O Commands

Product Version 23.1
September 2023

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

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.

Contents

1

O Commands

oa in	oa out	odb_out
offset copy	offset move	offset via gen
oops	open	opencd
opengl report	open project	optimize
optimize_ts	options	orbit import
orcad in	osdelete	

oa in

The `oa in` command lets you import die information from an OpenAccess (OA) database into APD. You can import information such as the IC outline, and die pin sizes and locations. You use the LEF Library Manager to provide technology information and cell data, including the cell outline as well as pin size, location, and layer information.

The `oa in` command preserves the OA terminal name as the logical pin name. It locks out net assignment changes for co-design die pins in the layout tool if the die pin is not implemented as a bump cell. Also, even if it is a bump cell, it locks out net assignment changes if there is RDL routing to the bump.

The design tool imports the following OA data:

- The design, which contains the design geometry, including the IC outline
- Cells of the following types:
  - oacCellTypePad (macro classes PAD, PAD INPUT, PAD OUTPUT, PAD INOUT, PAD POWER in LEF/DEF)
  - oacCellTypeCorner (macro classes ENDCAP TOPLEFT, ENDCAP TOPRIGHT, ENDCAP BOTTOMLEFT, ENDCAP BOTTOMRIGHT, in LEF/DEF)
  - oacCellTypeCover (macro classes COVER in LEF/DEF)
  - oacCellTypeCoverBump (macro classes COVER BUMP in LEF/DEF)
  - oacCellTypeBlockRing (macro classes RING in LEF/DEF)
  - oacCellTypeBlock (macro of class BLOCK in LEF/DEF)
- Pins
- Terminals
- Netlist information

With the `oa in` command, you can also apply scribe lines and an optical shrink to the imported die. For additional information, see Die Scribe Lines and Die Shrink in the *Placing the Elements User Guide.* You can also view the values for scribe lines and optical shrink on an existing design using the `die properties` command.

The `oa in` command depends on the OA runtime environment. The runtime environment is installed in a separate location from APD. For additional information on installation and configuration, see the *OpenAccess Installation and Configuration Guide* in the Cadence Help Library*.*

Test probe pins are imported as pads on the appropriate Probe_Top or Probe_Bottom subclass.

Refer to the *IO Planner Application Note* for additional details.

⚠ This command is available in APD on the UNIX platforms only.

The `oa in` command also generates symbol definitions, which includes a Design for Assembly (DFA) boundary. For additional information, see *Completing the Design* in the user guide.

Related Topics

- die properties

IC Import from OpenAccess Dialog Box

The IC Import from OpenAccess dialog box appears when you run the `oa in` command.

Before you import information from an OA database, you should have received a complete directory structure including the database, Library Exchange Format (`.lef`) files, Condensed Macro Library (`.cml`) files and at least one library definition (`.ldf`) file f*rom the IC designer. Additionally, you need this information:* OA definition file name (`lib.defs` file), library name, cell name, and view name related to the design.

> ⚠ Be sure to check the `lib.defs` file. If the file contains absolute paths to the library files, update the `lib.defs` file to use relative paths so that the design tool can access the library files.

defs file	Specifies the name of the OA definition file. The default name is `lib.defs.` The definition file is an ASCII file that contains a symbolic name for each library and the path to each library. It is recommended that you use the `lib.defs` file name. The tool automatically detects the file, and all required fields in the dialog box are automatically populated. The design tool searches for the *defs file* in this order: 1. Your current working directory: `./lib.defs` (default name and location) 2. Your login directory: `$HOME/lib.defs` 3. The data directory in the OA installation hierarchy: *<install_dir>/*`data/lib.defs.` Clicking *Browse* lets you choose a definition file in another directory. Once the correct definition file appears in this text box, the settings for *Library name, Cell name*, and *View name* automatically appear. ⚠ Some tools recognize only the `lib.defs` name as the *.defs file* name. Use other names with caution.

Library name	Specifies the OA library name. The library consists of design data files, auxiliary files, and the relationships among them. Use the drop-down list to view all the libraries listed in the specified `.defs` file. • If you initially imported the design from an OA database, the tool records the library name, from which the die was extracted, in the Allegro database. The name appears in this field as the default setting. • If the specified `.defs` file does not contain the library name recorded in the Allegro database, the tool displays a warning message in the dialog box, adds the name to the list of available libraries, and makes that library name the default setting. • If you did not originally import the die from an OA database, then the default setting is the first library name listed in the `.defs` file.
Cell name	Specifies the cell name (similar to the *Design Name* in the DEF file) within the current library. A single cell contains the data that describes a building block of a chip or system. • If you initially imported the design from an OA database, the tool records the cell name in the Allegro database, and that name appears in this field as the default cell name. • If this cell name does not exist in the specified library, the tool displays a warning message in the dialog box to indicate that the originally imported cell is missing from the specified library. • If you originally imported the design using the `def in` command or just created the design, then the default cell name is *die*.

View name	Specifies the view name within a specified cell to which you save your modifications. A view represents levels of abstraction for a design (behavior or gate), or different stages in the design process (Register Transfer Logic, post-synthesis, placed, or routed). Use the drop-down list to see all the views in the current cell. • If you import a design from an OA database, the tool records the view name in the Allegro database, and that name appears in this field as the default view name. • If the specified cell does not contain a view with the name recorded in the Allegro database, the tool displays a warning message in the dialog box to indicate that the originally imported view is missing from the specified cell. • If you originally imported the design using the `def in` command or created the design, then the default view name is the first view of the specified cell whose type is *layout*. If there are no *layout* views in the current cell, then the view name defaults to *layout*.
Refdes	Specifies the reference designator of the die component that you are importing.
Die and Pad placement	
IC Placement	
IC Center at	Specifies that the tool should place the center of the IC at the coordinates indicated in the *X and Y* coordinate fields. This is the default setting.
Lower Left Corner at	Specifies that the tool should place the lower left-hand corner of the IC at the coordinates specified in the *X and Y* coordinate fields.
As defined in OA	Specifies that the tool should use the origin as it is defined in the OA database, and place the die so that its coordinates match those in the OA database. When you choose this option, the tool disables the *X* and *Y* coordinate fields.
Location	
X coordinate	When enabled, you can specify the X coordinate of the IC location. The default setting is `0.0`.

Y coordinate	When enabled, you can specify the Y coordinate of the IC location. The default setting is `0.0`.
Pad layer	Lets you select the layer on which the imported pins reside. This field lists the layers from top to bottom in the current layer stackup. The default setting is the top layer in the layer stackup.
Apply IC Fabrication Optical Shrink	Check this box to shrink the die. The default setting is *Off*.
%	Enter a positive value (1 - 100) in the text box to indicate the percentage by which the original die size should be shrunk. For example, a 10% shrink means that the resulting die will only be 90% of the size of the original die. The default setting of this field is 0%, which means that no shrink will be applied and that the die symbol will be the size of the original die.
Add Scribe Width	Check this box to add scribe line information to the specified die. The default setting is *Off*. ⚠ For placement purposes, the extents are fixed per side and rotate when you change the orientation of the die. For example, if you rotate the die ninety degrees clockwise, the north side now exists on the east side of the representation.
North	Enter a value to indicate the amount that the actual physical die is larger than the represented extents on the North side of the die.
South	Enter a value to indicate the amount that the actual physical die is larger than the represented extents on the South side of the die.
East	Enter a value to indicate the amount that the actual physical die is larger than the represented extents on the East side of the die.
West	Enter a value to indicate the amount that the actual physical die is larger than the represented extents on the West side of the die.

