PULSONIX

# Pulsonix Design System

## V2.0 Update Notes

## Copyrights



Pulsonix, a division of WestDev Ltd.

Printed in the UK. Issue date: 10/03/03

**Pulsonix**
Oak Lane
Bredon, Tewkesbury
Glos, GL20 7LR
United Kingdom

Phone +44 (0)1684 773881 Fax +44 (0)1684 773664
Email info@pulsonix.com
Web www.pulsonix.com

# Contents

# Chapter 1. Getting Started With V2.0

## About these notes

These update notes are provided for existing users as a supplement to their existing Pulsonix Users Guides. These notes are to highlight new features in version 2.0 and to briefly describe their use.

Each chapter is broken down into logical functional descriptions based on the application type, Schematics, PCB etc.

## Installing the new version

It is recommended that you back-up all libraries, designs, technology files, profile files, reports and netlists before installing the latest version. Other than for any reason, this is good working practice, although you should have a backup already of all this data!

As with any other installation of Pulsonix, insert the CD and wait for a short time. The *Autorun* facility will start the installation procedure. Follow the on-screen messages from the install wizard. You should install Pulsonix V2.0 on top of your existing installation, you do not need to uninstall any old version first.

There are some small changes to the install procedure that mainly affects the licensing scheme used, whether it is node locked (as it currently is) or whether it is a network license. Network licensing is a new cost option and you can upgrade your existing node locked licensing to this scheme.

For existing users it is recommended that you simply click the **No Change In Licensing** check box on the licensing page of the wizard. Any new licenses and changes to network licensing can be made after the installation using the License Manager.

## Licensing V2.0

Version 2.0 has a new license file that will be supplied with your update CD. Copy the license file from the floppy disk supplied (or an email) to your Pulsonix installation folder. From the license Manager use the Browse Master File option to locate this license. If the license supplied has the same name as your old license, you can overwrite the existing one. License files are backwards compatible and can be used with Version 1.0, 1.1 or 1.2. The older License Manager programs will ignore any new license options. This means you can run both Version 2.0 and Version 1.2 if you wish. Obviously, you need to install Version 2.0 into a different folder to that of Version 1.0 if you wish to run both.

# Network licensing

## What is network licensing?

Network licensing is the ability to run any number of licenses from one central server without the need for a dongle on each machine which will run the product. This is called ‘floating’ licensing.

## How it operates

The license mechanism allows licenses to be ‘floating’ on any nominated machines within a local network. The scheme works on a book out / book in system for each available product license. The central Pulsonix server administrator system allows users or machines to be nominated for use or no use of the Pulsonix tools. Parts of the tool set can be allowed or disabled for use as well for specific users, for example, user Bert might be able to use a Pulsonix Schematics license but not a PCB one, while user Fred can use Schematics, PCB and the Autorouter. This example also applies for machines, so that a user can log in as his name on any machine but will still only have the products available to him, ones which have been allocated to him through the administrator tool.

The floating license uses a special license, you can’t ‘float’ normal licenses.

## System permissions

The server needs the License Manager and associated files to run the Pulsonix network licensing, it does not require a full Pulsonix installation.

### Permissions and the System Registry

The Pulsonix Setup program may add or modify entries in your System Registry. There are some issues relating to access permissions that are described below.

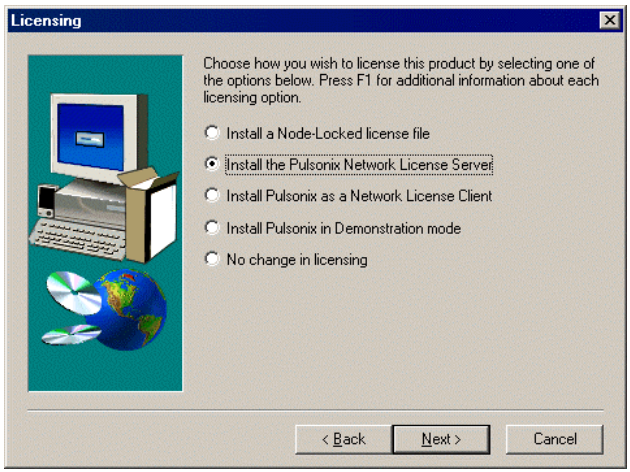
Two areas of the System Registry are written by Pulsonix Setup. One is for the 'current user', and the other is for the 'local machine'. To write to the latter section of the registry, you must be logged in as a user that has permission to write to this part of the registry.

For users of Windows 9x-based operating systems (Windows 95, Windows 98 and Windows ME) this is not usually an issue. However, for users of Windows NT-based systems (Windows NT, Windows 2000 and Windows XP) each user who logs into a computer generally has various access permissions assigned to them, and in order to install Pulsonix successfully these permissions must include Administrator rights on the local computer itself.

If you are in any doubt about whether or not your system is set up to allow you to access the registry correctly, please consult your System Administrator.

## Installing the License Manager on the server

There is an option to install the Network License Server on the main Pulsonix installation CD. This is available from the Licensing page in the install wizard.




With the **Install the Pulsonix Network License Server** option checked you will only get the licensing items required to centrally license Pulsonix.

If you wish to run both the License Server and a copy of the Pulsonix application on the same computer, you will first need to install the License Server, then re-run Setup and choose the next option (Network License Client) to install Pulsonix itself.

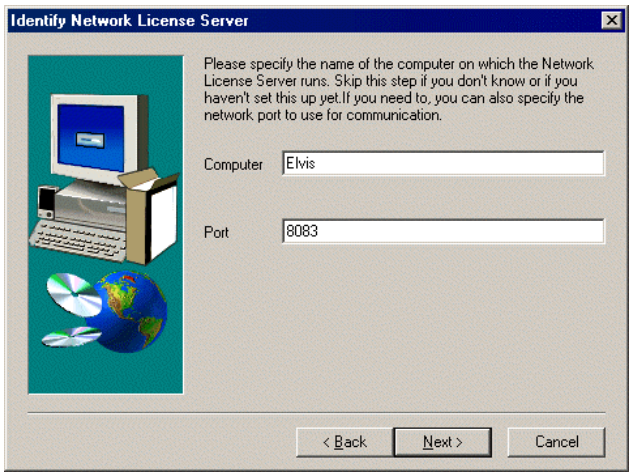
Depending on your system configuration, you may also be prompted to install a HASP driver for a server installation. If you are using a dongle, an appropriate Hasp driver must be installed otherwise the program will not run. Setup will install the HASP driver for you.

## Installing a network ‘client’ machine

When installing Pulsonix as a 'Network License Client', you are asked to specify the name of the computer on which the License Server will run (or is already running). The License Server does not need to be running at the time, as the Setup program will not actually attempt to communicate with the server.

If you don't know the name of the License Server computer, you can skip this step by leaving the Computer name blank. You can set up the name of the License Server later, by using the Pulsonix License Manager program once you have finished installing Pulsonix.

If necessary, you can also change the network 'port' used by Pulsonix to communicate with the License Server. This should only be necessary if you are already using other software that uses a network 'socket' on this port number. This number is generally only changed with the assistance of a member of our Technical Support Team.



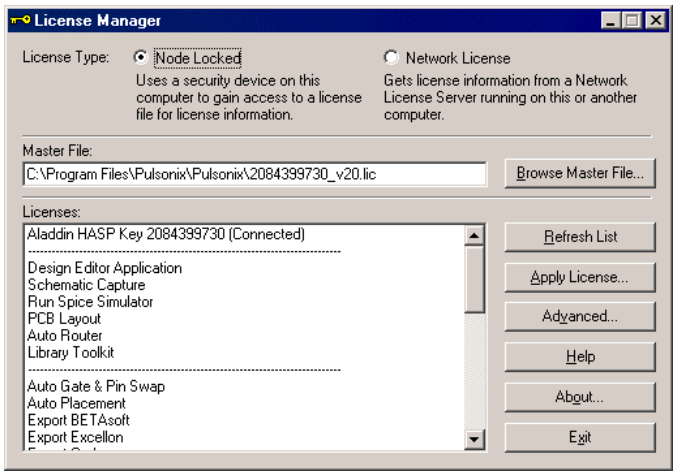

Once you have set up the Computer and/or Port, click Next to proceed. The values you have specified will be stored in the system registry for later use by the various Pulsonix programs that need to use the Network Licensing scheme.

## Using the License Manager for network licensing

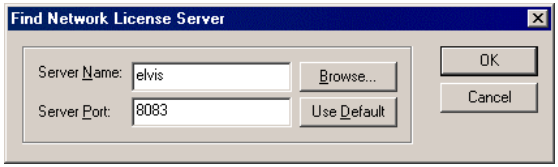
As with node locked licenses, the License Manager can be used to locate and manage network licenses. The process is very straightforward once the license server has been set up and each license client has been installed.

### ▶ To use a network license

1. Click on **Start** on the taskbar, click **Programs,** click on **Pulsonix**
2. Click on the **License Manager** program icon
3. You are presented with a dialog from which you can manage the license files.




4. Click on the **Network License** radio button.
5. Click on the **Find Server** button (this was named Browse Master File).




6. You will need to indicate the name of the **Server** on which the network license and dongle reside. You will also need to supply a **Server Port** number for it. If you don't know the number, select the **Use Default** button. This will automatically allocate a number for you.
7. At this point the License Manager looks for the network license server which must be running.
8. The list will refresh and the **Licenses** box will show a network license connection.

## Updating Pulsonix from the web

As reminder, don't forget that you can also update Pulsonix from our web site www.pulsonix.com with the latest updates which become available from our development centre. It is highly recommended that when notified, you keep up-to-date with the latest version. This not only keeps your product current but ensures that in the event of a problem or technical question we can direct you through the product using the same version that we run. This facility is available for all users with a current maintenance contract.

To access the latest updates from the web site, go to www.pulsonix.com and click on **Updates**. The updates are self-extracting InstallShield setup programs.

If you click on the update name it will self extract to your local drive after detecting where the Pulsonix product is installed. You can also save the .exe file to your local disk if you wish to install it at a later time or on another machine or over a local intranet network.

# Chapter 2. General Options

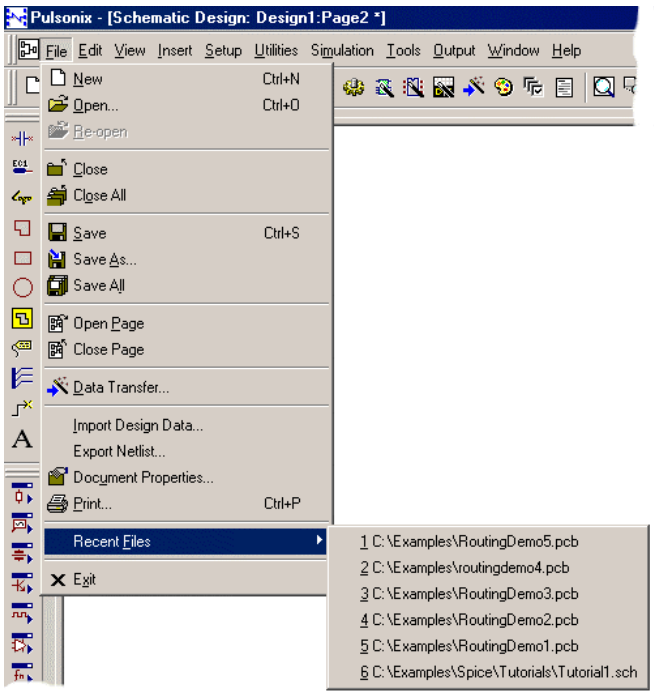
## Introduction

This section covers all item changes common to both Schematics, PCB and non-design related options such as the Library Manager or Pulsonix interface.

## Changes to File Menu

The **File** menu has been changed to make it less cluttered. The **Save Technology** and **Save Profile** options have been moved to the **Setup** menu and the **Recently Used Files** list has been made into a sub menu (see below).

From the **Setup** menu you will now find, **Save Technology** / **Load Technology**, **Save Colours** / **Load Colours**, and **Save Profile** / **Load Profile**. (Save and Load Technology and Colours files are also available from **Technology** and **Colours** dialogs respectively).



# Save and Close All options

New options for **Save All** and **Close All** appear on the **File** menu.

**Save All** allows you to save all currently open documents in Pulsonix. This includes all designs and libraries.

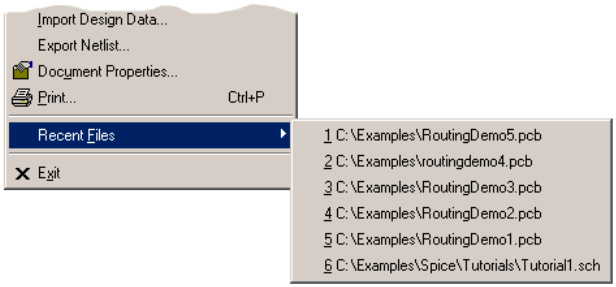
**Close All** will close all currently open documents. Where documents or libraries have not been saved, you will be prompted to save or cancel.

# Re-Open File

**Re-Open File** is used to close the current design and open it again from the saved copy on disk. If the design has been changed but the changes not saved then a message box will appear.

# New Recent Files list

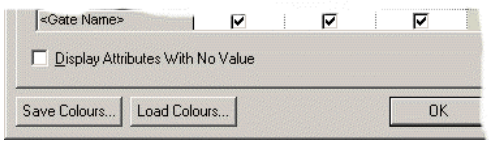
A sub-menu **Recent Files>** has been created to de-clutter the **File** menu. By hovering the mouse over Recent Files >, the sub-menu will be displayed.




# Save/Load Colour Files

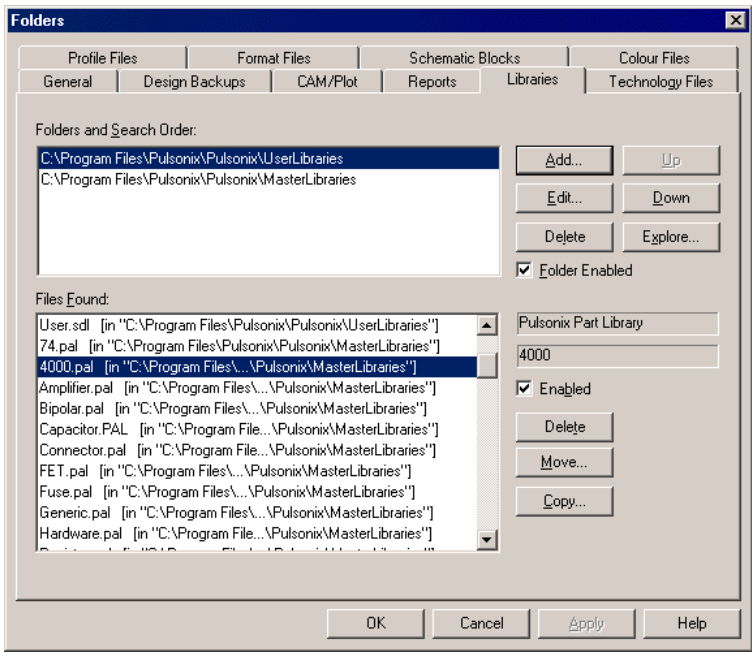
It is now possible to save and load colour files from disk. You can save and load files from two locations, from the **Setup** menu, and from the **Colours** dialog itself.

The location to which the Colour files are saved and loaded is defined in the **Folders** dialog using the **Colour Files** tab.

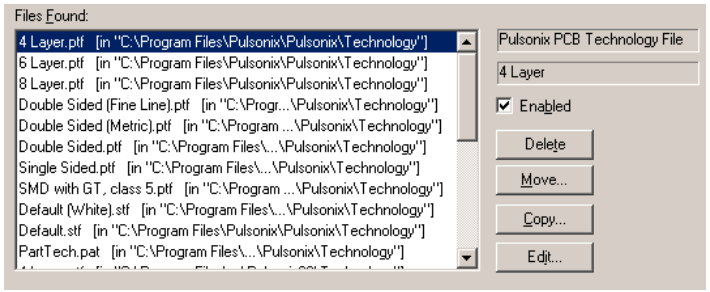


# Changes to Folders Dialog

The **Folders** dialog has been changed to include new items for **Colour Files** and **Schematic Blocks**.




There are two other changes, you can now disable folders, (as you could already with the files within the folders), and a new **Edit** button has also been included on this dialog for editing files directly from the **Files Found** list.




The file must be selected from the list in order for the Edit button to be active. This button appears on the **Profile Files**, **Format Files**, **Schematic Blocks** and **Technology Files** dialogs. When a file is selected and the Edit button pressed, the file is loaded into the respective editor.

# New Dockable Bars

## Introduction to New Dockable Bars

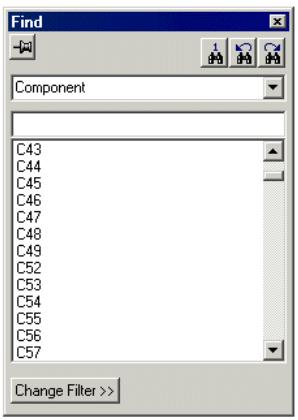
There are three new types of Dockable bars; **Find**, **Layers** and the **ERC Errors Browser** (Electrical Rules Checker). Each of these bars is modeless which means it can 'float' over the application, or it can be docked. The Find bar is discussed here in detail, the Layers and ERC Errors Browser discussed later on in this document under their respective applications.

## Find

The Find *item* options have been replaced with a more sophisticated **Find Tool** operated from a Dockable bar. Using the shortcut key <Ctrl-F>, you can find:

| | | |
|---|---|---|
| Attribute | Copper | Pad or Pin |
| Block Port | Error | Template |
| Component | Mounting Hole | Testpoint |
| Connection | Net | Wire |

Each of the item types is selected from a drop down list.




The dialog presents you with a list of items from which to choose.

The list will show named items of the selected type in the design. In the example above, components are shown. If the item type does not have names (e.g. Copper) then a list of nets containing items of the chosen type will be shown. If the item type is **Errors**, a list of the possible error codes is shown. If the item type is **Attribute**, a list of the used attribute names is shown (see below).

Click on a named item to find it in the design. It will be highlighted (or optionally selected) and optionally centred in the display window. If the list contains a list of nets, click on the required net or use <Search entire design>. The first item of the required type on the chosen net (or entire design) will be found.

On the browser bar are two additional sets of icons (these are item sensitive and are not enabled for every item type):


The ‘pin’ icon on the left hand side is for ‘latching’ this browser if ‘floating’ and is discussed below. The three other icons are for Finding the First item in the list, Finding for Next item and Finding for Previous item.

Once items have been found (using any conditions from the Filter options, if Filtering), the Find First will then start the search on the first item. Next and Previous will toggle through the list of found items if there is more than one item found.

## The item filter

The Find dialog has a Filter mechanism so that item searches can be made on specified criteria.

To use this option, click on the **Use Filter** or **Change Filter** button if you want to further define the search criteria and reduce the number of items to find. If the button has the text **Change Filter**, then a filter is currently being applied to the list. When the button is selected a set of extra filter controls will appear to use, these are detailed below. When the **Hide Filter** button is selected, these filter controls will disappear.

