Design Rules Reference

Summary

This comprehensive reference provides detailed information on setting up design rules for a PCB design. It also covers addition of rule-based parameters to objects in the schematic and Design Rule Checking. Detailed information for each of the individual rule types and their associated constraints is also provided.

PCB design is no longer a matter of placing tracks to create connections. High speed logic combined with smaller and more complex packaging technologies place new demands on the PCB Designer. It is not possible to satisfy all the requirements of the design by only considering the clearance between tracks, pads and vias. Designs today can also require that you apply specific requirements to individual nets, components or regions of the board as well as considering such issues as crosstalk, reflections and net lengths. Altium Designer's PCB Editor allows you to define design rules that monitor and test for these and other requirements.

Design rules collectively form an instruction set for the PCB Editor to follow. Each rule represents a requirement of your design and many of the rules, eg. clearance and width constraints, can be monitored as you work by the online Design Rule Checker (DRC). Certain rules are monitored when using additional features of the software, for example routing-based rules when using the Situs Autorouter to route a design, or signal integrity-based rules used by the Signal Integrity Analyzer when performing a detailed signal integrity analysis of a design.

The PCB Editor provides a powerful interface from where you can define the various design rules as required. The rules themselves are divided into the following categories:

- Electrical
- Placement
- Routing
- Manufacturing
- SMT
- Plane
- Mask
- Testpoint
- High Speed
- Signal Integrity

With a well-defined set of design rules, you can successfully complete board designs with varying and often stringent design requirements. This is further enhanced by the fact that the PCB Editor allows you to export and import rule sets, enabling you to store and retrieve your favorite design rule configurations, depending on the job at hand.
Creating and Editing Design Rules in the PCB Document

To set up new design rules or edit existing ones for the current design, from the PCB document select Design » Rules to open the PCB Rules and Constraints Editor dialog.

In the folder-tree pane on the left side of the dialog, each of the ten design rule categories are listed under the Design Rules folder. Click on the root folder (Design Rules) to access a summary listing - in the main editing window of the dialog - of all specific rules that have been defined for all design rule types, across all categories (as shown in the previous image).

Click on a category to list all specific rules that have been defined for all associated design rule types of that category.

Click on a rule type to list all specific rules that have been defined for that type.

In each case - whether you have clicked on the root folder, a category or a type - the main editing window of the dialog will display the following summary information for each defined rule:

- the rule name
- the type of rule it is
- the rule category it belongs to
- the scope of the rule (i.e. what object(s) it applies to)
- the constraint attributes that have been defined for the rule
- the rule's priority.

You can also enable/disable a rule from within these summary list views.
Creating a New Rule

A new rule can be created by right-clicking on the required rule type in the folder-tree pane and choosing **New Rule** from the pop-up menu.

Alternatively, pressing the **Rule Wizard** button at the bottom of the dialog will launch the **New Rule Wizard**. Follow the pages in this wizard to quickly create a new design rule of any type.

The new rule will be added to the folder-tree and will also appear in the summary list for that rule type. The rule name will appear bold to distinguish it as being new and yet to be ‘applied’.

Editing a Rule

To edit the scope and constraint attributes for a rule, either click on the entry for the rule in the folder-tree pane or double-click on its entry in a summary list. The main editing window of the dialog will change to give access to the controls for defining the scope and constraint attributes for that rule. The example image below shows the controls accessed for a Solder Mask Expansion rule.

Changes made to existing rule definitions are highlighted in both the folder-tree pane and the applicable summary lists. Such entries are distinguished by the rule name becoming bold and an asterisk displayed to the right of the name. The asterisk is used to reflect that the rule is an existing rule that has been modified, rather than a newly created rule (which is displayed bold without an asterisk).
**Deleting a Rule**

To delete a design rule, right-click on its entry in the folder-tree pane of the PCB Rules and Constraints Editor dialog and choose the **Delete Rule** command from the pop-up menu that appears.

The rule name will appear bold with strike-through highlighting to distinguish it as being a deletion that is yet to be ‘applied’.

**Effecting Rule Changes**

To effect changes, either press the **Apply** button or press **OK**. Use of the former will allow you stay in the dialog to carry out further changes.

**Notes**

When a new rule is added from within the PCB Rules and Constraints Editor dialog it will initially be given a default name based on the specific type of rule. For example, if you add a new Clearance rule, the default name will be Clearance. If this default naming is not changed, adding another new rule of the same type will result in the same rule name with an incremented numerical suffix (i.e. Clearance_1, Clearance_2, and so on).

When a new rule is created for a particular rule type, it is automatically given priority 1. If any other rules of that type exist, their priorities will be shifted, by one, accordingly. They are then considered to be modified - even though you may not have specifically modified them at the scope/constraint level. All such existing rules of that type will therefore be displayed in the modified state (bold with asterisk).

If you do not want to use a design rule, but may wish to use it in the future, rather than delete it you can disable it. Toggle the corresponding **Enable** option for the rule in one of the relevant summary lists.

Many rule types have default rules created when you open a new PCB document. In a similar fashion, if you delete all specific rules for one of those rule types, the default rule will be re-added automatically. For information on the default rules that are created, see the **Default Design Rules Created with a New PCB Document** topic.
When defining the scope of a design rule - the extent of its application - you are essentially building a query to define the member objects that are governed by the rule. Use the options available in the dialog to build the query required.

Depending on whether the rule is unary or binary, you will need to define one or two scopes respectively. For more information, see the Unary and Binary Design Rules topic.

Basic options allow you to quickly generate scope queries that target:

- all design objects
- all objects in a specific net
- all objects in a specific net class
- all objects on a specific layer
- all objects in a specific net and on a specific layer.

The available drop-down fields will populate in accordance with the basic option chosen. When using the Net, Net Class or Layer option, use the top drop-down field to choose from:

- all defined nets in the design,
- all defined net classes in the design or
- all currently enabled layers in the design, respectively.

When using the Net and Layer option, the top drop-down will contain the list of nets in the design, with the bottom drop-down containing the list of enabled layers.

As you select an option, the corresponding query will appear in the Full Query region of the dialog.

The Advanced (Query) option enables you to write your own, maybe more complex, but also more specific query. Two facilities are available to provide aid in the creation of queries - the Query Builder and the Query Helper. These facilities can be very useful if you are unsure of the syntax of a query or the possible keywords that you may want to use.

### Using the Query Builder

Click the Query Builder button to open the Building Query from Board dialog, which enables you to create a query for targeting specific objects in the design document, by simple construction of a string of ANDed and/or ORed conditions.

The left-hand section of the dialog is where you specify the condition(s) that you require to target the set of objects needed. Initially the entry in the Condition Type/Operator column will be Add first condition. Clicking on this field will reveal a list of condition types.

The condition types listed will only reflect those relevant to building the scope for the current rule type.

Choose the first condition and click in the Condition Value column to access a drop-down list of possible values for that condition type. As you define a condition in the left-hand section of the dialog, a preview of the currently built query is shown in the right-hand section.

Continue to add further conditions to narrow down your target set of design objects as required. Conditions can be ANDed or ORed together. The default logical operator is AND. To change the logical operator between conditions, click on the AND or OR entry in the Condition Type/Operator column and select the required operator.
Design Rules Reference

The preview of the query will update accordingly.

When the expression for the query has been defined as required, clicking OK will load the expression into the Full Query region of the PCB Rules and Constraints Editor dialog. Note that the scope option will automatically be switched to Advanced (Query).

Specifying Precedence

The and buttons at the top of the dialog essentially allow you to add and remove brackets around the presently selected condition (increasing and decreasing indent). This allows you to create precedence for certain logically ANDed or logically ORed conditions. For example, consider the following query:

\[(\text{InNet}('+12V') \text{ AND } (\text{OnLayer}('TopLayer') \text{ AND } \text{IsTrack}))\]

In this case, the first condition has been set to the condition type Belongs to Net, with value +12V.

Another condition has then been added, using the condition type Exists on Layer, with the value TopLayer.

At this stage, with the second condition selected in the dialog, the right arrow button has been clicked. Brackets have been added around the second condition and now the possibility to add a condition within that pair of brackets is given.

The third condition with condition type Object Kind and value Track is then added within the brackets.

Use the Show Level list to control the display of levels in your structured string of conditions, essentially expanding/collapsing the display of brackets. Adding brackets creates a new level. You can display levels 1-5. For any further levels added, use the Show All Levels option. Alternatively, click on the expand (+) or contract (-) symbols (associated with a bracketed condition) to
show the next level(s) or hide the current level (and all levels below) respectively. The and buttons can also be used to expand or collapse the currently selected condition.

Use the and buttons to move a selected condition in the query string being built. For a condition that has sub-levels (i.e. a bracketed condition), any condition in the level structure can be moved. When levels are expanded, a condition can be moved down or up through the levels. When levels are collapsed, a condition will be moved over the level structure.

To delete a condition, select it and either click the button, or use the Delete key.

Using the Query Helper

To use the Query Helper, ensure that the Advanced (Query) option is enabled and then click on the Query Helper button to open the Query Helper dialog. The underlying query engine analyzes the PCB design and lists all available objects, along with generic keywords for use in queries.

Use the Query region of the dialog to compose a query expression. You can type directly within the region. As you type, a context-sensitive prompt list of possible keywords or objects will appear as an aid.

The Categories region of the dialog provides access to available PCB Functions, PCB Object Lists and System Functions, which can be used to create the query expression. As you click on a sub-category within each of these three areas, the corresponding list of keywords or objects will be displayed in the region to the right.

Locate the keyword or object that you wish to use in the query string, using the Mask field if need be, and then double-click on the entry. The entry in the Name column will be inserted at the cursor position in the Query region of the dialog.

The mid-section of the dialog provides a range of operators (as buttons) for use when constructing an expression. Use the Check Syntax button to verify that an expression is syntactically correct.

When the expression for the query has been defined as required, clicking OK will load the expression into the Full Query region of the PCB Rules and Constraints Editor dialog.
**Design Rules Reference**

**Notes**

You can type your own specific query for the rule scope directly into the **Full Query** region of the PCB Rules and Constraints dialog, or elaborate on one of the basic query entries. In either case, the **Query Kind** option will change to **Advanced (Query)** and access to the Query Helper dialog will be made available, should you wish or need to use it.

The Query Builder (Building Query from Board dialog) is a simpler method of constructing a query, using sensitive condition types and values that only allow you to build using relevant ‘building blocks’. For advanced query construction, with full keyword specification and operator syntax, use the Query Helper dialog.

When using the Query Helper dialog to construct a query, clicking inside a typed keyword or on a keyword in one of the available lists and pressing **F1**, will launch help for that particular keyword.

When using the Query Builder:

- You can adjust any condition in your query string, at any time, by clicking on the entry for that condition in the **Condition Type/Operator** column and choosing the required new condition from the available entries in the drop-down list. The preview of the query expression will update accordingly.
- The currently selected condition or logical operator in the left side of the dialog is visually confirmed in the preview section of the dialog by the entry appearing in red text.
- The **CTRL + Up Arrow** and **CTRL + Down Arrow** keyboard shortcuts can also be used to move the selected condition entry up or down in the structure respectively.
- The **CTRL + Right Arrow** and **CTRL + Left Arrow** keyboard shortcuts can also be used to increase or decrease indent at the selected position in the structure (add/remove brackets) respectively.

When building query expressions using the Query Helper:

- It is highly advisable to use brackets whenever there is any possibility whatsoever that the query might not be correctly interpreted.
- Brackets have the highest precedence within an order of precedence that has been defined for the various operators provided and which determines how queries are interpreted by the software (whenever the user has not provided brackets). The sequence of this order is as follows:

  **Brackets**
  Not
  ^, *, /, Div, Mod, And
  +, -, Or, Xor
  =, <>, <, >, <=, >=
  &&, ||

  This order of precedence is similar to that used in Pascal type languages. However, generous usage of brackets removes doubt and makes the resulting queries easier to read by others.

- Ambiguities are resolved by working from left to right.
- Parentheses are evaluated from inside to outside and equal levels are done left to right.