Chip attachment and orientation	
Attachment	Specifies the attachment type (*Flip-chip* or *Wire bond*) for the IC that you are importing. The default setting for this frame is *Flip-chip*. If there is no *diestack* layer in the design, then the tool disables the *Wire bond* button.
Orientation	Specifies the orientation of the IC with respect to the package. The graphical display shows the choices you make.The default selection for this frame is *Chip-down.* If you toggle the chip attachment from *Flip-chip t*o *Wire bond*, then the tool automatically changes the chip orientation to *Chip-up*.
Import	Imports the specified `.defs` file and the LEF library files, and creates a new or updates an existing die. The die's reference designator (*Refdes*), component, and symbol name are equal to the *Cell name* (similar to the *Design Name* in the DEF file).
Cancel	Exits the command without importing any data from the OA database.
Help	Activates the Help Window.

Wire Bond Die Replace Dialog Box

If you are replacing a wire bond die, the Wire Bond Die Replace dialog box appears. It lets you control how wire bonds are updated based on the results of a die exchange.

Replacement Options	
Reconnect wires by pin number	If selected, the design tool reconnects wires to die pins based on the pin number that was previously associated with each bond finger.
Reconnect wires by net name	If selected, die pads are rewired based on the net assignments to the die pins and the recorded net assignments on the wire bond pads prior to the die replacement. If you add a new net to the design, this pin is not mapped to a bond finger initially under this scheme. This is the default choice.
Reconnect wires by pin location	If selected, this causes wires to be reconnected to die pins based on the pin number that was previously associated with each bond finger.
Rip up existing wires	If selected, all wires connected to this die are deleted and are not replaced during the die replacement operation. This does not affect the bond finger pattern itself, and should be used in situations where you want to rewire the die manually.
Ripup Options	
Delete dangling wires	When you check this box and click on either *Reconnect wires by pin number* or *Reconnect wires by net name*, the tool automatically removes any wires that are no longer connected to the die after the die replacement. The default setting is enabled.
Delete disconnected fingers	If selected, the tool automatically removes any disconnected fingers.
Next	Clicking this button prepares the design based on your settings and invokes the appropriate wizard for die definition update.

Cancel	Exits without making a die replacement.
Help	Displays context-sensitive help for this command.

Importing a Die from an IC Tool to APD

To import a die created in an IC tool for packaging in APD:

1. Create a design in the design tool with the appropriate layer information.
2. Ensure that you set up the LEF Library Manager files.
   Before importing an OA database, you must set up the LEF Library Manager to correctly map to the specified LEF files supplied by the IC designer. This allows the design tool to interpret the cells used by the IC design.
   If you do not have LEF files, use the `oa2lef` tool to convert the library cell into the LEF format. You can find this tool in the area where you installed the runtime version of OA. The tool is located in the `OA_HOME/bin` directory. Then you need to add these files to the `.ldf` file and create `.cml` files using the LEF Library Manager.
3. Type `oa in` in the Command window.
   The IC Import from OpenAccess dialog box appears. The design tool automatically populates the fields if the `lib.defs` file is located in the current working directory. Otherwise you have to browse to choose the appropriate .defs file.
4. Use the drop-down list to choose the *Library name*, if necessary
   The values for *Cell name* and *View name* automatically appear in the boxes.
5. Complete the other settings in the dialog box or accept the default values.
6. Click *Import*.
   The data is imported using the OpenAccess library data and the appropriate cells (macros) from a LEF library file. The existing LEF Library Manager manages the LEF files and creates the Condensed Macro Library (`.cml`) files. This library has a hierarchy of LEF files in which to search for each cell.
   As a result of OA import processing, a die component and symbol are created. Netlist information is included for each die pin, if it exists.

Related Topics

- die properties

oa out

The `oa out` command lets you export die data, including a physical description of the IC outline and the I/O pad sizes and locations to an OpenAccess (OA) database.

With this command, you can export the bump pattern from APD for package-driven flows.

The `oa out` command depends on the OA runtime environment. The runtime environment is installed in a separate location from APD. For additional information, see the *OpenAccess Installation and Configuration Guide*.

The design tool exports the following data:

- Macros of the following classes:
  - PAD
  - PAD INPUT
  - PAD OUTPUT
  - PAD INOUT
  - PAD POWER (oacCellTypePad in OpenAccess)
- Macros of the following classes:
  - ENDCAP TOPLEFT
  - ENDCAP TOPRIGHT
  - ENDCAP BOTTOMLEFT
  - ENDCAP BOTTOMRIGHT (oacCellTypeCorner in OpenAccess)
- Macros of the class COVER (oacCellTypeCover in OpenAccess)
- Macros of the class COVER BUMP (oacCellTypeCoverBump in OpenAccess)
- Pins
- Netlist information

⚠ This command is available in APD on the UNIX platforms only.

For additional information about OA, see *Getting Started with Physical Design in the user guide.*

OpenAccess Export Dialog Box

The OpenAccess Export dialog box appears when you run the `oa out` command.

> ⚠ Be sure to check that the `lib.defs` file uses relative paths so that when files are moved, the tool can still access the library files.

Die component	
Refdes	Lets you select a die component to export.
OpenAccess	
.defs file	Specifies the name of the definition file, which is an ASCII file that contains a symbolic name for each library and the path to each library. If you do not specify a definition file, the tool automatically creates it. The design tool searches for the OA`.defs file` in this order: 1. Your current working directory: `./lib.defs` (default name and location) 2. Your login directory: `$HOME/lib.defs` 3. The data directory in the OA installation hierarchy: *<install_dir>/*`data/lib.defs` Clicking *Browse* lets you choose a definition file in another directory. Once the correct definition file name appears in this text box, the settings for *Library name, Cell name*, and *View name* automatically appear. ⚠ Some tools recognize only the `lib.defs` name as the .defs file name. Use other names with caution.

Library name	Specifies the OA library name. The library consists of design data files, auxiliary files, and the relationships among them. Use the drop-down list to view all the libraries listed in the specified `.defs` file. • If you initially imported the design from an OA database, the design tool records the library name, from which the die was extracted, in the Allegro database. It uses this name as the default library name. • If the specified `.defs` file does not contain the library name recorded in the Allegro database, the tool displays a warning message in the dialog box, adds the name to the list of available libraries, and makes that library name the default setting. • If you did not originally import the die from an OA database, then the default setting is the first library in the specified `.defs` file.
Cell name	Specifies the cell name (similar to the *Design Name* in the DEF file) within the current library. A single cell contains the data that describes a building block of a chip or system. This field shows a cell name for the current library. • If you initially imported the design from an OA database, the tool records the cell name in the Allegro database, and uses the name as the default cell name. • If this cell name does not exist in the specified library, the tool displays a warning message in the dialog box to indicate that the originally imported cell is missing from the specified library. • If you originally imported the design using the `def in` command or just created the design, then the default cell name is *die*.

