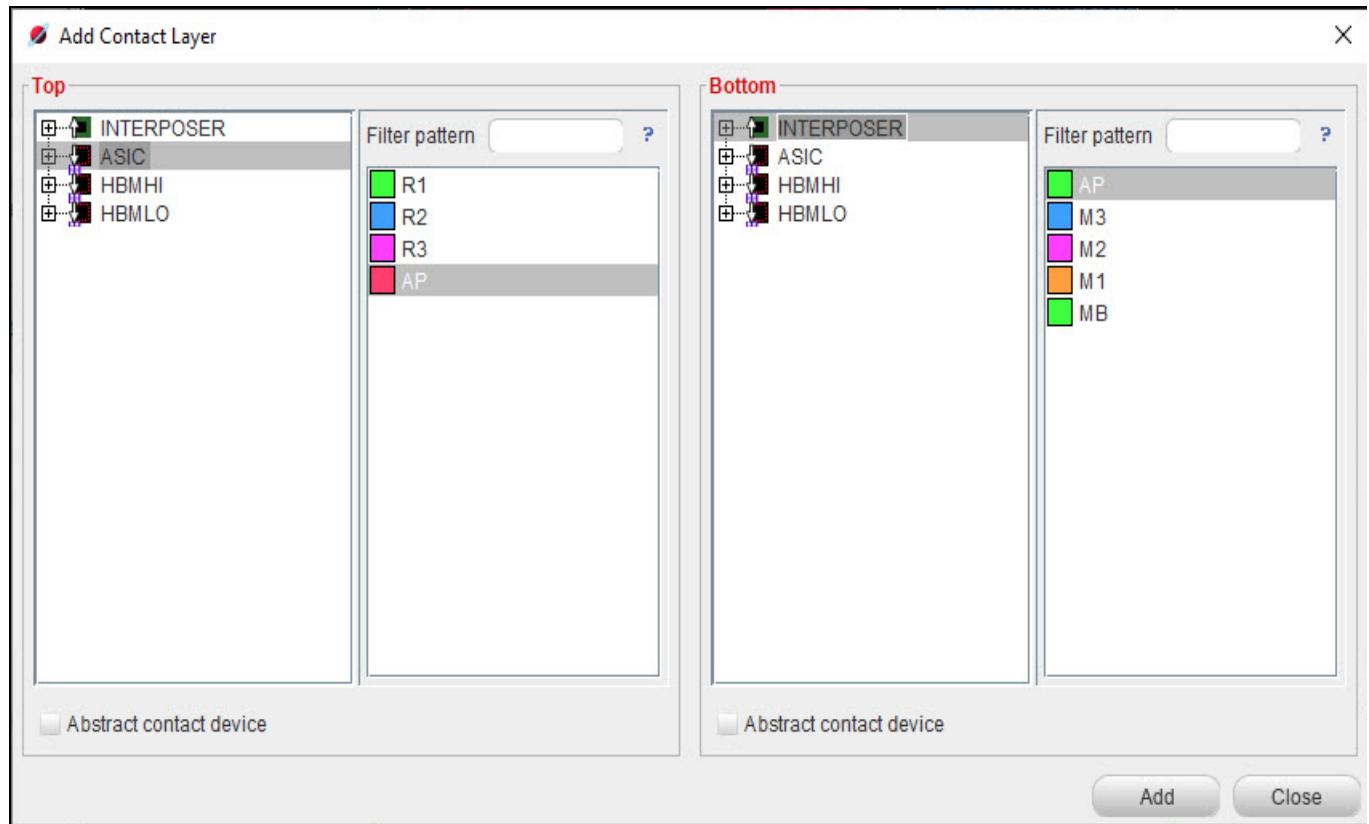
- *Refresh*: Refreshes the chart to reflect any recent changes.
- *Import/Export Contact Layer CSV*: Allows import and export of contact layers.
- *Add Contact Layer*: Invokes the *Add Contact Layer* dialog to add a new contact layer.
- *Delete Contact Layer*: Removes the selected contact layer.
- *Write contact pin map file*: Exports a CSV file containing the pin mapping between the

devices for the selected contact layer.

-  **Auto-Connect Nets:** Automatically derives connectivity for all aligned pads on the two contact devices.
-  **Details:** Invokes the Details dialog for the selected device.

To create a new contact layer:

1. Click the *Add Contact Layer* () icon. The *Add Contact Layer* dialog is presented, as shown below.

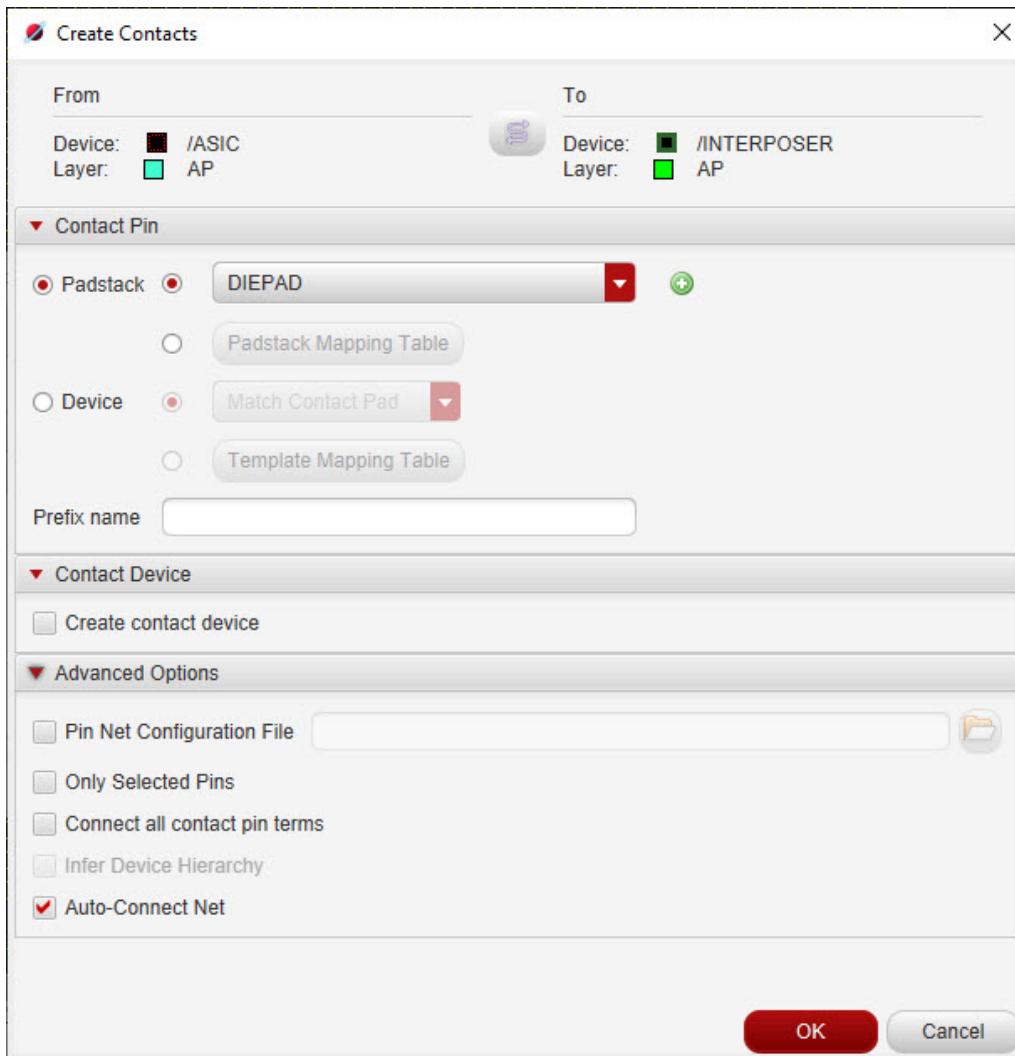


2. Select the *Top* device and *Layer* on the left and the corresponding *Bottom* device and *Layer* on the right, as shown above with the two *AP* layers.
3. Click the *Add* button to create the contact layer and display the following dialog to create contact pads or a contact device.



4. Clicking *No* will simply establish the contact layer without creating contact pads or a contact device. They can be added later, if desired. The child device nets are not automatically propagated to the parent-level netlist. They do appear under the parent and are available for propagation and mapping to parent-level nets.

Clicking *Yes* will present the Create Contacts dialog to create either contact pins or a contact device, as shown below.



5. Select the *Contact Pin* type :

- a. *Padstack* - Used for Allegro and will show all substrate padstacks either imported from Allegro or created manually. Select the desired padstack from the pulldown menu or click the green icon to create a new padstack.
- b. *Device* - Used for Innovus and will show all substrate bump devices either imported from LEF files or created manually. Select the desired bump device from the pulldown menu.

The *Padstack Mapping Table* and the *Template Mapping Table* can be used when multiple pad sizes are needed. The template names for the corresponding pads can be entered in the table.

Enter a *Prefix name* for the pins. This is helpful to identify or uniquify nets. For example, HBM1_, HBM2_,... prefixes could be used to uniquify the HBM interface nets

Select *Create contact device* if the *To* device is being implemented with Allegro. The child contact device for the *From* device is used to create the Allegro symbol.

The *Advanced Options* consist of the following:

- *Pin Net Configuration File* - Presents a browser to select a pin configuration file. Pin configuration files are used to customize the generated contact pin's name or the net/term of the contact pins. The format of the CSV file is same as the pin map CSV, as shown below. Columns "Pin A" and "Pin B" are required, others are optional.

| PinPath A | Pin A | Term A | Net A | PinPath B | Pin B | Term B | Net B |
|-----------|-------|--------|-------|-----------|-----------|------------|------------|
| . | A128 | VSS | VSS | . | ASIC_A128 | ASIC_VSS | ASIC_VSS |
| . | A139 | VDDIO | VDDIO | . | ASIC_A139 | ASIC_VDDIO | ASIC_VDDIO |

- *Only Selected Pins* - Generates contact pins for only the pre-selected pins. This is often used when multiple pad sizes are needed.
- *Connect all contact pin terms* - Assigns terms to the bumps being generated. Used for term-based Verilog netlist and design structure for Innovus.
- *Infer Device Hierarchy* - Infers child parent hierarchy (rarely used)
- *Auto-connect Net* - Automatically creates top-level nets connecting the two devices. It uses generic net1-netX names if matching top-level nets do not exist.

6. Click *OK* to create the contact pins or contact device.

Creating Contact Pins

The use of Contact Pins is recommended for structuring System Planner designs for Innovus implementation. Contact Pins are created using the methods described above and when the *Create contact device* option is turned *off*. Contact pins can be created as either bump *Device* type pins for Innovus or *Padstack* type pins for Allegro. Typically, a contact device would be used for Allegro and not contact pads. Once contact pins are created, the contacted substrate layer will receive pads with the desired sizes and net naming.

Creating a Contact Device

The use of contact devices is highly recommended for structuring System Planner designs for Allegro implementation. They enable proper device and padstack configuration for the implemented substrate.

Contact devices are created using the methods described above and when the *Create contact device* option is turned *on*. Once the contact device is created, the contacted substrate layer will receive pads with the desired sizes and net naming. The Device Hierarchy will now contain a child-level copy of the source device, as shown below.



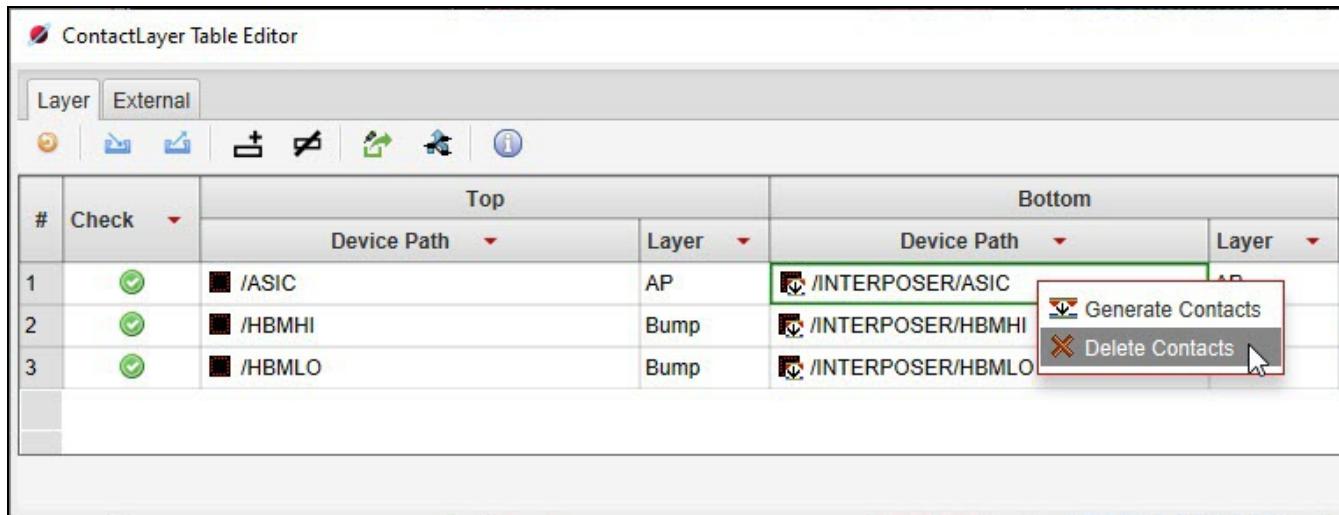
Managing Contact Layers

System Planner will attempt to maintain contact pins and contact devices through changes in the design. When a device with contact pads is moved, a prompt is presented to also move the contact pads. As net names are changed or removed, the contact pads are adjusted accordingly. If major placement changes are being made, delete the contact pads and regenerate them after the moves are completed.

Often after the source device is updated, the contact device may need to be regenerated. Refer to the Inter-Substrate Checks as an indicator whether the contact device is in sync with the source device.