Check the **Filter Names** checkbox to enable the name filter control. If the Find list contains a list of named items (e.g. components) or net names, type the filter string containing wildcard characters '*' and '?' into the control underneath the **Filter Names** checkbox. Use the drop down list to obtain filter strings previously used for this item type. When the filter string has been changed the **Apply** button will be enabled. Press **Apply** to apply the filter string to the list of names.

Further filtering of items can be done using an attribute condition. Check the **Attribute:** checkbox to enable the attribute condition controls. Use the dropdown list underneath the **Attribute** checkbox to choose the attribute name you wish to use. Next use the **Is** dropdown list to select the type of test to apply to the attribute value. Then, in the remaining edit box, type in a string representing the condition to test against.

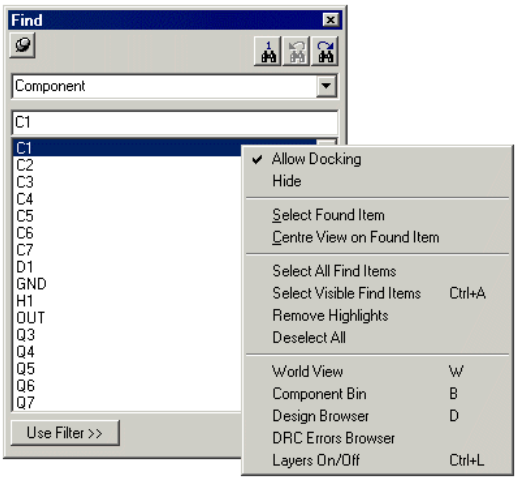
Press **Apply** to reduce the items to be found to items that satisfy the attribute condition. The attribute filter applies to unnamed items as well as named items.


## The ‘pin’ status of Dockable bars

The pin on some Dockable bars is used to show whether the dialog is latched or not. In an unlatched status (shown in the picture below but only applicable if ‘floating’) once an item is selected from the list, the dialog dismisses itself. If the pin is latched, the dialog is persistent and always displayed until closed by you.


## The context menu

From within the Find Browser you also have options available on the right hand mouse menu.



**Search All Pages** - Applies to a schematics design. Check this to look for items on all pages in the design, or uncheck to only search the current page.

**Select Found Item** - Check this to select the item when it is found. Leave unchecked to just highlight the item.

**Include Default Nets** - Only applicable if the list contains a list of net names. Check this to include default nets in the search. Leave unchecked to only list user named nets.

**Include Locked Errors** – This is only applicable if the item type is errors. Check this to find locked errors. Leave unchecked to find all errors.

**Only List Errors in the Design** – This is only applicable if the item type is errors. Check this to only list the error codes of error markers that are currently in the design. Leave unchecked to list all the available error codes.

**Centre View on Found Item** - Check this to always change the view so that the found item is in the centre of the display window. Leave unchecked to only change the view if the found item was not on the screen.

**Check All** - Only applicable if the static list contains check boxes. Use this to check all the boxes in the list.

**Uncheck All** - Only applicable if the static list contains check boxes. Use this to uncheck all the boxes in the list.

**Select All Find Items** - Use this to select all items that satisfy the search criteria, (item type, name filter and attribute condition).

**Select Visible Find Items** - Use this to select all visible items that satisfy the search criteria, (item type, name filter and attribute condition).

**Remove Highlights** - Use this to remove highlighting from all items in the design.

**Deselect All** - Use this to deselect all items in the design.

**Switching to other dockable windows** - The next section contains a list of other dockable bars that you may wish to use. If you click on one of these, it will become visible and replace the Layers On/Off bar. This allows you to use a dockable area for just 2 bars ( for example) and keep replacing them with the next bar you wish to use.

## Layers

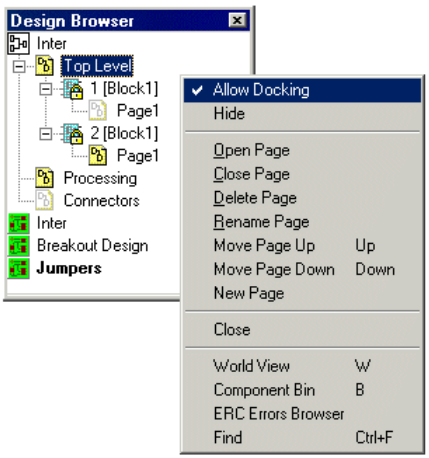
The new Layers Browser window is specific to the PCB design editor application and PCB Footprint editor. This is discussed later in the chapter on *PCB Design*.

## ERC Errors Browser

The Electrical Rules Error browser bar is specific to the Schematic design editor and is discussed later under the chapter on *SCM Design*.

# Changes to Design Browser

The Design Browser has been changed to now show hierarchical blocks (in the Schematic Design Editor). Changes have also been made on the right mouse context menu available when the cursor is over the Design Browser window.



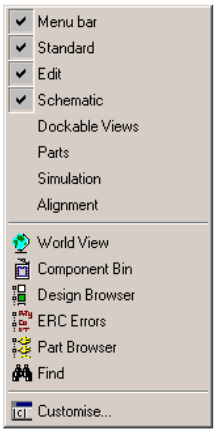

Multiply instanced blocks are shown with different icons; a block with a pencil for an unlocked block, and a block with a padlock for locked blocks. Block instances show their block names in square brackets.

You can now re-order pages using **Move Page Up** and **Move Page Down** on the browser context menu. You can also re-order block instances in a page using **Move Block Up** and **Move Block Down**. For both these options, using the cursor up and down keys will do the same.

Multiply-instanced blocks can be unlocked using an option on the context menu in the browser. These can also be updated to match the modified ‘unlocked’ block (so they all are the same) using the context menu in the browser.

# Changes To Toolbar Menu

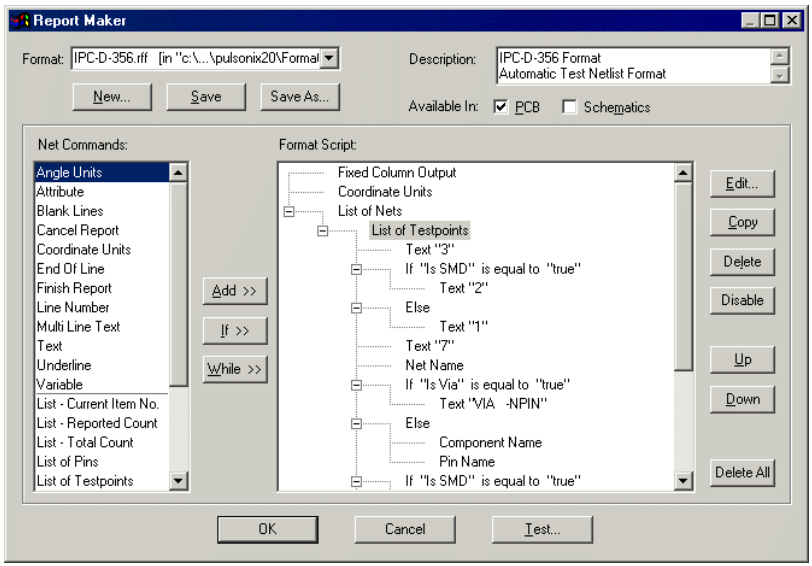
The Toolbar context menu that is available on the Pulsonix toolbars or main menu has been modified so that it is shorter and is design context sensitive. This makes it easier to read and provides more space for the new items, such as ERC Errors browser and Find dialog.



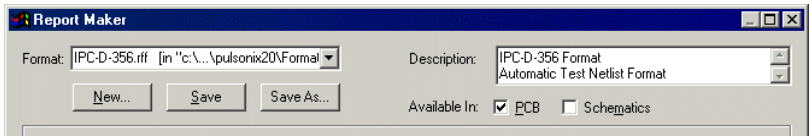

# Report Maker

The **Report Maker** option replaces both the Formattable Partslist output option and the Formattable Netlist output option on the **Output** menu. If you have files saved in these formats, they can still be run from the Reports option, but can only be edited using the **User Formatted Part List** or **User Formatted Net List** option run interactively from the Run Command option on the Edit menu. Commands are available to directly replace all commands previously available in the Formattable Parts and Netlist outputs.

The Report Maker is much easier to use with the dialog being more logically laid out than the Formattable output mechanism. It also now combines both the Parts and Netlist outputs for reports of all design types which require both types of information, e.g. IPC-356 test format.

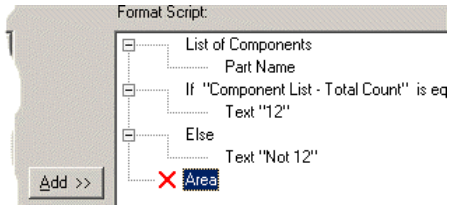



The dialog is broken down into a number of functional areas.



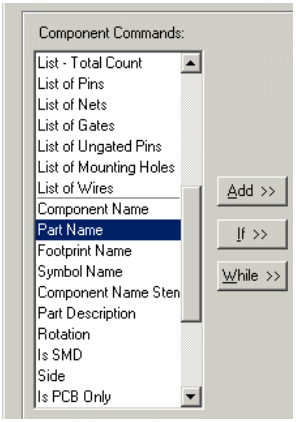

The **Format** drop down list is a list of all the available report files. These are found by the system using the search criteria from the **Folders**, **Format Files** page. **New**, **Save** and **Save As** are utilities for maintaining the format files. The **Description** entry is for labelling the format file. This description is displayed in the **Reports** option when the file is selected in the list. It is provided to give an indication of the activity of the file when run. Format files can be made to be available in either or both of the **Schematic** or **PCB** design editors by selecting their respective check boxes.

If you create a format file for both design types then later switch one of the type off, you may find that some commands do not appear on the selection. If used, these will be presented with a small red cross to indicate that they are unavailable for this design type.



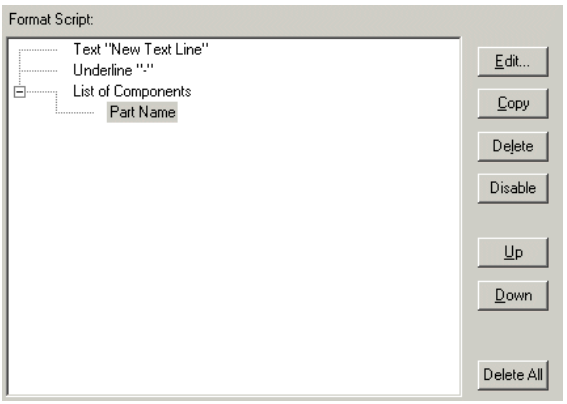

The **Commands** are used to make up the scripts. The process is to select a command and add it to the **Format Script** area using the **Add>>** button or by double-clicking the command from the list. Commands are grouped in logical

functions and in hierarchical design order importance, starting with general commands first and ending with specific item commands at the bottom of the list.




The extra two buttons in the centre panel under the Add>> button are for inserting conditional operations to the commands. These are represented with **If>>** and **While>>**. The **Else** operator is inserted as part of the **If** command and is selected from that command's dialog along with other choices.

Once in the **Format Script** area, the commands are listed in a tree style and are nested to show their hierarchy.

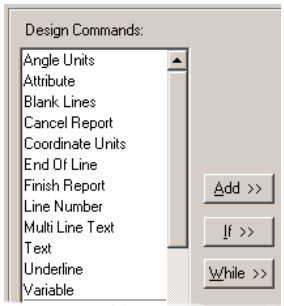



The buttons down the right hand side of the dialog control the Format Script items. When using the **Edit** button, the contents of the edit dialog will be in context of the item selected.

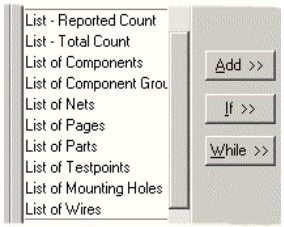
## Using the Report Maker

The principle for using this dialog is to decide what output format you require. Make the selection from the Commands list which then writes the command to the Format Script list. Once in the list, apply the specific command format to the command and sort it in the order required to make the output format correct.

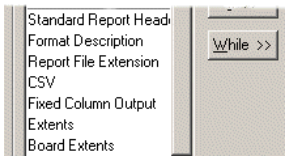
Initially, the Commands list will display 3 sets of commands related to the design. Each set of commands is separated by a thin horizontal line.




The first set of commands are general, and not specific to any context or command. These are always available for use in the script.



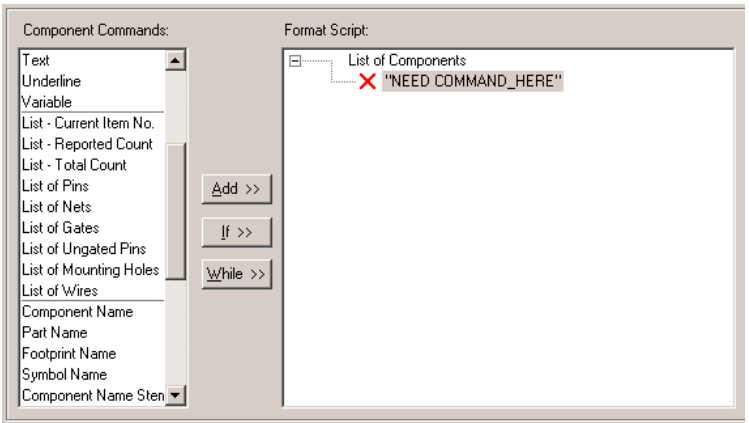

The second set of commands are the lists. These commands will be used to create the 'body' of the script. This is split into two sub categories, one for list variables for **List – Reported Count** and **List – Total Count**, and the other for listing items within the design, **Components**, **Nets** and **Wires** etc.




The third set of commands is always context sensitive usually dependant on the item currently selected in the Format Scripts list. The initial set presented is applicable to the design as a whole.

Picking one of the list commands and clicking **Add >>** will add it to the Format Scripts list, double clicking will also add it without having to use the

Add >> button. For example, double clicking on the **List – Components** will give you this:




You can see that the List of Components is now available and is waiting for a command.

The Command list and is now changed to be context sensitive for the list of Component fields, shown with the heading **Component Commands**.

The Design commands are still available at the top of the list (as they always are), the middle list now shows List commands in context of what is available for Components, and the bottom list are fields for items within the list of Components, such as Part Name etc.

Where you see the words X “NEED_COMMAND_HERE”, you are expected to add a command from the list to complete the script in order for it to run.


If you selected this line (by default it is selected after adding the List command) and then selected the field **Part Name** and added it to the Format Scripts the format would now look like this:


By clicking the **Test** button at the bottom of this dialog you can try the script being created. If using the **Inter.pcb** design supplied with Pulsonix, the report would look like this:

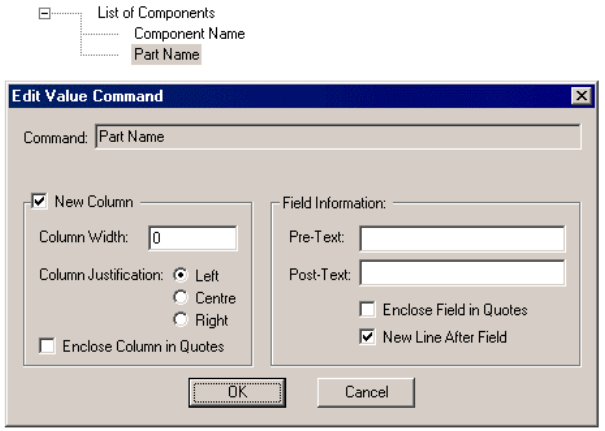
```
C CP3 41612-M-64 3WP 5WP 26WDP 26WDP 3WP BC184L BC184L R
R R R R8C R R R R R R SPDT 74LS251 74LS251 74LS32 74LS125
74LS244 74LS244 74LS251 74LS244 74LS125 74LS251 74LS42
74LS244 74LS42
```

As you can see, the report is a list of Part Names from the design, there is no formatting as such and no filtering for specific names.

Ideally for a simple Parts list output you would add some Component Names and format the report into simple columns.

By selecting the Part Name field, the Component Name is promoted from the Component Command field list to the format script. It appears in the script under the Part Name field. As the Component Name field is still highlighted it can be moved up above the Part Name by using the **Up** button located on the right hand side of the dialog.

Formatting of the output file can be determined by using the **Edit** button (or double clicking) on the Part Name field. Because the Part Name appears after the Component Name, this will be the order that they appear in the output report.




By selecting the New Line After Field check box, a new line is entered after the Part Name. As there is no formatting on the Component Name field this means that the report will now first find a list of components, then output the Component Name followed by the Part Name for each unique instance, then start a new line and report the next item.

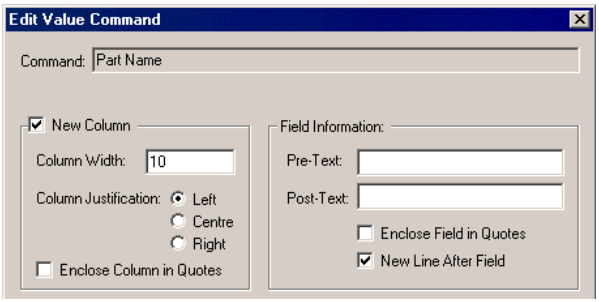
The report now looks like this when the **Test** button is clicked:

```
C1 C
C2 CP3
CONN1 41612-M-64
PL1 3WP
PL2 5WP
PL3 26WDP
PL4 26WDP
PL5 3WP
Q1 BC184L
Q2 BC184L
R1 R
R2 R
```

This list can be formatted into evenly spaced columns even though the text fields are different sizes.

Again, using the **Edit** button, the formatting will need to be on both the Component Name and the Part Name. This means you would need to edit both items in the list.

For each field the **Column Width** would need to be defined and set. You would always need to allow enough space for the largest name in the list.

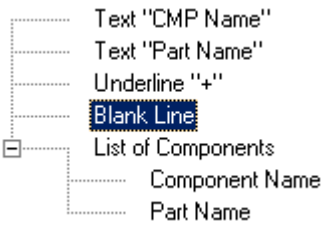



The report now looks like this when the **Test** button is clicked:

```
C1          C
C2          CP3
CONN1       41612-M-64
PL1         3WP
PL2         5WP
PL3         26WDP
PL4         26WDP
PL5         3WP
Q1          BC184L
Q2          BC184L
R1          R
R2          R
```

To add some headers to the formatted list you can add free **Text** before the List of Components statement. We've also added the **Underline** field and the **Blank Line** field. The Underline field will look at the previous field and will automatically underline this field matching the field width. The underline character can be defined by editing this field. The Blank Line field will add a blank line (effectively a <CR>) to the report. The number of blank lines can be changed by editing this field.

As you can see, to make a header appear once at the start of the list all header fields must be on the same 'level' as the List of Components and must be above it.