View name	Specifies the view name within a specified cell to which you save your modifications. A view represents levels of abstraction for a design (behavior or gate), or different stages in the design process (Register Transfer Logic, post-synthesis, placed, or routed). Use the drop-down list to see all the views of the current cell. • If you import a design from an OA database, the tool records the view name in the Allegro database and uses the name as the default view name. • If the specified cell does not contain a view with the name recorded in the Allegro database, the tool displays a warning message in the dialog box to indicate that the originally imported view is missing from the specified cell. • If you originally imported the design using the `def in` command or created the design, then the default view name is the first view of the specified cell whose type is *layout*. If there are no layout views in the current cell, then the view name defaults to *layout*.
Tile export	
Tile instances	Clicking this radio button lets you export tile instances. This is the default setting.
Flattened tiles	Clicking this radio button lets you flatten the tiles by replacing the tile instances with their contents, and export the flattened tiles.
Physical pin layer	
Layer name	Lets you specify the physical pin layer name used in the IC database.
OK	Exports the data to the OA database.
Cancel	Exits the command without exporting data to the OA database.
Help	Activates the Help Window.

Exporting Die Information from the Design Window

You probably will export information to the OA database from a different directory than the one in which you imported data from the OA database. To export die information from the Design Window:

1. Type `oa out` in the Command window. The OpenAccess Export dialog box appears.
2. Use the drop-down list to choose the reference designator of the die you are exporting.
   The design tool automatically populates the fields if the `lib.defs` file is located in the current working directory. Otherwise you have to browse to choose the appropriate `.defs` file.
3. Use the drop-down list to choose the *Library name*.
   The values for *Cell nam*e and V*iew name* automatically appear.
4. Complete the other fields in the dialog box or accept the default values.
5. Click *Export*.
   The design tool creates or updates the OA database using the appropriate cells (macros) from a LEF library file. This library has a hierarchy of Library Exchange Format (LEF files in which to search for each cell.
   If you do not have a `.ldf` (dot `ldf`) file, use Library Manager to find the LEF files, create a `.ldf` file, and create `.cml` files if necessary.

Related Topics

- lef lib

odb_out

Only available from the design layout editor to export database information into a Valor ODB++ database. The ODB++ format contains all CAD/EDA database, assembly, and manufacturing data.

You can use the ALLEGRO_BRD2ODB_USER_OPTIONS environment variable to contain command line options to the interface `brd2odb` executable that you want to add, or to override existing settings made by the interface. You can define it directly as a UNIX/Windows environment variable or in the `allegro.env` file.

If you attempt to export, and dynamic shapes are out-of-date, `odb_out` fails. Run `status` to use the Status tab of the Drawing Options dialog box to verify the current state of dynamic shapes and DRCs and update them if they are out of date. You cannot export until you update dynamic shapes and/or DRC. Valor has a limit of 64 characters on most entities in ODB++.

Access using:

- Menu Path: *File – Export – ODB ++ inside*
- Toolbar Icon:

  ODB

Related Topics

- status

Running the Valor ODB++ Inside Package

1. Open the design file from which you want to export data.
2. Choose *File – Export – ODB ++ inside.*
   Alternatively, type `odb_out` in the Command window.

   If the ODB++ Inside package is installed, the Valor-supplied Cadence Allegro BRD to ODB++ Export dialog box displays, and you can set the parameters to export your current design to a Valor database. See the Valor online documentation for information about these parameters and setting them.

Installing the Valor ODB++ Inside Package

1. Open the design file from which you want to export data.
2. Choose *File – Export – ODB ++ inside.*
   Alternatively, type `odb_out` in the Command window. If the ODB++ Inside package is not installed, a dialog box displays that describes ODB++ Inside.
3. Click Install/Download to launch a browser with a Valor-created and -maintained URL that provides ODB++ Inside details and a registration form.
4. Complete the registration form (in the browser).
5. Follow the installation instructions on the download page for the operating system you are running.
6. Restart once you complete the installation and download.
7. Choose *File – Export – ODB ++ inside.*
   The Valor-supplied Cadence Allegro BRD to ODB++ Export dialog box appears, and you can set the parameters to export your current design to a Valor database. See the Valor online documentation for information about these parameters and setting them.

offset copy

The `offset copy` command creates multiple copies of various elements (arc, circle, rectangle, frectangle, line, and text) that are offset from the original element. You can specify the distance between the elements and the number of occurrences in the *Options* tab.

Offset copy command: Options Panel

Access using

- Menu Path: *Manufacture – Drafting – Offset Copy*

X Offset	Specifies the distance between the original and new element in X direction. By default, it is set to 0.
Y Offset	Specifies the distance between the original and new element in Y direction. By default, it is set to 500.
Repetitions	Specify the number of copies for new elements. By default, it is set to 1.
Update Line	Sets the line width and font for creating new elements. By default this option is On.
Width	Specifies the width of the Solid lines for creating new elements in user units. All other line fonts remain at 0 width.
Font	Specifies the line pattern used in creating new elements. The choices are Solid , Hidden , Phantom , Dotted , and Center . The default is Solid .

Related Topics

- offset move

Creating Multiple Copies of Shapes

1. Either choose *Manufacture – Drafting – Offset Copy* or set *General Edit* application mode and select an element and then right-click and choose *Drafting – Offset Copy.*
1. Specify the X *Offset* in the *Options* tab.
2. Specify the Y *Offset* in the *Options* tab.
3. Specify number of copies in the *Repetitions* field.
4. Specify the line width for creating new element in the *Width* field.
5. Specify the line pattern for creating new element in the *Font* field.
6. Select an element.
   A copy of the selected element is added with specified line width and font at a distance set in the Offset fields.
7. Right-click and choose *Next* to continue or *Done* to complete the operation.

Related Topics

- Moving Shapes

offset move

The `offset move` command moves different types of elements (arc, circle, rectangle, frectangle, line, and text) to a new location that is a distance from the original location. You can specify the X and/or Y offsets in the *Options* tab.

Related Topics

- Moving Shapes

Offset move command: Options Panel

Access using

- Menu Path: *Manufacture – Drafting – Offset Move*

X Offset	Specifies the distance between the original and new location in X direction. By default, it is set to 50.
Y Offset	Specifies the distance between the original and new location in Y direction. By default, it is set to 50.

Moving Shapes

1. Do one of the following:
   - Choose *Manufacture – Drafting – Offset Move.*
     Alternatively, type `offset move` in the Command window.
   - Set *General Edit* application mode and select an element. Right-click and choose *Drafting – Offset Move.*
2. Specify the X *Offset* in the *Options* tab.
3. Specify the Y *Offset* in the *Options* tab.
4. Select an element to move.
   The element is moved from its current location by the offset specified in the Options tab.
5. Right-click and choose *Next* to continue or *Done* to complete the operation.

Related Topics

- Creating Multiple Copies of Shapes
- offset copy

offset via gen

The `offset via gen` command, available only with the `Allegro X Advanced Package Designer+` license, lets you create vias for one, some, or all pins in a package. Before generating the offset via, you can choose:

- The angle of the offset vias to be directed at the package origin or at a 45-degree angle to the origin.
- Whether vias are on the inside (between the pin and the origin) or on the outside (with the pin between via and origin).
- whether offset vias are only created to pins assigned to a net. You can also create offset vias for selected pins using the Find tab filter.
- The distance between the center of the pin and the center of the via. You can also place vias directly at the pin location, extending its penetration through the layers.
- Automatic generation of a simple two-pad padstack (and save to disk), searching through a drawing database for a desired padstack, or loading a padstack from disk.
- Automatic creation of fillets between the pins and the vias.