Deleting Contact Pins or a Contact Device

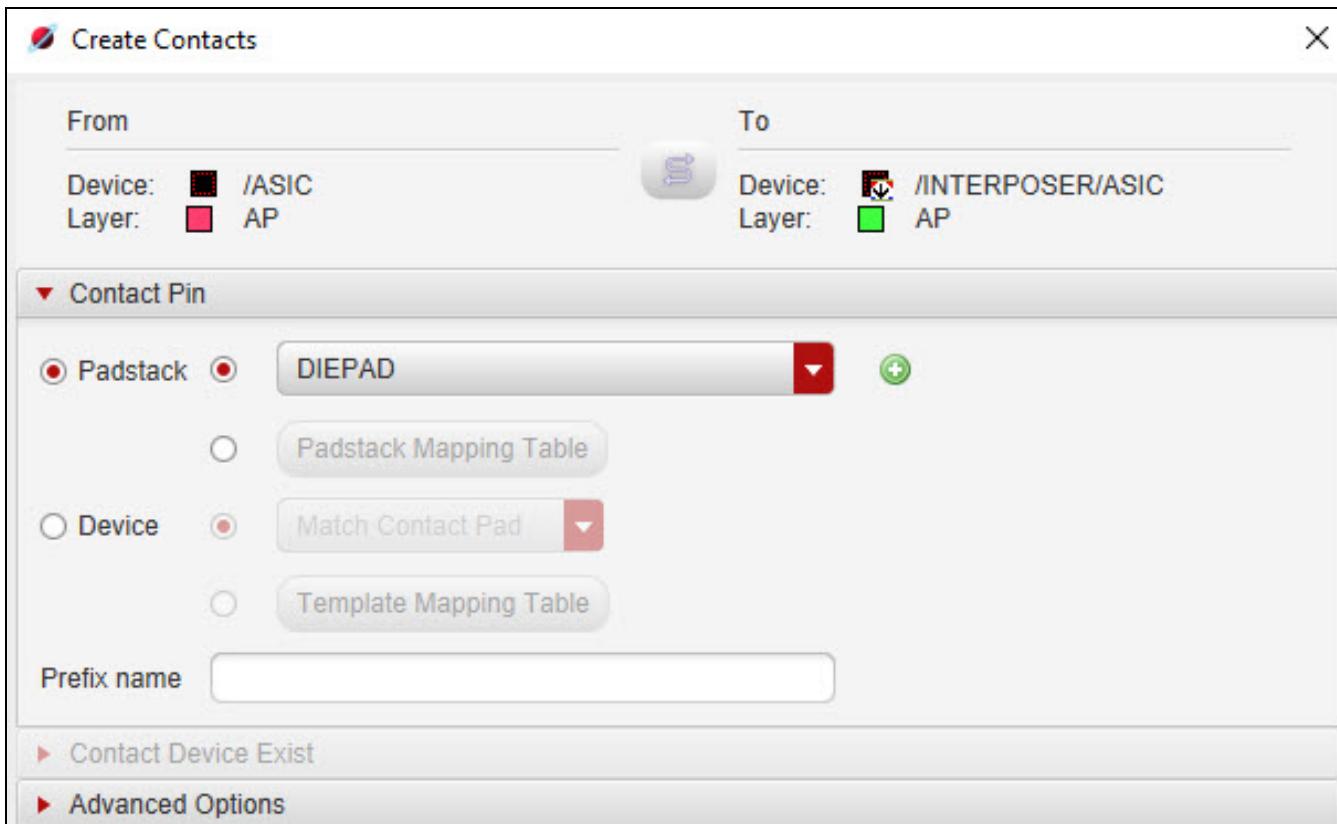
Occasionally, contact pads need to be removed or regenerated. Before regenerating, it is advised to first remove the existing contact pads. To remove contact pads, right-click the appropriate *Contact Device* in the ContactLayer Table Editor and choose *Delete Contacts*, as shown below. It does matter which column the pulldown menu is used from. Typically pads are removed from the *Bottom Device Path*.



If a contact device exists, a prompt is presented to choose between the deleted contact device or its pins.

Generating Contact Pins or a Contact Device

Use the same pulldown menu above to choose *Generate Contacts* to invoke the *Create Contacts* dialog, as shown above.



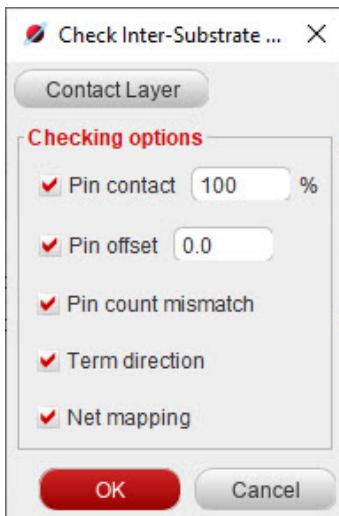
Related Information

[Creating Contact Layers, Contact Pins, and Contact Devices](#)

Running Inter-Substrate Connectivity Checks

The System Planner includes a set of inter-substrate design rule checks aimed at ensuring the system design integrity.

The checks can be performed by selecting *Tools – Check Inter-Substrate Connectivity*. The *Check Inter-Substrate Connectivity* dialog is presented, as shown below.



You can select the rules required to be run as part of the inter-substrate connectivity check in this dialog. For example, select the *Pin contact* check to ensure that the pins are contacting each other on each device. Similarly, select the *Pin offset* option to verify that the contacted pins meet the maximum allowable pin offset limit, which you can set in the field provided.

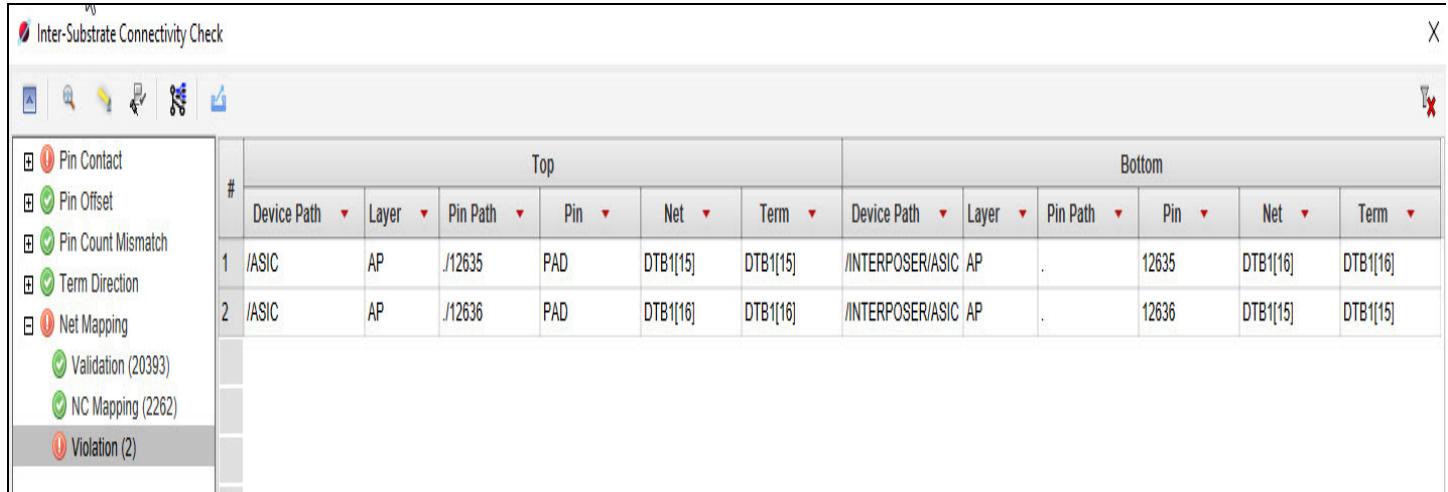
As contact layers are defined, the Inter-Substrate Connectivity Checks are automatically performed. The *Check* column in the ContactLayer Table Editor displays a red or green circle to indicate the status of these checks.

| # | Check | Top | | Bottom | |
|---|-------|-------------|-------|-------------------|-------|
| | | Device Path | Layer | Device Path | Layer |
| 1 | ✓ | /ASIC | AP | /INTERPOSER/ASIC | AP |
| 2 | ✓ | /HBMHI | Bump | /INTERPOSER/HBMHI | AP |
| 3 | ✓ | /HBMLO | Bump | /INTERPOSER/HBMLO | AP |

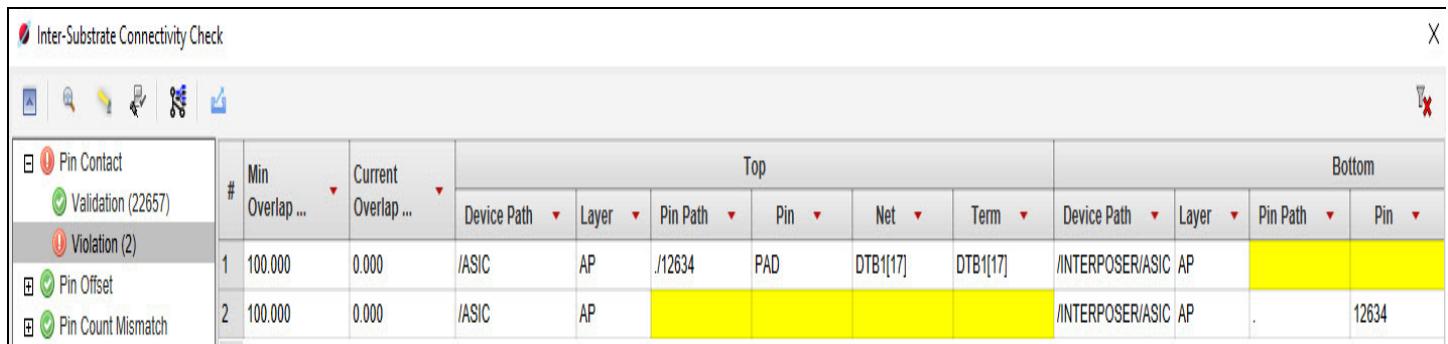
Double-clicking on the circle will present the results in the *Inter-Substrate Connectivity Check* table.

Exploring Results

The Inter-Substrate Connectivity Check chart is presented after the rules have been run, as shown below.



| # | Top | | | | | | Bottom | | | | | |
|---|-------------|-------|----------|-----|----------|----------|---------------------|-------|----------|-------|----------|----------|
| | Device Path | Layer | Pin Path | Pin | Net | Term | Device Path | Layer | Pin Path | Pin | Net | Term |
| 1 | /ASIC | AP | J12635 | PAD | DTB1[15] | DTB1[15] | /INTERPOSER/ASIC AP | | | 12635 | DTB1[16] | DTB1[16] |
| 2 | /ASIC | AP | J12636 | PAD | DTB1[16] | DTB1[16] | /INTERPOSER/ASIC AP | | | 12636 | DTB1[15] | DTB1[15] |



| # | Min Overlap ... | Current Overlap ... | Top | | | | | | Bottom | | | |
|---|-----------------|---------------------|----------|-----|--------|------|-------------|----------|---------------------|-----|--|-------|
| | Device Path | Layer | Pin Path | Pin | Net | Term | Device Path | Layer | Pin Path | Pin | | |
| 1 | 100.000 | 0.000 | /ASIC | AP | J12634 | PAD | DTB1[17] | DTB1[17] | /INTERPOSER/ASIC AP | | | |
| 2 | 100.000 | 0.000 | /ASIC | AP | | | | | /INTERPOSER/ASIC AP | | | 12634 |

Expanding the rule categories on the right and selecting *Violation* will display those violations in the table, as shown above. The violations are often shown highlighted in yellow.

The toolbar presents the following functionality:

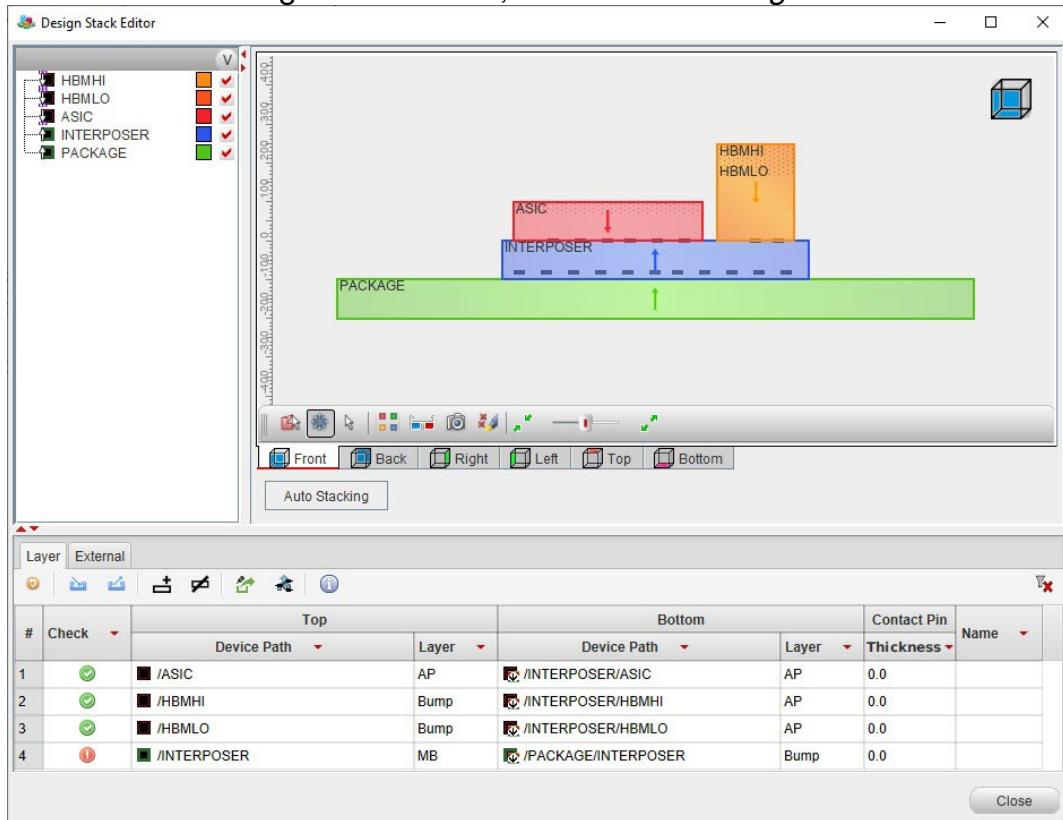
- *Zoom to* will zoom into the canvas to show the design object violating the selected rule.
- *Show me* will highlight the object violating the selected rule on the canvas, often with a dragged connection line.
- *Select* will select the design object that the violation is reporting.
- *Graph net* will open the Graph net dialog with the offending net.
- *Export to CSV* will export the current results to a CSV file.
- *Clear both sort order and filter* will remove all sort order and filters from the results table.

Defining the Design Stackup

The Design Stack Editor is used to visualize the die physical locations in the assembled design.