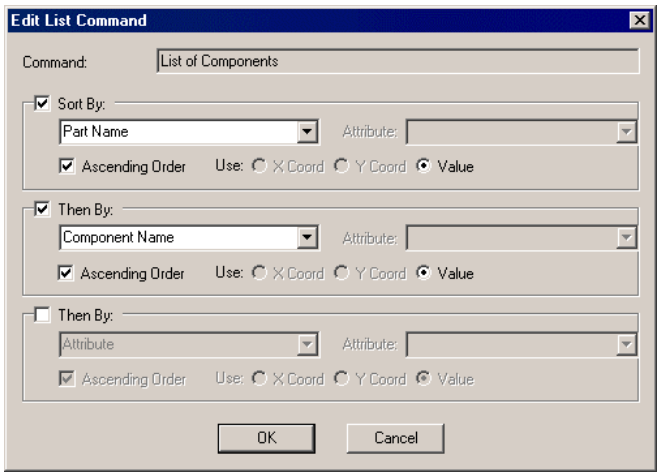
Both Text fields used could have been created using a single Text field with both headers spaced at the correct width. The disadvantage of using this method is that if any of the field widths under the List of Components are changed then you then have to adjust the text fields as well.

The report now looks like this when the **Test** button is clicked:

```
CMP Name  Part Name
++++++++++++++++++

C1        C
C2        CP3
CONN1     41612-M-64
PL1       3WP
PL2       5WP
PL3       26WDP
PL4       26WDP
PL5       3WP
Q1        BC184L
Q2        BC184L
R1        R
R2        R
```

The list can also be sorted. This is achieved by using **Edit** with the List of Components selected. By default the List of Components is sorted by Component Name first.

We will change this so that the Part Name is sorted first followed by the Component Name. This is achieved by selecting the **Sort By** check box and picking a field from the list provided. There are three possible sorting levels available. For additional sorting use the **Then By** check boxes. When sorting, the output can be sorted in **Ascending** or **Descending** order depending on the selection of the **Ascending Order** check box.



The report now looks like this when the **Test** button is clicked:

```
CMP Name  Part Name
++++++++++++++++++

PL3       26WDP
PL4       26WDP
PL1       3WP
PL5       3WP
CONN1     41612-M-64
PL2       5WP
U4        74LS125
U9        74LS125
U5        74LS244
```

You can see that the Part Name 26WDP appears in the report with Component Name PL3 listed first followed by PL4, then the next **Part Name** with Component Names are listed, this is how the entire list if formatted.

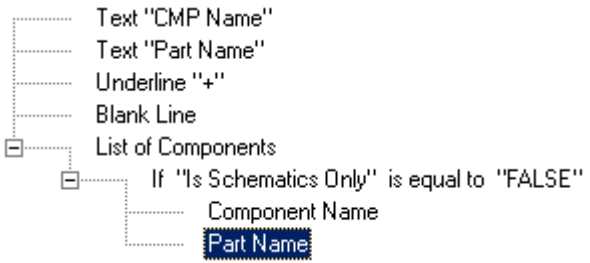
## Advanced Format Scripts

Adding **IF** and **ELSE** conditional statements to the format script adds further control on the data being processed. You can also add **Variable** fields to the script which can be combined with **WHILE** to test for data conditions.

### IF and ELSE

IF statements can be added to test a field.

For our example we will test the List of Components to ensure that our Parts list does not contain any Schematics Only Parts. We will need to make the script look like this:

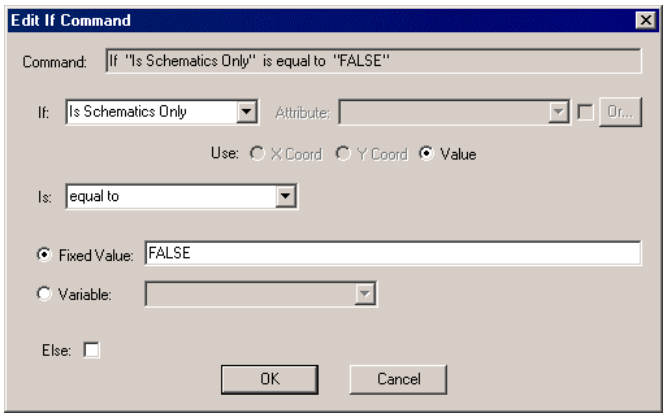



To achieve this you must go through a number of stages. You will need to select the Component Name field first. This puts the IF statement at the correct ‘level’. The test must be under the List of Components, so the script reads, “request a list of components”, “now test to see if the (List of) Components are Schematics Only”, “if they are not” then output “Component Name and Part Name (with formatting)”.

The **If** statement must be added below the List of Components. Select the Component Name field to do this and click the **If >>** button. The **Edit If**

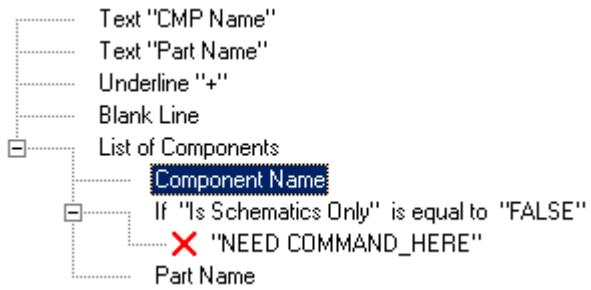
**Command** dialog is displayed. From this you can edit the commands to make the test condition.

For this example, on the **If** list we will select **Is Schematics Only** to test. We will test to see if **Schematics Only** Components **Is Equal to** and then a **Fixed Value** of **FALSE** (so the test is for the Component not being Schematics only).




For the statement **Not Equal to TRUE**, you could also say use **Equal to FALSE**, in other words, test are they Schematics only? If not, output them, if they are, ignore them.

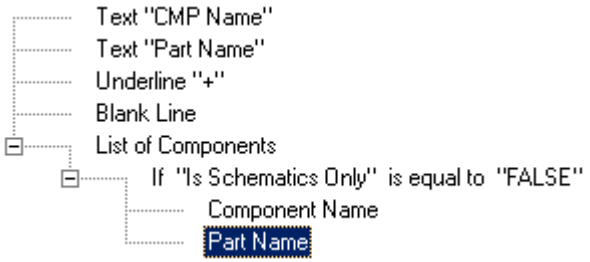
The script now looks like this:




Because the If statement have been added below the Component Name field (you selected the Component Name field to get the IF statement on the correct 'level'), you must now use the **Up** and **Down** buttons to move the Component Name and Part Name fields to be under the IF statement.

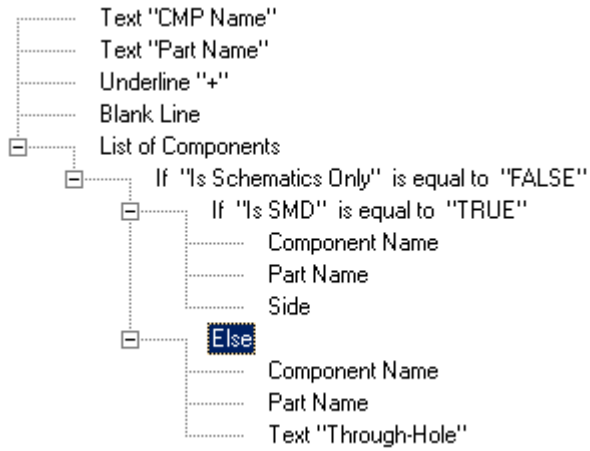
Once the Component Name has been moved down you will notice that the Part Name is now on the wrong nested level. It must be under the IF statement but in-line with IF. Use the **Up** button to move it.

*Note that the X "NEED COMMAND_HERE" message now disappears.*




Using the Test button the report will now look pretty much the same as before (it is a PCB design which obviously can't contain SCM only Components). Save the format file using the **Save As** button. It can be used on a Schematic design, Inter.sch for example.

The **Else** statement can also be used when using **If** to provide an alternative if the **If** test was not true. An example of **Else** is shown below.




For our example file we've added another If statement to test whether the Components are SMD (surface mounted devices) using **If "Is SMD equal to TRUE"**, and if they are we will first output the **Component Name**, **Part Name** and their **Side**. Then if the Component is not an SMD (tested using **Else**) we will output the **Component Name**, **Part Name** and a **Text** field, the text contains the words **Through-Hole**.

Our report now looks like this when the **Test** button is clicked:

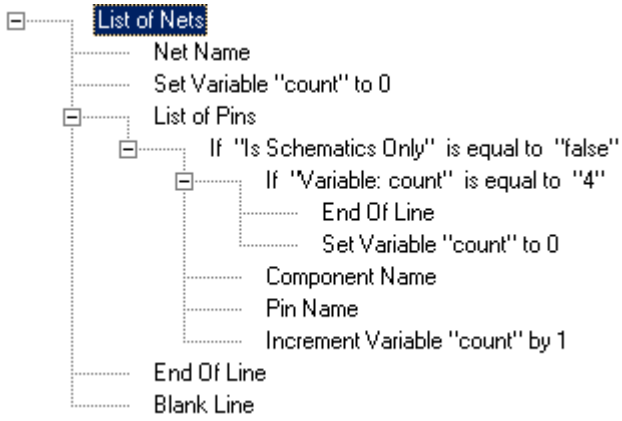
```
CMP Name  Part Name
++++++++++++++++++

Q1        BC184L         Top
Q2        BC184L         Top
PL3       26WDP          Through-Hole
PL4       26WDP          Through-Hole
PL1       3WP            Through-Hole
PL5       3WP            Through-Hole
CONN1     41612-M-64     Through-Hole
```

## Variables

**Variables** allow you to set a status on an item, then test that status and perform commands on it.

In the example below we have used a netlist output which will be driven from within the Schematic editor. Use the standard output file Net List.rff and Inter.sch if you wish to try this.

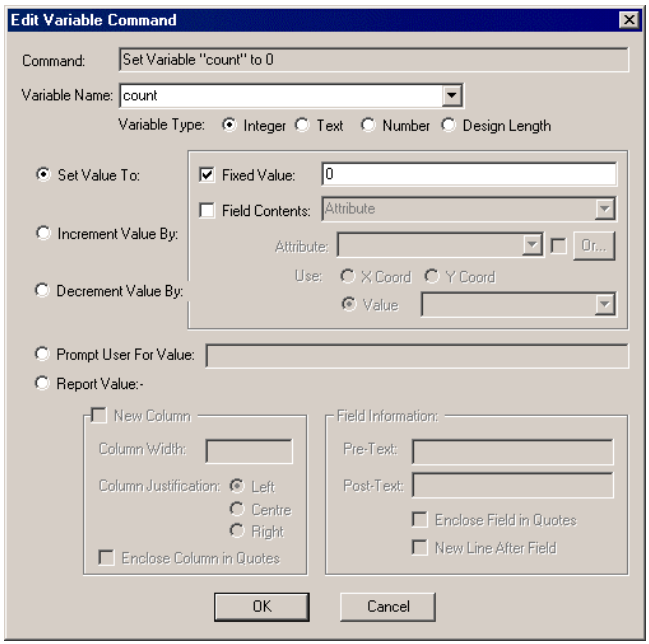
This script will output a netlist with the Net Name first then a list of pins on the net. By using the count variable, the number of pins reported per net name can be restricted to 4 per line, thus keeping them readable when viewed in Notepad.




In this example we initially set the name to **count**, the **Variable Type** to **Integer** and a value of **0**. It will be set to an Integer, as the value reported back on use will be a count number. By setting the **Variable** before the List of Pins it can be tested for at any point after it.

Inside the List of Pins and following the initial test to see if the Components are Schematics Only, the next test is to check to see if the Variable is 0. Initially it will still be set to zero until the script has run through one pass. The first pass will output the **Component Name** and **Pin Name**, the **Variable** will then **Increment** the **count** by **1** and go back through the list again. It will do this for each **Component Name** and **Pin Name** of the same **Net Name** until the **count** of **4** is reached. It then performs an **End Of Line** and continues with the same net name. Once it has finished listing the **Component Names** and **Pin Names** for that net it starts a new **Net Name** after inserting an **End Of Line** and a **Blank Line**. The process then starts again until there are no more **Net Names**.

The **Edit Variable Command** dialog is used to set the **Variable Name**, its **Type** and **Value**.




When the report is run it will be formatted like this:

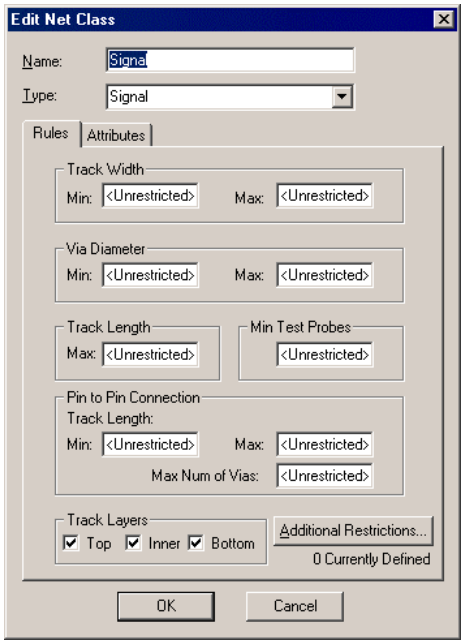
```
Net COMPUTER
PL2.5        U11.4

Net GND
C1.2         C2.2         CONN1.1      CONN1.31
CONN1.33     PL1.1        PL1.3        PL3.12
PL3.24       PL3.25       PL3.26       PL4.12
PL4.24       PL4.25       PL4.26       PL5.2
Q1.1         Q2.1         SW1.2        U1.8
U2.8         U3.4         U3.5         U3.7
U3.9         U3.10        U4.2         U4.5
U4.7         U4.9         U5.10        U6.10
U7.8         U8.10        U9.7         U10.8
U11.8        U11.12       U11.13       U12.10
U13.8

Net ID0
CONN1.14     R5.2         U2.5         U5.11
U6.11        U9.6
```

The formatting is best demonstrated on nets with many pins, for example the GND net.

# Net Class changes to Technology Dialog

## Net Class Rules

The **Net Class Rules** define restrictions on how nets are to be routed. They would normally be defined for critical nets in the Schematic then passed through to the PCB. This gives the Schematic engineer some control over the routing rules that apply to the nets.




Rule values default to **<Unrestricted>**, in other words, no rule is applied. You can define a rule simply be typing in the appropriate value. Clearing a value will set it back to **<Unrestricted>**.

Rules are checked by selecting the appropriate options in the **Nets** section of the Design Rules Checker in PCB. None of these rules are currently applied during the design process, and are only checked when explicitly selected in the **Design Rule Checker**.

### The Rules

**Minimum & Maximum Track Width** defines the limits of the width of all tracks used on a net. Any number of different track styles can used, but their width must all fall between these limits.

**Minimum & Maximum Via Diameter** defines the limits on the sizes of pad styles used for vias on a net. The limits apply to the minimum and maximum dimensions of any shape, and includes all shapes explicitly applied to any electrical layer.

**Minimum & Maximum Track Length** defines the minimum and maximum length of track on a given net. This does not include the length of connections through power planes, copper or vias, but it does include wires.

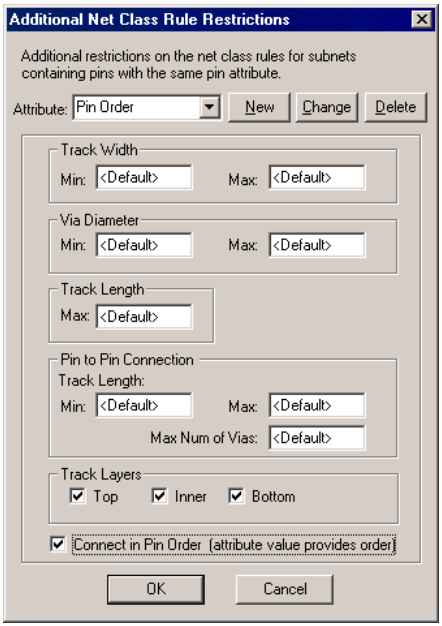
**Minimum Test Probes** defines the minimum number of probe points (Testpoints) for a net.

**Pin to Pin Connection** defines the **Maximum Track Length** and **Maximum Number of Vias** in the shortest path between each component pin and the nearest other pin in the net.

**Track Layers** defines which layers are allowed to contain routes for the net. You can restrict it to any combination of the **top** electrical layer, any **inner** electrical layers, and the **bottom** electrical layer.

**Additional Restrictions** allows restrictions to be defined between a subset of pins on a net. The **Additional Restrictions** dialog is used to define these.

## Additional Restrictions




This dialog allows additional rules to be defined between selected pins on a net.

### How to create Additional Restrictions

Press the Additional Restrictions button on the **Net Class Rules** tab. Firstly; you must nominate a pin attribute to hold the restrictions. To do this, press the **New** button. You can select an existing attribute, or type a new name in the **New Net Class Rules Attribute** dialog. Once this has been done, you can edit the restrictions, as defined below.

Rule values are all initially set to **<Default>**, in other words, the rule defined for the Net Class apply, and there is no additional restriction. You can define a rule simply be typing in the appropriate value. Clearing a value will set it back to **<Default>**.

The **Delete** button will delete the current additional restriction set.

The **Change** button will change the attribute used to define the additional restriction set.

## How the Additional Restrictions are applied

By adding the nominated attribute to a subset of the pins in a net, the additional restrictions will apply to these nominated pins and the connections between them. The value of the attribute is only relevant to the **Pin Order** check.

A minimum set of paths between the nominated pins will be calculated and the rules applied to these paths. The paths may pass through other, unnominated pins, which will be considered as vias in the path. Paths to other parts of the net, or which make a longer alternative path between pins in the subnet, are not included.

## The Rules

**Minimum & Maximum Track Width** defines the limits of the width of all track used in the subset. Any number of different track styles can used, but they must all fall between these limits.

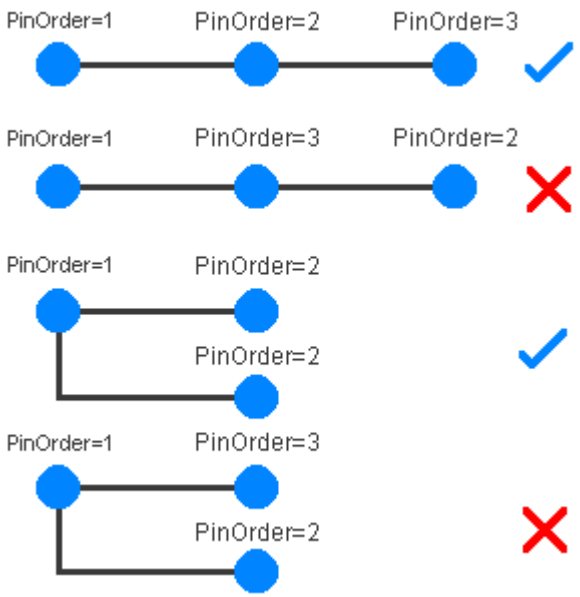
**Minimum & Maximum Via Diameter** defines the limits of pad styles used on vias in the subset (this includes pads not in the subset which are used as vias). The limits apply to the minimum and maximum dimensions of any shape, and includes all shapes explicitly applied to any electrical layer.