The rule scope (Full Query) is checked automatically. If you switch from a rule with an incorrectly defined scope to another rule, the former will appear highlighted in red in the folder-tree pane of the dialog.

Similarly, if you close the PCB Rules and Constraints Editor dialog and rules with incorrect scope definitions exist, a dialog will appear alerting you to this fact and asking if you wish to correct them. If you click Yes, the folder-tree pane will be fully expanded, enabling you to glance down and easily catch the offending, red-highlighted rules.
Browsing Rules using the PCB Panel

A list of currently defined rules can be displayed in the PCB panel by enabling the Rules mode for the panel.

All rules can be viewed or only those rules associated with a particular rule type. The Rule Classes region will only include a rule type if a specific rule of that type has been defined for the active design.

Double-clicking on a rule entry (or right-clicking on a rule entry and selecting Properties from the pop-up menu) will open the relevant Edit PCB Rule dialog, from where you can edit the scope of the rule and also the specific rule constraints.

As you click on a specific rule in the Rules region of the panel, filtering will be applied, using the rule as the scope of the filter. Only those design objects that fall under the scope of the rule will be filtered, the visual result of which (in the main design window) is determined by the highlighting options enabled (Mask/Dim/Normal, Select, Zoom). Multiple rule entries can be selected using standard SHIFT + Click and CTRL + Click features.

This allows you to examine which objects an enabled rule applies to when creating your query for the rule scope(s). Because you can edit a rule directly from the panel, you can tweak your query until the desired objects are captured by the scope(s).

For more information about the PCB panel, press F1 when the cursor is over the panel.
Unary and Binary Design Rules

There are two types of design rules - unary and binary. Unary rules apply to one object, or each object in a set of objects. As a consequence, unary design rules have one rule scope. Binary rules apply between two objects, or between any object in one set to any object in a second set. As a consequence, binary design rules have two rule scopes.

An example of a unary rule is the Solder Mask Expansion rule. This rule applies individually to each pad identified by the rule scope.

A selection of examples of a binary rule is the Clearance rule, which applies between any copper object in the first set and any copper object in the second set, as identified by the separate queries of the two rule scopes.

Viewing Applicable Unary Rules

For any placed object in the current design, you can quickly access information about which unary design rules apply to that object. Position the cursor over the object (or select it), right-click and select Applicable Unary Rules from the pop-up menu. The Applicable Rules dialog will appear.

All defined design rules that could be applied to the selected object are analyzed and listed in the dialog. The specific constraints for each rule are also displayed. Each rule that is listed in the dialog will have either a tick (✓) or a cross (✗) next to it. A tick indicates that this is the rule with the highest priority out of all applicable rules of the same type and is the rule currently applied. Lower priority rules of the same type are listed with a cross next to them, indicating that they are applicable but, as they are not the highest priority rule, they are not currently applied. Any rules that would apply to the object but are currently disabled also have a cross next to them and are shown using strike through highlighting.
Viewing Applicable Binary Rules

In a similar fashion, you can also access information about the binary design rules that apply between two placed objects in a design. Position the cursor over any object, right-click and select **Applicable Binary Rules** from the pop-up menu. You will be prompted to select two objects in the design. Position the cursor over each object in turn and click or press **ENTER**. The **Applicable Rules** dialog will appear, displaying all binary design rules that apply to those objects.

![Applicable Binary Rules Dialog](image)

**Note**: If the two objects that you select do not have any binary rules applied to them, the **Applicable Rules** dialog will not open.

**Notes**

If, rather than seeing which rules apply to an object (or between two objects) you would prefer to pick a rule and see which objects that rule applies to, this can be achieved from the **PCB** panel, when configured in **Rules** mode. As you click on a specific rule in the **Rules** region of the panel, filtering will be applied, using the rule as the scope of the filter. Only those design objects that fall under the scope of the rule will be filtered, the visual result of which (in the main design window) is determined by the highlighting options enabled (**Mask/Dim/Normal, Select, Zoom**).

![PCB Panel](image)

For more information about the **PCB** panel, press **F1** when the cursor is over the panel.
**Design Rules Reference**

**Rule Priority**

As you create a new rule, it is given a priority setting. This setting defines the order in which multiple rules of the same type are applied when, for example, performing a Design Rule Check.

<table>
<thead>
<tr>
<th>Name</th>
<th>Priority</th>
<th>Enabled</th>
<th>Type</th>
<th>Category</th>
<th>Scope</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>VCC</td>
<td>1</td>
<td></td>
<td>Width</td>
<td>Routing</td>
<td>Inf</td>
<td>Width = 10mil</td>
</tr>
<tr>
<td>+12V</td>
<td>2</td>
<td></td>
<td>Width</td>
<td>Routing</td>
<td>Inf</td>
<td>Width = 10mil</td>
</tr>
<tr>
<td>-12V</td>
<td>3</td>
<td></td>
<td>Width</td>
<td>Routing</td>
<td>Inf</td>
<td>Width = 10mil</td>
</tr>
</tbody>
</table>

Each new rule you add for the same rule type, will be given the highest priority setting, i.e. 1. You can change the priority order that exists for rules of the same type by clicking on the Priorities button in the PCB Rules and Constraints Editor dialog. The Edit Rule Priorities dialog opens.

Initially, the dialog will list all rule instances for the rule type that is currently selected in the PCB Rules and Constraints Editor dialog. Use the Rule Type field to change the rule type and hence list the specific rules defined for that type. The defined rules are listed in order of current priority - from 1 (highest priority) downwards.

Select a rule entry and use the Increase Priority and Decrease Priority buttons to move it up or down in the priority order respectively.

**Notes**

Multiple rules of the same type can be set up. It may arise that a design object is covered by more than one rule with the same scope. In this instance, a contention exists. All contentions are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

When a new rule is created for a particular rule type, it is automatically given priority 1. If any other rules of that type exist, their priorities will be shifted (lowered), by one, accordingly. They are then considered to be modified - even though you may not have specifically modified them at the scope/constraint level. All such existing rules of that type will therefore be displayed in the modified state (bold with asterisk).
Application of Design Rules

Different design rules are applied in different situations. Certain rules can be applied as you design, by enabling the online design rule check (DRC) feature. A violation of a rule is flagged as soon as the violation occurs during placement. You may prefer to design first and check for violations later. If this is the case, you could enable the batch DRC feature, which will apply a certain set of rules when launched and provide feedback via a report. Some rules are only applied at certain times and during software operations, such as autorouting, autoplacement and manufacturing output generation.

The PCB Editor only applies each rule when it is appropriate. A rule's definition specifies when that particular rule is applied. To reiterate, each rule is applied in one or more of the following situations:

- **Online Design Rule Check (DRC):** running in the background, as you work, flagging and/or automatically preventing design rule violations of certain rule types
- **Batch DRC:** allows you to manually run a DRC at any time during the board design process, on all enabled rule types, and obtain a report
- **During a software operation:** certain rules are monitored during a software operation including: polygon pour, autorouting, autoplacement and output generation. Examples of these include the mask expansion rule which is monitored during output generation and the routing via style rule which is monitored during autorouting.

The following table summarises where each of the individual rule types are applied:

<table>
<thead>
<tr>
<th>Rule</th>
<th>Category</th>
<th>Auto-router</th>
<th>Online DRC</th>
<th>Batch DRC</th>
<th>Output Generation</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clearance</td>
<td>Electrical</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>Interactive routing, Polygon placement</td>
</tr>
<tr>
<td>Short-Circuit</td>
<td>Electrical</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unrouted Net</td>
<td>Electrical</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unconnected Pin</td>
<td>Electrical</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parallel Segment</td>
<td>High Speed</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td>High Speed</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matched Net Lengths</td>
<td>High Speed</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>Equalize Net Lengths command</td>
</tr>
<tr>
<td>Daisy Chain Stub Length</td>
<td>High Speed</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vias Under SMD</td>
<td>High Speed</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maximum Via Count</td>
<td>High Speed</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Minimum Annular Ring</td>
<td>Manufacturing</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Acute Angle</td>
<td>Manufacturing</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hole Size</td>
<td>Manufacturing</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Layer Pairs</td>
<td>Manufacturing</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>Interactive routing</td>
</tr>
<tr>
<td>Solder Mask Expansion</td>
<td>Mask</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Paste Mask Expansion</td>
<td>Mask</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Room Definition</td>
<td>Placement</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>Arrange Within Room command</td>
</tr>
<tr>
<td>Component Clearance</td>
<td>Placement</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>Autoplacement (Cluster Placer)</td>
</tr>
<tr>
<td>Component Orientations</td>
<td>Placement</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Autoplacement (Cluster Placer)</td>
</tr>
<tr>
<td>Permitted Layers</td>
<td>Placement</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Autoplacement (Cluster Placer)</td>
</tr>
<tr>
<td>Nets to Ignore</td>
<td>Placement</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Autoplacement (Cluster Placer)</td>
</tr>
<tr>
<td>Height</td>
<td>Placement</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>Autoplacement PCB3D Editor</td>
</tr>
<tr>
<td>Power Plane Connect Style</td>
<td>Plane</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓ Internal Planes</td>
</tr>
<tr>
<td>Power Plane Clearance</td>
<td>Plane</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓ Internal Planes</td>
</tr>
<tr>
<td>Rule</td>
<td>Category</td>
<td>Auto-router</td>
<td>Online DRC</td>
<td>Batch DRC</td>
<td>Output Generation</td>
<td>Other</td>
</tr>
<tr>
<td>-------------------------</td>
<td>------------</td>
<td>-------------</td>
<td>------------</td>
<td>-----------</td>
<td>-------------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>Polygon Connect Style</td>
<td>Plane</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Polygon placement</td>
</tr>
<tr>
<td>Broken plane</td>
<td>Plane</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td>Additional split plane error report</td>
</tr>
<tr>
<td>Dead Copper</td>
<td>Plane</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td>Additional split plane error report</td>
</tr>
<tr>
<td>Starved Thermal</td>
<td>Plane</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Additional split plane error report</td>
</tr>
<tr>
<td>Width</td>
<td>Routing</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>Interactive routing</td>
</tr>
<tr>
<td>Routing Topology</td>
<td>Routing</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Routing Priority</td>
<td>Routing</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Routing Layers</td>
<td>Routing</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Routing Corners</td>
<td>Routing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Third party Autorouters (e.g. Spectra)</td>
</tr>
<tr>
<td>Routing Via Style</td>
<td>Routing</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td>Interactive routing</td>
</tr>
<tr>
<td>Fanout Control</td>
<td>Routing</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td>Interactive routing</td>
</tr>
<tr>
<td>Differential Pairs Routing</td>
<td>Routing</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>Interactive routing</td>
</tr>
<tr>
<td>Signal Stimulus</td>
<td>Signal Integrity</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Signal Integrity Analysis</td>
</tr>
<tr>
<td>Overshoot - Falling Edge</td>
<td>Signal Integrity</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Signal Integrity Analysis</td>
</tr>
<tr>
<td>Overshoot - Rising Edge</td>
<td>Signal Integrity</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Signal Integrity Analysis</td>
</tr>
<tr>
<td>Undershoot - Falling Edge</td>
<td>Signal Integrity</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Signal Integrity Analysis</td>
</tr>
<tr>
<td>Undershoot - Rising Edge</td>
<td>Signal Integrity</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Signal Integrity Analysis</td>
</tr>
<tr>
<td>Impedance</td>
<td>Signal Integrity</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Signal Integrity Analysis</td>
</tr>
<tr>
<td>Signal Top Value</td>
<td>Signal Integrity</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Signal Integrity Analysis</td>
</tr>
<tr>
<td>Signal Base Value</td>
<td>Signal Integrity</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Signal Integrity Analysis</td>
</tr>
<tr>
<td>Flight Time - Rising Edge</td>
<td>Signal Integrity</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Signal Integrity Analysis</td>
</tr>
<tr>
<td>Flight Time - Falling Edge</td>
<td>Signal Integrity</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Signal Integrity Analysis</td>
</tr>
<tr>
<td>Slope - Rising Edge</td>
<td>Signal Integrity</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Signal Integrity Analysis</td>
</tr>
<tr>
<td>Slope - Falling Edge</td>
<td>Signal Integrity</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Signal Integrity Analysis</td>
</tr>
<tr>
<td>Supply Nets</td>
<td>Signal Integrity</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>Signal Integrity Analysis</td>
</tr>
<tr>
<td>SMD To Corner</td>
<td>SMT</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMD To Plane</td>
<td>SMT</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMD Neck-Down</td>
<td>SMT</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Testpoint Style</td>
<td>Testpoint</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>Find and Set Testpoints command</td>
</tr>
<tr>
<td>Testpoint Usage</td>
<td>Testpoint</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>Find and Set Testpoints command</td>
</tr>
</tbody>
</table>
Design Rule Checking (DRC)

Design Rule Checking (DRC) is a powerful automated feature that checks both the logical and physical integrity of your design. Checks are made against any or all enabled design rules and can be made online, as you work, and/or as a batch check, with results listed in the Messages panel and a generated report. This feature should be used on every routed board to confirm that minimum clearance rules have been maintained and that there are no other design violations. It is particularly recommended that you always perform a design rule check prior to generating final artwork.