Related Topics

- Creating Offset Vias

Offset Via Generator Dialog Box

Access using

- Menu Path: *Route – Offset Via Generator*

Pattern Tab - Create options

Allow DRCs	Select this check box if you want to create vias only in the regions with no DRCs.
Create offset vias/fillets for assigned pins only	Select this check box to create offset vias or fillets only for pins that are assigned to a net. Otherwise, unassigned pins also have offset vias/fillets generated for them
Generate fillet lines to existing vias where possible	Specifies that for pins to which there is a single-segment cline connection between the pin and a via on the specified layer, the tool fillets this connection instead of adding a new via, clines, and set of fillets. If the tool does not find a via, it continues to create a new offset via with fillets, based on the settings in the dialog box.
Add via at Pin Location	Enable this check box to place the via directly at the pin location to provide layer penetration from the pin location. This option eliminates the need for trace wires and fillets, but may cause manufacturing problems
Pitch	Specify a number to indicate the distance from the center of the pin to the center of the via.
Create fillet lines	Enable this option to automatically add fillets between the pin and the offset via.
Non-fillet line width	Editable only when *Create fillet lines* is inactive. Lets you set line widths for non-filleted lines.

Pattern Tab - Orientation

Focus	Specify an origination type that determines the focus of the via pattern. The *nearest corner* option makes the tool orient the offset vias toward the corners instead of the center of the symbol. Using this option can provide a better pattern in high-density designs. Your selection is reflected in the bitmap illustration in the dialog box. Notice that the options in the Angle and Direction lists are determined by your Focus selection.
Angle	Specify the type of angle of the offset vias. Your selection is reflected in the bitmap illustration in the dialog box. The direction of a selected pin depends on the package origin (center) and the quadrant in which the selected pin is located. For example, using the 45-degree angle, a pin in the –X, Y quadrant will be at a 315-degree angle (or –45 degrees) from the Y axis.
Direction	Specify the direction of the offset vias. Your selection is reflected in the bitmap illustration in the dialog box.

Padstack Tab - Method

To connect between the current layer to the connection layer, specify one of the following padstack options:

New	Specifies a new padstack definition.
Available Padstack	Choose from a list of available padstacks from the current database.
Load from Disk	Locates and loads an existing padstack from a different path from the current database.

Padstack Tab - Specifications

Name	Specifies a unique identifier for the padstack.
To Layer	Specifies the connection layer to which the generated via will be drilled.
From Layer	Specifies the connection layer from which the generated via originated.

Padstack Tab - Shape

Circle	Creates a circular padstack.
Rectangle	Creates a rectangular padstack.

Padstack Tab - Dimensions

Width	Specifies the width of the padstack.
Height	Specifies the height of the padstack.

Creating Offset Vias

1. Choose *Route – Offset Via Generator.*
   Alternatively, type `offset via gen` to display the Offset Via Generator dialog box.
2. Choose the component or selected pins for offset via generation.
   - To generate offset vias for the all pins in the package, you must set the *Find filter* t to *Comps* (default) and choose a component.
   - To generate offset vias for selected pins, you must set the *Find filter* tab to *Pins*, and choose the desired pins. (This requires deselecting *Comps*.)
   - To generate offset vias for selected nets, choose *Nets* in the Find Filter. The console window displays a prompt for you to select a component. If you select multiple components, they should all be on the same layer.

     ⚠ If you are generating offset vias for selected nets, the tools prompts you to select the component if there is more than one component in the design.
3. Specify the parameters on the *Pattern* tab.
4. Specify the parameters on the *Padstack* tab. The padstack to use for the offset via may be an existing one in the design, one from the library, or a new one may be created.
5. Click OK or Apply to generate the offset vias.

Related Topics

- offset via gen

oops

The `oops` command, available on the pop-up menu when an interactive command is active, cancels the last selection made during the current interactive command but does not undo any actions on which Done has already been executed. After a group, window, or cut operation has been completed, Oops undoes the entire operation.

open

The `open` command opens an existing design file in the current directory. You are prompted to save or discard changes in the current open file. A file browser lets you search for the specified design file if you do not provide a file name.

A list of your most recently used (MRU) files appears. See opencd.

You no longer are warned if you are switching your design from the tier where it was last saved to another tier. To display the warning, set the `db_tier_nomsg` environment variable using the *Setup – User Preferences – Drawing* (enved) menu command.

Co-Design Environment

When you open a new .`mcm` design, and an existing drawing that has co-design dies with unsaved changes is open, you are asked whether you want to save the design. If you choose to save the changes, then the *File – Save* command is invoked to save the data. If you choose to discard the changes, then any temporary Open Access (OA) library/cell/views containing unsaved data are deleted before the new design is opened.

Opening System-Level Configurations

System-level configurations (SCs) are multiple drawings of different types that, connected together, define a system. You can open SCs in your Allegro product when the primary drawing of the configuration is a native design; for example, if you are running APD and the primary drawing is a .`mcm`, it will display on the APD canvas. The secondary or background drawings (.`brd`) will not be visible or editable, but the data from those background drawings will be accessible to APD.

Opening Different System-Level Configurations

You can open in one application designs created in other applications; for example, opening from Allegro PCB SI a .`mcm` design created in APD. The typical purpose of doing so would be to set up the designs for use in a system configuration, but because designs created in another tool may not contain signal integrity data that your application requires (voltage properties, cross-section data, model assignments, and so on), only limited editing of the non-native drawing is allowed. The following functions remain active:

File commands	**Open**
	Save
	Save As
	Script
	Recent Designs
	Exit
View commands	All
Display commands	Color/Visibility
	Layer Priority
	Element
	Hilight/De-highlight
	Highlight By Pick
	De-highlight By Pick
	Shadow toggle
Setup commands	Materials
	Cross-section (Limited functionality. You cannot add or delete layers to the stack-up.)
Logic commands	Identify DC Nets
	Pin Type
	Assign Differential Pair
Analyze – SI/EMI Sim commands	Initialize
	Library

	Model Assignment
	Model Dump/Refresh
	Preferences
	Audit (Limited functionality. You cannot perform a design audit.)

When you run the *File – Open* command, the file browser defaults to the design types created by your application; for example, .`brd` files for PCB SI, .`mcm` files for APD, and so on. To open a non-native design in your application, select the *All Files* option from the *Files of type* drop-down.

The *Save* and *Save As* commands will automatically save the design in its existing design type.

Access using

- Menu Path: *File – Open*
- Toolbar Icon:
  📂

```
open [<design to open>]
```

If you do not provide the *<design to open>* argument, a browser window opens in the current directory.

Examples

```
open master.brd
```

The `master.brd` file opens in the current directory.

```
open \boards\master.brd
```

The `master.brd` file, located in the `boards` directory in the current directory, opens.

Related Topics

- open project
- opencd
- opengl report
- Opening an Existing File

Open Dialog Box

The Open dialog box is a standard file browser. Two buttons appear below the *Help* button. The left button lets you display a text preview of the current design; the right button lets you display the graphics preview of the design. The preview area appears on the right side of the list box.

Opening an Existing File

1. Choose *File – Open.*
   Alternatively, type `open` with the design name you want to open in the Command window.
   The file opens in the current directory. If you do not provide a *design to open* argument, the Open dialog box opens in the current directory. If you are opening a design type that is not primary to the application you are in (for example, a `.mcm` in PCB SI), select the *All Files* option from the *Files of type* drop-down to display the non-native designs.
2. Choose a file from the list.
   You can also enter the file name in the *File name* field.
3. Click the *Preview Image (P)* button to display the graphics preview of the specified file.
4. Click *Open* to open the file.

Related Topics

- Opening an Existing Project
- open

opencd

The `opencd` command opens a file using the specified path name and also changes the current directory to the directory where the file resides.