**Minimum & Maximum Track Length** defines the minimum and maximum length of track in the subset. This does not include the length of connections through power planes, copper or vias, but does include wires.

**Pin to Pin Connection** defines the **Maximum Track Length** and **Maximum Number of Vias** in the shortest path between each component pin and the nearest other pin in the subset

**Track Layers** defines which layers are allowed to contain routes for the track paths. You can restrict it to any combination of the **top** electrical layer, any **inner** electrical layers, and the **bottom** electrical layer.

**Pin Order** defines the order in which pins should be routed. This rule uses the **Value** of the nominated attribute. The order is defined by an alphanumeric comparison of the values. The connections between the pins should form a chain or tree, such that each pin forms a link between pins with lower values and those with higher values.

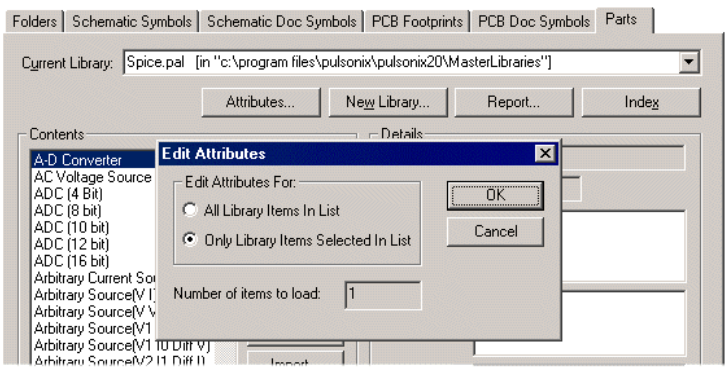



# Parts Library Editor Changes

## Part Attribute Editor

Within the Parts library there is now the facility for editing **Attributes** using the attribute editor. This is available using the Attributes button on the Parts dialog.

By clicking the **Attributes** button you can select all of the library items in the list or the current selection (from the **Contents** list). The **All Library Items In List** radio button will select everything in the contents list, be it the whole library (<All Libraries>), or just a selected library. Care should be taken when <All Libraries> is being used as this may take some time to load into the Attribute Editor.

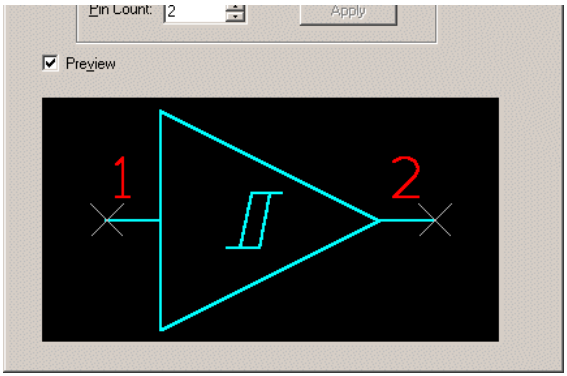


Clicking **OK** will display the Attribute Editor with the Parts Libraries loaded. For Parts you can edit attributes for both Parts and Pins.

*This option is discussed in full detail in the Attribute Editor section below.*

## New Preview mode on Insert Gates

Within the **Part Editor** on the **Gates** page, there is now a Preview mode on **Insert Gates** and **Change Gates** to aid location and recognition of the gates.



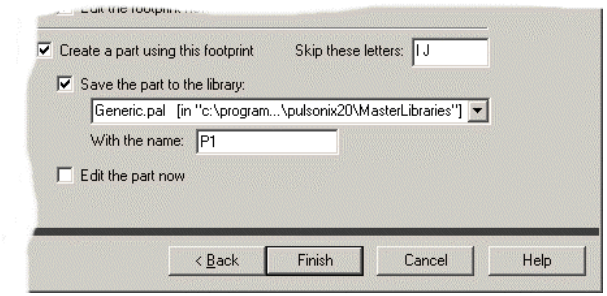

## Auto-updating of Symbol Preview Windows

A change to the **Parts Editor** window means that the **Schematic Symbols** and **PCB Footprint** preview windows are now automatically updated when the respective symbols have been changed in their editors.

## Skip letters in Part pin names for BGA Footprint Parts

There is an extra edit box on the **Finish** page of the **Footprint Wizard**, which is only visible when creating **BGA**-type Footprints.

Use this to specify any letters to be "skipped" when naming pins, so for example, you can enter **I J** to name pins as: A1 B1 C1 D1 E1 F1 G1 H1 K1 etc. missing out rows **I** and **J**.



# Associated Parts

Where Parts are defined in the Parts libraries, you may also include additional Parts which although are added to the manufactured PCB, do not appear on the Schematic or PCB design as electrical items. For example, a transistor which when built must have a heatsink, a thermal washer, a nut and bolt and a plastic washer. The design may also contain Associated Parts which must be ordered as part of the Bill of Materials but which aren't electrical items, for example mounting columns and plastic insulation washers and nuts.

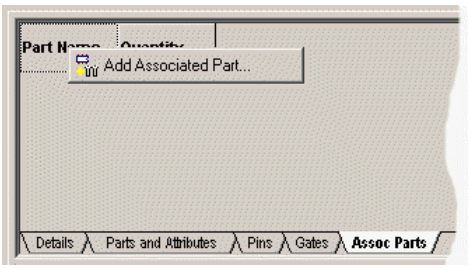
These Parts are called Associated Parts and can be of Schematic or PCB or both usages. All Associated Parts can then be extracted using the Report Maker facility.

Where Associated Parts are 'Footprint' specific, for example in the case of some transistors where there may be up to four different mountings, and hence four different associated Part variants, these Associated Parts can be defined with the Footprint itself. This uses exactly the same mechanism as the Design level Associated Parts using the **Design Properties** dialog.

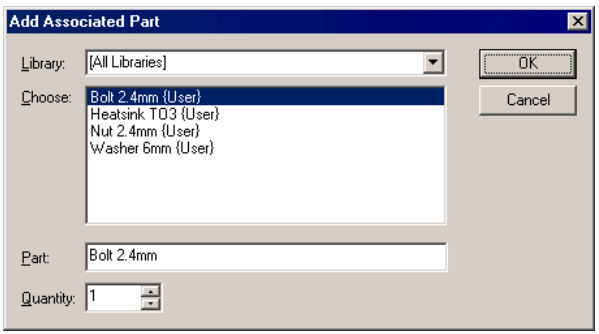
## Associated Parts

From within the Parts Editor dialog you can associate Parts using the Associate Parts dialog. This will provide you with a list of all the additional Parts which will be called whenever the main Part is used in the design.

By right clicking in this dialog you are presented with the **Add Associated Parts** option.



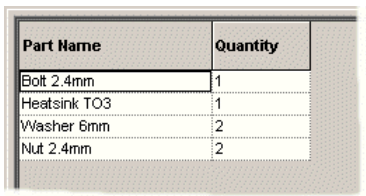

When this option is selected you are presented with the dialog:



You can use this dialog by either typing in the **Part Name** into the **Part** entry box or by selecting from the list presented in the **Choose** box. The Choose box will list all Parts in the Parts library which have been defined with no pins. If the Part has pins then you must type in its name to use it.

The **Quantity** can of each associated Part can also be defined so more than one Part can be used for each main Part.

The associate Parts for a Part definition might then look like this:



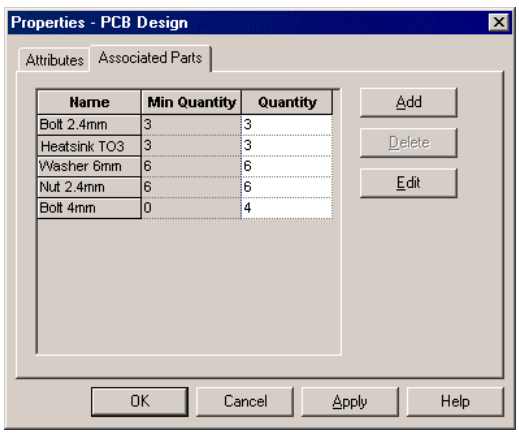
| Part Name | Quantity |
|---|---|
| Bolt 2.4mm | 1 |
| Heatsink TO3 | 1 |
| Washer 6mm | 2 |
| Nut 2.4mm | 2 |

The Part is then saved.

Each time this Part is used in the design it will also include an instance(s) of all the associated Parts. The full list of Associated Parts for the design can be viewed at any time by using the **Design Properties** option from the **Edit** menu.

## Design Level Associated Parts

Quantities can be changed on a design basis using the **Design Properties** dialog. This dialog displays a list of all the Part based Parts Associated, and design based Associated Parts which have been added to the design.




In the above example, all items except the *Bolt 4mm* Part have been added as Part based Associated Parts. The *Bolt 4mm* has been added through this dialog as a board (or design) Associated Part.

The Associated Parts list shows the name of the Part, the minimum number of instances (which are associated with Components in the design), and the

total number which includes any additional instances which have been defined through this dialog. You cannot have less than the minimum number.

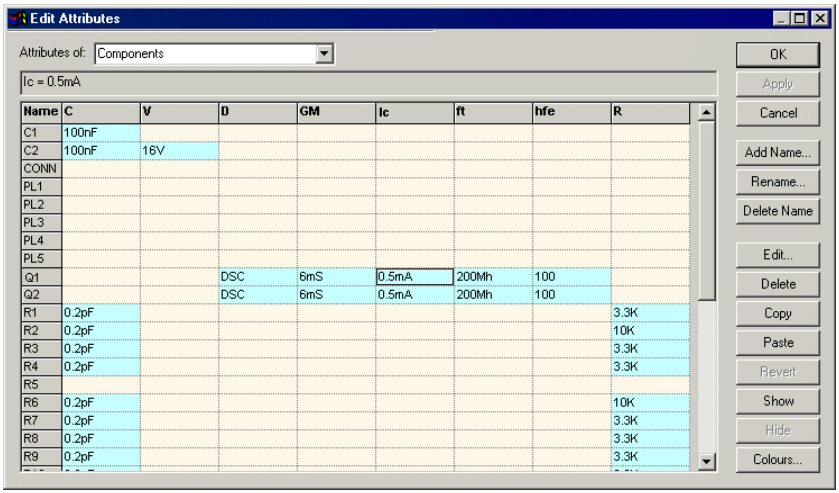
The **Add** button allows you to add an associated part to the design. This brings up the **Edit Associated Part** dialog.

To edit an existing Associated Part, select the appropriate row in the grid and press **Edit**. This brings up the **Edit Associated Part** dialog, with the data from the current row pre-selected. You can also change the quantity directly in the grid.

Use the **Delete** button to delete the selected row. You can only delete Associated Parts that have a minimum quantity of 0.

# Attribute Editor

The Attribute Editor option presents all the attributes in a design (or library) 'spreadsheet style' table, this is available as **Attributes…** from the **Edit** menu, or the **Attributes** button on the Libraries dialog.




At the top of the dialog, choose the type of item from the drop-down list. The types of items available will depend on the type of design or library from which you have invoked this dialog.

Down the right-hand side of the dialog is a set of buttons that allow you to modify the attributes.

The remainder of the dialog is occupied by a grid or table of the attributes currently in the design. Cells in the grid are coloured depending on their 'state', indicating for example whether a particular item has the attribute defined but empty, or defined but hidden, or defined and visible.

Unlike other dialogs, the Edit Attributes dialog can be resized to allow you to view more attribute information in one go. The size and position of the dialog will be remembered for next time.

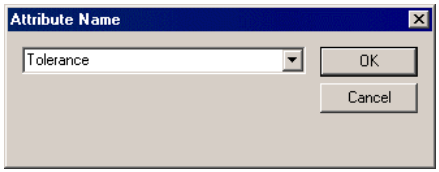
Editing attribute values can be done directly in the grid, just like working with a spreadsheet. They can also be modified using the function buttons provided.

The order in which items are displayed in the dialog can also be changed. Clicking on the column heading will sort the grid by the values in that column. For example, clicking on the "Name" label at the top of the first column will sort the grid by item name (this is the default sort order).

The buttons down the right-hand side of the dialog provide you with tools to edit the attributes and their values. Some buttons may only be enabled if appropriate cells are selected in the grid.

## Add Name

This will add another attribute name (Attribute) to the grid, allowing you to create a new attribute to use for attributes be added to items in the design.



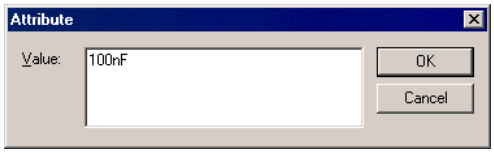

Enter the attribute name to be used for the new column and press OK. An error message will appear if you enter an attribute name that already exists. Otherwise, a new column will be added to the grid, with the new name shown in the heading row at the top of that column.

## Rename/Delete Name

Use this to rename or delete an existing attribute name .

## Edit Values

This will open another dialog, allowing you to edit the value of the attribute in the selected cells. If more than one cell is selected, and the contents of the cells are different, the attribute value displayed will be the special value "<Different>". Otherwise, the current value of the selected cells will be displayed for you to edit.




Edit the Value to contain the text you want, and press OK. The grid will be updated to show the new value in all the selected cells.

### Delete

This will remove the attributes from the selected cells.

### Copy and Paste

As well as entering values directly, you can also copy and paste values between cells (or ranges of cells) in the grid. For example, to apply the value in one cell to a number of other cells, start by selecting the first cell and pressing Copy. Then select the other cells (hold the mouse button down to select a range of cells), and press Paste. The same value will be pasted into all the selected cells.

### Revert

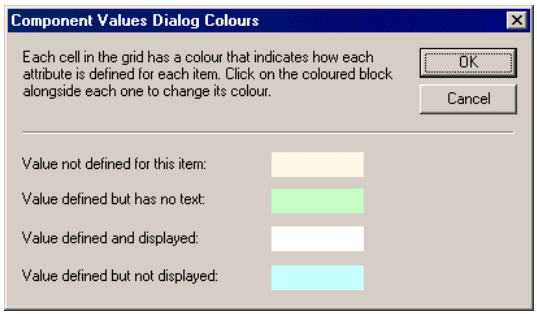
Use this button if you have accidentally modified the wrong cells but don't want to cancel the whole dialog. Pressing Revert will return the selected cells to their original values, leaving other cells untouched. This button will only be enabled if one or more of the selected cells have been modified.

### Show and Hide

These two buttons allow you to show or hide the selected attribute(s) in the design.

### Colours

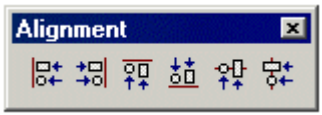
This button opens another dialog that allows you to customise the colours used to differentiate between the different states or types of attribute on the main editing dialog.




The colours are retained in the system registry, so that your choice of colours is preserved for future use.

# Alignment Tools

There are 8 new options for aligning items with other items. Tools are available for, Align Left, Right, Top, Bottom, Centres Vertically, Centres Horizontally, Origins Vertically and Origins Horizontally.

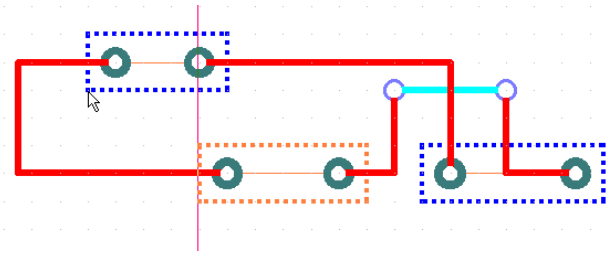



Six of the main tools are shown on the new Alignment Toolbar. These are:

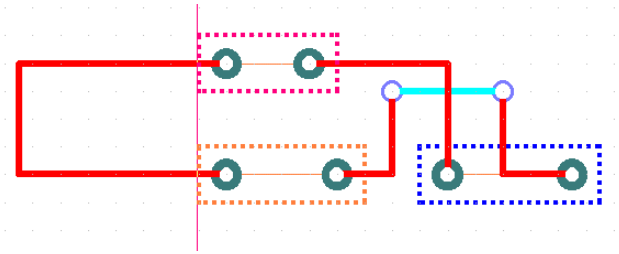
Align Left, Align Right, Align Top, Align Bottom, Align Centres Horizontally and Align Centres Vertically.

These tools allow you to align whole items, segments or ends of linear dimensions.

When using any of the align tools in ‘latch’ mode, the first item selected will be highlighted and a thin line displayed showing the alignment origin. A pad centre to component edge for example. Otherwise they are aligned with the last selected item.



You can see in the example below that the bottom jumper has been selected as the reference and the top jumper has moved to align it’s left edge with the other jumpers left edge (the **Align Left** tool has been selected).



# New Shapes

New shape modes have been added to allow a **Triangle** and a simple **straight line** to be created.

**Add Square** is also available by restricting **Free Movement** on the right mouse menu during **Insert Rectangle**.

**Insert Triangle** and **Insert Line** are available on the **Insert Copper** and **Insert Shape** options. These have two modes, orthogonal and free.

During **Insert Triangle** you can create a triangle in a number of modes; **Free Movement** allows uneven lengths of sides, the **Equilateral Triangle** mode allows all sides to be equalised.

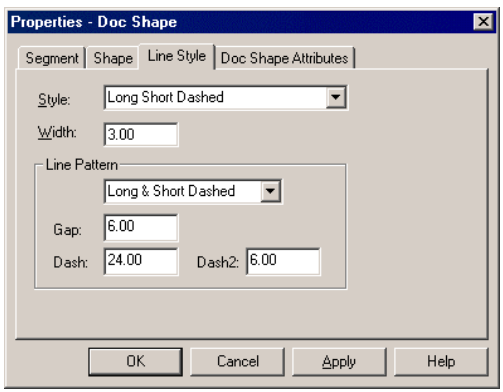
Using this mode you can create a proper triangle without calculating the angles and side lengths to make the correctly formed triangle. After the first side has been added, Orthogonal mode snaps the moving segment side to a 90 degree triangle or to the centre to create two other equal sides of the triangle with a 45 degree angle. With the triangle corner dynamic, a right click will enable the triangle to be snapped into the Equilateral Triangle mode to fix this corner.

Insert Triangle is also available for **Cutouts**.

The simple **Insert Line** option has been created to allow just a straight line to be added to the design or library item (the previously available Insert Polygon as an open shape is still available). During **Insert Line**, from the right mouse menu you can change the mode from **Orthogonal** and **angled** to **Free Movement**.

# New Line Styles

New line styles of **dotted** and **dashed** lines and various combinations have been added to line styles. This applies to doc shapes and unconnected copper. There are various dot and dash line styles available - dotted, dashed, dot/dash, and user defined dot/dash spacing. Each of the standard line styles is available from a drop down list.