The design stack is used to configure some of the analysis tools in Integrity 3D-IC. For Allegro implementation, the diagram is merely a visual aid and it does not define relationships between the devices. Contact Layers are displayed with dashed lines, as shown below.

To invoke the Design Stack Editor, select *Edit – Design Stack*.



The Design Stack Editor has two panes:

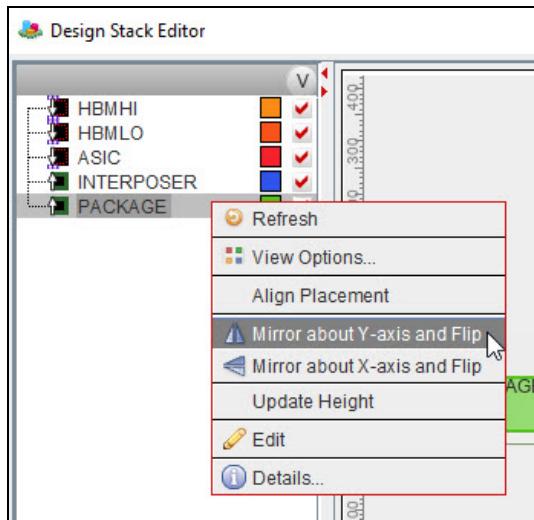
- The upper pane depicts the die physical locations in the design stack as a diagram.
- The lower part displays the Contact Layer Editor.

The default stack interactive behavior mode is *Select*. Click the *Select* icon  to change to *Select* mode.

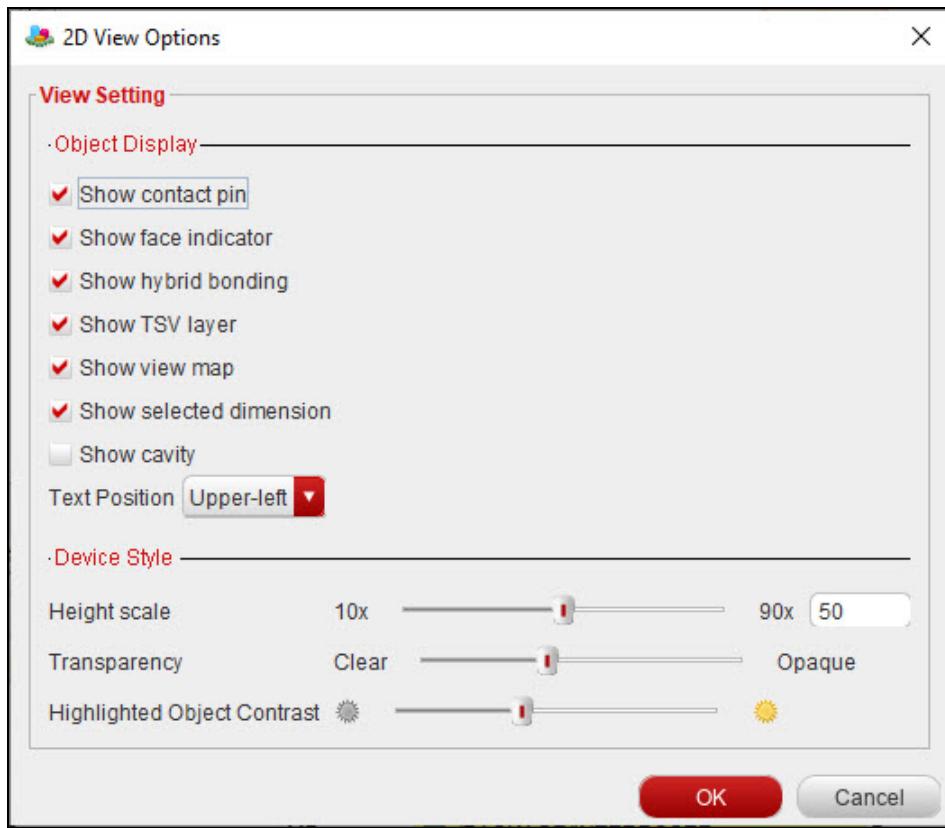
Devices can be moved up and down in the stack by selecting them from the list on the left, clicking the Move mode icon , and dragging them in the diagram.

The height and shape of the devices can be interactively modified by clicking the Shape Editing mode icon . Edges of the devices can then be interactively dragged to change the device shape.

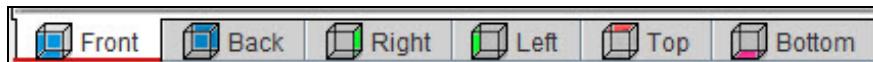
Right-click options enable device modification such as mirroring and height adjustment, as shown below.



The Design Stack Editor *View Options* enable customization of the information displayed, as shown below.



By default, the *Front* perspective is displayed. The view perspective can be changed by clicking any of the other buttons, such as *Back* or *Right*, at the bottom of the pane, as shown below.

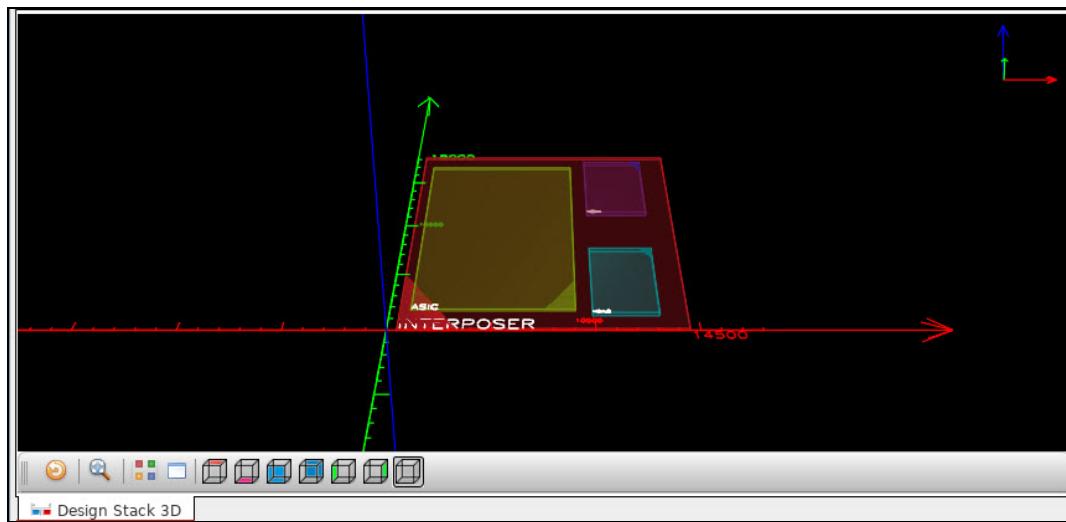


The view zoom level can be adjusted with the slide bar.



The *3D Layout* button launches a 3D representation of the die stack. The view perspective can be interactively manipulated.

The resulting window should be closed if it fails to render the design.



Creating Pin Patterns, Bumps, and TSVs

System Planner has several capabilities to create bumps and bump patterns for use with bottom layer pads, power and ground patterns, and TSVs.

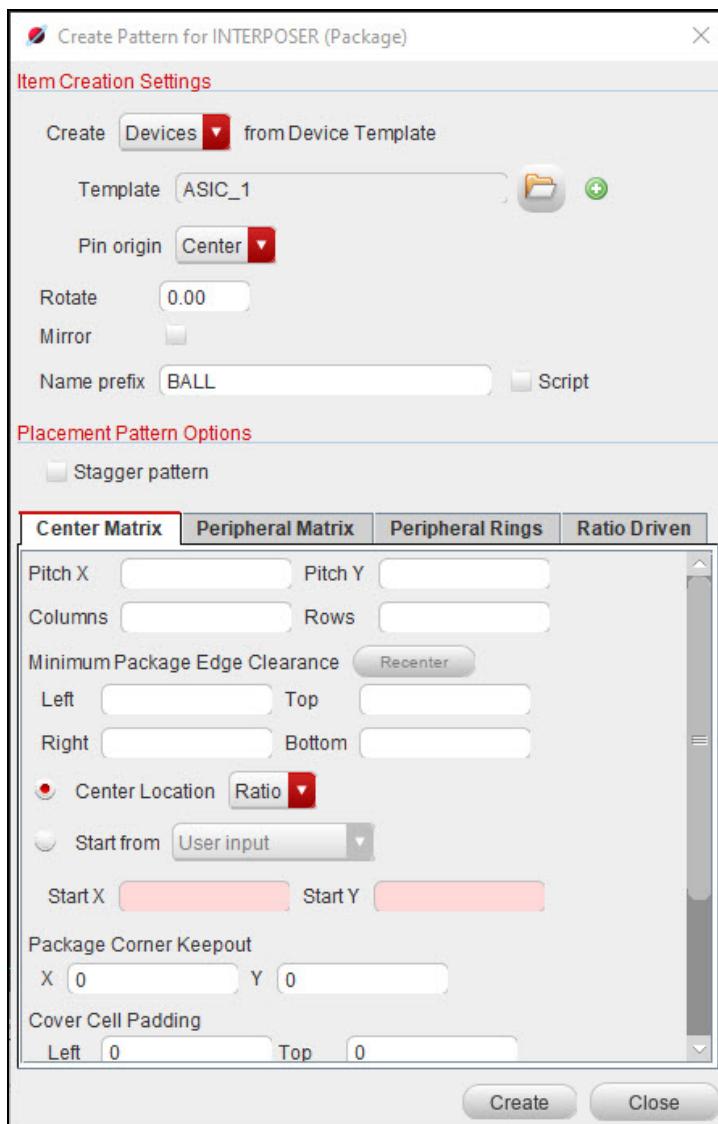
- Create patterns of pins for power and ground matrixes, signals, and TSVs by using the *Create Pin or Device Pattern* command. See [Creating Pin Patterns for Pins, Bumps, and TSVs](#).
- Create bumps on any layer by selecting a group of pins and using the *Bumps for All Layer Shapes* command. See [Using the Bumps for All Layer Shapes Command](#).
- Generate TSVs for a selected set of pins by using the *Fanout Vias* command. See [Creating TSVs Using the Fanout Vias Command](#).

Creating Pin Patterns for Pins, Bumps, and TSVs

Patterns of pins can be created to be used for a variety of reasons. The pattern itself can be created in several styles. Pattern styles ranging from simple matrixes to complex shapes that automatically avoid placed die devices can be generated. Bump device pads for Innovus and Pins style pads for Allegro X layout editors can both be created.

To generate a pin pattern:

1. Select the device in which pins are to be created in the Device Hierarchy and right-click to choose the *Create Pin or Device Pattern* command. The Create Pattern for <selected device> dialog is presented, as shown below.



In this dialog, you can specify various options, including the placement pattern, for generating the pin pattern.

2. Click the *Create* button to generate the pin pattern based on the specified options.
3. Click *Close* to close the dialog.

The pin pattern groups are created under the selected device in the *Device Hierarchy* in a folder called *COVERs*. For more details on *COVERs*, refer to [Working with the COVER Pin Groups](#).

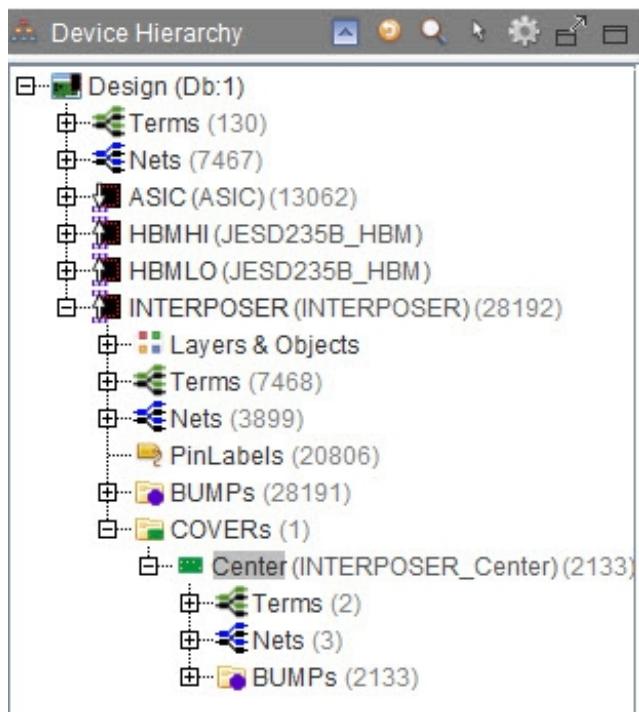
Create Pattern Options

| Option | Description |
|---|---|
| <i>Item Creation Settings</i> | |
| <i>Create</i> | Presents the options to generate <i>Device</i> style pins for Innovus design or <i>Pins</i> for Allegro design. |
| <i>Template</i> | Enables you to select an existing bump device or padstack template. Alternatively, you can click the green icon to create a new one. |
| <i>Pin origin</i> | Specifies the pin origin. |
| <i>Rotate</i> | Rotates the pads by the specified value as they are generated. |
| <i>Mirror</i> | Mirrors pads by Y-axis. |
| <i>Name prefix</i> | Appends the specified name prefix to the bump pad name. The default prefix is <i>Ball</i> . Leave this field blank if you do not want the default prefix. |
| <i>Script</i> | Provides the option to generate the pin names by a script. |
| <i>Placement Pattern Options</i> | |
| <i>Stagger pattern</i> | Generates the pins in a staggered fashion, if selected. |
| <i>Center Matrix</i> | <p>Generates a uniform rectangular pin matrix starting at the center of the device. The group of pins can be moved after they are generated.</p> <p>The <i>Minimum Die Edge Clearance</i> fields define the spacing for voids created around exiting placed die, which is often used for power and ground cover patterns. Entering 0 will ignore placed die and create the pattern under the die.</p> <p>The <i>Count</i> at the bottom shows the number of pins to be generated based on the options entered.</p> |
| <i>Peripheral Matrix</i> | <p>Generates a matrix ring pattern of pads around the periphery of the device. The size of the pattern depends on options entered with pads created from the outside die edges toward the center of the device.</p> <p>The <i>Minimum Die Edge Clearance</i> fields define the gap between the outside edge of the selected die to the first ring of pads created. The values need to be negative values. For example, entries of -50 will create a 50-micron space between the outside edge of the die and the first ring of pads.</p> |

| | |
|-------------------------|---|
| <i>Peripheral Rings</i> | Creates pin rings with varying rotation options for each side of the die. For example, rectangular pads can have different rotation values specified for each side of the die. This is primarily used for wire bonds or package pins and is not recommended for uniform pad geometries. |
| <i>Ratio Driven</i> | This option is not applicable for Integrity 3D-IC. |

Working with the COVER Pin Groups

Each pin pattern group created using the *Create Pin or Device Pattern* command is created as a pin group and listed under the *COVERs* folder in the Device Hierarchy, as shown below with the *Center* pattern group. This example shows a power and ground matrix, which explains why only three *Nets* are shown under it (VSS, VDD, NC).