The `opencd` command maps to the *File – Recent Designs* menu bar command. When you use the menu bar command, a list (up to 20) opens of your most recently used (MRU) files with their specified path names. The default number of file names is 10. When you choose a file name from the list, the file opens and changes the current directory to the directory where the specified file resides.

Use the `recentFileList` environment variable to control the maximum number of files that appear on the menu. You can also choose *Setup – User Preferences*, then click the *UI* category and type in a value from 0 to 20 in the *recentfilelist* field. The list of these files are saved in the *<program>.mru* file located in your `pcbenv` directory.

Related Topics

- open
- open project
- Opening Recently Used Projects

Opencd Dialog Box

Access using

- Menu Path: *File – Recent Designs*

```
opencd [<pathname of file>]
```

If you do not provide the correct *<pathname of file>* argument, a browser window opens in the current directory.

The Opencd dialog box is a standard file browser. Two buttons appear below the *Help* button. The left button lets you display a text preview of the current design; the right button lets you display the graphics preview of the design. The preview area appears on the right side of the list box.

Examples

The following are examples using the `opencd` command.

- `opencd master.brd`

  The `master.brd` file opens in the current directory.

- `opencd \projects\project1\example.brd`

  The working directory changes to the `project1` directory and opens the `example.brd` file.

Related Topics

- Open Dialog Box
- Opening an Existing File

Opening Recently Used Projects

1. Choose *File – Recent Designs.*
   Alternatively, type `opencd` in the Command window.
   The file opens. If the pathname is not correct, the Opencd dialog box opens in the current directory.
2. Click the *Look in* list to choose a different directory and highlight the file name.
3. Click the left button below the *Help* button to display a text preview of the specified design. The preview area appears on the right side of the *File name* list.
4. Click the right button below the *Help* button to display the graphics preview of the design.
5. Click *Open* to open the file.

Related Topics

- open
- opencd
- open project

opengl report

The `opengl report` command checks system graphic information and creates a report listing vendor card type and version.

⚠ OpenGL must be enabled for this command to function.

open project

The `open project` command opens a Capture project file (`.opj`) or HDL design file (`.cpm`). The name of the project directory and design name should be specified in the project files.

Related Topics

- Opening an Existing Project

Open Project Dialog Box

Access using

- Menu Path: *File – Open Project*

The Open dialog box is a standard file browser. Two buttons appear below the *Help* button. The left button lets you display a text preview of the current design; the right button lets you display the graphics preview of the design. The preview area appears on the right side of the list box.

Opening an Existing Project

1. Choose *File – Open Project.*
   Alternatively, type `open project` in the Command window.
2. Choose a file from the list.
   You can also enter the file name in the *File name* field.
3. Click *Open* to open the file.

Related Topics

- open
- opengl report

optimize

Internal command.

optimize_ts

The `optimize_ts` command optimizes the location of Tpoints. Optimal Tpoint location is critical to the performance of the router and the signal integrity of your design.

When you enter the command, PCB Router is invoked in background mode with no router commands in the Do file. Upon optimizing the location of Tpoints based on physical criteria, a Session file containing the new Tpoint locations is written.This file translates back into the layout editor to update the design.

> ⚠ Specctra removes connections to power layers resulting in floating power vias for routers. To prevent floating power vias, mark all standalone vias as fixed before you run any routing operation including optimize_ts.

Access using

- Menu Path: *Route – Optimize Rat Ts*

Optimizing Tpoint Locations in Your Design

1. Choose *Route – Optimize Rat Ts.*

   Alternatively, type `optimize_ts` in the Command window.

   PCB Router runs in the background and displays the following messages in the Command window as it optimizes Tpoint locations.

   ```
   Optimizing Rat Ts. Please wait... Updating Rat Ts. Please wait... Optimizing Rat
   Ts Done.
   ```

   When the operation is complete, the design updates and displays each Tpoint at its new location.

   > ✓ Use this command only for complex topologies for PCB Router to automatically manage Tpoint placement and routing without further intervention. However, PCB Router performance might be affected while automatically optimizing Tpoint locations in designs containing dense H-tree and Tpointed differential pair topologies. For best results with these designs, you should help optimize Tpoints by using your knowledge of these topologies along with an auto-interactive use of the tools. For complete details, see *Routing the Design in the user guide*.

options

The `options` command lets you set any of the visible fields in the active Options tab of the Control panel.

`options` <field name> <value>

<field name>	The name of the parameter you are setting in the Options tab. To find out the field names, record a script and change each field to all possible values. Then choose *File – File Viewer* to view the script and locate the field names. **For example, the field name for Line lock is lock_mode.**
<value>	**The values of the specified field. For example, the values for lock_mode are Line and Arc. This field is case-sensitive. Be sure that when you type the text for value at the command line, you enter it the same way as it appears in the Options tab.**

Examples

The following example shows that you are setting the Bubble field on the Options tab to *Shove preferred.* This means that after you type the `options` command and then enter into these modes: *Add – Connect, Slide,* and *Edit – Vertex,* the field is set to *Shove preferred.*

```
    options bubble_space Shove preferred
```

The following shows an example of assigning an alias (Ctrl-b) for commands that you have chained together, separated by a semicolon.

The first command changes the `shove_mode` environment variable to the next value in the sequence (Off, Hug preferred, Shove preferred). The second command sets the bubble_space field to the value that was just stored in the `shove_mode` environment variable. If you press Ctrl-b while you are in one of these modes: *Add – Connect, Slide,* and *Edit – Vertex,* the field is set to the new value set stored in the environment variable.

alias ::B 'settoggle shove_mode Off "Hug preferred" "Shove preferred"; options bubble_space $shove_mode'

Related Topics

- settogle

orbit import

The `orbit import` command imports Integrity System Planner (`.oio`) databases.

Related Topics

- Importing Integrity System Planner Databases

Orbit Import Dialog Box

Access using

- Menu Path: *File – Import – Integrity System Planner*

Import/Override Data	
Netlist	Overrides all netlists. Selected by default.
Placement (includes location, rotation)	Overrides existing placement, including location and rotation. Selected by default.
Routing (includes wires, vias, metasl; excludes wire bonds)	Overrides all routing including wires, vias and metals. Not selected by default.
Wire bonds (includes bond fingers, wire profiles, bond wires)	Overrides bond wires, including bond fingers and wire profiles. Not selected by default.
Ignore FIXED property	Ignores the FIXED property assigned to any object, such as a pin. Not selected by default.
Padstack Import Options (includes pins, vias, bond fingers)	Overrides padstacks including pins, vias, and bond fingers.
Bundle Import Options	Lists the import options for bundles.
Net Schedule	
Override (Use net schedule defined in Integrity System Planner)	Choose to use the net scheduled defined in Integrity System Planner. Selected by default.
Reuse (Use net schedule defined in Allegro)	Choose to use the net scheduled defined in APD. Not selected by default.
Bundle	

Remove all exisitng Allegro bundles	Choose to remove any existing bundles from the APD design to which the Integrity System Planner database is being imported. Selected by default.
Merge/Rename existing bundles	Choose to merge or rename any existing bundles of the APD design to which the Integrity System Planner database is being imported. Not selected by default.

Importing Integrity System Planner Databases

1. Choose *File – Import – Integrity System Planner*
   Alternatively, type `orbit import` in the Command window. By default, the Integrity System Planner layers are used.
2. Click OK.
3. Specify the overrides.
4. Click *OK*.
   The Orbit Layer Map appears.
5. Specify the layers and bundle import options.

   > ⚠ Imported components with invalid stackup information are placed on the top layer.