Configuring the DRC

Configuration for the check is carried out in the Design Rule Checker dialog, accessed by selecting the Design Rule Check command from the PCB Editor's Tools menu.

In the folder list on the left side of the dialog, each of the design rule categories whose rule types can be checked are listed under the Rules To Check folder. Click on the Rules To Check folder to list all checkable design rule types, across all categories, in the right side of the dialog.

Click on a category to list all associated (and checkable) design rule types for that category.

Enable/disable Online (where available) and/or Batch checking options for each rule type you wish to check. Use the options available from the right-click pop-up menu to enable/disable checks of all rule types, or to enable checks of all used rule types only.
Using Online DRC

To turn on the Online DRC feature, enable the **Online DRC** option on the **PCB Editor - General** page of the **Preferences** dialog (Tools » Preferences). Online Design Rule Checking runs in the background, as you work, flagging and/or automatically preventing design rule violations. Errors are highlighted in the document by outlining the violating object(s) in the current **DRC Error Markers** color, defined in the **System Colors** region of the View Configurations dialog (Design » Board Layers & Colors).

Using Batch DRC

Batch Design Rule Checking allows you to manually run a check at any time during the board design process. When setting up a batch DRC, various additional options can be defined by clicking on the **Report Options** folder, in the folder-tree pane of the Design Rule Checker dialog. These options include generation of a report. A batch DRC is initiated by clicking the **Run Design Rule Check** button, at the bottom left of the dialog. After the check has completed, all violations will appear listed as messages in the **Messages** panel.

If the **Create Violations** report option is enabled, clearance, length and width errors will be highlighted on the PCB document.

DRC Reports

Enable the **Create Report File** option in the Design Rule Checker dialog to generate a DRC report. Options available on the PCB Editor - Reports page of the Preferences dialog allow you to specify in which format the report is generated and whether a report is automatically displayed in the main design window. The following report formats are available:

- **TXT** - producing the Design Rule Check - PCBDocumentName.drc file
- **HTML** - producing the Design Rule Check - PCBDocumentName.html file
- **XML** - producing the PCBDocumentName.xml file.

By default, TXT and HTML formats are generated, with the HTML report being displayed after generation.
The report lists each rule that was tested, as specified in the Design Rule Checker dialog. Each violation that was located is listed with full details of any reference information, such as the layer, net name, component designator and pad number, as well as the location of the object. In the HTML format report, click on the entry for an offending object to cross probe directly to that object in the workspace.

**Interrogating Design Violations**

There are essentially three methods of interrogating design violations - from the Messages panel, from the PCB panel and directly within the design workspace. The first method is solely associated with having run a Batch DRC.

**From the Messages Panel**

After running a Batch DRC, double-clicking a violation message in the Messages panel will cross probe to the object(s) causing that violation in the main design window. **Note:** The Create Violations option must be enabled as part of the DRC Report options in the Design Rule Checker dialog, for the cross probing to work.

**From the PCB Panel**

When running an Online or Batch DRC, any rule violations associated with a rule class or individual rule will be listed in the Violations region of the PCB panel, when the panel is configured in Rules mode.

Clicking on a violation entry will apply filtering using the offending object(s) as the scope of the filter. The resulting view in the main design window will depend on the highlighting options enabled (Mask/Dim/Normal, Select, Zoom) at the top of the panel.

Double-clicking on a violation entry (or right-clicking on an entry and choosing Properties from the subsequent menu) will open the Violation Details dialog, which provides information about the rule being violated and the primitive(s) responsible.

From this dialog you can highlight the offending object (causing it to flash in the workspace) and jump to it, effectively providing zoom and center.
Each specific rule can be enabled or disabled with respect to Design Rule Checking - directly from the PCB panel - using the corresponding option under the On column. With this option disabled, the rule will not be included in the DRC and no violations of it will be listed.

Directly in the Workspace
You can interrogate violations associated with a particular design object directly within the PCB workspace. Position the cursor over the offending object you wish to interrogate, right-click and select Violations from the pop-up menu. In the example image, the offending track near the top-left corner (denoted by a yellow marker for ease of reference) is being investigated.

You can either choose to investigate individual violations associated with that object, or all violations. Choosing the former will cause the object(s) involved in the indicated violation to be zoomed and centered in the main design window. The zoom level can be adjusted by clicking the Zoom Level button in the PCB panel and using the slide control.

Irrespective of your choice, the Violation Details dialog will appear, providing details about the particular design rule that is being violated and the offending object(s). If you chose to Show All Violations, each of the individual violations will be listed in the dialog, from which to choose.

Highlight and jump to the object(s) causing the violation as required using the Highlight and Jump buttons respectively.

Resolving Design Rule Violations
DRC reports can appear quite daunting to the new PCB designer. The secret to keeping the process manageable is to develop a strategy. One strategy is to limit the number of violations that are reported. When setting up the report options in the Design Rule Checker dialog, set the Stop When Found feature to a small number. Another strategy is to run the DRC in a number of stages. If you find that the design contains a large number of violations, begin by enabling the rules one at a time. With experience you will develop a preferred approach to testing the various design rules.

Tracking Down Broken Nets
When a net is not completely routed, it is reported as a violation of the applicable Unrouted Net rule. The net is considered to be broken. The violation details for such a net will show how many sub-nets the net is broken into and the percentage of the net that is currently routed. Broken into two sub-nets indicates one break, broken into three sub-nets indicates two breaks, etc.

A break, or rather an unrouted connection, can be quickly found using the PCB panel. You could of course cross probe from the relevant violation message in the Messages panel, but by using the PCB panel, you can take advantage of the masking feature. To highlight the break, follow these steps:

- Configure the PCB panel in Rules mode
- In the Rule Classes region of the panel, click on the Un-Routed Net Constraint entry
- Click on the required entry in the Violations region of the panel. Filtering will be applied using the associated net as the basis for filtering. Ensure that the Mask or Dim option at the top of the panel is enabled. In the workspace, only the violating net will be displayed with all other objects masked or dimmed out. Click the Mask Level button at the bottom right of the main design window and use the controls to increase the masking or dimming levels of contrast as required.
• To highlight just the unrouted connection, double-click on the violation entry in the panel and use the Jump button in the Violation Details dialog that appears.

Notes
Turn the Online DRC feature on when manually routing to immediately highlight clearance, width and parallel segment violations.

Disabling a rule has the same effect as deleting the rule in terms of how it is handled by the Online and Batch DRC.

All currently displayed DRC error markers can be cleared from the document using the Reset Error Markers command, available from the main Tools menu. Clearing the error markers also clears the violations reported in the PCB panel. The violation messages that appear in the Messages panel after running a Batch DRC, will remain however. Bear in mind that this command just clears the error markers, it does not fix the violations. If you run a Batch DRC again, all violations will reappear in the PCB panel, along with the error markers in the workspace.

With respect to Batch DRC of signal integrity design rules:
• you must include a Layer Stack rule to be able to perform a signal integrity analysis
• for the design analysis to be correct you need to include appropriate Supply Nets design rules
• The DRC tests are worst-case. Each net is simulated from all possible output pins and the worst result is displayed.
Design Rules Reference

Default Design Rules Created with a New PCB Document

The following design rules are created by default with a new PCB document. Except for specific Fanout Control rules, all default rules have a scope (Full Query) of All, meaning they apply to the whole board. For default rule constraints, refer to the individual design rule topics.

Electrical
- Clearance
- Short-Circuit
- Un-Routed Net

Routing
- Width
- Routing Topology
- Routing Priority
- Routing Layers
- Routing Corners
- Routing Via Style
- Fanout Control
  * Fanout_BGA - with Full Query IsBGA
  * Fanout_LCC - with Full Query IsLCC
  * Fanout_SOIC - with Full Query IsSOIC
  * Fanout_Small - with Full Query (CompPinCount < 5)
  * Fanout_Default - with Full Query All
- Differential Pairs Routing

Mask
- Solder Mask Expansion
- Paste Mask Expansion

Plane
- Power Plane Connect Style
- Power Plane Clearance
- Polygon Connect Style

Testpoint
- Testpoint Style
- Testpoint Usage

Manufacturing
- Hole Size
- Layer Pairs

Placement
- Component Clearance
- Height
Adding Design Rule Directives to a Schematic Document

Design constraints (rules) can be defined prior to PCB layout, by adding parameters that are configured as design rule directives to the schematic source document(s). The scope of the corresponding PCB design rule, created when the design is transferred to the PCB document, is determined by the nature of the object to which the parameter (added as a rule) is assigned. The following table summarises the schematic parameter-to-PCB rule scope options that are supported:

<table>
<thead>
<tr>
<th>Add a Parameter (as a rule) to a...</th>
<th>From...</th>
<th>For a PCB rule scope of...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pin</td>
<td>the Parameters tab of the Pin Properties dialog</td>
<td>Pad</td>
</tr>
<tr>
<td>Port</td>
<td>the Parameters tab of the Port Properties dialog</td>
<td>Net</td>
</tr>
<tr>
<td>Wire</td>
<td>the Parameters dialog, after placing a PCB Layout Directive (Parameter Set object) on the wire using the Place » Directives » PCB Layout command</td>
<td>Net</td>
</tr>
<tr>
<td>Bus</td>
<td>the Parameters dialog, after placing a PCB Layout Directive (Parameter Set object) on the bus using the Place » Directives » PCB Layout command</td>
<td>Net Class</td>
</tr>
<tr>
<td>Component</td>
<td>the Parameters region of the Component Properties dialog</td>
<td>Component</td>
</tr>
<tr>
<td>Sheet Symbol</td>
<td>the Parameters tab of the Sheet Symbol dialog</td>
<td>Component Class</td>
</tr>
<tr>
<td>Sheet</td>
<td>the Parameters tab of the Document Options dialog (Design » Document Options)</td>
<td>All Objects</td>
</tr>
</tbody>
</table>

In each case, the method of adding a rule-based parameter is the same. From the respective tab or dialog, perform the following:

- use the Add as Rule button - the Parameter Properties dialog will appear, with the Name and Type fields set to Rule and STRING respectively and uneditable

- click the Edit Rule Values button to open the Choose Design Rule Type dialog. This dialog lists each of the rule categories and types that are available in the PCB document and for which you can validly add as a rule parameter in the schematic document.

- select a rule type and click OK (or double-click on it) to open its corresponding Edit PCB Rule (From Schematic) dialog, from where you can define the constraints for the rule.
Synchronicity through Unique IDs

When adding design rule parameters to objects on a schematic, a unique ID is given to each rule parameter. The same IDs are given to the corresponding design rules that are created in the PCB. With this Unique ID, the constraints of a rule can be edited on either the schematic or PCB side and the changes pushed through upon synchronization.