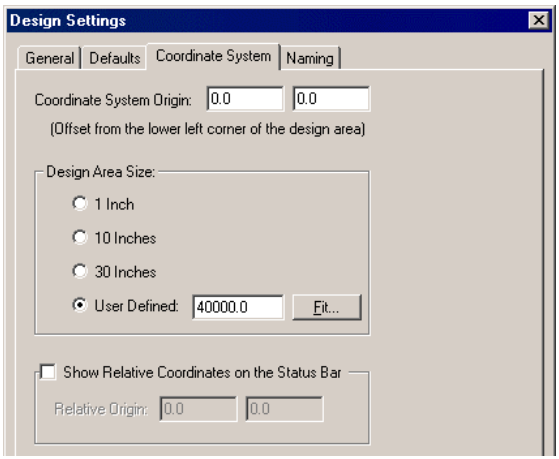
The diagram below shows how the entries in the dialog relate to the actual dashes and gaps produced in the design.


# Design Settings

## New Design Setup dialog

The **Design** tab on the **Options** dialog have been moved to the new **Design Settings** dialog (on the **Setup** menu). The **Defaults** tab has also be moved to this new dialog. This dialog also includes some of the options found on the previous **Design Properties** dialog on the **Edit** menu. These changes are designed to make the design settings accessible in one location.

There is only new functionality on the Naming dialog, this is discussed under the section on *Hierarchy*. The other tabs are just a consolidation of existing options into the Design Settings dialog.

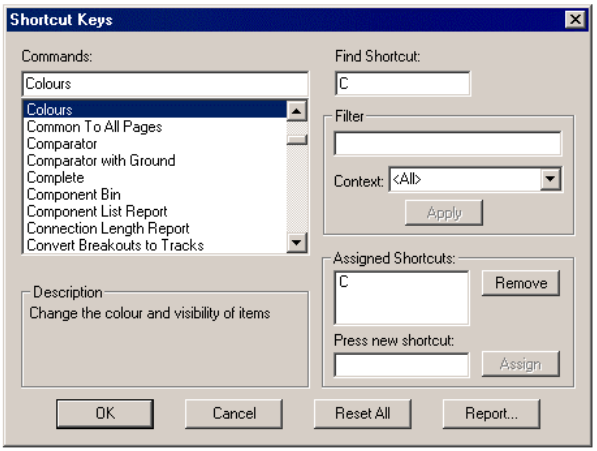



## Design Settings - Matching Styles

An extra radio button added **Match By Value Only** has been added to the Design Settings dialog. The old **Match By Value** option has been renamed to **Match By Name And Value** which is what it used to do.

# Shortcut Key Dialog

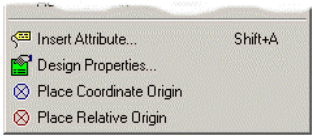
The **Shortcut Keys** dialog has been re-organised to make it more logical to use. You can now **Filter** the list of commands for more specific searching and there is no separate dialog for defining the key to assign.




# Minor General Changes

## Context Menu Change in Design

Right clicking the mouse with no selection within a design will now show an extra selection on the menu. The **Place Coordinate Origin** will position the design origin at the end of your cursor ready for placing. Once on the end of your cursor you can place it interactively, to a specified coordinate or with a specific offset.

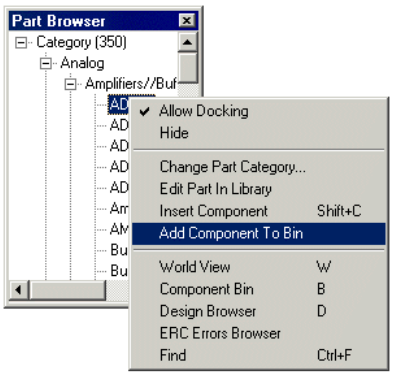



## Pre-selection of Items in Technology dialog

If items in the design are selected, opening the **Technology** dialog will preselect a row on the dialog that corresponds to the first suitable selected item. For example, if you select a whole **Component**, the **Pad Styles** page will preselect the row for the pad style of the first pad on the Component, the **Line Styles** page preselects the row for the first doc shape, and so on.

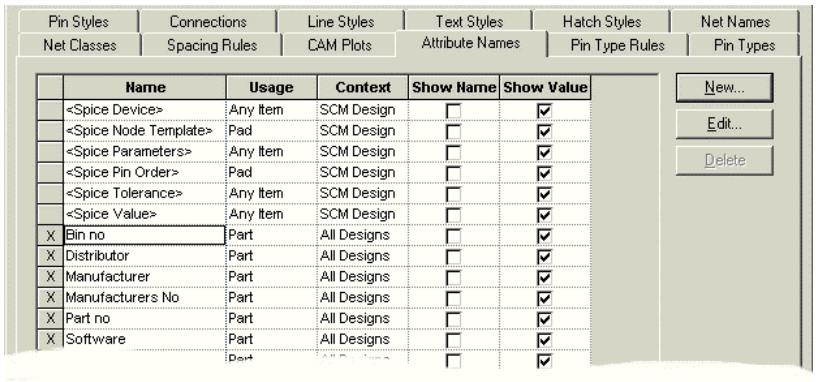
## Changes to Part Browser

From within the existing **Part Browser** window context menu there is a new option - **Add component to Bin**. This allows you to add any Parts displayed in the Parts Browser directly to the Component Bin.




## Visibility of Attribute Names

Attribute names and values can be selected for display using the **Technology** dialog and **Attribute Names**. You can choose to display Attribute Values, Names or both Name and Value shown as Name=Value, e.g. R=22K




| | Name | Usage | Context | Show Name | Show Value |
|---|---|---|---|---|---|
| | <Spice Device> | Any Item | SCM Design | ☐ | ☑ |
| | <Spice Node Template> | Pad | SCM Design | ☐ | ☑ |
| | <Spice Parameters> | Any Item | SCM Design | ☐ | ☑ |
| | <Spice Pin Order> | Pad | SCM Design | ☐ | ☑ |
| | <Spice Tolerance> | Any Item | SCM Design | ☐ | ☑ |
| | <Spice Value> | Any Item | SCM Design | ☐ | ☑ |
| X | Bin no | Part | All Designs | ☐ | ☑ |
| X | Distributor | Part | All Designs | ☐ | ☑ |
| X | Manufacturer | Part | All Designs | ☐ | ☑ |
| X | Manufacturers No | Part | All Designs | ☐ | ☑ |
| X | Part no | Part | All Designs | ☐ | ☑ |
| X | Software | Part | All Designs | ☐ | ☑ |

## Next Cursor Command

There is now a command **Next Cursor Type** for cycling through the application cursor styles such as crosshair or standard (setup in the **Options** dialog and **General**). You can assign this command to a shortcut key if required.

## Security Dump Cursor

When security dumps are being performed a Save icon cursor is displayed.

## Paste into original position

When you paste an item into a design, you can use the command **Cancel Move** from the right mouse menu to place the item in its original position. Previously, the item was dynamic during Move on the end of your cursor and was difficult to position.

## Net Completion Report

The report has been redesigned to divide the nets into lists to make it easier to check.

```
Split Nets
----------
GND
    Gap between (25350.0 21285.0) and (25900.0 21190.0).
    1 connection to complete.

Single Pin Nets
---------------
None

Empty Nets
----------
None

Fully Connected Nets
--------------------
None

Summary
-------
Checked 1 of 1 nets.
1 split nets found.
1 connections to complete.
0 single pin nets found.
0 empty nets found.
```

## Changes to Attributes

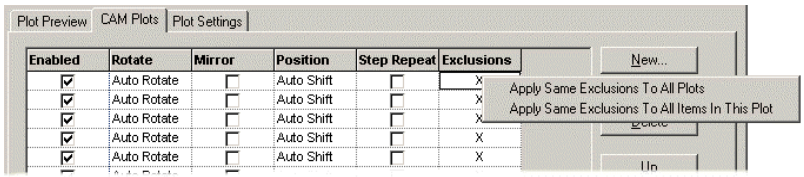
The attribute types of **Component** and **Symbol** have been removed from the lists of ones available; you should use **Part** and **Any** instead.

## Rotate One Step vs Rotate By 90

The **Rotate By 90** degrees command is now the command associated with the default <R> key. The **Rotate One Step** command is now associated with <Alt+R>. This changes means that you have to use the Rotate One Step command for specific rotational steps if the steps are not increments of 90 degrees.

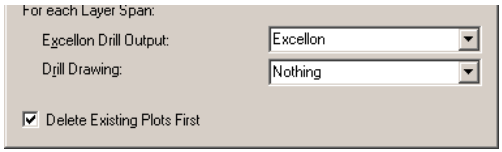
## Cam Plot - Exclude Items

Excluding items from plots still works as it did (items excluded on a per-layer basis, not for all layers in a plot), but a new item on the context menu (right-click on **Exclude** column in grid) allows you to copy Excludes from 'master' layer in a plot to all other layers in a plot.



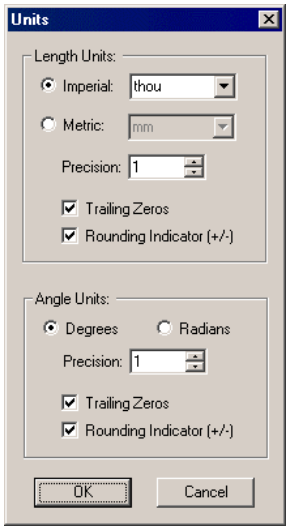

## Cam Plot - Autogen Plots Dialog Change

A checkbox has been added at the foot of the **Auto Generate Plots** dialog, to allow you to choose whether or not to **Delete** all **Existing Plots First** before auto-generation. If unchecked, it will attempt to create plots only for items (layers, pages, etc) that are not already covered.




## Units Dialog Change

Two switches have been added to the **Units** dialog: **Trailing Zeros** and **Rounding Indicator**, these are for both **Length** and **Angle** units (both these switches were previously hidden from you and both set to true).



Switching Trailing Zeros off means that if precision is 4 and the value is 24.1 you don't see the extra zeros, e.g. 24.1000. Switching the rounding indicator off removes the + or - from the end of any displayed number.

*Note, this change allows you to switch the rounding indicator off for inbuilt reports as well as these use the design units definition for their output.*

## Special actions for the Alt key

During the initial drag in **Move Corner** or **Move Segment**, if you keep the **Alt** key pressed you will always get free movement of the attached segments. Holding down the **Alt** key down in **Insert PCB track** will temporarily switch continuous **Online DRC** off (to allow you to select a pad to end on).

# Chapter 3. Schematic Design

## Close Page

The **Close** page option appears on the **File** menu along with **Open** page. It is active once a Schematic page to close a page which has been opened. You can also use Close page from the Design Browser on a selected open page.

## Load/Save Drawing Profile

**Load Profile** in Schematics has been renamed to **Load Drawing Profile**. Along with **Save Drawing Profile,** this has moved and now appears on the **Settings** menu.

**Load Drawing Profile** now deletes the old profile first and replaces it with the selected one (this also happens with the Load Profile option in the PCB design editor).

## Schematic Symbol Wizard

To complement the existing **PCB Footprint** and **Part Wizards**, the **Schematic Symbol Wizard** has been introduced.

This is used to create Schematic Symbols which are used in the Schematic design editor when included in a Part definition. This new option is in addition to the existing Symbol editor.

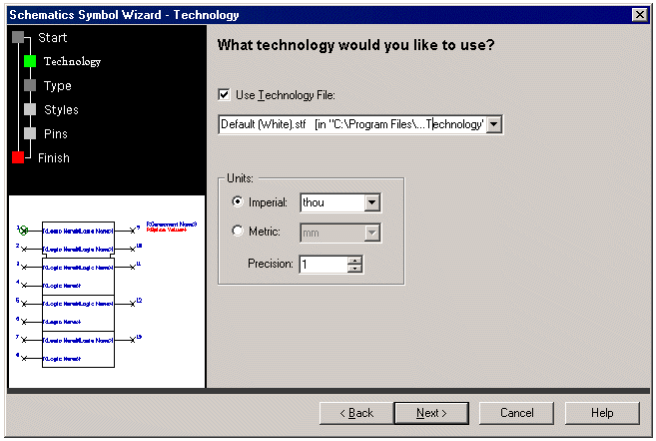
### Running the symbol wizard

The symbol wizard can be run from a number of locations:

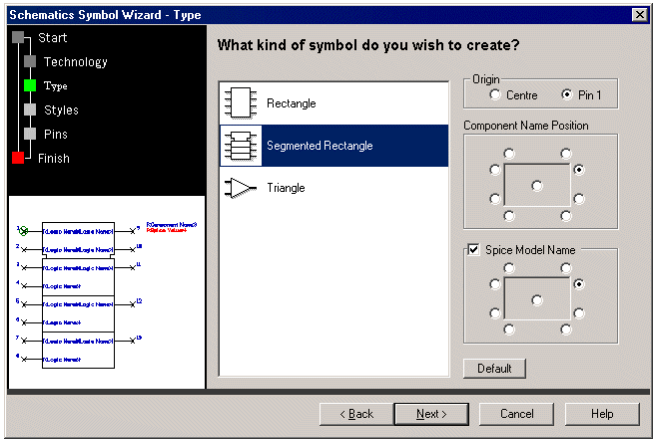
- From the File menu and New option;
- From the Library Manager dialog on the Schematic Symbols page.
- From the Symbol Wizard button on the standard Edit toolbar.

Each page of the wizard represents different facets of the symbol. Each page is summarised below:

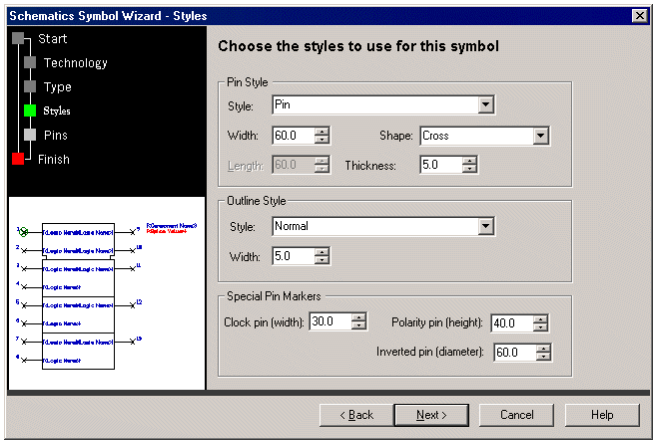
The **Technology** page is used to define the technology file used to create the Symbol. You can also change the Symbols **Units** on this page.



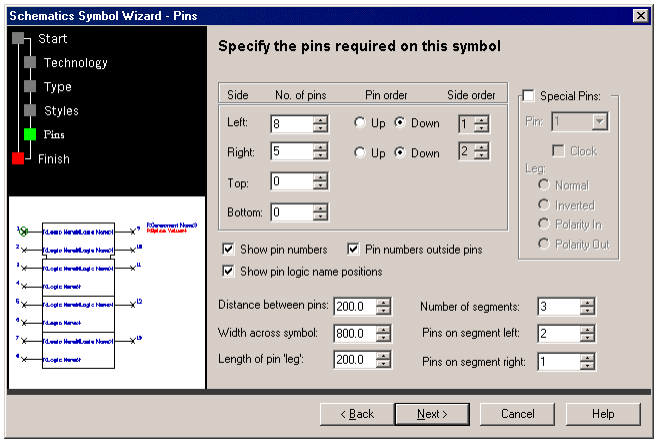

The Type page is used to define the overall 'look' of the Symbol by choosing the shape type from the list of representative pictures. You also define the **Origin** and location of **Component** and **Spice** names.



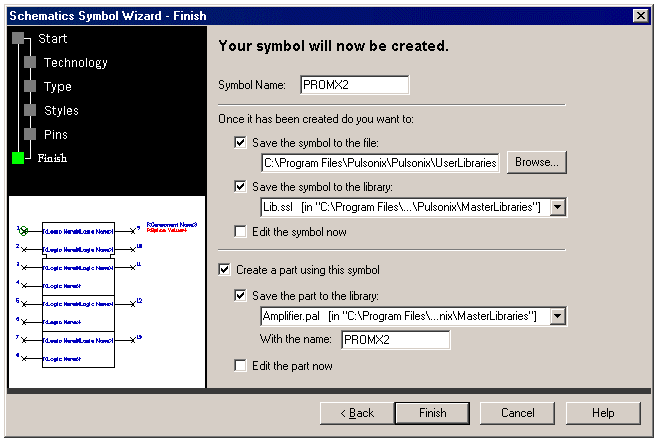

The **Styles** page allows the **Pin** and **Outline** styles to be defined.



The **Pins** page is used to define the number of pins that the Symbol has, the pin pitch and order. It can also be used to add markers to the pins, such as **Pin Numbers** and **Logic Names Positions**.




The **Finish** page is used to define the **Symbol Name** and the **Library** it is saved to. You can also create a **Part** definition directly from this dialog as well.



# Electrical Rules Checking

## Defining The Rules

There is a new checking mechanism which allows pin type rules to be set, then checked on Components which have been allocated these rules. The checking is carried out using an online option, or the batch option.

The order in which this process works is:

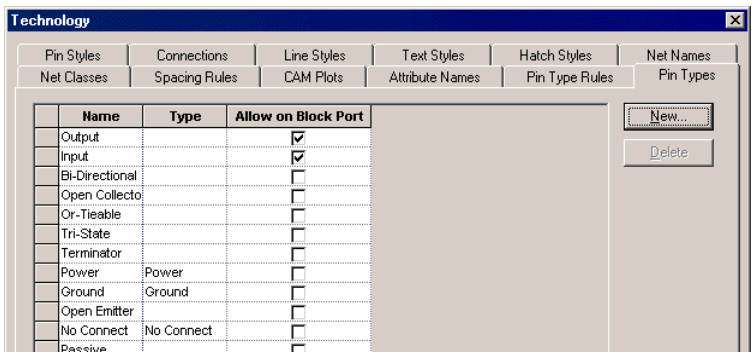
1. Define and add the **Pin Type** to the pin number in the **Part** editor.
2. Define the **Pin Type Rules** in the **Schematic** design or its technology. This is done using the **Pin Types** and **Pin Type Rules** dialogs.
3. Add the Component(s) to the design, these must be Components which have **Part** definitions with **Pin Types** defined.
4. Enable the **Online Electrical Rules Checking** (**ERC**) and start adding connections to the design. Errors found during this process will be displayed, or;
5. Use the **Electrical Rules Checking** option from the **Tools** menu which will check all of the connections and Components in the design.

## Pin Type Rules

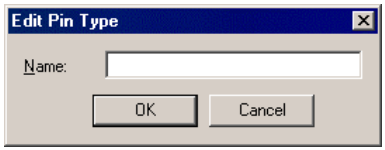
You must set up two items within the **Technology** dialog: **Pin Types** and **Pin Type Rules**. One defines the Pin Type Name and the other defines how each of the Pin Types react with each other.