The pins in a pin pattern group can be manipulated as a group in the Device Hierarchy. You can select a group and move, copy, or rotate it as required. You can also import and export CSV netlist and pins list files to assign the pin nets.

To select all the pins in the group, right-click the group and choose the *Select – All Pins* command. Selecting all pins is often required to feed other commands, such as the *Assign Power and Ground Nets* command.

Pin patterns are created with no connect (NC) pins. Signals can be assigned to the pads using a variety of methods.

- Power and Ground assignment patterns can be assigned using the *Assign Power and Ground Nets* command.
- Individual nets can be assigned to any selected set of pins by using the *Set Selected Pins' Nets* command in the right-click or shortcut menu.
- Nets can be automatically assigned in a route-optimized manner by using the *Connectivity Optimization* command.

Ungrouping Cover Pin Groups

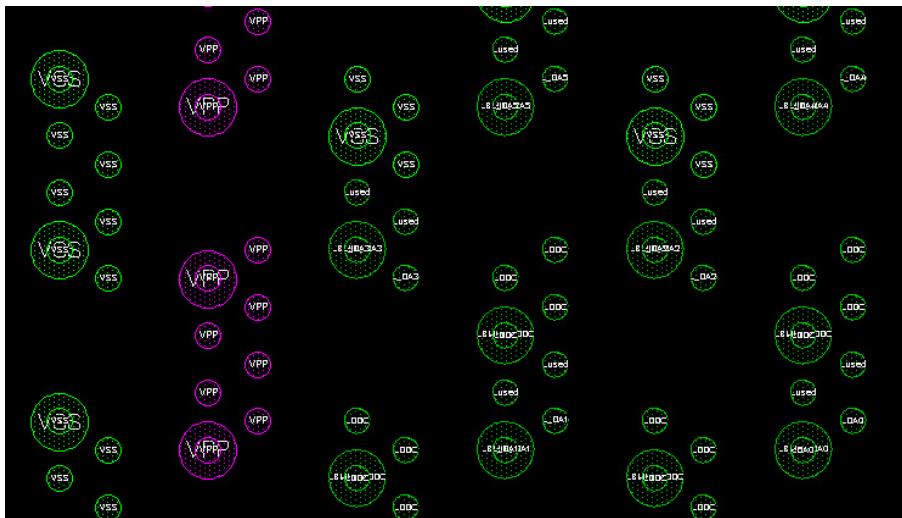
Pin groups listed under the COVERs folder are also located in a COVERs level of hierarchy in the netlist. This is often undesired from a logic netlist perspective. These pin groups can be dissolved and merged into the parent-level netlist. This is recommended after the pins are placed and assigned to nets. To dissolve a pin cover group, right-click the group and choose the *Ungroup* command. Once done, the *COVERs* group will disappear and the pins will be displayed in the parent-level netlist and the *Pins* folder.

Related Information

[Defining Netlist Connectivity](#)

Using the Bumps for All Layer Shapes Command

Bumps can be automatically created on any layer for selected pins. You must create a padstack or bump device for the bump on the target layer before using the *Bumps for All Layer Shapes* command. These bumps are vertically aligned with the selected pins. This command is often used to create larger bottom-side bumps with larger spacing that are aligned with the external signal bumps on the top-side, as shown below.



To use the *Bumps for all Layer Shapes* command:

1. Select a group of pins to generate bumps using any selection method. Adjusting visible layers and using the *Selection Filter* is often useful for this purpose to ensure only the source layer pins are selected.
2. Once the desired pins are selected, right-click on the canvas to choose the *Bumps for All Layer Shapes* command. The *Create Bumps For All Layer Shapes* dialog is displayed.



3. Most of the dialog options are optional and are explained below. Click *Apply* to generate the bump pads based on the specified options.

Create Bumps for All Layer Shapes Form

| Option | Description |
|------------------------------|---|
| <i>Include All Instances</i> | Creates bumps for all pins with the same net. |

| | |
|------------------------------------|---|
| <i>Include Unassigned Pins</i> | Creates bumps for unconnected pins. |
| Source | |
| <i>Source Layer</i> | Identifies the substrate layer for the selected source pins. |
| <i>Shape Size Filter</i> | Filters the shapes on the selected source layer to use for creating bumps. |
| Bump | |
| <i>Bump Type – Device Template</i> | Identifies the bump device to be used for the bumps for IC design. |
| <i>Bump Type – Pad Template</i> | Identifies the padstack to be used for the bumps for package design. |
| <i>Bump Term</i> | Identifies the term of bump device template to be used for connecting the source pin. In most cases, a bump template only has one term. However, some special testcases have multiple terms in a bump template. |
| <i>Rotation</i> | Defines the rotation value for the bump pad. |
| <i>Mirror</i> | Mirrors bumps by Y-axis. |
| <i>Name Prefix</i> | Enables a name prefix to be entered for the bump pad name. |
| <i>Alignment</i> | Defines the alignment between the selected pins and the bump pads. |
| <i>Min Bump Pitch</i> | Defines the minimum spacing between generated bump pads. |

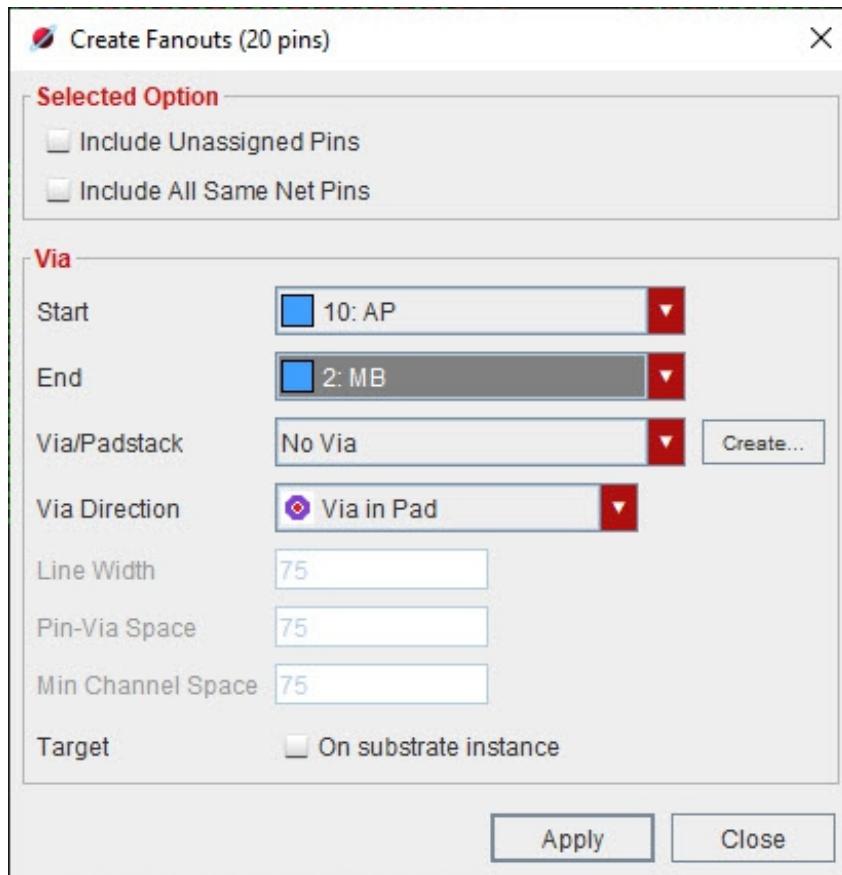
| | |
|-------------------------|--|
| <i>Pitch X</i> | Defines a specific X/Y grid pitch to generate the bump pads. |
| Fanout | |
| <i>Reference</i> | Identifies the starting anchor point to be used for placing bumps. |
| <i>Fanout Algorithm</i> | Defines the direction for the fanout routes. |
| <i>Parameters</i> | Defines a specific X/Y grid pitch to generate the fanouts. |
| Term | |
| <i>None</i> | Does not generate terms for the bump pads. |
| <i>Name Prefix</i> | Defines a name prefix for the bump pad terms. |
| <i>Term List CSV</i> | Defines a term list file to be used for the bump pads. Existing terms are assigned to the newly created bumps. |

Creating TSVs Using the Fanout Vias Command

TSVs can be generated automatically on any selected set of pins. A padstack or bump device can be created beforehand for the TSVs, or it can be created interactively.

To use the Fanout Vias command to create TSVs:

1. Select the desired pins and right-click to choose the *Fanout Vias* command. The Create Fanouts dialog is displayed.



2. Selecting the desired *Start* and *End* layers will display any padstacks that are defined between them in the Via/Padstack pulldown menu.
3. Set the following options in the dialog, as required:

| Option | Description |
|----------------------------------|---|
| <i>Include Unassigned Pins</i> | Generates TSVs on selected pins, including the ones without nets assigned. |
| <i>Include All Same Net Pins</i> | Generates TSVs on all selected pins with the same net name. For example, if 10 VSS pins are selected, a TSV will be created for each pin instead of a single TSV for the net. |
| <i>Start</i> | Specifies the layer from which the TSV should originate. This is typically the layer that has the selected pads. |

| | |
|---|---|
| <i>End</i> | Specifies the layer on which to end the TSV. |
| <i>Via/Padstack</i> | <p>Specifies the padstack to be used. If padstacks exist in the design that match the <i>Start</i> and <i>End</i> layer selections, they will be displayed in the list.</p> <p>To create a new padstack, select the <i>Create</i> button. The <i>Edit Padstack</i> dialog is presented to enter padstack values. Once a new padstack is created, it will be listed in the pull-down menu.</p> |
| <i>Via Direction</i> | Defines the location for the TSVs to be created. This command is also used for other methodologies, including package design. Selecting any option other than <i>Via in Pad</i> will create vias offset from the pad center with routes on the <i>Start</i> layer. The <i>Via in Pad</i> option is recommended for IC design and will create the TSVs in the center of the selected pins. |
| <i>Line Width, Pin-Via Space, Min Channel Space</i> | Define parameters for the fan out routing and via spacing when using options other than <i>Via in Pad</i> . The options are self-explanatory. |
| <i>Target – On substrate instance</i> | If enabled, creates vias in the substrate instance (DIE, PACKAGE, BOARD) template. If not selected, System Planner will create vias in the pin's device template. |

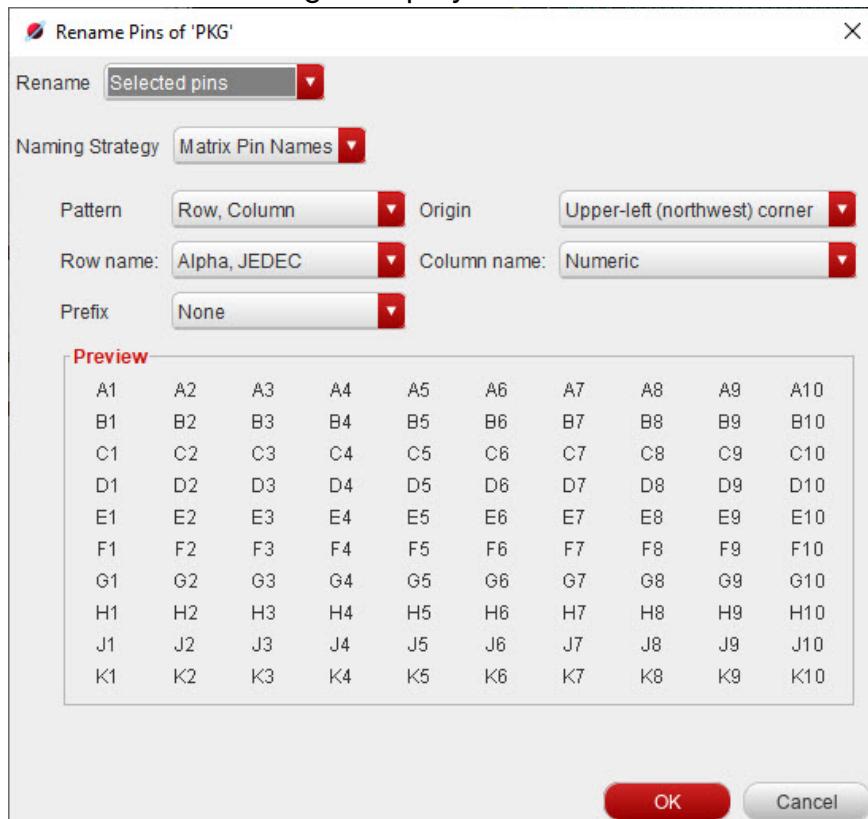
4. Click *Apply* to generate TSVs based on the specified options.

Renumbering Pins

Pins can be renumbered to a uniform naming pattern better facilitate implantation and analysis or to eliminate duplicates.

To renumber pins:

1. Turn off all layers accept the desired pin layer to rename.
2. Set the *Selection Filter* to *Pins* and click the *Select All* button or select the desired pins to rename.
3. Right-click the device in the Device Hierarchy and choose *Device - Rename - Pins*. the *Renames Pins* dialog is displayed.



4. Set the *Rename* option to *Selected pins*.
5. Set the pin rename options as required.
The *Preview* pane shows how the pins will be renamed with the options defined.
6. Click *OK* to rename the pins.

Interfacing with Allegro X Advanced Package Designer

System Planner is well integrated within the Integrity 3D-IC platform for use with Innovus and Allegro® X layout and package editors. Allegro X layout and package editors can be launched from System Planner with the design automatically imported.

The standalone Integrity System Planner is well integrated with the Allegro X Advanced Package Designer (APD) implementation tool. The integration includes importing the System Planner projects into the Allegro X implementation tools. Allegro X designs of type .brd, .sip, and .mcm files can also be imported into System Planner

File transfer interaction is also enabled through a variety of file format import and export capabilities. The following are the typical 3D-IC methods:

- Importing System Planner designs into Allegro X layout and package editors. Refer to [Importing System Planner Designs into Allegro X Package Designer](#).
- Creating an input file usable by \$PRODUCTAPDSHORT to build a package symbol. This is for use in an established \$PRODUCTAPDSHORT flow. System Planner is not used as the netlist source using this method. Refer to [Creating a Die Symbol for APD](#).
- Importing existing \$PRODUCTAPDSHORT designs into the System Planner. Refer to [Importing Devices from Allegro X Designs](#).
- Back Annotating APD Changes to System Planner.