For example, consider adding a width rule parameter to a particular wire (associated with the net NETS2_1) on a schematic sheet, by placing a PCB Layout directive:

When you edit the default parameter entry for the directive, you will notice that the Unique ID field in the corresponding Parameter Properties dialog has a specific entry, as illustrated in the image below:

When the design change is passed on to the PCB - using the Synchronizer and generating and executing the relevant Engineering Change Order (ECO) - the rule will be created and added to the defined Width rules for the PCB and will have the same Unique ID assigned to it:
Exporting and Importing Design Rules

Design rules can be exported from, and imported to, the PCB Rules and Constraints Editor dialog. This allows you to save and load favorite rule definitions between different designs. To export, right-click anywhere within the folder-tree pane of the dialog and select Export Rules from the popup menu.

The Choose Design Rule Type dialog will appear. Select the rule types you wish to export and click OK. The Export Rules to File dialog will then appear, from where you can determine where, and under what name, the exported rules file (*.RUL) is to be stored.

To import, choose the Import Rules entry from the same right-click menu. The Choose Design Rule Type dialog will again appear. Select the rule types you wish to import and click OK. The Import File dialog will then appear, from where you can browse to and open, the particular PCB Rule file you wish to import.

Notes

Multiple rule types may be chosen for export or import using standard multi-select features (CTRL + Click and SHIFT + Click).

When importing, if rules of a chosen type already exist, you will be asked if you wish to clear the existing rules prior to import. If you click Yes, all existing rules of that type will effectively be deleted and the rules in the .rul file then brought in. If you click No, the existing rules will remain. Note however that in this latter case, if existing rules and imported rules have the same name, the imported rules will overwrite the existing ones.
Design Rule Reports

A report of currently defined design rules for all rule categories, a specific rule category or a specific rule type, can be generated. Right-click in the respective summary list, or over the respective entry in the folder-tree and choose the Report command in the pop-up menu.

The Report Preview dialog will appear, with the appropriate report already loaded.

Use this dialog to inspect the report using various page/zoom controls, before ultimately exporting it to file or printing it.

Notes

When exporting the report from the Report Preview dialog, the following file formats are supported:

- Microsoft Excel Worksheet (*.xls)
- Adobe PDF (*.pdf)
- Rich Text Format (RTF) (*.rtf)
- Web Page (*.htm; *.html)
- Web Layer (CSS) (*.htm; *.html)
- JPEG Image File (*.jpg)
- Window Bitmap File (*.bmp)
- TIFF Image File (*.tif)
- Quattro Pro Worksheet (*.wq1)
- Lotus 123 Worksheet (*.wk1)
Electrical Rules

Clearance

Description
Defines the minimum clearance allowed between any two primitive objects on a copper layer. Use this rule to ensure that routing clearances are maintained.

Constraints

Minimum Clearance - the value for the minimum clearance required. (Default = 10 mil)

Connective Checking - the scope of the rule with respect to the nets in the design. Can be set to one of the following:

- Different Nets Only (default) - constraint is applied between any two primitive objects belonging to different nets (e.g. two tracks on two different nets)
- Same Net Only - constraint is applied between any two primitive objects belonging to the same net (e.g. between a via and pad on the same net)
- Any Net - constraint is applied between any two primitive objects belonging to any net in the design. This is the most comprehensive of the three options and covers the possibility of the objects belonging to the same net or different nets.

Rule Classification
Binary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Online DRC, Batch DRC, interactive routing, autorouting and during polygon placement.

Notes
When defining the constraints for the rule, the Connective Checking option would typically be set to Different Nets Only. An example of when Same Net Only or Any Net could be used is to test for vias being placed too close to pads or other vias on the same net, or any other net.

When defining a clearance rule for a polygon, it is the primitives of the polygon that the rule is actually applied to, rather than the polygon itself. The keyword entry InPolygon (or InPoly) should be included in the Full Query in this case, instead of IsPolygon (or IsPoly). The specific polygon clearance rule must also be given a higher priority than any general clearance rule, if it is to have any effect.
**Short-Circuit**

**Description**
Tests for short circuits between primitive objects on the copper (signal and plane) layers. A short circuit exists when two objects that have different net names touch.

**Constraints**

- **Allow Short Circuit** defines whether the target nets falling under the two scopes (full queries) of the rule can be short-circuited or not. If you require two different nets to be shorted together, for example when connecting two ground systems within a design, ensure that this option is enabled. (Default = disabled).

**Rule Classification**
Binary

**How Duplicate Rule Contentions are Resolved**
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**
Online DRC, Batch DRC and during autorouting.

---

**Un-Routed Net**

**Description**
Tests the completion status of each net that falls under the scope (full query) of the rule. If a net is incomplete then each completed section (sub-net) is listed along with the routing completion. The routing completion is defined as:

\[
\text{(connections complete} \div \text{total number of connections)} \times 100.
\]

**Constraints**
None

**Rule Classification**
Unary

**How Duplicate Rule Contentions are Resolved**
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**
Batch DRC.

**Notes**
Some **split planes** DRC checks require the Un-Routed Net rule to be **Batch** enabled for them to work.
Unconnected Pin

Description
Detects pins that have no net assigned and no connecting tracks.

Constraints
None

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Online DRC and Batch DRC.
## Routing Rules

### Width

![Diagram of Width with Min Width, Preferred Width, and Max Width]

#### Description

Defines the width of tracks placed on the copper (signal) layers.

#### Constraints

<table>
<thead>
<tr>
<th>Constraint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min Width</td>
<td>Specifies the minimum permissible width to be used for tracks when routing the board.</td>
</tr>
<tr>
<td></td>
<td>Specifying a value here will apply to all signal layers. (Default = 10 mil).</td>
</tr>
<tr>
<td>Max Width</td>
<td>Specifies the maximum permissible width to be used for tracks when routing the board.</td>
</tr>
<tr>
<td></td>
<td>Specifying a value here will apply to all signal layers. (Default = 10 mil).</td>
</tr>
<tr>
<td>Preferred Width</td>
<td>Specifies the preferred width to be used for tracks when routing the board.</td>
</tr>
<tr>
<td></td>
<td>Specifying a value here will apply to all signal layers. (Default = 10 mil).</td>
</tr>
<tr>
<td>Characteristic Impedance Driven Width</td>
<td>If the design needs to be routed to strict impedance requirements, ensure that this option is enabled. When the rule is configured in this mode, the routing width required on each routing layer is calculated based on the specified impedance, using the appropriate equation (microstrip or stripline) and the physical parameters of the layer stack. Once the rule is defined, as you route a net that falls under the scope of the rule, the track width will automatically be set to the width required to meet the specified impedance for that layer. (Default = disabled).</td>
</tr>
<tr>
<td>Layers in layerstack only</td>
<td>Allows you to display and edit the width constraints for just the defined signal layers in the layer stack. When enabled, only the layers in the stack will be displayed in the Layer Attributes Table. When disabled, all signal layers will be displayed. (Default = enabled).</td>
</tr>
<tr>
<td>Layer Attributes Table</td>
<td>Displays all signal layers or only those defined in the layer stack, as controlled by the Layers in layerstack only option. The minimum, maximum and preferred routing widths are displayed, as well as other layer-specific information. The routing width fields can be set globally by defining a value in the individual width constraint fields, or individually by typing a width value directly into the table. When the Characteristic Impedance Driven Width option is enabled, the required width entries will be automatically calculated and entered for each layer in the table.</td>
</tr>
</tbody>
</table>

#### Rule Classification

Unary

#### How Duplicate Rule Contentions are Resolved

All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

#### Rule Application

The Preferred Width setting is obeyed by the Autorouter.

The Min Width and Max Width settings are obeyed by the Online DRC and Batch DRC. They also determine the range of permissible values that can be used during interactive routing (press TAB key while routing to change the trace width within the defined range). If a value is entered outside of this range a dialog will appear alerting you to this fact. You will be prompted to either continue, in which case the value will automatically be clipped, or cancel and change the value yourself.
Notes

The impedance equations used for calculating the impedance and trace width are accessed from the Layer Stack Manager dialog, by pressing the Impedance Calculation button. The subsequent dialog that appears - the Impedance Formula Editor dialog - contains impedance calculators for both microstrip and stripline impedance calculations.

Default equations are in place to calculate the impedance and the required trace width in order to satisfy that impedance when routing. Clicking the Helper button associated with an equation will open the Query Helper dialog, from where you can edit the equation if required.

When defining values for the minimum, maximum and preferred routing widths, the Layer Attributes Table will highlight any invalid entries by using red text. This could happen, for example, when you specify a minimum constraint value that is greater than the maximum constraint value.

The incorrect rule definition is further highlighted by the rule name becoming red in both the folder-tree pane and the respective summary lists.
Routing Topology

Description
The topology of a net is the arrangement or pattern of the pin-to-pin connections. By default, pin-to-pin connections of each net are arranged to give the shortest overall connection length. A topology is applied to a net for a variety of reasons; for high speed designs where signal reflections must be minimized the net is arranged with a daisy chain topology; for ground nets a star topology could be applied to ensure that all tracks come back to a common point.

Constraints

Topology - defines the topology to be used for the net(s) targeted by the scope (full query) of the rule. The following topologies can be applied:

- **Shortest**
  This topology connects all nodes in the net to give the shortest overall connection length.

- **Horizontal**
  This topology connects all the nodes together, preferring horizontal shortness to vertical shortness by a factor of 5:1. Use this method to force routing in the horizontal direction.

- **Vertical**
  This topology connects all the nodes together, preferring vertical shortness to horizontal shortness by a factor of 5:1. Use this method to force routing in the vertical direction.

- **Daisy-Simple**
  This topology chains all the nodes together, one after the other. The order they are chained is calculated to give the shortest overall length. If a source and terminator pad are specified, then all other pads are chained between them to give the shortest possible length. Edit a pad to set it to be a source or terminator. If multiple sources (or terminators) are specified, they are chained together at each end.

- **Daisy-Mid Driven**
  This topology places the source node(s) in the center of the daisy chain, divides the loads equally and chains them off either side of the source(s). Two terminators are required, one for each end. Multiple source nodes are chained together in the center. If there are not exactly two terminators the Daisy-Simple topology is used.

- **Daisy-Balanced**
  This topology divides all the loads into equal chains, the total number of chains equal to the number of terminators. These chains then connect to the source in a star pattern. Multiple source nodes are chained together.

- **Starburst**
  This topology connects each node directly to the source node. If terminators are present, they are connected after each load node. Multiple source nodes are chained together, as in the Daisy-Balanced topology.

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
During autorouting.

Notes
When using the Autorouter, routing completion time may be longer when using topologies other than Shortest.
Routing Priority

Description
Assigns a routing priority to the net(s) targeted by the rule. The Autorouter uses the assigned priority value to gauge the routing importance of each net in the design and hence determine which nets should be routed first.

Constraints
Routing Priority - the priority value assigned to the net(s) targeted by the scope (full query) of the rule. Enter a value between 0 and 100, whereby the higher the number assigned, the greater the priority when routing. (Default = 0).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
During autorouting.
Routing Layers

Description
Specifies which layers are allowed to be used for routing when using the Autorouter.

Constraints
Each of the signal layers currently defined for the design - as defined by the layer stackup - are listed. By default, the Allow Routing constraint for each layer will be enabled.

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
During autorouting.

Notes
The routing direction for each enabled signal layer in the design is defined as part of the Situs Autorouter setup. Directions are specified in the Layer Directions dialog, which in turn is accessed by clicking the Edit Layer Directions button in the Situs Routing Strategies dialog.

Note: Setting the routing direction for a layer to ANY can affect performance when autorouting. More efficient use of board area may be achieved by choosing a specific routing direction.
Routing Corners

Description
Specifies the corner style to be used during autorouting.

Constraints

Style - specifies which routing corner style to use. The following three styles are available:
- 90 Degrees
- 45 Degrees (default)
- Rounded

Setback - these two fields allow you to define a minimum and maximum value for the setback, when using the 45° and Rounded corner styles. The setback is the distance from the 'true' corner location (that which would exist if using the 90° style) to the point at which the Autorouter should begin its chamfering or rounding, in effect, controlling miter size or corner radius. (Default = 100mil for both fields).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
This rule is intended for use by third party Autorouters that implement 45° routing as a post process. It is not followed by the Situs Autorouter, which implements 45° routing as a native process.
Routing Via Style

Description
Specifies the routing via diameter and hole size.