### Pin Types Dialog

The Pin Type name is added to the **Pin Types** dialog under **Technology**. By default, Pulsonix is supplied with a list of industry standard names, there are no built-in Pin Types as such but a set predefined for you. Any of the names can be edited but care must be taken if you do this as the names will potentially not match Part library Pin Type entries. This list also includes any user-defined Pin Types.



Use the **New** Pin Type button to create a new Pin Type name.




Any names added can be checked once they have been added to the Pin Type Rules definitions and provided they have been assigned to a pin on a Part.

From the Pin Type dialog you can specify if a Pin Type is power/ground/no connect and if it's allowed on block ports.

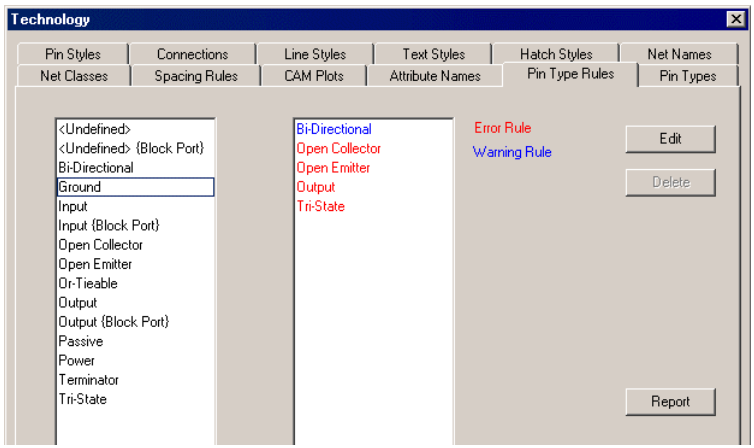
When a Pin type is defined as **Power** or **Ground** then when the Part is brought into the design, if the design has more than one net name assigned to any power net classes then a dialog will prompt you for the net name for which to connect to. A Pin type of **No Connect** will not allow any connection to connect to it.

The **Allow On Block Port** check box allows the Pin type to be used for Block Ports and applies block port rules to block ports when using hierarchy. Block instance pins use the normal pin rules.

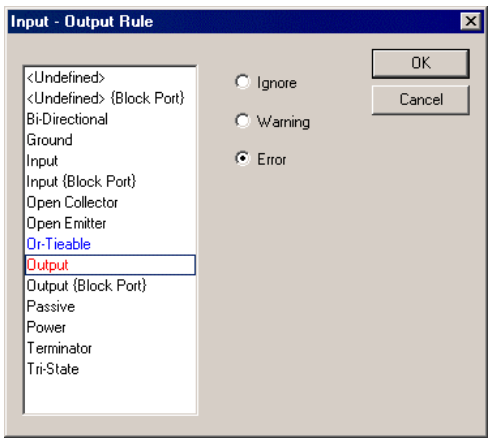
## Pin Type Rules

The **Pin Type Rules** dialog is used to define how the Pin Types will react with each other when connected together. The first column displays a list of all the Pin Types defined (taken from the Pin Types dialog). The second list displays the rule against the corresponding Pin Type, these are colour coded depending on the rule. Any text in **blue** is noting a **Warning Rule** with **Red** noting an **Error Rule**.

A rule can be defined by selecting the Pin Type and clicking the **Edit** button, or by double clicking the Pin Type name.



The **Edit** dialog allows single or multiple selections of Pin Types on which to then apply a rule. The **Ignore** status is the default and is shown in the normal text colour, only **Warning**s and **Error** rules are displayed in the appropriate colour.




Pin Type Rules can be **reported** using the Pin Type Rules report found on the **Output** menu and **Reports**.

An example report looks like this:

```
<Undefined>
<Undefined> {Block Port}

Bi-Directional
     Ground                          Warning
     Input {Block Port}              Warning
     Power                           Warning
     Output                          Warning
     Output {Block Port}             Warning

Ground                          Ground
     Bi-Directional                  Warning
     Open Collector                  Error
     Open Emitter                    Error
     Tri-State                       Error
     Output                          Error
```

**Load Technology** now loads **Pin Types** and **Pin Type Pair Rules**.

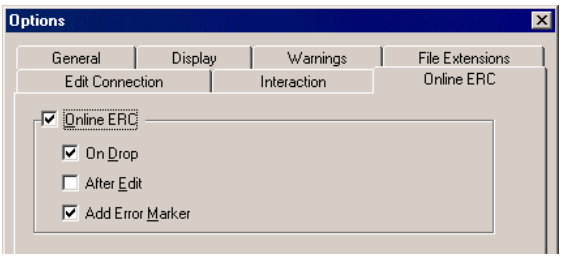
# Electrical Rules Checking (ERC)

There are two modes of checking the electrical rules; online and batch checking. The batch check is more comprehensive and checks all rules set. The online checking option checks a subset of the full batch checking option and is restricted to Pin Types only.

## Online ERC

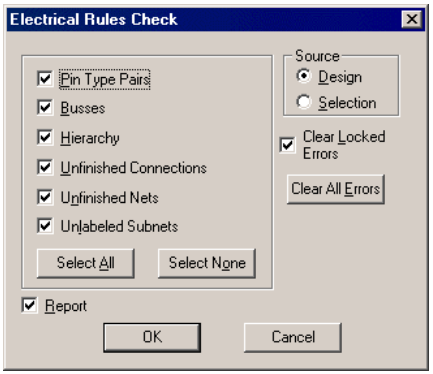
Online rules can be switched on and off in two ways; through the **Options** dialog and **Online ERC**, and by use of the **right mouse context menu** during **Insert Connection**.

The Online ERC dialog on the Option dialog allows the mode to be switched on, and then as with Online DRC mode in PCB, allows it to be continuous or enabled **On-Drop** or **After Edit**. You can also opt to **Add** an **Error Marker** to the design on the detection of an error.




## Electrical Rules Checking Dialog

As well as the Pin Type Rules there are also additional rules which can be checked when the Electrical Rules Checking option is run.




**Unlabeled Subnets** which checks that all subnets of Named Nets have at least one Net Name shown.

**Busses** this is used to check that all the Nets attached to a Bus are connected at least twice on a Bus.

**Hierarchy** – this is used to check that there are matching block pins to block ports in the hierarchy. This also checks that block ports are connected.

**Unfinished Connections** – this is used to check if there are connections which have not been finished off (i.e. the connection starts and end on a pin or terminal).

**Unfinished Nets** – this is used to check that all nets have at least two Component pins in them.

## The ERC Error Browser

When the design contains error markers (added when the ERC check is run and errors are found), you can show a list of all the markers by displaying the **ERC Error Browser**.

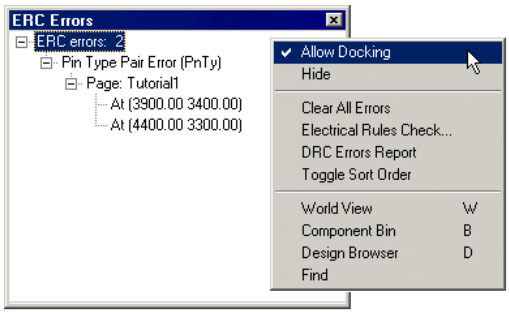
The browser can be invoked from either the **View** menu or from the **Dockable Views** toolbar.

As with the PCB Error Browser, you can select an error marker and highlight it in the design.

### Viewing Errors

To view an error of a particular type, click on the box next to **ERC Errors**. This will then display all the actual error types found in the design. Click on the box next to the error type that you want to look at. This will then give a list of the pages that have this error. Click on the box next to the page name. This will then list the position of all errors of this type on this page. If you then double click on one of the positions in the list, the view centre of the page will move to this position. The error marker will be drawn in the current highlight colour.

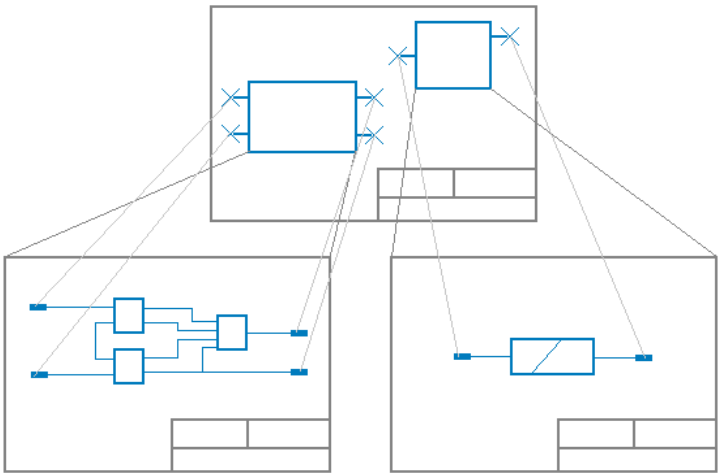
*Note: it is better to use a closer zoom level rather than viewing the whole page.*




Right clicking inside the ERC Errors Browser will display a context menu. From this there are menu items which allow various complimentary options to be run.

# Hierarchy

The introduction of hierarchy means that ‘blocks’ can be created within a design which in-turn can contain circuitry. Blocks are simply shapes that represent a lower level of the design. Pages or Sheets in Pulsonix can be multi-sheet or flat and can be true instanced in all identical blocks.



There are two, not unrelated, reasons for using hierarchy.

- To allow a schematic design to be structured, so that it can be broken down into its functional elements. This promotes Top Down design; break down a problem into functional elements, then define the detail of each element.
- To facilitate the re-use of commonly used circuit elements. This promotes Bottom Up design; use predefined elements to build a solution.

In practice, you may want to use a combination of both Top Down and Bottom Up design.

## Overview Of Hierarchy

### Blocks

A block should be a functional element which has a well defined purpose and interface. In design terms, a block is a special schematic design which describes the implementation of its function, it also has a symbol associated with it which represents the interface.

A block can be instanced in a schematic design (or another schematic block design). The symbol representing the block is placed in the circuit, and can be connected to, in a similar way to a component. However, as well as the symbol being added to the design, all the pages of the block design are also

added. Component and net names are resolved so that there are no conflicts. Connecting to the symbol pins joins the nets with those defined in the block instance.

In many respects, a hierarchical schematic design is no different to a normal multi page design. The net lists would be the same, and the drawings are largely the same. However, it is easier to navigate, because you can **Push** into the pages representing the block instance, from it's symbol, and **Pop** out of the block instance, back to the symbol. It is also possible to update the contents of a block instance, from the original block design.

## Top Down Design

To work top down, add a Generated block instance for each part of the design which is, as yet, undefined. At a later stage, you can **Push** into the pages of the block instance to define its implementation. You can then connect to the block instance using **Insert Connection**, either connecting to an existing pin on the block symbol, or adding a new one by pressing the **Ctrl** key while clicking anywhere on the symbol outline.

## Bottom Up Design

You may want to create libraries of commonly used blocks, such as power supplies or signal converters. You can do this by creating a Schematic Hierarchy block design and saving it in one of the nominated Schematic block folders. When creating a Schematic Design it is then possible to insert an instance of one of these existing blocks into the design.

## Navigating A Hierarchical Design

To navigate the hierarchy of a design, use the **Push** and **Pop** commands. To **Push** into a block instance, select any part of the symbol and select the **Push into block** command from the context menu. To **Pop** out of a block instance, to the symbol, select the **Pop out of block** option from the context menu. You do not need to select anything it is sufficient that the active page is part of a block instance.

## Connecting Through Hierarchy

Instancing hierarchical blocks gives you the internal connectivity of the block, but you will want to connect these nets up though the hierarchy. There are two ways to do this.

If the net in a block is defined as not **local** (global), then it may be used anywhere in the entire design. So the same global net name can be used inside and outside a block instance, and the attached items will be connected together. This is fine, but you will want to define connectivity inside a block instance which cannot be connected to outside the block instance. This is achieved by using a **local** name. Net names are defined in the Technology file.

To connect a local net to items outside the block instance, you will need to use a Block Port inside the block. This will match with a pin on the block instance symbol. The mapping between the two is defined in the **block Properties**.

## Multiple Instances

It is possible to add more than one instance of the same block design into a schematic. Once they have been instanced, it is possible to edit them independently so that they diverge from the original definition. More likely, you would want to keep the definitions in step, either with the original block design, or with each other. This can be achieved, by Reload Block updating from the original block design, or by applying changes from one instance, to all the others.

A block instance has a Locked instance property which prevents the contents of the instance from being edited or changed in any way. This ensures that it remains a copy of the original definition. A block instance also has a **Can Be Multiple Instanced** property. By enabling this, multiple instances of the block are locked together. Only one of these instances can be unlocked and edited. They can be brought back into line using **Update Multiple Instanced Block**

*Note that to use Multiple Instances, you must enable them in the **Design Settings**.*

## Updating Block Instances

Use the **Reload Block** command from the context menu, to reload the block definition from a block design. This will make all instances of the block become the same as that stored in the block design.

Use the **Update Multi-Instanced Block** command from the context menu to cause all multiply instanced copies of the block to become the same as the unlocked instance (only one unlocked instance is allowed).

You can also update the block instance symbol. For a symbol stored in the library, use Reload From Library. For a generated symbol, use Regenerate Block Symbol.

## Starting with Top Down design

Starting with Top Down design means you will start with a new block and after pushing down into it, will add the circuitry.

The process for creating a new block from the top down is as follows (and this assumes a new block rather than an existing block will be used):

1. Select the relevant **Technology file** or edit the **Design Settings**.
2. Use **Insert Block Instance** to add a block
3. Select **Auto-Generate** block. If the number of ports (input and output pins) is known then these can be set. If not set it to zero (0).
4. Add the block to the design.
5. Double click on the block to **Push** down into it, or right click and select **Push Into Block** from the menu.
6. Create the circuit or copy and paste it from another design.
7. Insert the required number of block ports using **Insert Block Port**.
8. Pop back up to the upper level using the **Pop** command, and add Block Ports to the block.

The block now has a circuit at its lower level, each of the ports at the lower level match the number of ports at the upper level. This is the basic hierarchical design.

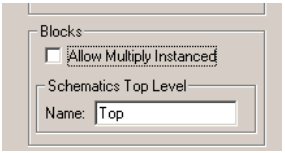
### Pre-hierarchical design checks

If you are using an existing design, then there are some settings which you must set in order to use hierarchy successfully.

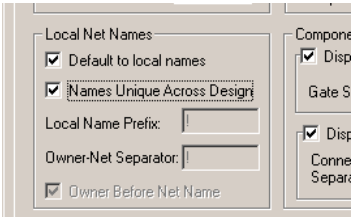
Firstly, you should select the right technology file to use. If using an existing design then set the switches in the **Design Settings** dialog as described below. If creating a new design with hierarchy and using one of the supplied **Technology files** then the default ones, **Default.stf** and **Default (White).stf** are both setup to allow hierarchical designs. If using true instanced hierarchy then you'll need to either use the **Multi Instanced.stf** Technology file which is already set up or make the necessary changes to the **Design Settings** dialog.

Secondly, the Design Settings should be set so that the program handles the Blocks and Net Names correctly.

From within the **Design Settings** dialog under **General** you should set the **Allow Multiple Instance** only if using multiple instanced hierarchy. If using this for the first time or if unsure, leave it unchecked.




On the **Design Settings** dialog under **Naming** you should set the way in which Local Net Names are used. Set **Default to Local Names** and **Name Unique Across Design** to on (checked). This will ensure that net names used within each hierarchical block will be local (and not Global) and that net names across the design will be unique.




In summary, the Design Settings should be like this:

For Hierarchy but <u>not</u> multiple-Instanced:

| | |
|---|---|
| Allow Multiple Instance | Off |
| Default to Local Names | On |
| Name Unique Across Design | On |

For Hierarchy but using multiple-Instanced:

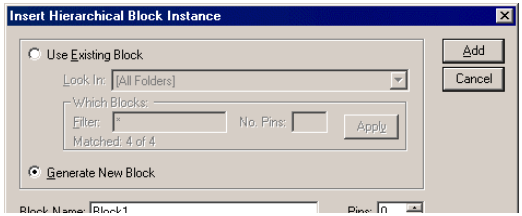
| | |
|---|---|
| Allow Multiple Instance | **On** |
| Default to Local Names | On |
| Name Unique Across Design | **Off** |

## Using Blocks

Blocks are used to represent a lower level set of circuitry. The block contains two main elements, the block shape or Symbol, and the block contents. The block contents can be represented at the upper level by any block Symbol, either auto-generated or from a saved symbol.

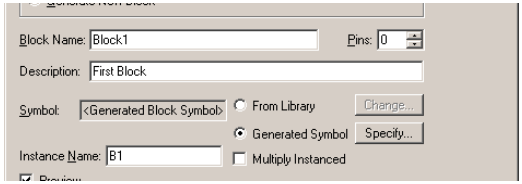
The **Insert Block** dialog allows these combinations to be catered for. Initially its use will be on the auto-generation of Symbols on the fly. Later on, it will be used for specifying **Existing Blocks** once saved.

The dialog is split into three sections. The top section is used for selecting whether a block is taken from an existing saved block (**Use Existing Block**) or a new one will be generated (**Generate New Block**).



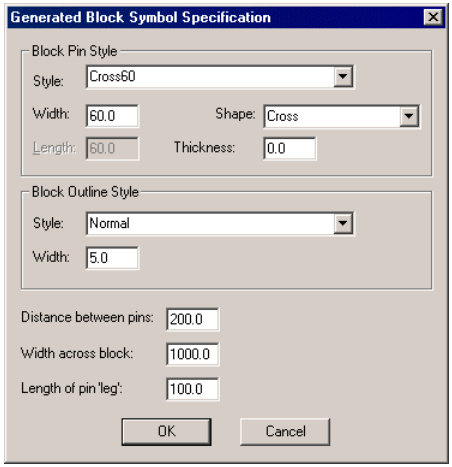

The middle section is used for selecting the **Block Symbol** or for specifying the saved block (**Block Name**).

When generating blocks, you can define the number of **Pins** that the block will have. If the number isn't known then it can be specified with 0 (zero) as the number of Pins. You can add Block pins at a later stage but by specifying the number up-front it will generate a block with the pins already positioned.

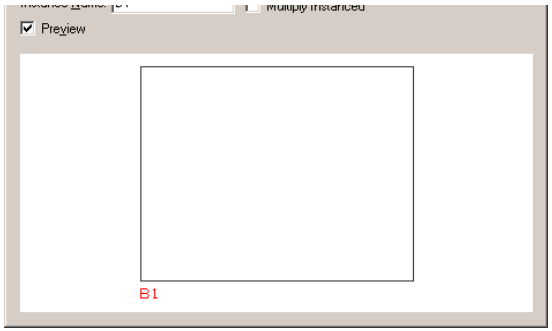



To add the pins at a later stage you must right click on the block and use the **Regenerate Block Symbol** option from the menu. Because the block has been automatically generated, another automatically generated block Symbol containing the correct number of block pins will be added. If the Symbol has been edited to be a specific shape before regeneration, then you should save this to the block Symbol library first.

The **Specify** button is used to specify the parameters from which the generated Symbols are created. This is used for generation and regeneration of Symbols.