Importing System Planner Designs into APD

System Planner designs can be imported directly into the Allegro Package Designer Plus implementation tool. The netlist, layer stackup, device symbols, and padstacks are all created automatically and the design is ready for constraint definition, implementation and analysis. Active designs can be updated with any changes made in System Planner using the same import command. Updates can be limited to specific devices only.

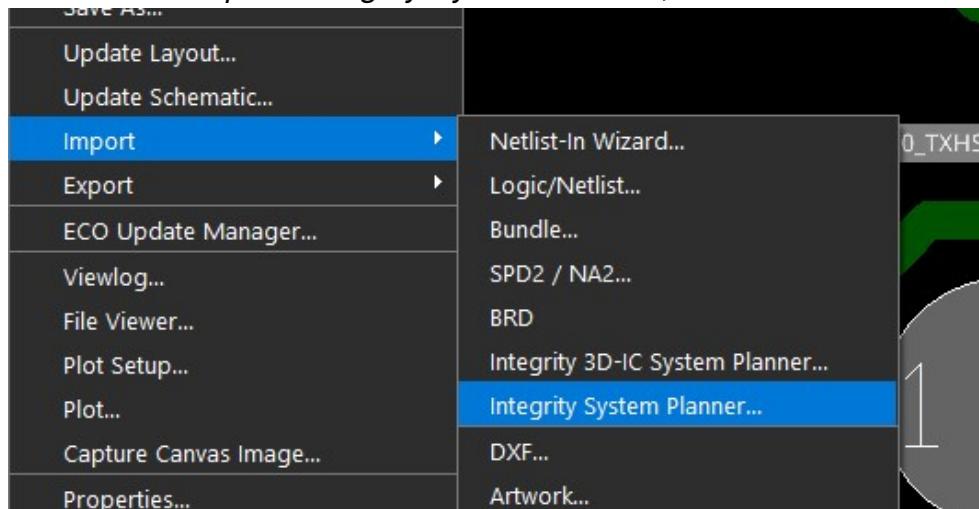
Some users prepare an empty APD seed design with the layer stackup, padstacks, constraints, and so on. That design can be imported into System Planner as a starting point. An option exists during import to use the Allegro padstacks if the names match.

The layer names used in System Planner should match exactly the names used in Allegro in order to import more easily. Layer mapping can be done during import if they differ.

System Planner designs (.oio) can be imported directly into APD.

To import a System Planner design into APD:

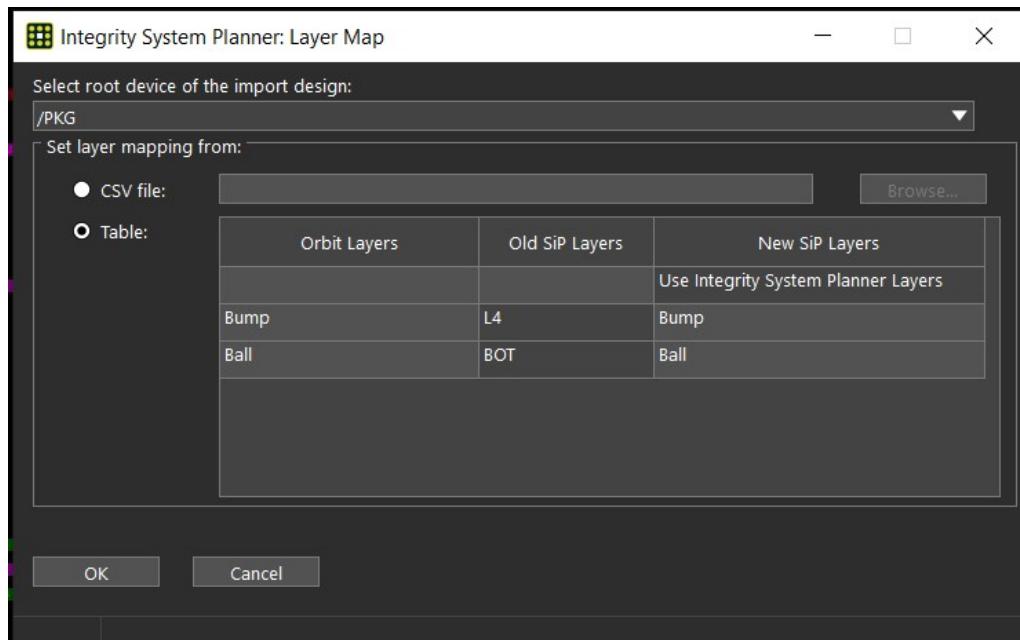
1. Open APD with either an empty design, seed design, or an active design to be updated.
2. Select *File - Import - Integrity System Planner*, as shown below.



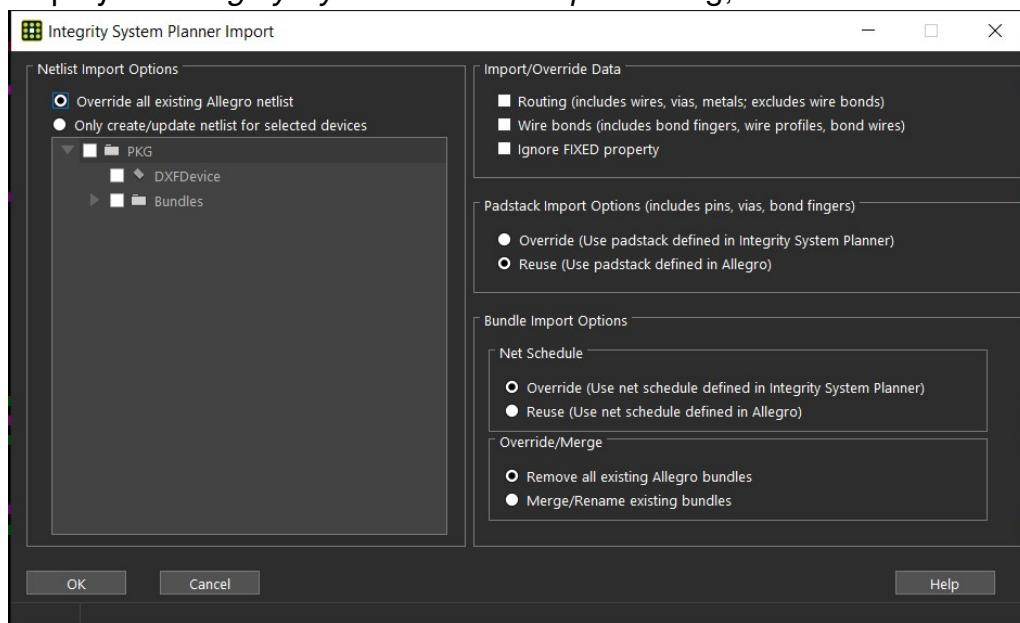
3. Use the browser to select the System Planner .oio file for import.
4. The *Integrity System Planner Layer Map* dialog is displayed, as shown below.

Integrity System Planner User Guide

Interfacing with Allegro X Advanced Package Designer--Importing System Planner Designs into APD



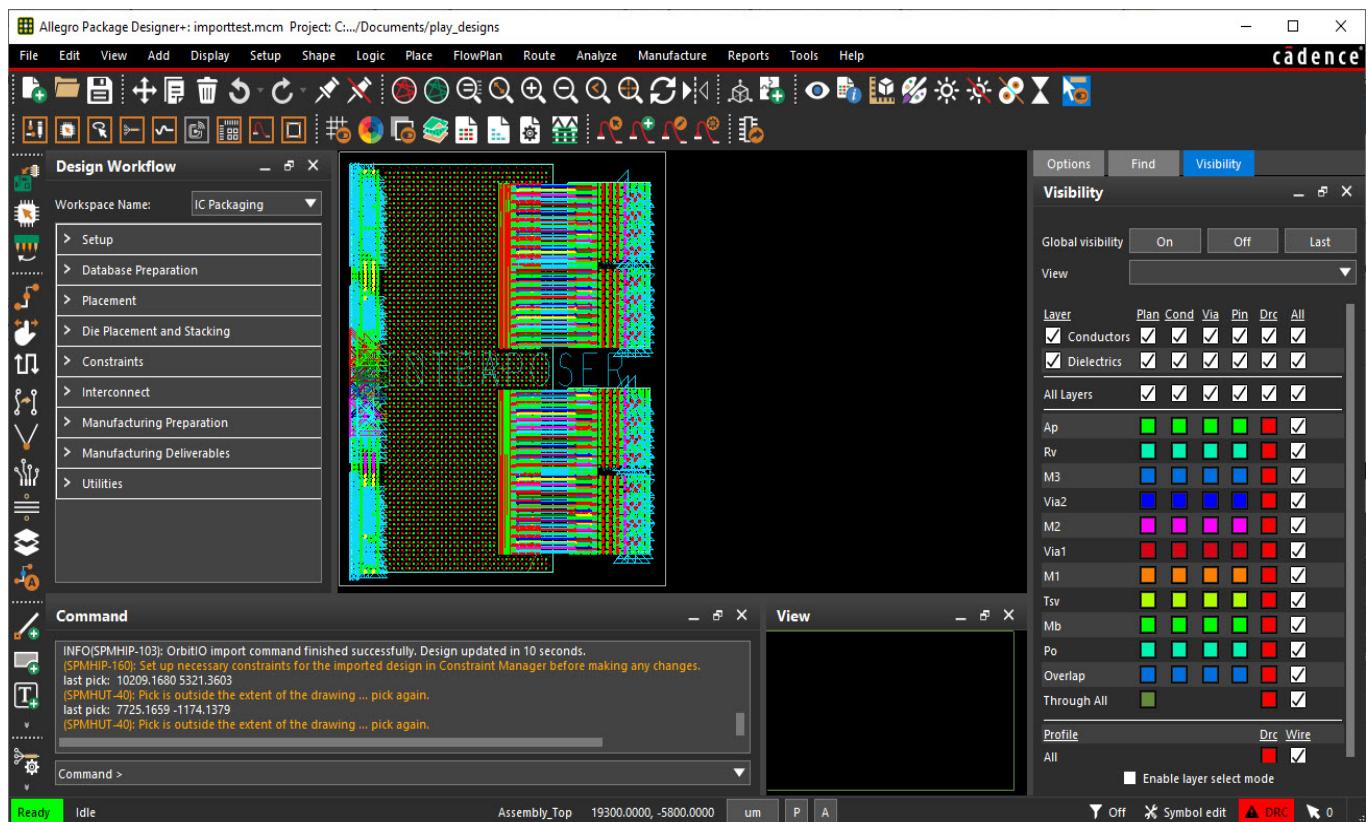
- a. Select the desired import device from the *Select root device of the import design* pulldown menu.
 - b. Verify the layers are mapped properly in the Table. Make any adjustments needed. Layers names that match are automatically mapped.
5. Once the root device has been selected and the layers verified in the *Table*, click *OK* to display the *Integrity System Planner Import* dialog, as shown below.



The *Integrity System Planner Import* dialog has the following options:

| Option | Description |
|--|---|
| Netlist Import Options | |
| <i>Override all existing Allegro netlist</i> | Imports the entire design. |
| <i>Only create/update netlist for selected devices</i> | Imports the symbols and netlist only for selected devices. |
| Import/Override Data | |
| <i>Routing (includes wires, vias metals; excludes wire bonds)</i> | Imports routing, vias, and metals from the System Planner design. |
| <i>Wire bonds (includes bond fingers, wire profiles, bond wires)</i> | Imports wire bond objects from the System Planner design. |
| <i>Ignore FIXED property</i> | Overrides the Allegro FIXED property during update. |
| Padstack Import Options (includes pins, vias, fingers) | |
| <i>Override (Use padstack defined in Integrity System Planner)</i> | Uses the padstacks defined in the System Planner. |
| <i>Reuse (Use net schedule defined in Allegro)</i> | Uses the padstacks defined in the Allegro design. Padstacks with matching names are replaced with the Allegro padstack. |
| <i>Bundle Import Options</i> | Defines bundle behavior during import. The options are self explanatory. |

6. Click *OK* to import the System Planner design into Allegro with the options selected. This may take a few minutes depending on the design size and options selected. The layer stackup, padstack, device symbols and design are displayed, as shown below.



Creating a Die Symbol for APD

You can export a die text file from the System Planner in order to build a die or interposer symbol for a package design that is not using the netlist from the System Planner. Some package design teams have established flows and may not be importing the System Planner design as the netlist source. They only need to create symbols for the design. The process involves creating a "fake" package device that is the same size as the die or interposer, creating contact pins on it, and then exporting the die text file.

- [Creating a Package Device](#)
- [Creating Package Substrate Contact Pads](#)
- [Exporting a Die Text File for the Die Symbol Creation](#)

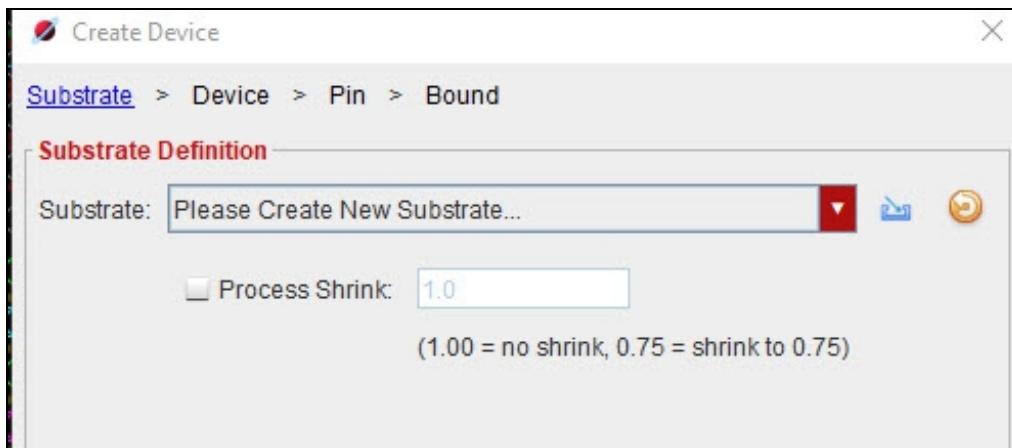
Creating a Package Device

The first step in the process of creating die symbols is to create an empty package device in the design.