Constraints

**Via Diameter**
- Specifies constraint range values to be adhered to with respect to the diameters of vias placed when routing the board. The following individual values are definable:
  - **Minimum** - the minimum permissible value for the via diameter. (Default = 50mil)
  - **Maximum** - the maximum permissible value for the via diameter. (Default = 50mil)
  - **Preferred** - the preferred value for the via diameter. (Default = 50mil).

**Via Hole Size**
- Specifies constraint range values to be adhered to with respect to the hole sizes of vias placed when routing the board. The following individual values are definable:
  - **Minimum** - the minimum permissible value for the via hole size. (Default = 28mil)
  - **Maximum** - the maximum permissible value for the via hole size. (Default = 28mil)
  - **Preferred** - the preferred value for the via hole size. (Default = 28mil).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
The **Preferred** via attributes are used by the Autorouter.

The **Minimum** and **Maximum** via attributes determine the range of permissible values that can be used during interactive routing, when you press the * shortcut key to toggle routing signal layers, or when you press the / shortcut key to connect to a plane layer. Press the TAB key while routing to change a value within its defined range. If a value is entered outside of its range, a dialog will appear alerting you to this fact. You will be prompted to either continue, in which case the value will automatically be clipped, or cancel and change the value yourself.

Notes
In order to control the size of blind and buried vias, individual rules can be set up targeting the different layer pairs. For example, to control the via size for blind vias between the top layer and mid layer 1, the following scope (Full Query) can be used:

```plaintext
(StartLayer = 'TopLayer') and (StopLayer = 'MidLayer1')
```

to control the via size for buried vias between mid layer 2 and mid layer 3, the following scope would be used:

```plaintext
(StartLayer = 'MidLayer2') and (StopLayer = 'MidLayer3')
```

Alternatively, instead of creating individual rules, you can expand the one rule query using ORs:

```plaintext
((StartLayer = 'TopLayer') and (StopLayer = 'MidLayer1')) or
((StartLayer = 'MidLayer2') and (StopLayer = 'MidLayer3'))
```
**Design Rules Reference**

**Fanout Control**

**Description**
Specifies fanout options to be used when fanning out the pads of surface mount components in the design that connect to signal and/or power plane nets. Fanout essentially turns an SMT pad into a thru hole pad, from a routing point of view, by adding a via and connecting track. This greatly increases the probability of successfully routing the board as a signal is made available to all routing layers instead of just the top or bottom layer. This is particularly needed in high-density designs where routing space is very tight.

**Constraints**

**Fanout Options**

**Fanout Style**
- specifies how the fanout vias are placed in relation to the SMT component. The following options are available:
  - **Auto** (default) - chooses the style most appropriate for the component technology and in order to give optimal routing space results
  - **Inline Rows** - fanout vias are placed within two aligned rows
  - **Staggered Rows** - fanout vias are placed within two staggered rows
  - **BGA** - fanout occurs in accordance with the specified BGA Options
  - **Under Pads** - fanout vias are placed directly under SMT component pads.

**Fanout Direction**
- specifies the direction to use for the fanout. The following options are available:
  - **Disable** - do not allow fanout with respect to the SMT components targeted by the rule
  - **In Only** - fanout in an inward direction only. All fanout vias and connecting track will be placed within the component's bounding rectangle
  - **Out Only** - fanout in an outward direction only. All fanout vias and connecting track will be placed outside of the component's bounding rectangle
  - **In Then Out** - Fanout all component pads in an inward direction to begin with. All pads that cannot be fanned out in this direction should be fanned out in an outward direction (if possible)
  - **Out Then In** - Fanout all component pads in an outward direction to begin with. All pads that cannot be fanned out in this direction should be fanned out in an inward direction (if possible)
  - **Alternating In and Out** (default) - Fanout all component pads (where possible) in an alternating fashion, first inward then outward.

**BGA Options**

**Direction From Pad**
- specifies the direction to use for the fanout. When a BGA component is fanned out, its pads are sectioned into quadrants, with fanout applied to the pads in each quadrant simultaneously. The following options are available:
  - **Away From Center** (default) - fanout for pads in each quadrant is applied following a 45° angle away from the component's center
  - **North-East** - All pads, in each quadrant, are fanned out in a North-Easterly direction (45° anti-clockwise from the horizontal)
  - **South-East** - All pads, in each quadrant, are fanned out in a South-Easterly direction (45° clockwise from the horizontal)
  - **South-West** - All pads, in each quadrant, are fanned out in a South-Westerly direction (135° clockwise from the horizontal)
  - **North-West** - All pads, in each quadrant, are fanned out in a North-Westerly direction (135° anti-clockwise from the horizontal)
  - **Towards Center** - fanout for pads in each quadrant is applied following a 45° angle toward the component's center. In most cases, uniformity of direction will not be possible due to required fanout space already taken by another pads' fanout via. In these cases, fanout will occur in the next available direction (North-East, South-East, South-West, North-West).
**Via Placement Mode**

- specifies how the fanout vias are placed in relation to the pads of the BGA component. The following options are available:
  - **Close To Pad (Follow Rules)** - fanout vias will be placed as close to their corresponding SMT component pads as possible, without violating defined clearance rules
  - **Centered Between Pads** (default) - fanout vias will be centered between the SMT component pads.

**Rule Classification**

Unary

**How Duplicate Rule Contentions are Resolved**

All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**

During interactive routing and autorouting.

**Notes**

The following default Fanout Control design rules are automatically created, covering the typical component package types available (listed in descending order of priority):

- Fanout_BGA - with a query of IsBGA
- Fanout_LCC - with a query of IsLCC
- Fanout_SOIC - with a query of IsSOIC
- Fanout_Small - with a query of (CompPinCount < 5)
- Fanout_Default - with a query of All

These rules can be edited or others defined, in accordance with your individual design requirements.

The style used for the fanout vias will follow the applicable Routing Via Style design rule(s). Additional track laid down as part of the fanout process from pad to via will follow the applicable Routing Width design rule(s).
Design Rules Reference

Differential Pairs Routing

Description
Specifies the maximum, minimum distances between tracks in a differential pair routing and the maximum distance that the pair is allowably uncoupled (i.e. less than minimum gap or more than maximum gap).

Constraints

- **Min Gap** - the value for the minimum permissible distance between the tracks in the differential pair. (Default = 10mil).
- **Max Gap** - the value for the maximum permissible distance between the tracks in the differential pair. (Default = 10mil).
- **Preferred Gap** - the value for the preferred distance between the tracks in the differential pair. (Default = 10mil).
- **Max Uncoupled Length** - the value for the maximum permissible distance in the differential pair where the tracks are considered uncoupled. (Default = 500mil).

Rule Classification
Unary

Rule Application
Online DRC and Batch DRC.

Notes
Differential pair routing should be performed independently of the Autorouter.
SMT Rules

SMD to Corner

Description
Specifies the minimum distance from the edge of the surface mount pad to the first routing corner.

Constraints
Distance - the value for the minimum permissible distance from the SMD pad edge to the start of the first routing corner. (Default = 0 mil).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Online DRC and Batch DRC.

SMD to Plane

Description
Specifies the maximum routing length from the center of the surface mount pad to the center of the pad/via connecting to a power plane.

Constraints
Distance - the value for the maximum permissible distance from SMD pad to pad/via connecting to the power plane. (Default = 0 mil).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Online DRC and Batch DRC.
Design Rules Reference

SMD Neck-Down

Description
Specifies the maximum ratio of the track width to the SMD pad width, expressed as a percentage.

Constraints

**Neck-Down** - the percentage value for the maximum permissible ratio of track width to SMD pad width. Entering a larger value will allow for the use of greater width track. (Default = 50%).

Rule Classification

Unary

How Duplicate Rule Contentions are Resolved

All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application

Online DRC and Batch DRC.
Mask Rules

Solder Mask Expansion

Description
The shape that is created on the solder mask layer at each pad and via site is the pad or via shape, expanded or contracted radially by the amount specified by this rule.

Constraints
Expansion - the value applied to the initial pad/via shape to obtain the final shape on the solder mask layer. Enter a positive value to expand the initial pad/via shape, enter a negative value to contract it. (Default = 4mil).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
During output generation.

Notes
Partial and complete tenting of pads and vias can be achieved by defining the appropriate value for the Expansion constraint:

- To partially tent a pad/via - covering the land area only - set the Expansion to a negative value that will close the mask right up to the pad/via hole.
- To completely tent a pad/via - covering the land and hole - set the Expansion to a negative value equal to or greater than the pad/via radius.
- To tent all pads/vias on a single layer, set the appropriate Expansion value and ensure that the scope of the rule (the Full Query) targets all pads/vias on the required layer.
- To completely tent all pads/vias in a design, in which varying pad/via sizes are defined, set the Expansion to a negative value equal to or greater than the largest pad/via radius.

The solder mask expansion can be defined for pads and vias on an individual basis, in the associated properties dialog. Options are available to follow the expansion defined in the applicable design rule, or to override the rule and apply a specified expansion directly to the individual pad or via in question.
Design Rules Reference

Paste Mask Expansion

Description
The shape that is created on the paste mask layer at each pad site is the pad shape, expanded or contracted radially by the amount specified by this rule.

Constraints
Expansion - the value applied to the initial pad shape to obtain the final shape on the paste mask layer. Enter a positive value to expand the initial pad shape, enter a negative value to contract it. (Default = 0mil).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
During output generation.

Notes
The paste mask expansion can be defined for pads on an individual basis, in the associated properties dialog. Options are available to follow the expansion defined in the applicable design rule, or to override the rule and apply a specified expansion directly to the individual pad in question.
**Plane Rules**

**Power Plane Connect Style**

**Description**
Specifies the style of the connection from a component pin to a power plane.

**Constraints**
- **Connect Style** defines the style of the connection from a pin of a component, targeted by the scope (Full Query) of the rule, to a power plane. The following three styles are available:
  - **No Connect** - do not connect a component pin to the power plane
  - **Direct Connect** - connect using solid copper to the pin
  - **Relief Connect** (default) - connect using a thermal relief connection.

**Relief Connect Style Options**
- **Conductors** - the number of thermal relief copper connections (2 or 4, Default = 4)
- **Conductor Width** - how wide the thermal relief copper connections are. (Default = 10mil)
- **Expansion** - the radial width measured from the edge of the hole to the edge of the air gap. (Default = 20mil)
- **Air Gap** - the width of each air gap in the relief connection. (Default = 10mil).

**Rule Classification**
Unary

**How Duplicate Rule Contentions are Resolved**
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**
During output generation.

**Notes**
The images below show the various combinations for the Relief Connect style:

Power planes are constructed in the negative in the PCB Editor, so a primitive placed on a power plane layer creates a void in the copper.
Design Rules Reference

Power Plane Clearance

Description
Specifies the radial clearance created around vias and pads that pass through but are not connected to a power plane.

Constraints
Clearance - the value for the radial clearance. (Default = 20mil).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
During output generation.
**Polygon Connect Style**

**Description**
Specifies the style of the connection from a component pin to a polygon plane.

**Constraints**

- **Connect Style**
  - defines the style of the connection from a pin of a component, targeted by the scope (Full Query) of the rule, to a polygon plane. The following three styles are available:
    - *No Connect* - do not connect a component pin to the polygon plane
    - *Direct Connect* - connect using solid copper to the pin
    - *Relief Connect* (default) - connect using a thermal relief connection.

- **Relief Connect Style Options**
  - **Conductors**
    - the number of thermal relief copper connections (2 or 4, Default = 4).
  - **Conductor Width**
    - how wide the thermal relief copper connections are. (Default = 10mil).
  - **Angle**
    - the angle of the copper connections (45° or 90°, Default = 90°).

**Rule Classification**
Binary

**How Duplicate Rule Contentions are Resolved**
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**
During polygon pour.

**Notes**
The images below show the various combinations for the Relief Connect style:

- **2 and 4 conductors - 45° angle**

- **2 and 4 conductors – 90° angle**
**Design Rules Reference**

**Split Planes Design Rule Checks**

**Description**
In addition to the Power Plane Connect Style and Power Plane Clearance rules for plane connections, the Batch Design Rule Checker can test for and report the following faults for internal split planes:

- Broken planes
- Dead copper regions
- Starved thermal connections.