6. Click *OK*.

Related Topics

- orbit import

orcad in

The `orcad in` command converts PCB designs created in OrCAD® Layout to Allegro® PCB Editor designs. You can translate PCB design databases created in any version of Layout and prepare them for use within Allegro X PCB Editor.

The translator converts a design (`.max` file) created in Layout to a design database (`.brd` file) Allegro X PCB Editor can read. The Layout`.max` file contains all footprint information.

For more information, see the *Transferring Logic Design Data* user guide in your documentation set.

Before translating your Layout design to Allegro X PCB Editor:

- Avoid using characters such as $, ::, @, #, %, ^, &, *, (,), -, =, ', \, ", [,] , ?, /, <, >, !, . , , ; , { , } , ` , + , | in reference designator and symbol names in Layout. Otherwise, the translator cannot convert the reference designator and symbol names properly.
- Clean up the design in Layout by using the *Auto – Cleanup Design* menu command or by exporting the design to a `.min` format and then importing it to a `.max` format.
- Enable all the layers on which routing has occurred in Layout.
- Avoid a "." (period) in footprint names.
- Avoid package names containing Microsoft® Windows® reserved words. For example, `Con, Nul, Aux, Prn`, and so on. Otherwise, the translator cannot create the required device file (`.txt`).

You can also run the translator in batch mode by specifying all required information on the command line.

`orcad in` *[-b] [-w] <input_file> <output_directory>*

Optional arguments:

[-b]	Specifies batch mode.
[-w]	Overwrites an existing translated file (if any) in the destination directory.
<input_file>	Specifies the full path and name of the Layout (`.max`) file.

<**output_file**>	Specifies the full path and name of the output directory where the translated Allegro X PCB Editor (`.brd`) file resides.

Related Topics

- Creating a Catalog of the Library Prior to Translating
- Converting Layout designs to Allegro X PCB Editor designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in Board Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of Your Translated Board Design
- Removing Ratsnests
- Restricting Layers to be Routed in the Translated Board Design
- Resolving False Pin-to-Pin Spacing (Clearance) Errors in the Translated Board Design
- Resizing Pad Size in the Board Design
- Displaying Footprint Information in Translated Board Designs

OrCAD Layout to Allegro Dialog Box

- Menu Path: *File – Import – CAD Translators – OrCAD Layout*

Layout MAX File	Enter or browse to the directory location where the Layout design (`.max`) file resides.
Translate	Click to start translation.
Viewlog	Click to view the log file created during translation.
Close	Click to close the dialog box.

Related Topics

- derive connectivity
- dbdoctor
- padeditdb
- Converting Layout Designs to Allegro X PCB Editor Designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in your Allegro X PCB Editor Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of your Translated Allegro X PCB Editor Design
- Removing Ratsnests
- Restricting Layers to be Routed in the translated Allegro X PCB Editor Design
- Resolving False Pin-to-Pin Spacing (Clearance) Errors in the Translated Allegro X PCB Editor Design
- Resizing Pad Size in Your Allegro X PCB Editor Design
- Displaying Footprint Information in Translated Allegro X PCB Editor Designs

Creating a Catalog of the Library Prior to Translating

1. Choose *Tools – Catalog – Create*.
   Alternatively, type `orcad in` in the Command window. The Create Catalog dialog box appears.
2. Specify the path and filename of the library from which to create a catalog and click *OK*. This creates `.max` files.
   You can open this `.max` file in Layout, which contains all the footprint information. In Layout, libraries contain the following four layers: TOP, BOTTOM, PLANE, and INNER. The rest of the layers are documentation layers. Similarly, the `.max` file created by the Catalog tool also contains the above mentioned four layers and the rest of the layers are documentation layers.

Related Topics

- orcad in
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in Board Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of Your Translated Board Design
- Removing Ratsnests
- Restricting Layers to be Routed in the Translated Board Design
- Resolving False Pin-to-Pin Spacing (Clearance) Errors in the Translated Board Design
- Resizing Pad Size in the Board Design
- Displaying Footprint Information in Translated Board Designs

Converting Layout Designs to Allegro X PCB Editor Designs

1. In the OrCAD Layout to Allegro dialog box, enter or browse to the directory location of the original Layout design file (`.max`) in the Layout MAX file text box.
2. Click *Translate* to run the translation.
   The translated Allegro X PCB Editor design file (`.brd`) is created in the same directory as the Layout design file. A log file (`.log`) explaining translation information displays after the translation. The log file gets created in the same directory as the translated Allegro X PCB Editor design. To view the log file, click *Viewlog*.

Related Topics

- orcad in
- OrCAD Layout to Allegro Dialog Box
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in your Allegro X PCB Editor Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of Your Translated Allegro X PCB Editor Design
- Removing Ratsnests
- Restricting Layers to be Routed in the Translated Allegro X PCB Editor Design
- Resolving False Pin-to-Pin Spacing (Clearance) Errors in the Translated Allegro X PCB Editor Design
- Resizing Pad Size in Your Allegro X PCB Editor Design
- Displaying Footprint Information in Translated Allegro X PCB Editor Designs

Using Footprint Information after Library Conversion

After the conversion, start Allegro X PCB Editor and perform the following steps to use the footprints.

Before using your translated Allegro X PCB Editor design, run *Tools – Derive Connectivity* (derive connectivity command), *Tools – Database Check* (dbdoctor command), and *Tools – Padstack – Modify Design Padstack* (padeditdb command).

1. Choose *Setup – Cross-Section*. The Cross-section Editor dialog box appears.

   > ⚠ To open the Layout Cross Section dialog box in OrCAD PCB Editor, choose *Tools – Setup Advisor.* Click *Next* in the Database Setup Advisor dialog box. Click *Edit Cross-section* in the Database Setup Advisor – Cross-section dialog box.

2. Delete PLANE and IS2 layers and save the `.brd` file.
3. Create the flash and shape symbols if you wish to update the same for the padstacks of your design. Otherwise, run *Tools – Database Check* (dbdoctor command).
4. Choose *Tools – Padstack – Modify Design Padstack* (padeditdb command). A list of all padstacks defined in your design displays.
5. Select the padstack to edit from the list and click *Edit.* The Padstack Designer dialog box opens with the padstack definition loaded.
6. Assign the created flash symbol to all the padstacks and save the `.brd` file.

   > ⚠ The DEFAULT INTERNAL layer in the Layers tab populates from the TOP Layer padstack data and ensures that when this padstack is instantiated on a new Allegro X PCB Editor board, the values for all the defined layers are present.

7. Run *Tools – Database Check* (dbdoctor command).
8. Choose *File – Export – Libraries* (dlib command) to export all the symbols from your `.brd` file. and provide you all with all the symbols present in the open `.brd` file.

Related Topics

- derive connectivity
- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro X PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in your Allegro X PCB Editor Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of Your Translated Allegro X PCB Editor Design
- Removing Ratsnests
- Restricting Layers to be Routed in the translated Allegro X PCB Editor Design
- Resolving False Pin-to-Pin Spacing (clearance) Errors in the Translated Allegro X PCB Editor Design
- Resizing Pad Size in Your Allegro X PCB Editor Design
- Displaying Footprint Information in Translated Allegro X PCB Editor Designs

Troubleshooting Translation Issues

The following procedures can assist with troubleshooting:

- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro X PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in your Allegro X PCB Editor Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of your Translated Allegro X PCB Editor Design
- Removing Ratsnests
- Restricting Layers to be Routed in the Translated Allegro X PCB Editor Design
- Resolving False Pin-to-Pin Spacing (clearance) Errors in the Translated Allegro X PCB Editor Design
- Resizing Pad Size in Your Allegro X PCB Editor Design
- Displaying Footprint Information in Translated Allegro X PCB Editor Designs

Related Topics

- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout designs to Allegro X PCB Editor Designs

Defining Manual Voids on the Inner Copper Shapes of Your Translated Design

If your Layout design contains thermal reliefs (or manual voids) on the inner copper shape of a nested copper shape, then the voids do not get converted. You have to define manual voids on the inner copper shapes of your translated Allegro X PCB Editor design.