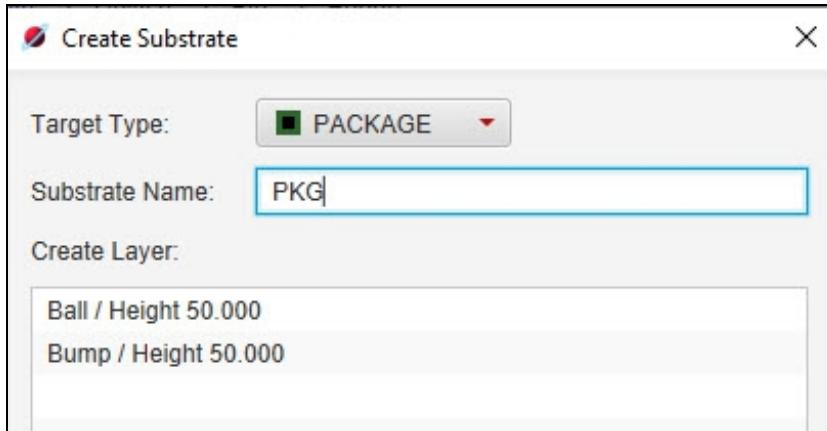
A simple default substrate layer stack is created since you are only concerned with the top-layer to generate the contact bumps. If you have a substrate for the package in a CSV or LEF tech file, it can also be imported and used but is not really needed.

To create an empty package device:

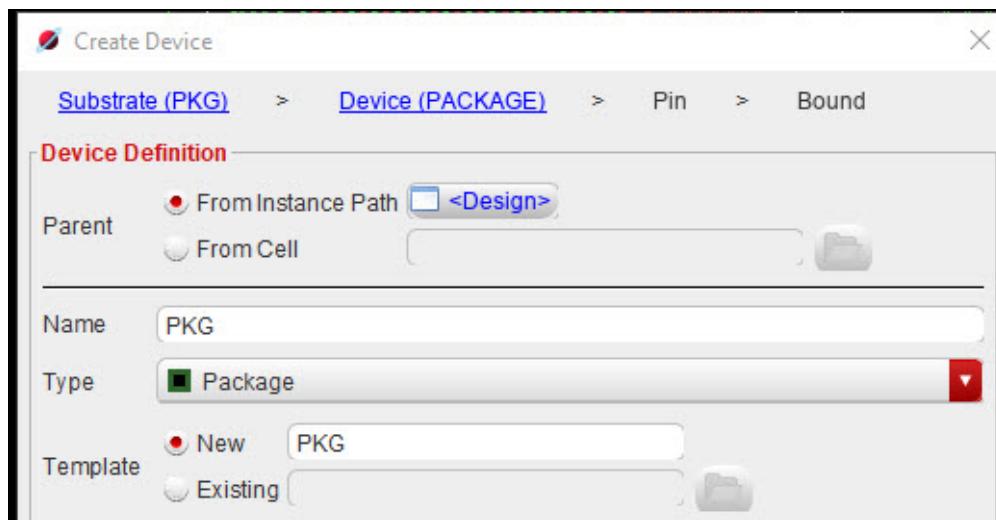
1. Select *Edit – Devices – Create Device* from the menu bar to invoke the *Create Device* dialog, as shown below.



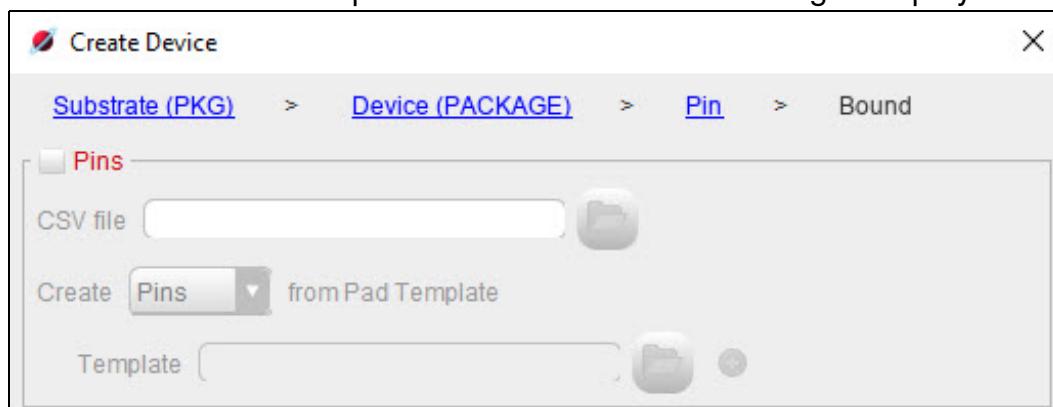
2. If a substrate for the package device has been previously imported or created, select it from the *Substrate* pull-down menu. Else, to create a new package substrate, click the *Create Substrate* icon (blue folder icon) and select *New*. The *Create Substrate* dialog is displayed.



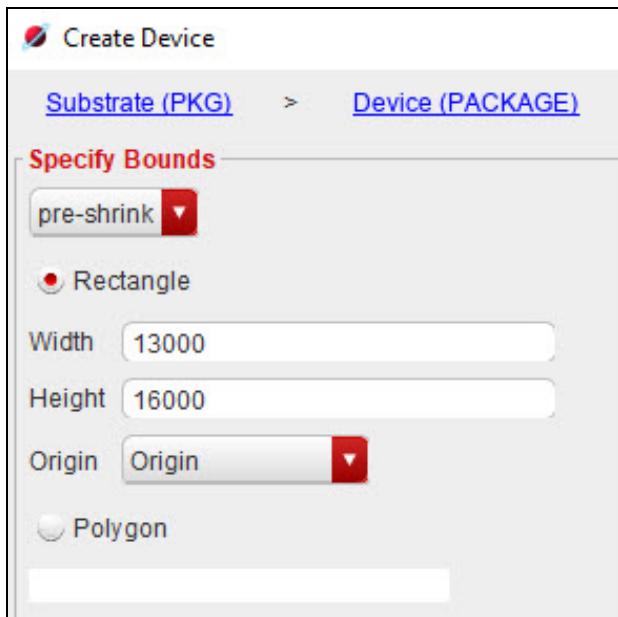
3. Select *PACKAGE* from the *Target Type* pull-down menu and enter a package device name in the *Substrate Name* field. Click *OK* to create the package substrate with the default layer stackup.
4. Notice the package substrate you created is now available in the *Substrate* pull-down menu in the *Create Device* dialog. Select the new *PKG* substrate from the *Substrate* pull-down menu and click *Next*. The Device pane of the dialog is displayed.



5. Select *Package* from the *Type* pull-down menu, enter names in the *Template* and *Name* fields and click *Next*. The Pin pane of the Create Device dialog is displayed.

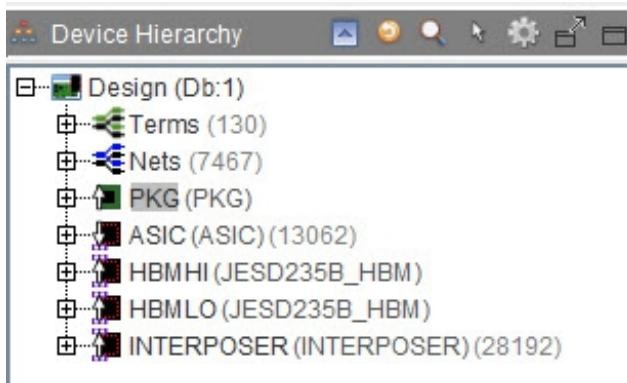


6. The Pin pane enables pins to be entered from a CSV pins file and pad geometries to be defined. For example, this is where package balls could be imported on the *Ball* layer. Since we are creating an empty package, click *Next* to continue.
7. In the Bound pane of the Create Device dialog, you can define the package size. Enter values in the *Width* and *Height* fields. The desired *Origin* location can also be selected.



For the proper die size and pin coordinates to be output in the die text file, enter the same size and origin of the die or interposer device. It also may not be important to use a realistic size, if not known. The output pin files for the APD symbols will use X-Y coordinates based on the origin defined for the "fake" package. If using a center origin, the device size can be changed easily by the package designer without affecting the pin coordinates.

8. Click *Create* to create the package device and click *Close*. The new package device is now displayed under the top-level design in the Device Hierarchy, as shown below.



The die or interposer should be the same size and in the same location as the newly created package device. If not, move or resize it to align.

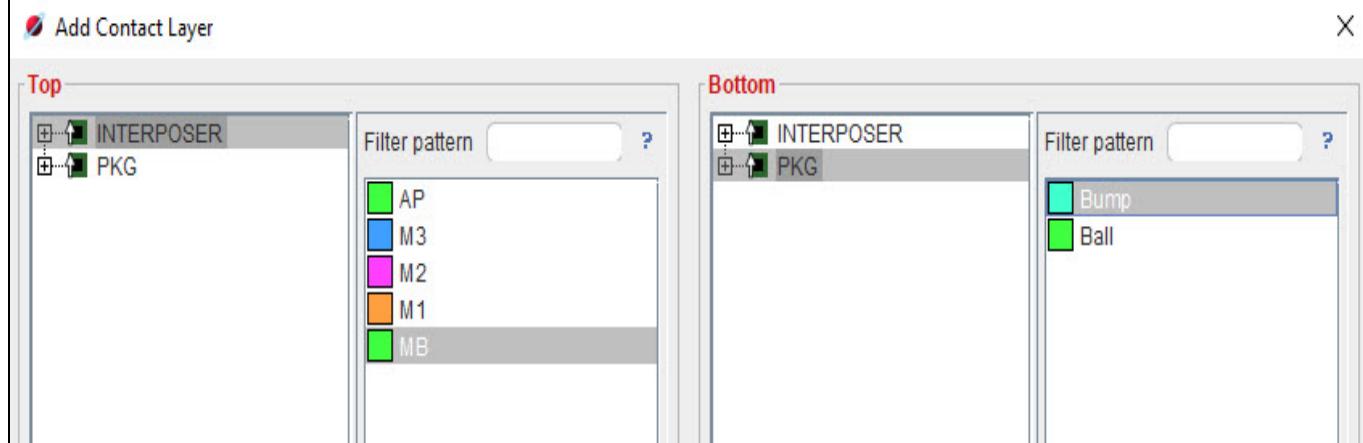
Creating Package Substrate Contact Pads

After aligning the package device origin with the die device, the next step is to generate contact pads on the package substrate to match the die locations. This is done by creating a contact layer between the two devices. As a part of creating the contact layer, contact pads can also be generated.

A padstack can optionally be created for the PKG die contact pins and to be used in the exported die text file.

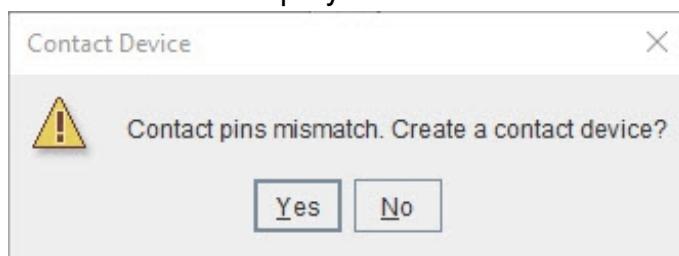
To create the contact layer and pads:

1. Select *Edit – Contact Layer* from the menu bar. The Contact Layer Editor is presented.
2. To create the new package to die contact later, click the *Add Contact Layer* icon . The *Add Contact Layer* dialog is displayed.

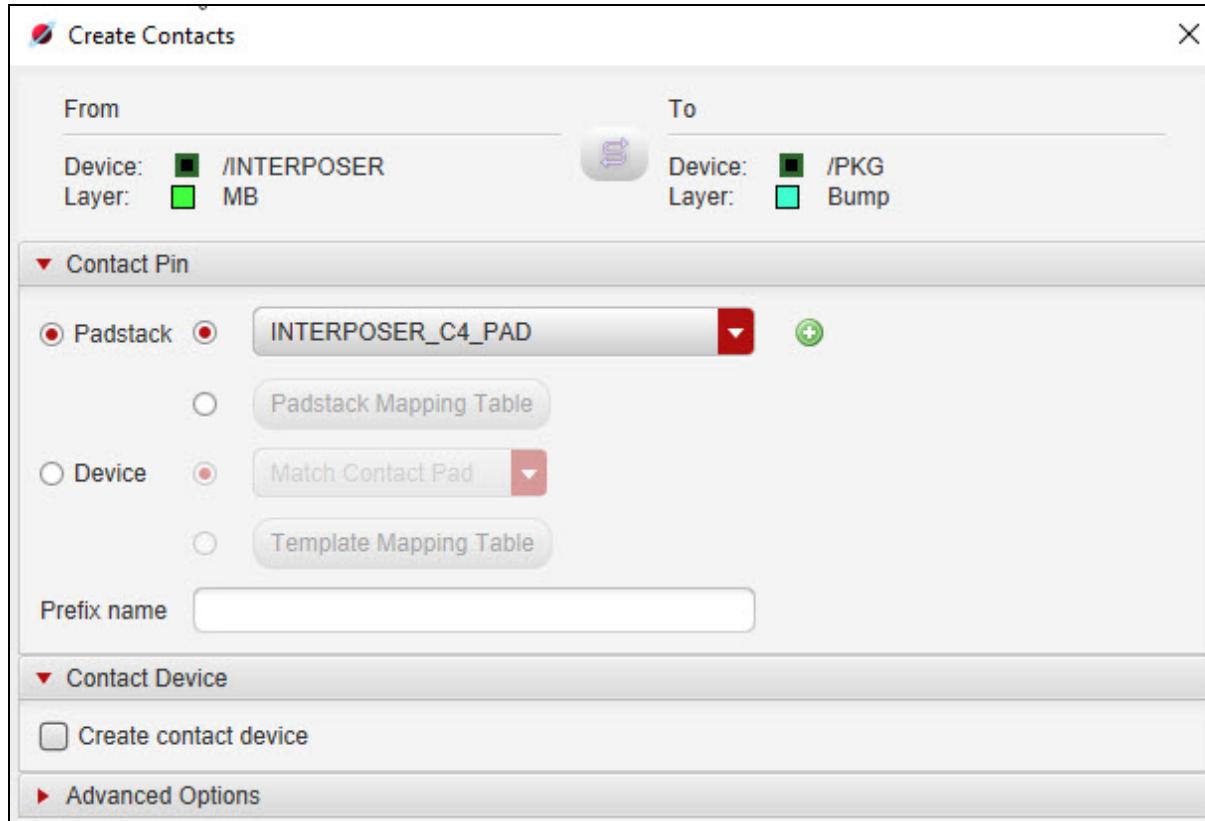


3. The Add Contact Layer dialog is used to establish contact relationships between two devices. To create a new contact layer, select the source device and existing pad layer from the *Top* column and the destination device and layer from the *Bottom* column. In the example above, *INTERPOSER* and *MB* are selected in the *Top* column and *PKG* and *Bump* are selected from the *Bottom* column. This will establish the physical relationship and contact layers between the two devices.