These options are available in the **Report Options** folder in the **Design Ruler Checker** dialog (Tools » Design Rule Check) under **Split Plane DRC Report** options. Enable the desired options to have them checked and reported during Batch DRC.

**Constraints**

**Report Broken Planes**
- specifies testing for and reporting broken planes. Broken planes occur when an area of the plane that has connectivity to the net becomes electrically disconnected from the rest of the plane. An example where this may occur is a connector that is placed across a split plane, but not connected to it. The voids around the pins join to completely cut through the plane copper, effectively breaking it into two parts.

**Note:** To check for broken planes, the **Un-Routed Net** rule (Electrical category) must be **Batch** enabled.

**Report Dead Copper larger than **xxx** sq. mils**
- specifies the minimum area size to report when testing for regions of unconnected copper in internal planes. Dead copper refers to sections of copper that have no connectivity to the net and which also become electrically disconnected from the original plane. An example where this may occur is a connector (not connected to the plane) with closely spaced pins, in which the voids around the pins join to isolate areas of plane copper from the rest of the plane.

Use the edit field to nominate the value in mil².

**Note:** To check for dead copper, the **Un-Routed Net** rule (Electrical category) must be **Batch** enabled.

**Report Starved Thermal with less than **xxx**% available copper**
- specifies the minimum area size to report when testing starved thermal connections. Thermals are connections to the plane with thermal relief 'cutouts' around them to reduce heat conductivity to the plane copper. Thermals can become 'starved' when the surface area of the copper spokes connecting it to the plane is reduced by void areas.

This rule also checks the surface area for the thermal (not just the spokes) against any voids areas that encroach into the thermal.

Use the edit field to nominate the value as a percentage.
Testpoint Rules

Testpoint Style

Description
Specifies the allowable physical parameters of pads and vias that are to be considered for use as testpoints.

Constraints

Style - the following options allow you to specify pad/via diameter and hole-size criteria when testing for valid testpoints:

- **Min Size** - specifies the minimum permissible diameter for a pad/via to be considered as a testpoint. (Default = 40mil)
- **Max Size** - specifies the maximum permissible diameter for a pad/via to be considered as a testpoint. (Default = 100mil)
- **Preferred Size** - specifies the diameter to be used for testpoint pads/vias placed by the Autorouter. (Default = 60mil)
- **Min Hole Size** - specifies the minimum permissible hole size for a pad/via to be considered as a testpoint. (Default = 0mil)
- **Max Hole Size** - specifies the maximum permissible hole size for a pad/via to be considered as a testpoint. (Default = 40mil)
- **Preferred Hole Size** - specifies the hole size to be used for testpoint pads/vias placed by the Autorouter. (Default = 32mil).

Grid Size - specifies the size of the grid to be used when attempting to find valid testpoint sites (pads and vias). (Default = 1mil).

Allow testpoint under component - enables the use of pads/vias located underneath components (on the same side of the board as the components) for testpoint purposes. (Default = enabled).

Allowed Side and Order - the Find Testpoint feature and the Autorouter use the **Allowed Side** settings to determine which types of pads and vias can be considered for testpoint usage and on which side of the board.

The Autorouter adheres to enabled entries in the list in the exact order that they appear, starting from the top. The ordering can be changed by selecting an entry and using the **Up** and **Down** commands from the right-click pop-up menu. Options can be enabled/disabled on an individual basis, using the right-click menus’ **Toggle** command, or by grouping using the quick select checkboxes below the list (Top, Bottom, Thru-Hole Top and Thru-Hole Bottom).

The following is the default list order (all options enabled by default):

- Use Existing SMD Bottom Pad
- Use Existing Thru-Hole Bottom Pad
- Use Existing Via ending on Bottom Layer
- Create New SMD Bottom Pad (Autorouter only)
- Create New Thru-Hole Bottom Pad (Autorouter only)
- Use Existing SMD Top Pad
- Use Existing Thru-Hole Top Pad
- Use Existing Via Starting on Top Layer
- Create New SMD Top Pad (Autorouter only)
- Create New Thru-Hole Top Pad (Autorouter only).

The Find Testpoint feature adheres to enabled options in the following predefined (non-definable) order:

- Use Existing SMD Bottom Pad
- Use Existing SMD Top Pad
- Use Existing Via ending on Bottom Layer
Design Rules Reference

- Use Existing Via starting on Top Layer
- Use Existing Thru-Hole Bottom Pad
- Use Existing Thru-Hole Top Pad.

Rule Classification

Unary

How Duplicate Rule Contentions are Resolved

All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application

This rule is obeyed by the Find Testpoint feature, the Autorouter, the Online and Batch DRC and during output generation. The Online DRC and Batch DRC test all attributes of the rule except the Preferred Size and Preferred Hole Size - these settings are used by the Autorouter to define the size of testpoint pads that the Autorouter places.

Notes

If you want to use a surface mount pad as a testpoint, the minimum hole size should be set to zero.

If a pad/via assigned as a testpoint is not on the grid specified by the Grid Size option, it will cause a violation when performing a Design Rule Check (DRC). For example, if you set the Grid Size to 25mil, then the testpoints must be on a 25mil grid. If the testpoints do not lie on any particular grid, you can enter a value for Grid Size that will accommodate all testpoints. The minimum setting is 0.001 mil.

The following should be considered when defining a strategy to incorporate testpoints into a design:

- When choosing the side of the board that testpoints will be allowed on, consideration should be given to the testing processes and associated fixtures that will be used. For example, will the board be probed from the bottom side only, the top side only, or both sides.
- A testpoint underneath a component (on the same side of the board as the component) is usually used at the bare-board testing stage. This should be taken into consideration when planning testpoint locations for assembled board testing.
- It is advisable to locate all testpoints on one side of the board only, using vias to achieve this if necessary. The reason for this lies in the fact that a dual-head test fixture incurs greater cost than a single-head test fixture.
- The more non-standard and complex your pattern of testpoints, the more costly it will be to configure a fixture with which to test the board. The best philosophy is to develop a methodology that will result in generic testability. A well-honed and adaptable testpoint policy will allow different designs to be tested efficiently and cost-effectively.
- Careful consideration should be given to any via tenting requirements of the design. Tenting a testpoint-designated via will effectively block test probe contact. Even partial tenting using a liquid photoimageable (LPI) solder mask will cause contact problems, as the mask liquid will tend to run away through the via hole. Peelable solder mask may indeed be used to provide temporary tenting of such designated vias, but this can often prove quite costly.
- Consult with your fabrication and assembly houses closely to make sure any specific design parameters are taken into account when specifying testpoints. These could include testpoint-to-testpoint clearances and testpoint-to-component clearances that may be more strict than normal placement and routing clearances.
Testpoint Usage

Description
Specifies which nets require a testpoint.

Constraints

- **Testpoint**
  - specifies whether or not a testpoint is required on the net(s) targeted by the scope of the rule. The following options are available:
    - **Required (default)** - each target net must have a testpoint assigned
    - **Invalid** - each target net must not have a testpoint assigned
    - **Don't Care** - each target net can have a testpoint assigned. It does not matter if a testpoint cannot be assigned to a net.

- **Allow multiple testpoints on same net**
  - allows more than one testpoint to be assigned (if possible) on the same net. (Default = disabled).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
This rule is obeyed by the Find Testpoint feature, the Autorouter, the Online and Batch DRC and during output generation.

Notes
As well as specifying whether or not a testpoint is required on a particular net, you can also specify whether multiple testpoints can be allowed on the same net. Multiple testpoints on a net might, for example, prove useful when checking routing connectivity on the unloaded board.

The DRC report, obtained from running a Batch DRC, can be used to identify each net that fails this rule.

A testpoint report can be configured and generated along with other fabrication outputs as part of an Output Job Configuration file (*.OutJob). Use this report to interrogate the locations of all valid testpoints assigned in the design.
Design Rules Reference

Manufacturing Rules

Minimum Annular Ring

Description
Specifies the minimum annular ring required for a pad or via. The annular ring is measured radially, from the edge of the pad/via hole to the edge of the pad/via.

Constraints

Minimum Annular Ring (x-y) - the minimum value for the annular ring around the pads/vias targeted by the rule. (Default = 10mil).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Online DRC and Batch DRC.

Notes
For very dense designs, the smaller the annular ring the better, as less space is taken by the pad or via and more space can be dedicated to routing the traces in highly populated areas of the board.

To take the annular ring constraint lower can have a greater impact on the cost when it comes to fabricating the board. The decision basically comes down to whether the benefit from greater routing space outweighs the price increase. Many designers will regularly specify an annular ring constraint of 5-7mil – happy to pay the extra cost for the freedom they have gained when it comes time to route their boards.

Different fabrication houses will undoubtedly use varied and differing manufacturing technologies and equipment. Average performance houses may offer design specifications allowing a 10mil minimum annular ring. High performance houses may be able to reduce that figure down to 5mil. If pad and via holes are laser-drilled, as opposed to mechanically drilled, then the value for the minimum annular ring may be reduced further still.

The class of board you are designing will also play a part in the value required for the minimum annular ring. For example, if your design is of IPC Class 3 standard, which refers to high reliability electronics products, the required minimum annular ring is 2mil.

If you do have to reduce the annular ring below the accepted standard of the fabrication house, try to limit the usage of such affected pads and vias. The more pads and vias on the board that use such annular ring specifications, the more chance there is of a board failing during the fabrication process.

To have no annular ring would, for one thing, cause poor solder joints, as there would be no copper for solder to flow onto after emerging from the pad/via barrel.
Standards define a minimum value for the annular ring, but these values can be reduced further. The reason why they are defined at the levels they are, is to guard against drill breakout. This phenomenon is fairly common when dealing with low values for the annular ring. Drill breakout occurs as a result of several manufacturing parameters (e.g. hole location, hole size, film expansion) interacting unfavorably with one another, leading to the hole being drilled in such a position as to cut through the connecting copper track(s).

It is possible to allow controlled drill breakout, without sacrificing board performance. One method of achieving this is to apply teardrops to required pads and vias. Teardropping (otherwise known as filleting or tapering) is the process of applying additional land area to the pad/via at the junction with any connecting track(s). This additional area protects the pad-track or via-track connection should breakout occur.

---

**Acute Angle**

**Description**

Specifies the minimum angle permitted at a track corner. Acute angles can be a problem when manufacturing, resulting in over-etching of the copper at the corner.

**Constraints**

- **Minimum Angle** - specifies the minimum permissible angle at a track corner. (Default = 90°).

**Rule Classification**

Unary

**How Duplicate Rule Contentions are Resolved**

All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**

Online DRC and Batch DRC.
Design Rules Reference

Hole Size

Description
Specifies the maximum and minimum hole size for pads and vias in the design. The hole size is the diameter of the hole to be drilled through the pad/via during fabrication.

Constraints

Measurement Method - specifies the method used in defining the minimum/maximum hole sizes:
- Absolute (default) - the values for minimum/maximum hole sizes will be absolute values
- Percent - the minimum/maximum hole sizes will be expressed as percentages of the pad/via size.

Minimum - the value for the minimum hole size with respect to pads and vias in the design. The value will appear as an absolute value (Default = 1mil) or percentage of the pad/via size (Default = 20%), depending on the Measurement Method selected.

Maximum - the value for the maximum hole size with respect to pads and vias in the design. The value will appear as an absolute value (default = 100mil) or percentage of the pad/via size (Default = 80%), depending on the Measurement Method selected.

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Online DRC and Batch DRC.
Layer Pairs

Description
Checks to ensure that the used layer-pairs match the current drill-pairs. The used layer-pairs are determined from the vias and pads found in the board, one layer-pair for each Start Layer-End Layer combination that is found.

Constraints

Enforce layer pairs settings - specifies whether the check is made or not. (Default = enabled).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Online DRC, Batch DRC and during interactive routing.
High Speed Rules

Parallel Segment

Description
Specifications the distance two track segments can run in parallel, for a given separation.