- Do one of the following to create voids:
    - Choose *Shape – Manual Void – Rectangular*.
    - Choose *Shape – Manual Void – Circular.*
    - Choose *Shape – Manual Void – Polygon*.

Related Topics

- shape void rectangle
- shape void circle
- shape void polygon
- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout Designs to Allegro X PCB Editor Designs
- Using Footprint Information after Library Conversion
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in Board Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of Your Translated Board Design
- Removing Ratsnests
- Restricting Layers to be Routed in the Translated Board Design
- Resolving False Pin-to-Pin Spacing (Clearance) Errors in the Translated Board Design
- Resizing Pad Size in the Board Design
- Displaying Footprint Information in Translated Board Designs

Deleting Shape Islands

When you convert a Layout design into Allegro X PCB Editor, you have to delete the islands manually by doing the followign step:

1. Choose *Shape – Delete Islands*.

Related Topics

- island_delete
- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout Designs to Allegro X PCB Editor Designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Removing Duplicate Vias
- Backannotating Translated Allegro PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in Board Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of Your Translated Board Design
- Removing Ratsnests
- Restricting Layers to be Routed in the Translated Board Design
- Resolving False Pin-to-Pin Spacing (Clearance) Errors in the Translated Board Design
- Resizing Pad Size in the Board Design
- Displaying Footprint Information in Translated Board Designs

Defining Thermal Relief Flash Symbols for Padstacks

The thermal relief flash symbols associated with padstacks are not retained during the conversion process. You have to create flash definition in Allegro X PCB Editor for each of the thermal padstacks in your design.

1. Choose *Tools – Padstack – Modify Design Padstack*
   Alternatively, type `padeditdb` in the Command window.
2. Select the padstack to edit from the list and click *Edit.* The Padstack Designer dialog box opens with the padstack definition loaded. The banner of the Padstack Designer lists the name of the padstack that you are modifying.
3. Select the appropriate layer.
4. Select *Flash* from the *Thermal Relief* list box.
5. Click the browse button. The Select Flash Symbol dialog box appears.
6. Select a flash symbol in the dialog box.
7. Choose *File – Update To Design* to load the padstack into your design.

⚠ The *Update to Design* option is available only if you invoke the Padstack Editor from a current design instead of using the Padstack Editor as a standalone program.

1. Choose *Setup – Design Parameters* (`prmed` command). The Design Parameter Editor dialog box appears.
2. Select the *Thermal pads* check box under the *Enhanced Display Modes* section in the *Display* tab.

Related Topics

- padeditdb
- prmed
- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout Designs to Allegro X PCB Editor Designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Backannotating Translated Allegro PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in Board Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of Your Translated Board Design
- Removing Ratsnests
- Restricting Layers to be Routed in the Translated Board Design
- Resolving False Pin-to-Pin Spacing (Clearance) Errors in the Translated Board Design
- Resizing Pad Size in the Board Design
- Displaying Footprint Information in Translated Board Designs

Removing Duplicate Vias

During the conversion, duplicate vias in layout are retained in the translated design. To remove the duplicate vias, do the following step:

1. Choose *Tools – Database Check*.

Related Topics

- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout Designs to Allegro X PCB Editor designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of your Translated Design
- Deleting Shape Islands
- Enabling Pin Number Visibility in Your Allegro X PCB Editor Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of your Translated Allegro X PCB Editor Design
- Removing Ratsnests
- Restricting Layers to be Routed in the translated Allegro X PCB Editor Design
- Resolving False Pin-to-Pin Spacing (Cearance) Errors in the Translated Allegro X PCB Editor Design
- Resizing Pad Size in Your Allegro X PCB Editor Design
- Displaying Footprint Information in Translated Allegro X PCB Editor Designs

Backannotating Translated Allegro PCB Editor Design Changes to Allegro Design Entry CIS Schematic

You use backannotation to synchronize the design file with the changes done in the board file. Backannotation ensures that the physical board design remains consistent with the logical schematic design. However, if you want to backannotate changes in your translated Allegro X PCB Editor design to your original Allegro X Design Entry CIS (also applies to OrCAD X Capture and AllegroORCADX Capture CIS schematic), then make sure that you use the steps given below.

⚠ Ensure that the inter-tool communication between Allegro X Design Entry CIS and Allegro X PCB Editor is working properly.

⚠ Ensure that you keep a backup of the schematic that was used in the Allegro X Design Entry CIS - Layout flow before backannotating your Allegro X PCB Editor changes to the Allegro X Design Entry CIS schematic.

1. Create a Layout netlist (`.mnl`) in Allegro X Design Entry CIS. For more information, see the Allegro X Design Entry CIS documentation.
2. Use the Layout netlist file to create a Layout board file (`.max`).
3. Convert the Layout design (`.max`) to a Allegro X PCB Editor design (`.brd`) using the Layout to Allegro X PCB Editor translator.
4. Start Allegro X PCB Editor and open the translated Allegro X PCB Editor design (`.brd`).
5. Choose *Tools – Reports*.
   The Reports dialog box appears.
6. Double-click the *Component Report* in the *Available Reports* list to select the report.
7. Click *Report* to generate the report.
8. Locate COMP_PACKAGE in the report.
9. Copy the footprint name from the component report and add it to the PCB footprint section in the *Property Editor* window of Allegro X Design Entry CIS.
10. From Allegro Design Entry CIS, create an Allegro X PCB Editor netlist. The following netlist files are generated: `PSTCHIP.DAT, PSTXNET.DAT`, and `PSTXRPT.DAT`.
11. Start Allegro X PCB Editor (if not already started).
12. Choose *File – Export – Logic*.

The Export Logic dialog box appears.

13. Select the *Design entry CIS* option under the *Logic* type section.
14. Enter or browse to the directory location of the Allegro X PCB Editor netlist file.
15. Click *Export Cadence* to export logic.
16. Save your Allegro X PCB Editor design.
17. Use Allegro X Design Entry CIS to backannotate changes from your translated Allegro X PCB Editor design. All the properties, such as PINSWAP and GATESWAP, update in your Allegro X Design Entry CIS schematic.

Related Topics

- report
- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout designs to Allegro X PCB Editor designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of your Translated Allegro X PCB Editor Design
- Removing Ratsnests
- Restricting Layers to be Routed in the translated Allegro X PCB Editor Design
- Resolving False Pin-to-Pin Spacing (clearance) Errors in the Translated Allegro X PCB Editor Design
- Resizing Pad Size in Your Allegro X PCB Editor Design
- Displaying Footprint Information in Translated Allegro X PCB Editor Designs

Enabling Pin Number Visibility in Board Design

After conversion, the pin numbers in your Layout design are not visible in your translated board design. You have to manually make the pin numbers visible in your translated design.