The bottom section of the dialog is the **Preview** area. This allows you to preview the Symbol before it is added to the design.




## Push/Pop to and from blocks

Once the block has been added to the design, you can move down into the block by using the '**Push**' command. This is available on the right mouse menu once the block is selected, or by using the <Pgdn> key on the keyboard.

At the lower level, you '**Pop**' back up into the hierarchical block using the option from the right mouse menu or by using the <PgUp> key on the keyboard.

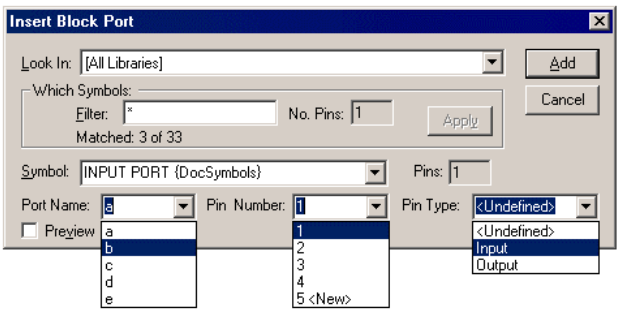
If you have the **Workbook** mode switched on, the blocks can be navigated using the tabs at the bottom of the screen. You can also use the **Design Browser** Window.

Once you've pushed into the lower level you can now add your circuit and Block Ports.

## Inserting Block Ports

At the lower level you add Block Ports to the design (using **Insert Block Port**) to create interface links back to the upper level. There will always be a direct 1:1 relationship between the lower **Block Ports** and upper level **Pins** of a hierarchical block.

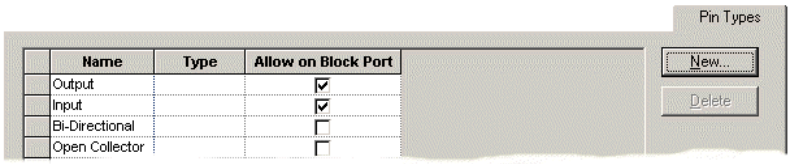
Block Ports are added from the **Documentation Symbol Library** and are defined as Block Ports during creation. Use the **Insert Block Port** option to add Block Ports to the design.



The Insert Block Port dialog allows you to specify the Libraries to be searched in and the Name of the Symbol.

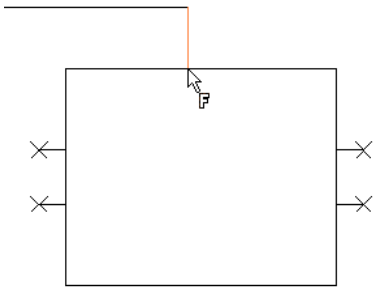
When adding Ports, if the number of pins was typed in during **Generate Block**, then the Port Name and Pin Number will match this number, e.g. if 4 pins were requested then **a**, **b**, **c**, and **d** will be available for **Port Names** and the **Pin Numbers** will show **1** to **4** and **5 <new>** as the next new pin number. **Port Names** are automatically created and incremented. You can type a name into the Port Name box for use. The next Port added will then be incremented using this name plus one, e.g. the name Port1 is incremented to Port2 etc.

You can also specify the **Pin Type** for the Block Port. By default this list is artificially restricted to **Input**, **Output** and **<Undefined>.** If you wish to use more Pin Types then they must be enabled through the **Technology** dialog under **Pin Types**. Checking any of the pin types in the **Allow on Block Port** column will enable them to be listed.

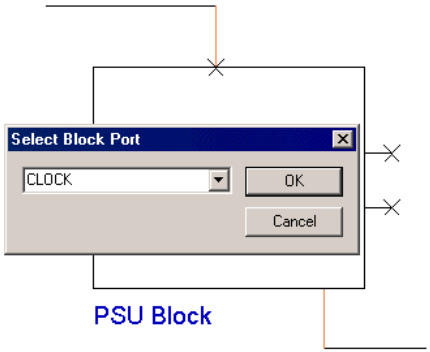



## Finish Markers

On the upper level, you can move a connection over a block, as you do this the '**F**' marker is displayed to indicate that you can automatically connect to the block using a block port.




If the block has no defined ports on the lower level then a new block pin is added without prompting. If the lower level has spare block ports defined, you will be prompted with the **Select Block Port** dialog. Use this to select or type the name of the block pin. When **OK** is clicked, the connection will be attached to the block with a block pin.



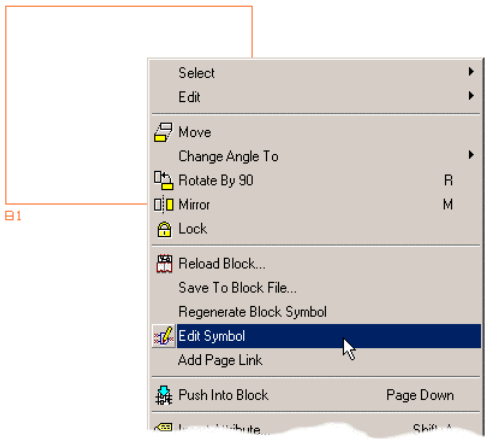

## Editing a block shape

An automatically generated block symbol shape can be edited. There are some points to note if doing this:

If the port has no block ports and you use the **Regenerate Block Symbol** option from the right mouse menu after adding block ports to the lower level, the shape will be replaced with another automatically generated block symbol.

If you wish to retain the modified shape, you may save the symbol to the Doc Symbol library.

After selecting the block, there are various options available for use from the right mouse menu.

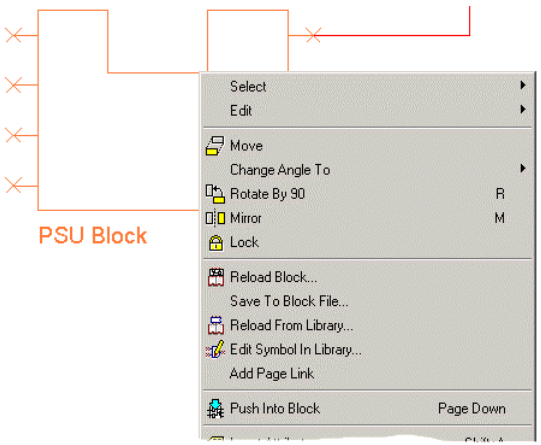



**Save To Block File** will allow the block (and its lower level contents) to be saved to a block file, to be used at a later date.

**Regenerate Block Symbol** will automatically generate a new symbol based on the current number of pins or block ports. If the **Edit Symbol** option has been used and the shape not saved, then this will be lost and overwritten with an automatically generated one.

**Edit Symbol** allows you to edit the current symbol. If you use this you can also save this symbol to the Doc Symbol library using **Save To Library** to be used and modified later on. If you save the Symbol (using **Save**) from this mode, the block will automatically load this Symbol by name from the library.

Once saved to library the right mouse menu on the block then reflects the Symbol being used rather than the automatically generated block shape.




The options now reflect that the block is a Symbol and is split into the block file and symbol:

**Reload Block** will reload the block with the one saved in a block file. If a block file has not been saved for this block then it will not allow you to reload.

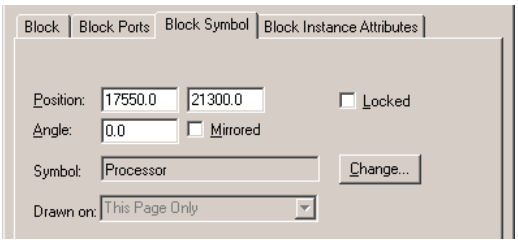
**Save To Block File** allows you to save this block to a block file.

**Reload From Library** allows you to reload the Symbol from the library.

**Edit Symbol In Library**, the program knows that the symbol has been saved to library and as such allows you to edit it.

## Saving Blocks

Selected blocks are saved within the design by using the **Save To Block File** from the right mouse menu. If the block was created using an automatically generated block, this shape will still represent the block symbol shape. It is advisable to save the block symbol to the library by using **Edit Symbol In Library** from the right mouse menu. Once in the Editor, save the symbol into the library using **Save To Library**. When you click Save to go back to the design, the block would have taken on this new name.



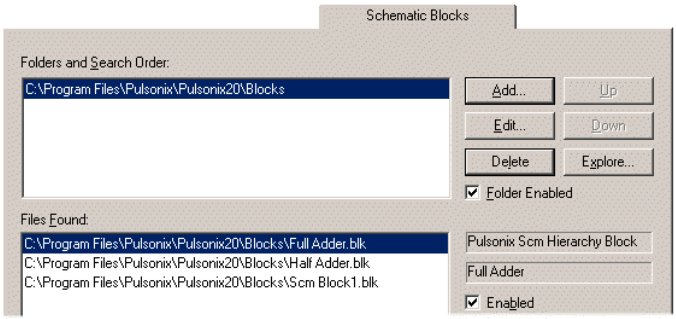

You can change the block name or change block symbols by using the right mouse menu option **Properties**. From the **Block Symbol** page, use the **Change** button to select a new **Symbol** name.

## Bottom Up Design

The process for 'Bottom Up' design is much the same as Top down but you use existing blocks that have been saved to the library to build the circuit. Common blocks can be used and reused to create circuits making the design process much easier.

### Folders and Schematic blocks

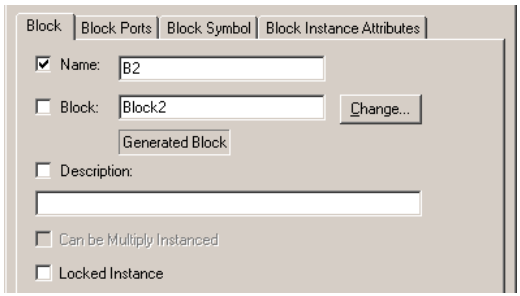
All previously saved blocks are located under the **Schematic Blocks** search path defined in the **Folders** option.




### Naming/Renaming blocks

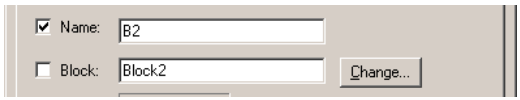
Blocks can be named, and once named, can be renamed.

Initially the block can be named by typing its name in the **Insert Block** dialog. Once in the design can be renamed using the **Properties** dialog from the right mouse menu and **Block**. **Name** is the block instance name, and **Block** is the block file name. You can also add a **Description** if you wish.

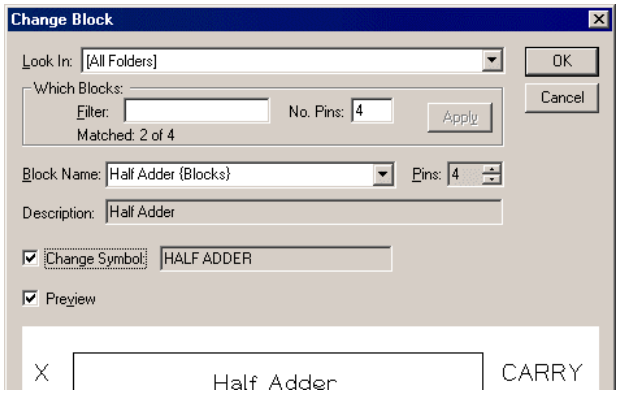



## Editing/Reloading blocks in designs

On the **Properties** of a block instance, the **Block** page has a **Change** button.

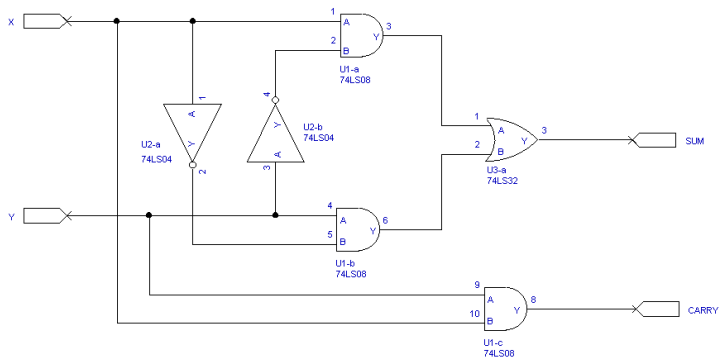



Use this to change the block use a different block file from disc. You have the choice whether to replace the symbol or not. The **Change Symbol** button means that the block can be changed with or without changing the Symbol which is saved with that block.

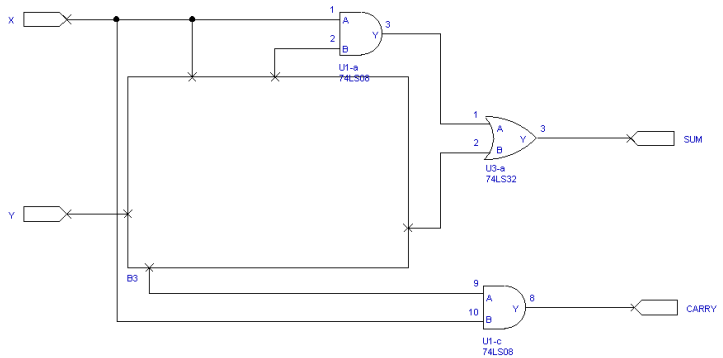



## Using Cut Circuit

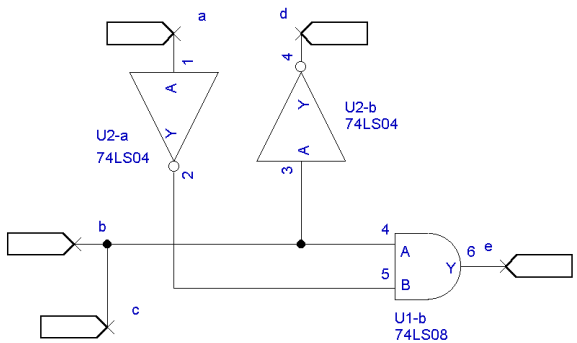
A section of a circuit can be ‘cut’ out to a block by using the **Cut To Block** option from the **Edit** menu. This option can also be assigned to a shortcut key or the button placed on a toolbar.




In the above example you will see that a section of the design will be cut out and pasted to a lower level block using **Cut To Block**. The selection of the design items will depend on the switch setting under **Options** and **Interaction**. On this dialog you have the option for **Frame Select**, being **Select If Completely Framed**. During **Cut To Block**, on the right mouse menu are the options for **Frame Select** or **Polygon Select**.



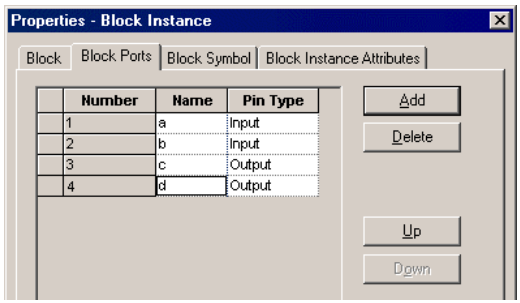

The resultant lower level now looks like this when the block is pushed into:




You can see that where connections existed, these have been terminated to the upper level using block ports. The block ports have been automatically named.

### Blocks and Pin Types

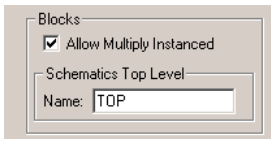
The **Block Ports** dialog allows you to change the **Name** of the pin, the pin **Number** sequence on the block and the **Pin Type** (used in the Electrical Rules Checking option).




The **Up** and **Down** buttons enable the order of the ports to be changed so that they physically change on the block symbol. These buttons are only enabled if the pin is not connected to a net.

## Instancing

If you want to use Multiple Instancing of blocks you must enable the **Allow Multiple Instanced** under **Block**s option in the **Design Settings** dialog first.




Once this has been selected, the block can be multiple instanced, without this selected, the **Properties** of a block will show the **Can Be Multiple Instanced** greyed out and you will not be able to instance it:


With the Multiple Instance enabled and the block allowed to be multiple instanced, once copied there are other additional options on the right mouse menu which are displayed.

The function **Unlock Multiply-Instanced Block** appears on the context menu when selecting a block instance, or with nothing selected when editing on a page in a locked block instance. This option allows you to **unlock** the block for editing. Within the Design Browser you can see the lock status of blocks (see below).

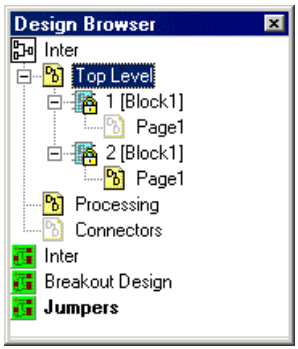
**Use Update Multiply-Instanced Block** to apply changes made to the design to all other instances of the block.

**Revert block option**. If you have unlocked and made changes to a block that is a multiple instance you can revert it back to look like the locked copies using this option.

If you unlock a block instance and switch off **Can Be Multiple Instanced** from **Properties**, you will not be able to apply the instancing back to the block.

### Design Browser

The **Design Browser** in the Schematic editor supports and shows **Hierarchical blocks**. It also shows the lock status of the block with a small padlock symbol. The use of hierarchical blocks is discussed previous under the section on **Hierarchy**.




## Pulsonix Spice Output Changes

You can now use the **Edit Spice Type** option on the **Simulation** menu to set up a block instance as a Spice subcircuit or model device. Prior to using this option, either select the block instance, or push down into the block. If the design is a block design then the option will be available with nothing selected in the top-level page.

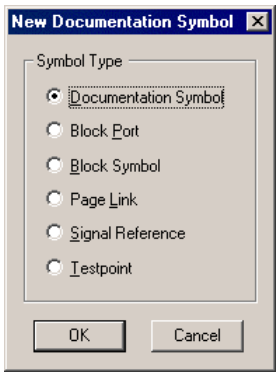
If the block instance is set up as a Spice device then it will be reported in the netlist as a model or subcircuit and all items within the block will not be sent to the netlist. If the block instance is part of a multiply instanced set, then unlock one of them, edit the spice type and use **Update Multi-Instanced Block** to update the rest.

The other item is to create the Spice subcircuit for the block contents. This was available before (from the **Simulation** menu and **Create SPICE Netlist File** sub-menu), but you needed to add special single pin **subcircuit pin** Components for the ports, block ports will now be used.

There is also a new option on the **Create SPICE Netlist File** sub-menu called **Block Instance to Subcircuit**. This is available when selecting a block instance within a design. Use this to create a subcircuit from the block instance's contents.

# New SCM Doc Symbol types

Schematic Documentation Symbols are used for several purposes. It is therefore necessary to specify the purpose of each symbol when it is created. In this way, the program can ensure that the symbol satisfies the requirements of its purpose (for example that it contains exactly one pin) and that you are only presented with symbols of the correct type when adding them.




There are a number of documentation symbol types:

**Documentation Symbol** which could be used in many ways. For example, it could contain a logo. You can add a documentation symbol to a design by using the Insert Documentation Symbol command.

**Block Port** which is used within a block to connect with the enclosing level. A block port symbol must have a single pin, and can contain any figures you wish. To add a block port, use the **Insert Block Port** command. Whilst adding a connection, use End Connection On from the context menu.