4. Click *Add*. This displays the Contact Device dialog.



5. Click *Yes* to create the contact pads on the package substrate. Clicking *No* will establish the contact relationship, but not generate contact pads. On clicking *Yes*, the Generate Contact Device dialog is displayed, as shown below.



6. Select a *Padstack* or select *Match Contact Pad* and then click *OK* to generate the contact pads on the package substrate.
7. Close the *Add Contact Layer* dialog.

In the *Layers* view in the View Options panel, right-click anywhere and choose the *All Layers off* command, and then select the package substrate layer used to generate the contact pads. In this example, it is the *Bump* Layer. Notice the pads are displayed on the package substrate layer.

Optionally, you can turn on net names by clicking the *Toggle display of pin net names* icon  on the View Options panel and zoom in to view the nets assigned to the pads.

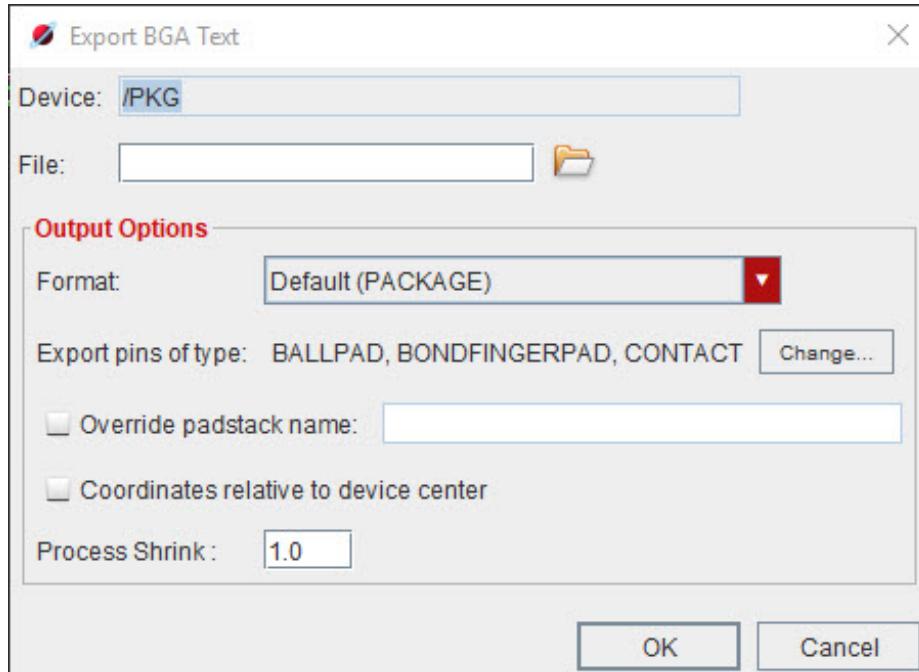
Optionally, the package pins can be renumbered using a variety of renaming options. For information, see [Renumbering pins](#).

Exporting a Die Text File for the Die Symbol Creation

The desired file format for PAD symbol creation is a die text file. The CSV pins file format can also be used, but it does not include the device outline.

To generate the die text file for the die pins:

1. Select the package device in the Device Hierarchy and right-click to choose *Export – BGA Text*. The Export BGA Text dialog is displayed.



2. Confirm or update the various option settings in the dialog, as required:

| Option | Description |
|---------------|--|
| <i>Device</i> | Displays the device from which to extract the pin locations. It should display the package device. |
| <i>File</i> | Specifies the name and location of the exported text file. |

| | |
|--|---|
| <i>Format</i> | <p>Specifies the output format for the text file. Choose the desired format from the pull-down menu, as shown below.</p> <p>The options are self-explanatory. Files can be exported with the various options to explore the formats. The <i>Default (PACKAGE)</i> option can be used for APD symbol creation.</p> |
| <i>Export pins of type</i> | <p>Lists the type of pins to be included in the file. The <i>Device</i> type pins that were created earlier for the package contact pads require BUMPPAD to also be included. Click the <i>Change</i> button and select BUMPPAD from the list.</p> |
| <i>Override padstack name</i> | <p>Enables an alternate padstack name to be used in the file.</p> |
| <i>Coordinates relative to device center</i> | <p>Produces X Y coordinates relative to the package device center.</p> |
| <i>Process Shrink</i> | <p>Adjusts coordinates based on the shrink value entered.</p> |

3. Click *OK* to create the file based on the options selected.

An example of the *Default* format die text file is displayed below.

```
File: C:\Users\Documents\interposer_die_text.txt
Date: Thu Jan 14 15:30:19 MST 2021
Units: microns, 3 decimal places
Name: PKG
RefDes: PKG
Origin: (0.0 0.0)
Rotation: 0.000000
Pad Layer: Bump
Extents: ((0.000000 0.000000) (25000.000000 25000.000000))
```

| Pin Number | Padstack | X Coord | Y Coord | Rotation | Net Name |
|------------|----------|-----------|-----------|----------|---------------|
| C4S_11 | C4_PAD | 1029.9995 | 13659.001 | 0 | ASIC_DTB1[24] |
| C4S_110 | C4_PAD | 1398.0 | 5261.5005 | 0 | ASIC_TAP_TCK |
| C4S_111 | C4_PAD | 1078.0005 | 3666.5005 | 0 | ASIC_DTB2[15] |
| C4S_112 | C4_PAD | 1238.0005 | 3666.5005 | 0 | ASIC_DTB2[16] |
| C4S_113 | C4_PAD | 1398.0005 | 3666.5005 | 0 | ASIC_DTB2[17] |
| C4S_114 | C4_PAD | 1078.0005 | 3506.5005 | 0 | ASIC_DTB2[12] |
| C4S_115 | C4_PAD | 1238.0005 | 3506.5005 | 0 | ASIC_DTB2[13] |
| C4S_116 | C4_PAD | 1398.0005 | 3506.5005 | 0 | ASIC_DTB2[14] |
| C4S_117 | C4_PAD | 1078.0005 | 3346.5005 | 0 | ASIC_DTB2[9] |
| C4S_118 | C4_PAD | 1238.0005 | 3346.5005 | 0 | ASIC_DTB2[10] |

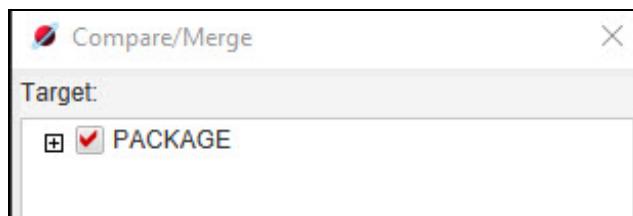
The die text file can then be used in APD to generate the symbols needed for the package design. The pads will include the net names needed for netlist creation.

Back Annotating APD Changes to System Planner

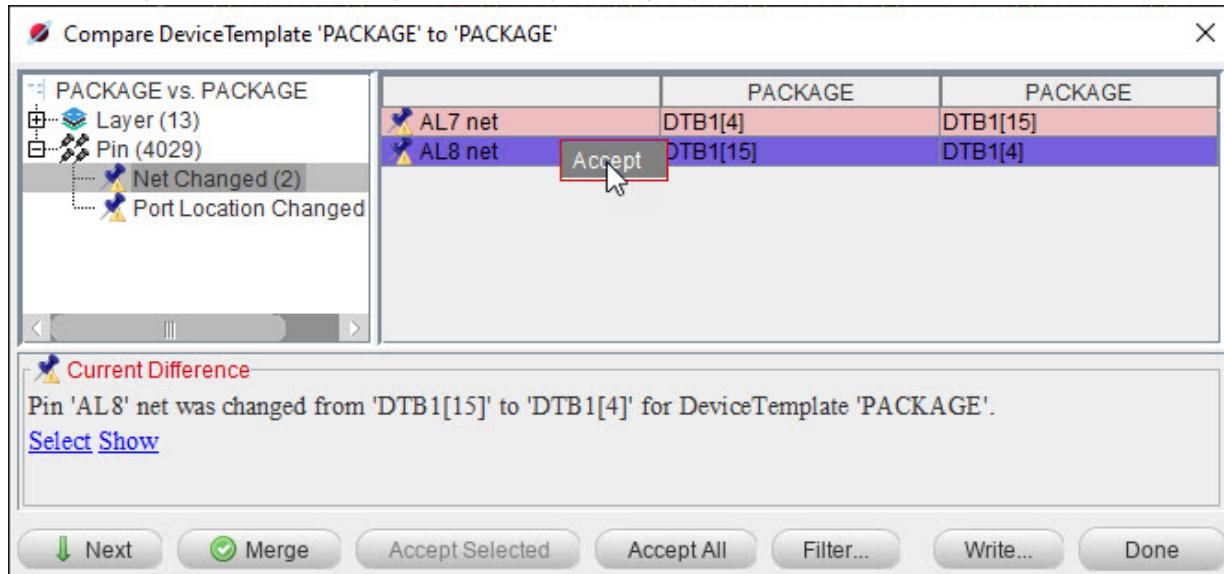
Package designers often need the flexibility to perform minor design changes such as pin swapping or component placement. Discrete capacitor component placement is often dictated by power plane locations and/or analysis results. These changes need to be back annotated to the System Planner to keep the designs in sync.

To back annotate APD changes to the System Planner:

1. Right-click the device in the Device Hierarchy and choose *Device - Merge Updated BRD, SIP, or MCM*.
2. Select the updated Allegro file and click *OK*. This will import the new Allegro design to prepare to compare it to the original System Planner design.
3. Select the *Target* and click *OK* to perform the comparison, as shown below.



4. The Compare Device Template dialog is displayed.



5. Any discrepancies between the two designs are displayed in the Compare Device Template dialog. It is common to see Layer differences since Allegro can add additional layers such as dielectric or soldermask that are not needed or used in System Planner. Pin swapping changes are shown under *Net Changed*. Part placement changes are listed under *Location*.
6. Click on each line to display the *Current Difference* information, as shown above.
7. *Accept* changes either individually by right-clicking the line as shown above, or by clicking *Accept Selected* or *Accept All*.
8. Click *Done* to finish the command and remove the temporary newly imported device.

Exporting Common Design Files

System Planner enables the export of a variety of file formats to exchange data with various layout tools. These files can be exported for specific devices in the design by right-clicking the device in the Device Hierarchy and selecting the desired format from the list shown in the *Export* submenu.

The following are some of the frequently used file formats:

- [Exporting a CSV Pins File](#)
- [Exporting a CSV Devices File](#)
- [Exporting LEF or DEF Files](#)
- [Exporting Verilog Files](#)
- [Exporting a Die Text File](#)

Exporting a CSV Pins File

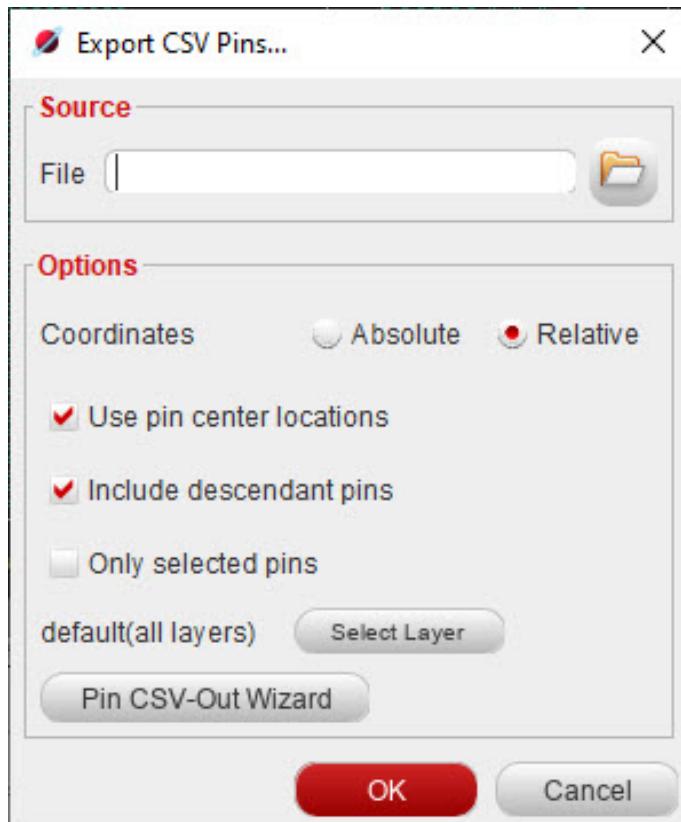
The CSV pins list file is often used to create new devices or to import and export pin or bump locations for a device. It is often used to assign or reassign nets to pins. It can be used to overwrite all existing pins on the device or to merge new assignments with existing pin assignments.

A CSV pins option is presented during the Create Device command to enable pin definition while creating a device.