1. Choose *File – Export – Logic*.
   The Export Libraries dialog box appears.
2. Enter or browse to the directory location to which to export the libraries.
3. Choose *Setup – User Preferences*.
   The User Preferences Editor dialog box appears.
4. Choose the *Design_path* category from the Categories list.
5. Set the path for `padpath` and `psmpath` environment variables to point to the directory location to which you exported the libraries.
6. Choose *Place – Update Symbols*. The *Update Symbols* dialog box appears.
7. Select the *Package symbols* check box in the *Select definitions for update* list.
8. Select the *Update symbol padstacks* check box.
9. Click *Refresh.* The design is updated, and the pin numbers are visible.
10. Click *Close* to close the Update Symbols dialog box.

Related Topics

- enved
- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout Designs to Allegro X PCB Editor Designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Specifying Spacing Constraints on Nets of your Translated Allegro X PCB Editor Design
- Removing Ratsnests
- Restricting Layers to be Routed in the Translated Allegro X PCB Editor Design
- Resolving False Pin-to-Pin Spacing (Clearance) Errors in the Translated Allegro X PCB Editor Design
- Resizing Pad Size in Your Allegro X PCB Editor Design
- Displaying Footprint Information in Translated Allegro X PCB Editor Designs

Changing Grid Settings

The route grid settings in your Layout design are not converted to your translated board design. You can change the grid settings manually for your translated design.

1. Choose *Setup – Design Parameters*.
   The Design Parameters Editor dialog box appears.
2. Click *Setup Grids* on the Display tab. The Define Grid dialog box appears.
3. Change the non etch and etch grid values in the dialog box.
4. Click *OK.* The grid settings update in your design.

Related Topics

- prmed
- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout Designs to Allegro X PCB Editor Designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro X PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Removing Ratsnests
- Restricting Layers to be Routed in the translated Allegro X PCB Editor Design
- Resolving False Pin-to-Pin Spacing (clearance) Errors in the Translated Allegro X PCB Editor Design
- Resizing Pad Size in Your Allegro X PCB Editor Design
- Displaying Footprint Information in Translated Allegro X PCB Editor Designs

Specifying Spacing Constraints on Nets of Your Translated Board Design

The spacing constraints defined for a particular net in your Layout design are not transferred to your translated layout or board design. You have to manually define them in your translated board design. However, the global spacing constraints such as, Track to Track, Via to Via, Pin to Pin, and Pin to Line in your Layout design are transferred to the translated board design.

To specify spacing constraints for nets in your translated Allegro X PCB Editor design, do the following steps:

1. Choose *Setup – Constraints – Spacing*.

Related Topics

- cmgr_spac
- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout Designs to Allegro X PCB Editor Designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro X PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in your Allegro X PCB Editor Design
- Restricting Layers to be Routed in the translated Allegro X PCB Editor Design
- Resolving False Pin-to-Pin Spacing (clearance) Errors in the Translated Allegro X PCB Editor Design
- Resizing Pad Size in Your Allegro X PCB Editor Design
- Displaying Footprint Information in Translated Allegro X PCB Editor Designs

Removing Ratsnests

1. Choose *Tools – Derive Connectivity* (command).
   The Derive Connectivity dialog box appears.
2. Select the *Convert Lines to Connect Lines* check box.
3. Select the *Convert Figure Stackups to Vias* check box.
4. Click *OK*. The redundant ratsnests are removed from your design.

Related Topics

- derive connectivity
- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout Designs to Allegro X PCB Editor Designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro X PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in your Allegro X PCB Editor Design
- Changing Grid Settings
- Resolving False Pin-to-Pin Spacing (clearance) Errors in the Translated Allegro X PCB Editor Design
- Resizing Pad Size in Your Allegro X PCB Editor Design
- Displaying Footprint Information in Translated Allegro X PCB Editor Designs

Restricting Layers to be Routed in the Translated Board Design

In layout, if you have set any restrictions for layers on nets, then these restrictions are not transferred to the translated board design. For example, if you have defined a constraint like, *Net A1* should be restricted to only the Top and Bottom layers, then this type of restriction of layers for nets is not transferred to your translated board design.

To restrict the layers to be routed in your translated board design, do the following steps:

1. Choose *Setup – Constraints – Spacing*.

Related Topics

- cmgr_spac
- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout Designs to Allegro X PCB Editor Designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro X PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in your Allegro X PCB Editor Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of your Translated Allegro X PCB Editor Design
- Resizing Pad Size in Your Allegro X PCB Editor Design
- Displaying Footprint Information in Translated Allegro X PCB Editor Designs

Resolving False Pin-to-Pin Spacing (Clearance) Errors in the Translated Board Design

1. Choose *Tools – Padstack – Modify Design Padstack*.
2. Select the padstack used for the pin from the list and click *Edit.* The Padstack Designer dialog box opens with the padstack definition loaded. The banner of the Padstack Designer lists the name of the padstack that you are modifying.
3. Select the appropriate layer.
4. In the *Regular Pad* group, specify the same padstack width as shown in the *Width* text box.
5. Choose *File – Update To Design* to load the updated padstack into your design.
6. The pin-to-pin spacing errors are removed.

Related Topics

- padeditdb
- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout Designs to Allegro X PCB Editor Designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro X PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in your Allegro X PCB Editor Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of your Translated Allegro X PCB Editor Design
- Removing Ratsnests
- Displaying Footprint Information in Translated Allegro X PCB Editor Designs

Resizing Pad Size in the Board Design

1. Choose *Tools – Padstack – Modify Design Padstack*.
2. Select the padstack you want to edit from the list and click *Edit.* The Padstack Designer dialog box opens with the padstack definition loaded. The banner of the Padstack Designer lists the name of the padstack that you are modifying.
3. Select a plane layer (GND, VCC).
4. Specify padstack width for *Regular Pad* in the *Width* text box. The pad width should be same as that of non-plane layers.
5. Choose *File – Update To Design* to load the padstack into your design.
6. The *Update to Design* option is available only if you invoke the Padstack Editor from a current design instead of using the Padstack Editor as a standalone program.

Related Topics

- padeditdb
- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout Designs to Allegro X PCB Editor Designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro X PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in your Allegro X PCB Editor Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of your Translated Allegro X PCB Editor Design
- Removing Ratsnests
- Restricting Layers to be Routed in the translated Allegro X PCB Editor Design
- Displaying Footprint Information in Translated Allegro X PCB Editor Designs

Displaying Footprint Information in Translated Board Designs

To update and replace quickview (footprint) information stored in your drawings, symbols, and board files:

1. Type `qvupdate` in the Command window and run this utility on your `.brd` and `.dra` files.

Related Topics

- qvupdate
- orcad in
- OrCAD Layout to Allegro Dialog Box
- Creating a Catalog of the Library Prior to Translating
- Converting Layout Designs to Allegro X PCB Editor Designs
- Using Footprint Information after Library Conversion
- Troubleshooting Translation Issues
- Defining Manual Voids on the Inner Copper Shapes of Your Translated Design
- Deleting Shape Islands
- Defining Thermal Relief Flash Symbols for Padstacks
- Removing Duplicate Vias
- Backannotating Translated Allegro X PCB Editor Design Changes to Allegro Design Entry CIS Schematic
- Enabling Pin Number Visibility in your Allegro X PCB Editor Design
- Changing Grid Settings
- Specifying Spacing Constraints on Nets of your Translated Allegro X PCB Editor Design
- Removing Ratsnests
- Restricting Layers to be Routed in the translated Allegro X PCB Editor Design
- Resolving False Pin-to-Pin Spacing (clearance) Errors in the Translated Allegro X PCB Editor Design

osdelete

The `osdelete` command deletes files or directories independently of the operating system.