Constraints

Layer Checking
- specifies where the two track segments to be checked should reside:
  - Same Layer (default) - the track segments for the targeted nets are both on the same layer
  - Adjacent Layers - the track segments for the targeted nets are on adjacent layers.

For a parallel gap of
- specifies the parallel gap that should exist between two track segments before they can be considered for test. (Default = 10mil).

The parallel limit is
- specifies the maximum permissible parallel length of two track segments (on different nets), when the parallel gap constraint is observed over the entire length. (Default = 10000mil)

Rule Classification
Binary

How Duplicate Rule Contentions are Resolved
Duplicate rules do not create contentions for this rule.

Rule Application
Online DRC and Batch DRC.

Notes
This rule tests track segments, not collections of track segments. Apply multiple parallel segment constraints to a net to approximate crosstalk characteristics that vary as a function of length and gap.
Length

Description
Specifies the minimum and maximum lengths of a net.

Constraints
Minimum - the value for the minimum permissible length of the net. (Default = 0mil).
Maximum - the value for the maximum permissible length of the net. (Default = 100000mil).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Online DRC and Batch DRC.
**Design Rules Reference**

**Matched Net Lengths**

**Description**
Specifies the allowable difference in net lengths. The set of nets targeted by the scope of the rule (as defined by its full query) are interrogated, with the length of each being compared to that of the longest net in the set. Those nets that are found to be outside of a specified tolerance (i.e. are too short) can be lengthened by running a separate command that places a defined accordion-shaped track extension, in order to equalize the lengths and so bring them within the specified tolerance.

**Constraints**

**Tolerance** - specifies a length tolerance when comparing each net with the longest net in the set. Any net whose length does not lie within the specified tolerance is deemed to be too short and will have track added to it should the Equalize Net Lengths command be run. (Default = 1000mil).

**Style** - specifies the accordion style to be used by the Equalize Net Lengths command, when adding track to nets that are found to be too short in comparison with the longest net. There are three styles available - 90° (default), 45° and Rounded, as shown in the following images:

- **90 Degrees**

- **45 Degrees**

- **Rounded**

**Amplitude** - specifies the height of the accordion section of track to be added. (Default = 200 mil).

**Gap** - this constraint is specific to the 90 and 45° styles only. It specifies the gap between the end of one vertical/45° track segment and the start of the next. (Default = 20mil).

**Rule Classification**
Unary

**How Duplicate Rule Contentions are Resolved**
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**
Online DRC, Batch DRC and the Equalize Net Lengths feature.

**Notes**
The PCB Editor can add "accordion" sections to nets to equalize their lengths. Having defined the Matched Net Lengths rule, from the PCB document select Tools » Equalize Net Lengths. The matched lengths rule will be applied to the nets specified by the full query of the rule and accordion sections will be added to those whose length falls outside the permissible tolerance.
The underlying algorithm that adds the accordion sections will only do so on vertical and horizontal tracks. If a net has been predominantly routed using 45° track segments, the possibility of successful equalization will be greatly reduced, depending upon the availability and extent of horizontal and vertical track on which to add the equalizing accordion lengths. If nets with such routing do exist, reroute them using short 45° track lengths and more orthogonal sections.

The degree of success depends on the amount of space available for the accordion sections and the accordion style being used. The 90° style is the most compact and the 45° style is the least compact.

---

**Daisy Chain Stub Length**

![Daisy Chain Stub Length](image)

**Description**

Specifies the maximum permissible stub length for a net with a daisy chain topology.

**Constraints**

- **Maximum Stub Length** - the value for the maximum stub length allowed. *(Default = 1000mil).*

**Rule Classification**

Unary

**How Duplicate Rule Contentions are Resolved**

All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**

Online DRC and Batch DRC.
Design Rules Reference

Vias Under SMD

Description
Specifies whether vias can be placed under SMD pads during autorouting.

Constraints
Allow Vias under SMD Pads - specifies whether vias can be placed under the pads of surface mount components. (Default = disabled).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Online DRC, Batch DRC and during autorouting.

Maximum Via Count

Description
Specifies the maximum number of vias permitted in the current design.

Constraints
Maximum Via Count - the number of vias allowed in the design. (Default = 1000).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Online DRC and Batch DRC.
Placement Rules

Room Definition

Description
Specifies a rectangular region where components are either allowed in, or not allowed in.

Constraints

Room Locked - allows you to lock the room in its current position within the design, preventing accidental movement either manually or by the Autoplacers. If you attempt to move the room when it has been locked, a warning dialog will appear asking whether you wish to go ahead with the move. The locked status of the room remains in force after such a manual-override movement. (Default = disabled).

Components Locked - allows you to lock the position of components arranged within, and associated to, the room. (Default = disabled).

Define button - enables you to define the area and location of the room. After clicking, you will return to the main design window, the cursor will change to a cross-hair and you will essentially enter room placement mode. Define the rectangular or polygonal room as required and at the location required. The component membership for the room has to be defined afterwards, it is not created automatically if the room area is defined around placed components in the design.

x1 and y1 - display the coordinates for the location of the lower-left corner of the room's bounding rectangle. These fields are non-editable - if placing the room from within the PCB Rules and Constraints Editor dialog, the Define button must be used.

x2 and y2 - display the coordinates for the location of the upper-right corner of the room's bounding rectangle. These fields are non-editable - if placing the room from within the PCB Rules and Constraints Editor dialog, the Define button must be used.

Layer - defines which layer the room can be placed on. (Default = Top Layer).

Confinement Mode - specifies whether the components targeted by the scope (Full Query) of the rule are to be kept inside the room or kept outside the room. (Default = Keep Objects Inside).

Rule Classification

Unary

How Duplicate Rule Contentions are Resolved

All rules are obeyed.

Rule Application

Online DRC, Batch DRC and during autoplacement with the Cluster Placer.

Notes

Rooms can be created/edited using the various commands available from the Design » Rooms sub-menu.

When placing a rectangular or polygonal room in the design, it can either be placed empty and components associated at a later stage, or it can be placed around components in the design, automatically associating them to the room:

- When an empty room is placed in the design, components required to be placed in the room should be grouped together by the use of a specific component class. A Room Definition rule will automatically be created and assigned to the room, with an initial scope (Full Query) of All. Edit this query to target the specific component class previously defined. The components can then be moved to the room by the use of the Tools » Interactive Placement » Place Within Room command.

- By placing a room around one or more components, so that they fall completely within its boundaries, the components will automatically be associated to the room. The scope or query for the room's definition rule depends on whether all components are part of an existing component class or not. If they are, then this component class will be used. If not, a new component class is created, with these components as its members. It is therefore possible to have multiple rooms, each
Design Rules Reference

with a scope that targets a particular component class, and have one or more mutual component members between those classes.

Use the Create Room from selected components-based commands to automatically generate a Rectangular, Orthogonal or Non-Orthogonal shaped room, whose members are the selected components. A component class is automatically defined to include the selection. A room is then created, the Room Definition rule of which is defined to associate the created component class. The room will be sized accordingly, in order to fit all components in the selection, as defined by the limits of their bounding rectangles.

Once components have been assigned to a room they move when the room is moved. To move a room without moving the components, temporarily disable the associated Room Definition rule.

As well as being a design rule in its own right (Room Definition), a room can also be used as an object when defining the scope of another rule, such as Clearance or Height. As the room is to be used as an object rather than a rule, you can disable the rule.

The following two queries can be used when using a room object in another rules' scope definition:

- **TouchesRoom(RoomName)** - use to find objects that are completely or partially within the room.
- **WithinRoom(RoomName)** - use to find objects that are completely within the room.
Component Clearance

Description
Specifies the minimum distance that components can be placed from each other. Component clearance includes clearance between 3D models used to define component bodies (both STEP and extruded (simple) types) and even non-component free-floating models in the 3D workspace, such as mechanical housings or PCB enclosures. In the absence of 3D bodies, the primitives on the silk and copper layers are used to define the object shape and size along with the height value specified in the component properties.

Component clearance is calculated using accurate 3D meshing to define shape and contour for the component through its associated 3D body objects. These may be imported STEP model files or extruded 2D shapes. It is evident that using 3D bodies provides greatest accuracy when it comes to clearance checking, particularly in the vertical sense and in the context of complex component shapes.

Note: The Component Clearance rule does not check for clearance violations between 3D bodies and the board surface.

Constraints

Minimum Horizontal Clearance - the value for the minimum permissible clearance, in the horizontal plane, between placed components in the design.
- Specified (default) - clearance checking is performed using the exact shape defined by the component 3D bodies or component footprint properties. (Default = 10mil).

Minimum Vertical Clearance - the value for the minimum permissible clearance, in the vertical sense, between placed components in the design.
- Specified (default) - clearance checking is performed using the exact shape defined by the component 3D bodies or component footprint properties. When using 3D bodies to make the check from, it is possible to have acceptable overhang between one component over another, provided they are not in violation. (Default = 10mil).
- Infinite - clearance checking is performed using a value representing infinity. This means that any components placed above or below will be in violation. An example of use would be a board that has an adjustment mechanism that must remain accessible. Using this rule on that component will cause a violation against any components or free-floating objects that protrude into the area above or below the component.

Show actual violation distances - enable this option to show lines between the points of greatest violation between components. The distance of the line is displayed and can be useful in calculating the distance required to move an object to resolve the violation.

Note: Enabling this option may reduce performance on some computer systems.

Rule Classification
Binary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.
**Design Rules Reference**

**Rule Application**
Online DRC and Batch DRC.

**Notes**
An extruded (simple) 3D body is a polygonal shaped object that can be placed in a library component or a PCB document, on any enabled mechanical layer. In a component footprint it can be used to specifically define the physical size and shape of a component in the X, Y and Z-axes. For more information, refer to the 3D Body topic in the *PCB Editor and Object Reference*.

Multiple 3D body primitives may be used to define shapes of any complexity. This can prove especially useful in the vertical sense, as it allows you to vary the height of a component in different regions of that component.

3D STEP models can be imported into component footprints to provide realistic representations during 3D visualization. The models can be either linked or embedded into the component footprint. Linked files remain attached to their original file, meaning that if the original file is altered, the changes are automatically reflected in Altium Designer.

3D STEP models can be imported as non-PCB mounted or component-based *free-floating* objects. This enables you to provide realistic representations of other objects (not part of the PCB) that are part of an assembled design. Examples being a housing for a PCB, or even another PCB as part of a stacked board assembly. Clearance checking between board-mounted components and free-floating objects is also carried out. This provides the ability to clearance check complete assemblies within Altium Designer.
Component Orientations

Description
Specifies allowable component orientations. Multiple orientations are permitted, allowing the Autoplacer to use any of the enabled orientations.

Constraints

Allowed Orientations - the chosen orientations that are made available for use by the Cluster Placer. The following orientation-based options are available:

- **0 Degrees** - allows the Autoplacer to rotate a component to the 0° orientation. Rotation is relevant to the orientation of the component in the source library. (Default = enabled)
- **90 Degrees** - allows the Autoplacer to rotate a component to the 90° orientation. Rotation is relevant to the orientation of the component in the source library. (Default = disabled)
- **180 Degrees** - allows the Autoplacer to rotate a component to the 180° orientation. Rotation is relevant to the orientation of the component in the source library. (Default = disabled)
- **270 Degrees** - allows the Autoplacer to rotate a component to the 270° orientation. Rotation is relevant to the orientation of the component in the source library. (Default = disabled)
- **All Orientations** - allows the Autoplacer to rotate a component to any of the four individual orientations. Rotation is relevant to the orientation of the component in the source library. (Default = disabled).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
During autoplacement with the Cluster Placer.
Permitted Layers

Description
Specifies which layers components can be placed on during placement with the Cluster Placer.

Constraints

**Permitted Layers** - the layers permitted to be used by the Cluster Placer during autoplacement. The following layer options are available:

- **Top Layer** - allow component placement on the top layer. (Default = enabled)
- **Bottom Layer** - allow component placement on the bottom layer. (Default = enabled).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
During autoplacement with the Cluster Placer.

Notes
The Cluster Placer does not change the layer a component is on - you must set the component layer prior to running the placer.