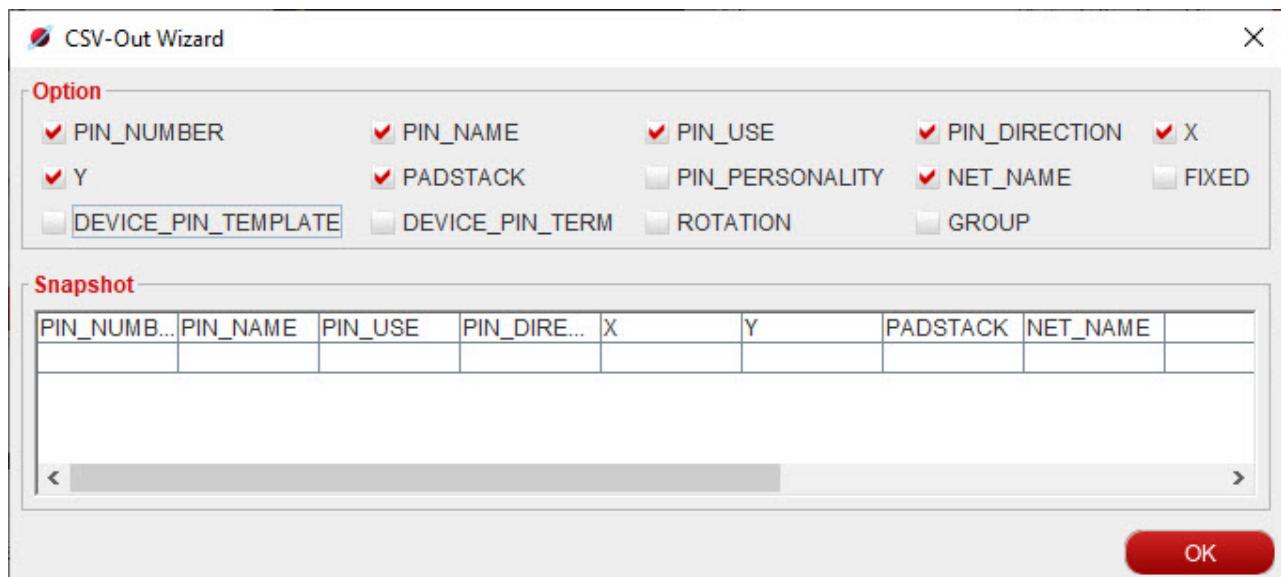
The CSV Pins and Die CSV are the same file format. The Die terminology is used in the Import menu only.

To export a CSV pins list file:

1. Right-click the device in the Device Hierarchy and choose *Export – CSV Pins*. The *Export CSV Pins* dialog is displayed.



2. Set the Export CSV Pins options below and click *OK* to export the file.
 - a. *File* - Define and export CSV pins file name and location.
 - b. In the *Options* section:
 - a. Set the *Coordinate* to base the pin coordinates as *Relative* to the device origin or *Absolute* to use design origin.
 - b. Select *Use pin center locations* to use the pin coordinates as the center of the pin with no origin offset.
 - c. Select *Include descendant pins* to include pins from child levels of device hierarchy.
 - d. Select *Only selected pins* to only export a pre-selected group of pins.
 - e. Click the *Select Layer* button to only export pins defined on specified layers. A dialog is displayed to enable layer selection.
 - c. Clicking the *Pin CSV-Out Wizard* button enables the content of the file to be defined, as shown below.



The format of the file to be created is displayed in the *Snapshot* area.

Options that are selected in the Wizard will remain as the user's default going forward.

Clicking *OK* in the *Export CSV Pins* dialog will create the file based on the options selected.

Exporting a CSV Devices File

The CSV devices file is used to save the placement locations for the devices exported. It can be imported to replicate the placement later in other designs. All devices under the selected device or level of the hierarchy will be included in the file.

During import, if devices exist in the file that are not in the design, they will be created in the design. This feature is helpful for replicating devices such as capacitors in the design and specific locations with specific names.

To create a CSV devices file:

1. Right-click the required device or level of hierarchy in the Device Hierarchy and choose *Export – CSV Devices*. The *Export CSV Devices* dialog is displayed, as shown below.

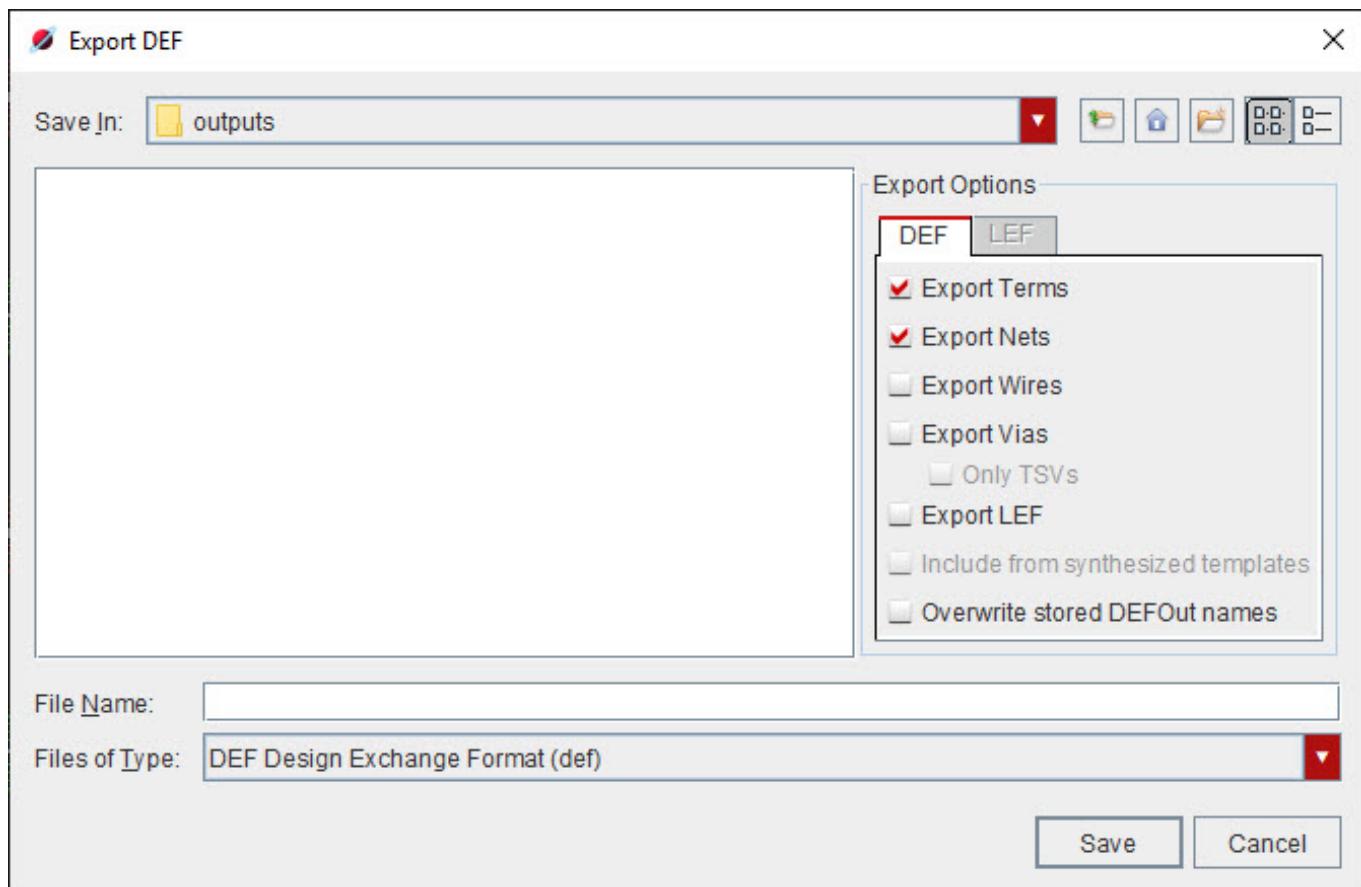


2. The *Export CSV Devices* dialog provides the following options:
 - a. *File* - Enter an output file name and location.
 - b. *Placement Origin* - set the desired origin to be used for the X Y coordinates in the file.
 - c. *Only select devices* - Only export the pre-selected devices in the CSV Devices file.
3. Click *OK* to export the CSV Devices file.

Exporting LEF/DEF Files

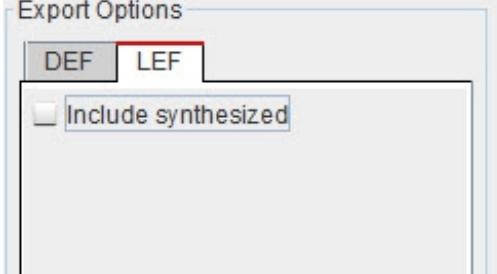
To export LEF or DEF format files:

1. Right-click the device in the Device Hierarchy and choose *Export – DEF*. The *Export DEF* dialog is displayed, as shown below.



2. Confirm or update the following option settings in the dialog, as required:

| Option | Description |
|-----------------------|---|
| <i>Save In</i> | Allows you to browse to the directory to which you want to write the exported files. The large pane below displays the files and directories in the specified <i>Save In</i> directory. |
| <i>File Name</i> | Specifies a file name for the exported file. |
| <i>Files of type</i> | Defaults to the DEF format to export the file in the proper format. |
| Export Options | |
| <i>Export Terms</i> | Includes terms in the exported DEF file. |
| <i>Export Nets</i> | Includes nets in the exported DEF file. |

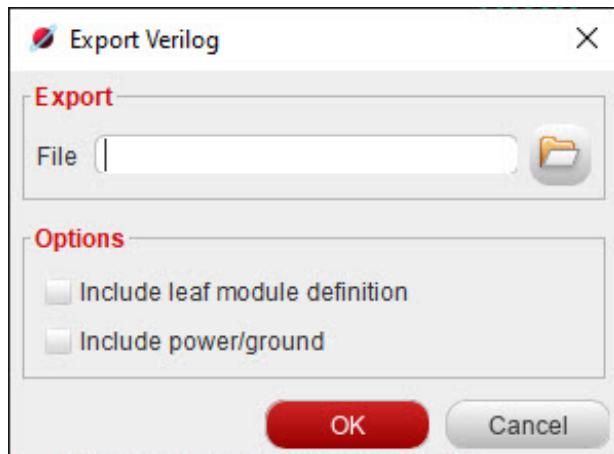
| | |
|---|---|
| <i>Export Wires</i> | Includes wires in the exported DEF file. |
| <i>Export Vias</i> | Includes vias in the exported DEF file. Select the <i>Only TSVs</i> option to filter vias to only TSVs. |
| <i>Export LEF</i> | Exports LEF format files as well for the tech file layer stackup, pads used on the selected device substrate, and other LEF design data. Selecting this option enables the LEF tab to be selected. The LEF tab includes the <i>Include synthesized</i> option, as displayed below.  |
| <i>Include from synthesized templates</i> | Includes devices of synthesized templates (non-LEF Macro) in the exported DEF file. |
| <i>Overwrite stored DEFOut names</i> | Overwrites DEFOut names previously stored in the design with new names; otherwise, existing names will be exported to the DEF file. |

3. Click *Save* to create the LEF/DEF files based on the options selected.

Exporting Verilog Files

To export Verilog format files:

1. Right-click the device or the top-level of hierarchy in the Device Hierarchy and choose *Export – Verilog*. The *Export Verilog* dialog is displayed, as shown below.



2. The *Export Verilog* dialog provides the following options:
 - a. *File*: Enter a Verilog file name prefix and location. Files will be named with the prefix defined appended to the top-level design names and device names.
 - b. *Include leaf module definition*: Outputs separate Verilog files for each leaf module in the design.
 - c. *Include power/ground*: Includes power and ground nets in the Verilog files.
3. Click *OK* to create the files based on the options selected.

A top-level Verilog file is created along with module level files for each device.

The content of the Verilog depends on the data defined in the design. For Innovus, a term-based netlist must first be defined prior to exporting the Verilog netlist.

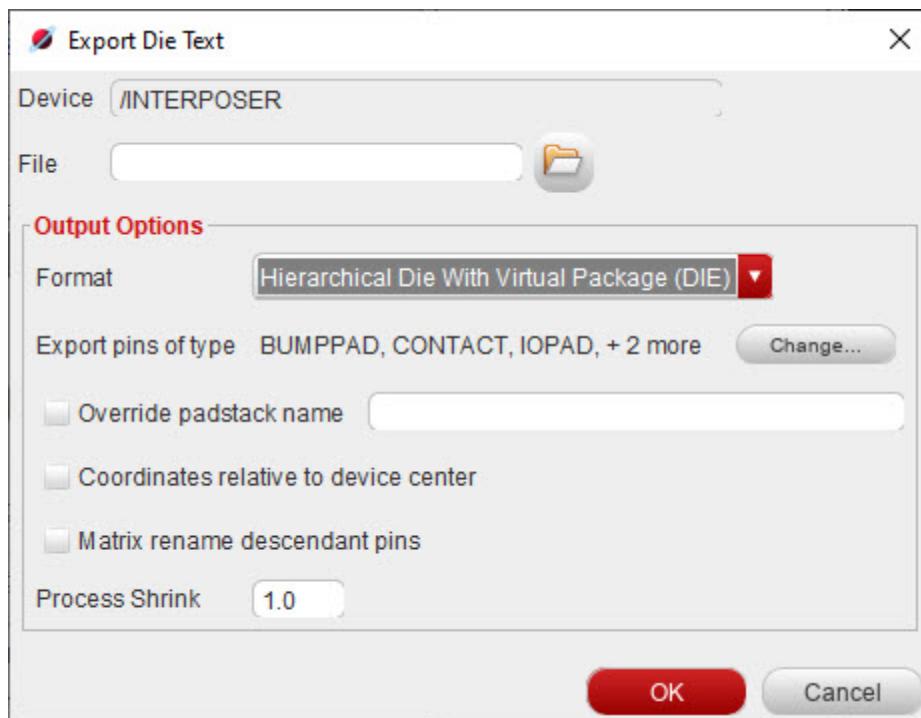
Exporting a Die Text File

Die text files are often used to create devices in both System Planner and APD. They can be configured to contain various types of data. One advantage of using the Die text files is that it can contain additional information such as die size, die origin, pad layers, and so on.

Die text files can only be exported for devices of Type - Die. Devices such as packages can be temporarily set to Die in order to create the file and then set back to package afterward.

To export a die text file:

1. Right-click the device in the Device Hierarchy and choose *Export - Die Text*. The Export Die Text dialog is displayed, as shown below.



2. The selected device is displayed in the *Device* field. This field is not editable.
3. The *Export Die Text* dialog provides the following options:
 - a. *File*: Enter the exported file name and location.
 - b. *Format*: Select the file format desired from the pulldown menu.
 - c. *Export pins of type*: Click the *Change* button to select the types of System Planner pins to include in the exported file.
 - d. *Override padstack name*: Enter a new padstack name to include in the exported file.
 - e. *Coordinates relative to the device center*: Ignore the device origin and use the device centers to define the pin coordinates.
 - f. *Matrix rename descendant pins*: Rename pins for descendant pins when multiple child device copies are used. The pin names and numbers need to be unqualified in order to avoid duplicates.
 - g. *Process Shrink*: Apply a shrink factor to the device size and pin coordinates as the file is exported. The offsets are relative to the device origin.
4. Click *OK* to export the die text file.