---

Nets to Ignore

Description
Defines which nets should be ignored during autoplacement with the Cluster Placer.

Constraints
None

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
Contentions are not possible.

Rule Application
During autoplacement with the Cluster Placer.

Notes
Ignoring power nets can assist in placement speed and quality. If the design has a large number of two pin components that connect to a power net, ignoring the power net will result in these components being clustered based on their other net, rather than the power net.
Height

Description
Defines height restrictions for components placed within the design. When placement of components in certain regions of the board is particularly height-critical, a height rule can be defined with a scope that targets one or more rooms in the design, with constraints set up to flag a violation if components over a certain height are placed in the room(s).

Constraints
Minimum - the value for the minimum permissible component height. (Default = 0mil).
Preferred - the value for the preferred component height. (Default = 500mil).
Maximum - the value for the maximum permissible component height. (Default = 1000mil).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
The Preferred setting is obeyed during autoplacement and when displaying the board three-dimensionally.
The Minimum and Maximum settings are obeyed by the Online DRC and batch DRC.

Notes
The height property for a component is defined in its associated properties dialog.

A component class can be created and used in the scope definition for a height rule, in order to flag any member components whose height violates the specified height constraint criteria of the rule.

If using a room to scope a height rule, remember that the room is itself a design rule (Room Definition). When you first place a room, a Room Definition rule is created. This rule will check to see if the scoped objects (by default, All) are within the room. As the room is to be used as an object rather than a rule, you can disable the rule. You can then use the room as a scope in your height rule. The following two queries can be used to define the scope of the rule:

- TouchesRoom(RoomName) - use to find objects that are completely or partially within the room.
- WithinRoom(RoomName) - use to find objects that are completely within the room.
Signal Integrity Rules

Signal Stimulus

Description
Specifies the characteristics of the stimulus signal used when performing a signal integrity analysis on the design. This is the signal that is injected at each output pin on the net under test. The worst-case result is returned during design rule checking.

Constraints

Stimulus Kind
- specifies the type of stimulus signal that is injected during signal integrity analysis. The following stimulus types are available:
  - **Constant Level** - the stimulus signal remains at a constant voltage - either High or Low - depending on the chosen Start Level option
  - **Single Pulse (default)** - the stimulus signal is a single pulse, whose characteristics are defined by the Start Level, Start Time and Stop Time options
  - **Periodic Pulse** - the stimulus signal is a continuous pulse train, whose characteristics are defined by the Start Level, Start Time, Stop Time and Period Time options.

Start Level
- specifies the voltage level used for the Constant Level stimulus signal, or the initial voltage level for the pulse-based stimulus signals. The following levels are available:
  - **Low Level (default)** - defined as the LOW level voltage for the output pin - dependent on the model used for the pin
  - **High Level** - defined as the HIGH level voltage for the output pin - dependent on the model used for the pin.

Start Time (s)
- the start time for a pulse-based stimulus signal. Used in calculating the width of the pulse. (Default = 10.00ns).

Stop Time (s)
- the stop time for a pulse-based stimulus signal. Used in calculating the width of the pulse. (Default = 60.00ns).

Period Time (s)
- the time between pulses in a periodic pulse train stimulus signal. After the period time has elapsed, another identical pulse of width \( \text{Stop Time} - \text{Start Time} \) is injected. (Default = 100.00ns).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Batch DRC and during Signal Integrity analysis.

Notes
When performing a Crosstalk analysis, an Aggressor net will be injected with the stimulus defined in the Stimulus design rule, the LOW and HIGH levels of which are dependent on the model used for the driving output pin. A Victim net will get a Constant Low level voltage injected into it, with the level again being dependent on the model used for the output pin.
Overshoot - Falling Edge

Description
Specifies the maximum allowable overshoot (ringing below the base value) on the falling edge of the signal.

Constraints
Maximum (Volts) - the value for the maximum permissible overshoot on the falling edge of the signal. (Default = 1.000).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Batch DRC and during Signal Integrity analysis.

Overshoot - Rising Edge

Description
Specifies the maximum allowable overshoot (ringing above the top value) on the rising edge of the signal.

Constraints
Maximum (Volts) - the value for the maximum permissible overshoot on the rising edge of the signal. (Default = 1.000).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Batch DRC and during Signal Integrity analysis.
Design Rules Reference

Undershoot - Falling Edge

Description
Specifies the maximum allowable undershoot (ringing above the base value) on the falling edge of the signal.

Constraints

Maximum (Volts) - the value for the maximum permissible undershoot on the falling edge of the signal. (Default = 1.000).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Batch DRC and during Signal Integrity analysis.

Undershoot - Rising Edge

Description
Specifies the maximum allowable undershoot (ringing below the top value) on the rising edge of the signal.

Constraints

Maximum (Volts) - the value for the maximum permissible undershoot on the rising edge of the signal. (Default = 1.000).

Rule Classification
Unary

How Duplicate Rule Contentions are Resolved
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

Rule Application
Batch DRC and during Signal Integrity analysis.
**Impedance**

**Description**
Specifies the minimum and maximum net impedance allowed. Net impedance is a function of the conductor geometry and conductivity, the surrounding dielectric material (the board base material, multi-layer insulation, solder mask, etc) and the physical geometry of the board (distance to other conductors in the z-plane).

**Constraints**

- **Minimum (Ohms)** - the value for the minimum permissible net impedance. (Default = 1.000).
- **Maximum (Ohms)** - the value for the maximum permissible net impedance. (Default = 10.00).

**Rule Classification**
Unary

**How Duplicate Rule Contentions are Resolved**
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**
Batch DRC and during Signal Integrity analysis.

---

**Signal Top Value**

![Signal Top Value](image)

**Description**
Specifies the minimum voltage level that a signal can settle to in the high state (the top value).

**Constraints**

- **Minimum (Volts)** - the value for the minimum permissible top value voltage. (Default = 5.000).

**Rule Classification**
Unary

**How Duplicate Rule Contentions are Resolved**
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**
Batch DRC and during Signal Integrity analysis.
**Design Rules Reference**

### Signal Base Value

**Description**
Specifies the maximum voltage level that a signal can settle to in the low state (the base value).

**Constraints**

- **Maximum (Volts)** - the value for the maximum permissible base value voltage. (Default = 0.000).

**Rule Classification**

Unary

**How Duplicate Rule Contentions are Resolved**
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**
Batch DRC and during Signal Integrity analysis.

### Flight Time - Rising Edge

**Description**
Specifies the maximum allowable flight time on signal rising edge. Flight time is the signal delay time introduced by the interconnect structure. It is calculated as the time it takes to drive the signal on the net to the threshold voltage (marking the transition from signal LOW to signal HIGH), less the time it would take to drive a reference load (connected directly to the output) to the threshold voltage.

**Constraints**

- **Maximum (seconds)** - the value for the maximum permissible flight time on the rising edge of the signal. (Default = 1.000ns).

**Rule Classification**

Unary

**How Duplicate Rule Contentions are Resolved**
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**
Batch DRC and during Signal Integrity analysis.
**Flight Time - Falling Edge**

**Description**

Specifies the maximum allowable flight time on signal falling edge. Flight time is the signal delay time introduced by the interconnect structure. It is calculated as the time it takes for the signal on the net to fall to the threshold voltage (marking the transition from signal HIGH to signal LOW), less the time it would take for a reference load (connected directly to the output) to fall to the threshold voltage.

**Constraints**

- **Maximum (seconds)** - the value for the maximum permissible flight time on the falling edge of the signal. (Default = 1.000ns).

**Rule Classification**

Unary

**How Duplicate Rule Contentions are Resolved**

All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**

Batch DRC and during Signal Integrity analysis.

---

**Slope - Rising Edge**

**Description**

Specifies the maximum allowable slope time on the rising edge of the signal. Rising edge slope is the time it takes for a signal to rise from the threshold voltage (VT), to a valid high (VIH).

**Constraints**

- **Maximum (seconds)** - the value for the maximum permissible rising edge slope time. (Default = 1.000ns).

**Rule Classification**

Unary

**How Duplicate Rule Contentions are Resolved**

All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**

Batch DRC and during Signal Integrity analysis.
**Design Rules Reference**

**Slope - Falling Edge**

![Slope - Falling Edge Diagram]

**Description**
Specifies the maximum allowable slope time on the falling edge of the signal. Falling edge slope is the time it takes for a signal to fall from the threshold voltage (VT), to a valid low (VIL).

**Constraints**
- Maximum (seconds) - the value for the maximum permissible falling edge slope time. (Default = 1.000ns).

**Rule Classification**
Unary

**How Duplicate Rule Contentions are Resolved**
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**
Batch DRC and during Signal Integrity analysis.

---

**Supply Nets**

**Description**
Identifies a supply net and specifies its voltage (or set of nets using the net class scope).

**Constraints**
- Voltage - the voltage value for the net(s) falling under the scope (full query) of the rule. (Default = 0.000V).

**Rule Classification**
Unary

**How Duplicate Rule Contentions are Resolved**
All rules are resolved by the priority setting. The system goes through the rules from highest to lowest priority and picks the first one whose scope expression(s) match the object(s) being checked.

**Rule Application**
Batch DRC and during Signal Integrity analysis.

**Notes**
The supply net(s) can be specified by enabling the Net or Net Class option in the Where the First object matches region of the PCB Rules and Constraints Editor dialog and choosing the required net or net class from the corresponding drop-down list. The corresponding Full Query for the rules’ scope will be as follows:

- `InNet('NetName')` - for a single net
- `InNetClass('NetClassName')` - for a net class

TR0116 (v1.8) May 20, 2008
### Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Version No.</th>
<th>Revision</th>
</tr>
</thead>
<tbody>
<tr>
<td>01-Dec-2004</td>
<td>1.0</td>
<td>New product release</td>
</tr>
<tr>
<td>18-Apr-2005</td>
<td>1.1</td>
<td>Updated and reformatted for SP3</td>
</tr>
<tr>
<td>10-Jun-2005</td>
<td>1.2</td>
<td>Updated for Altium Designer SP4</td>
</tr>
<tr>
<td>24-Aug-2005</td>
<td>1.3</td>
<td>Image enhancements</td>
</tr>
<tr>
<td>22-Sep-2005</td>
<td>1.4</td>
<td>Updated information for DRC Report generation</td>
</tr>
<tr>
<td>17-Dec-2007</td>
<td>1.5</td>
<td>Updated for AD 6.9</td>
</tr>
<tr>
<td>05-Feb-2008</td>
<td>1.6</td>
<td>Removed incorrect checking actions for component orientations rule. Minor updates and screen images for 6.9 added.</td>
</tr>
<tr>
<td>11-Feb-2008</td>
<td>1.7</td>
<td>Component body references changed to 3D body.</td>
</tr>
<tr>
<td>20-May-2008</td>
<td>1.8</td>
<td>Converted to A4 and introduce 3D component clearance rule and additional split plane DRC report options for Summer 08. Updated images for power plane connect style. Removed infinite horizontal component clearance rule option and updated infinite vertical spacing image.</td>
</tr>
</tbody>
</table>

Software, hardware, documentation and related materials:

Copyright © 2008 Altium Limited.

The material provided with this notice is subject to various forms of national and international intellectual property protection, including but not limited to copyright protection. You have been granted a non-exclusive license to use such material for the purposes stated in the end-user license agreement governing its use. In no event shall you reverse engineer, decompile, duplicate, distribute, create derivative works from or in any way exploit the material licensed to you except as expressly permitted by the governing agreement. Failure to abide by such restrictions may result in severe civil and criminal penalties, including but not limited to fines and imprisonment. Provided, however, that you are permitted to make one archival copy of said materials for back up purposes only, which archival copy may be accessed and used only in the event that the original copy of the materials is inoperable. Altium, Altium Designer, Board Insight, DXP, Innovation Station, LiveDesign, NanoBoard, NanoTalk, OpenBus, P-CAD, SimCode, Situs, TASKING, and Topological Autorouting and their respective logos are trademarks or registered trademarks of Altium Limited or its subsidiaries. All other registered or unregistered trademarks referenced herein are the property of their respective owners and no trademark rights to the same are claimed. v8.0 31/3/08.