**Block Symbol** which is used to represent a block instance. When you create a block, you can nominate a block symbol to represent that block. You can also choose the symbol when inserting a block. A block symbol must have a pin for each Port in the block it is to represent.

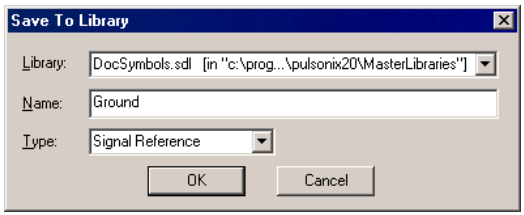
**Page Link** which is a symbol to represent a link to another page. When this symbol is used, it can be associated with another page in the schematic. You can then 'jump' to the specified page using the Jump to Page command, when the symbol is selected. Whilst adding a connection, use End Connection On from the context menu.

**Signal Reference** which is a symbol used to represent a break or link on the net. The symbol must contain a single pin. It would typically be used for a ground or power symbol. Whilst adding a connection, use End Connection On from the context menu.

**Testpoint** which is a symbol used to represent a testpoint. The symbol must contain a single pin. Testpoints can be added using the Insert Testpoint command. Whilst adding a connection, use End Connection On from the context menu.

### Changing Existing Doc Symbols

If you already have an existing **Documentation Symbol** which is to be used a **Signal Reference**, you must edit it in the **Doc Symbol Editor** and change its Type to a Signal Reference. To do this, read in the Symbol and click **Save to Library**. The dialog below will be shown. Change its **Type** to **Signal Reference** and click **OK**. Click **Yes** to overwrite the old one and it is now ready to use.




## New Insert Sig Ref Mode

As part of the changes for Doc Symbols, a new option has been introduced for **inserting Signal References** onto a design.

As stated above, this is used to represent a break or link on the net. The symbol must contain a single pin. It would typically be used for a ground or power symbol. Whilst adding a connection, use **End Connection On** from the context menu.

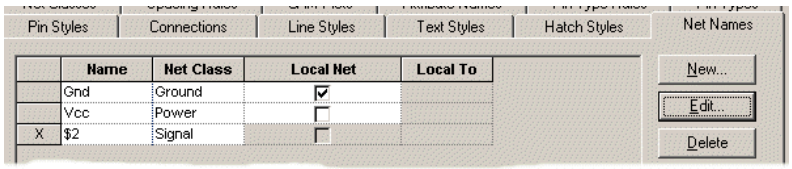
## New Insert Block Port & Block Symbol

These two new modes are covered in the previous section under *Hierarchy*.

# Minor Changes

# Changes to Net Names dialog

The Net Names dialog in the Technology file has been changed to allow hierarchical net names to be accommodated. The **Local To** box will display net names which are local to a specific hierarchical block.



# Autowelding SCM Connections

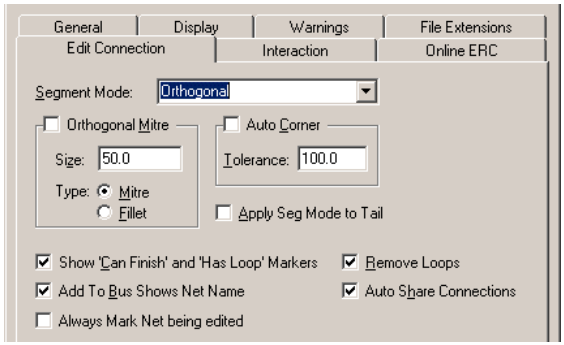
When finishing a **Move** of items with selected dangling connection ends or busses, an attempt will be made to weld these ends to pins beneath them. This welding is optional, by a check box in the **Options** dialog. The connection end must be exactly over the pin to weld it. If a connection is over a pin but fails to weld to it, an error message will be given. Choose to continue and drop the items without welding, or to cancel the finish.

You can also auto weld when moving the dangling end of a schematic Connection (Move Corner)

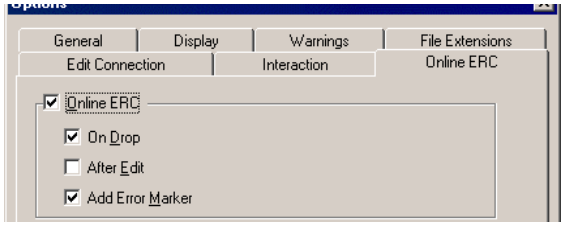
# Changes to Options

From the **Options** dialog and **Edit Connection** there is a new option **Always Mark Net being edited**.

When checked the net being edited or added will always be highlighted in the **Mark Net** colour.

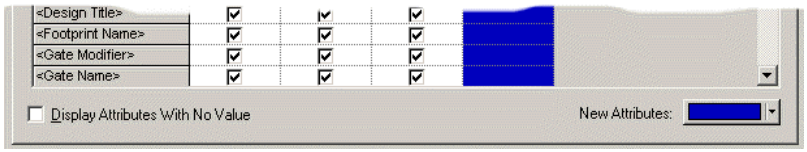



There is a new Online ERC tab for setting up the rules for use during Online Electrical Rules Checking, see previous section on *Electrical Rules Checking*.



# Colour for New Attributes

In **Colours** and the **Attributes** page, you can now specify the default colour for new attribute names.



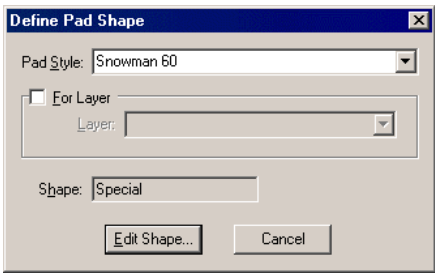
# Chapter 4. PCB Design

## User Defined Pad Shapes

User defined pads can be used where the standard Pulsonix pad shapes do not represent the actual pad shape required. A pad style editor is provided which is run from within the Pulsonix PCB design editor itself. This means you can create a special pad shape by drawing the actual pad shape required. This mode may be used to create new pad styles or edit existing ones.

The **Define Pad Shape** mode is entered using the option from the **Setup** menu.

A dialog is displayed which is used to define the name of the pad style, its layer and the name of the shape. The **For Layer** check box is the equivalent of clicking the **By Layer** button on the **Pad Styles** page of the **Technology** dialog.




You should think of this as entering a pad style creation mode where the default pad stack is created or where a layer specific shape is required for a pad style.

On entering the dialog you can type a new pad style name or select an existing one from the list. The list is taken from the existing pad styles from with the Technology dialog.

The **Edit Shape** button will enter you into the **Shape Editor**. This editor is similar to any other Pulsonix editor mode, such as the Footprint Editor. Obviously, it has limited functionality relative to creating pad shapes.

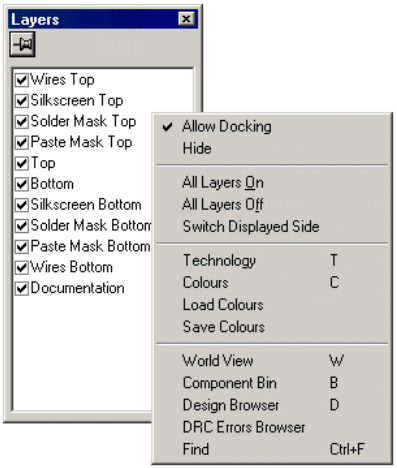
# Swap Layer Mode

The **Swap Layer** command is used for swapping track segments to the opposite layer. This is in addition to the **Change Track Layer** option. It can be used in latched or unlatched mode. In latched mode it can be used on multiple track segments as a manual tidying function. Where tracks segments are cleared out, any obsolete vias are also tidied. This option is available as a default button on the standard PCB toolbar.

This mode also works on **Components** for **Mirroring** and **Copper** etc.

# Layers Browser

A **Layers** browser has been introduced to allow fast display and switching of layers without having to use the Colours dialog. This modeless dialog is displayed as a configurable Dockable window containing a list of all the layers in the design. The checked layers are currently visible in the display window.




From the right mouse menu you can also select options from the context menu. These further enhance the selection of layers and layer combinations.

**Switch Displayed Side** is used to switch to the set of layers for the opposite side of the board, e.g. all Top or All Bottom. This is also available as a Command from the Run Command option.

Double clicking on one layer name will switch off all other layers.

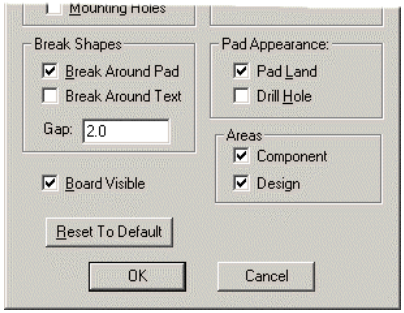
The list at the bottom of the context menu contains other dockable bars that you may wish to use. If you click on one of these, it will become visible and replace the Layers On/Off bar. This allows you to use a dockable area for just 2 bars ( for example) and keep replacing them with the next bar you wish to use.

# CAM Plot Improvements

Two new options have been added to improve existing CAM Plot outputs: **Break Silkscreen** and **Step & Repeat**.

## Break Silkscreen

**Break Silkscreen** is used to break the Silkscreen over holes and vias to stop the silkscreen paint going down the hole. This is available on the **Layer Class** dialog as layer type specific options – **Break Around Pad** and **Break Around Text**. The **Gap** between the item and silkscreen can also be defined using the current design units.



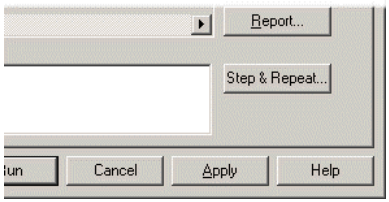

The picture below shows the effect of break silkscreen:

*Without Break Silkscreen pads*

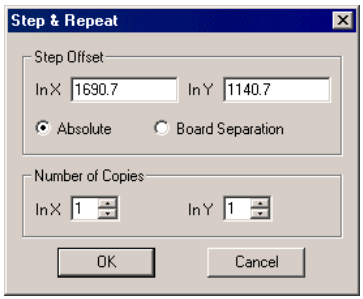
*With Break Silkscreen pads*

## Step & Repeat

The **Step & Repeat** option on the **CAM Plot** dialog allows you to plot multiple copies of the same plot with a specific offset or defined gap between them. It also allows an overlaid plot of different colours in one plot and allows true step and repeat for the same plot by using the **Add To Plot** button and offsetting it each time.




The **Step & Repeat** button opens up this dialog:




## Step Offset

The **Step Offset** defines the distance each copy is offset from the previous one. There are separate values for X and Y.

You can define the offset as **Absolute** which means that it is the total offset. Therefore an **Absolute** offset of 0 would put each copy exactly on top of the previous one (not recommended).

A **Separation** offset is the gap between copies. In PCB this is the separation between Board Outlines (if any), otherwise it is the separation between design areas. An offset of 0 would align the edges of each plot.

## Number of Copies

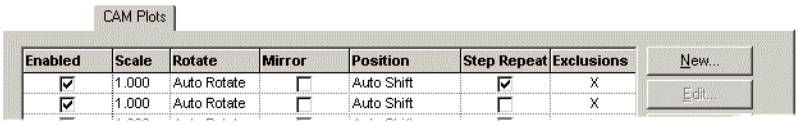
Defines the number of copies in the X & Y directions. These numbers include the original as the first copy (so they start from 1).

---

*Note: the offset and number of copies are defined in the X & Y directions before any rotation has been applied. The **Plot Preview** will give you an indication of how the final plot will appear.*

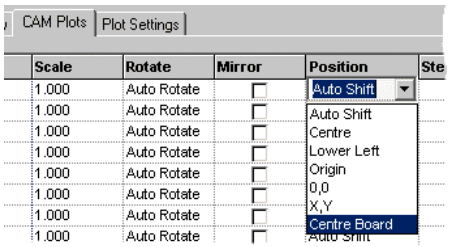
---

Once the Step & Repeat criteria have been set, you must enable each of the plot outputs to specify that it will be used. This is done using the **Step Repeat** check box available for each plot on the main **CAM Plots** dialog.



### Cam Plot Position by Board Centre

Positioning options for plots now includes ability to use the **Board Centre** for positioning as well as centre (of design extents).




# Jumpers and Wires

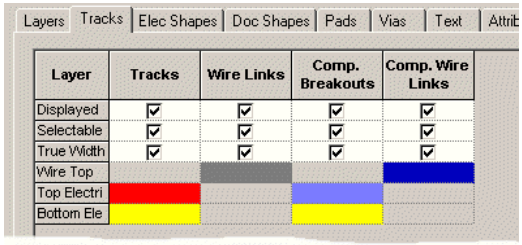
Jumper wires add the ability to make an invisible connective path between two isolated items but for both items to have the same net name. Typically, jumpers are used on single sided boards to jump over tracks where a track might have been used on a two-layer board. The use of wire jumpers enables low cost designs to be produced.

Jumper wires can also be used between die and bond pads. A connective path is required between the pads but the process for physically connecting them using a fine wire is either a manual or automatic one.

Jumpers and Wires have been implemented in three ways in Pulsonix;

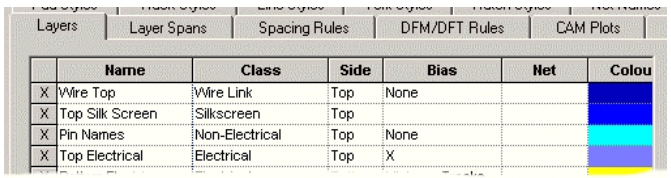
1. As **Wires** which can be inserted in any PCB design using the interactive feature (**Insert Wire**). They can be connected between any legal connective junction, such as pads or vias.
2. Still as **Wires** but contained within a **Footprint** for automatic insertion as a **Part**. These would be either a physical **Jumper Part** which would be 'bought', or used to indicate a specific size of wire jumper link to be inserted. This fixed size would be dependant on the Footprint pad 'pitch' used and can be varied using alternative Footprints for that Part.
3. As a **Bond Wire** as part of the Chip Packaging Toolkit. This allows die and bond pads to be connected together using a wire to indicate that they are on the same net.

The display of a wire can be set using its own colour in the **Colours** dialog.



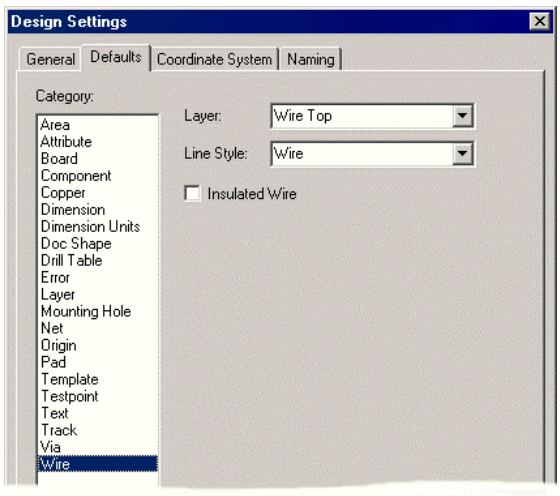

The dialog differentiates between wires used in a design (added interactively) and wires which are contained in a Footprint (Comp. Wire Links).

Wires have a special **Layer** either **Top** or **Bottom side**, by default none of the supplied Technology files have this special layer but when used will be automatically added. You can add and save the Wire layer to any Technology file if required.



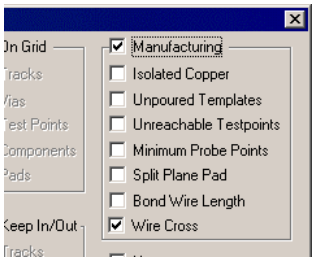

The style for the Wire is taken from the **Line Style** dialog. The style can be given a thickness and a dotted/dashed style.

When adding Wires the Default values are taken from the **Design Settings** dialog under the **Setup** menu. The **Layer** on which the Wire is added and the **Line Style** can be set. You can also specify whether a wire is **Insulated** or not (by selecting the check box).

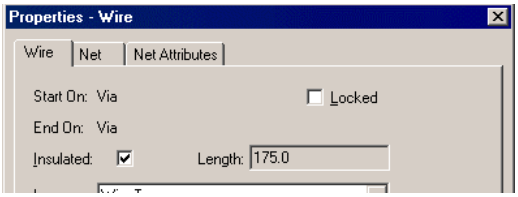



When running the **Design Rules Checking** option, there is a check for **crossing wires** under **Manufacturing**. If a wire is specified as **Insulated** and it crosses another wire then it will not flag an error. An un-insulated wire

which crosses with another wire will flag an error. Where one insulated wire crosses an un-insulated wire then no error is flagged.




The insulation status of a wire(s) can also be changed using the **Properties** dialog.




## Inserting Wires into the PCB design

The **Insert Wires** option is used in PCB design to create an interactive link between two items. It works in much the same way as Insert Connection.

A wire can be connected across any Pad of any type in the design (including vias, mounting holes etc.). Multiple wires can be added to any pad as well (the wire may not necessarily be assembled though the drill hole).

Wires can only exist on a special layer **Wire Top** or **Wire Bottom**. You can change the wire layer or style by selecting **Change Layer** or **Change Style** from the right mouse menu when inserting it.

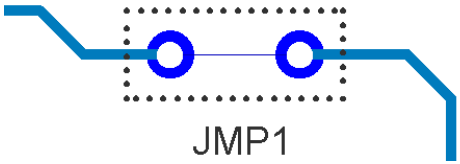
## Wire report

There is a wire report named **Wire.rff** which can be run from the Reports dialog. This reports the **Net Name** and the **X, Y** co-ordinates of the wire span.

## Jumper Parts

Wires can be inserted into Footprints to create a Jumper or to create common PCB pins (*using wires to create common pins is discussed below*).

A Footprint is created which includes Wires. For a **Jumper** this would mean that the two sides of the jumper are connected together using **Insert Wire**. The Footprint is saved as normal.



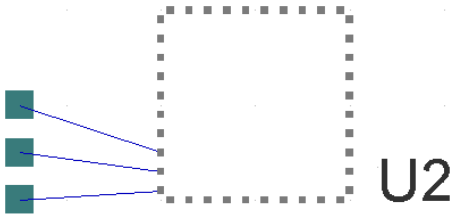

Nothing else is required in the library to be able to use the Jumper with this wire. The Part definition is created as normal with no special considerations.

For designs where the Jumper only appears on the PCB design and not in the Schematic, a PCB Only Component should be created. These are ignored during **Synchronise Design** and are not ripped out. Likewise, if you wish to show Jumper Parts in the Schematic design, you must create both the Schematic Symbol and PCB Footprint for the Part definition.

As with wires added interactively into a design, wire positions added into a Footprint can be output using the **Wires** report.

## Wires in the Chip Packaging Toolkit

Wires can be used to connect die pads with bond pads in the Footprint editor when the Chip Packaging Toolkit is licensed. Once placed on the design, the Component's bond pads move freely but the die pads remain fixed and retain their connectivity.



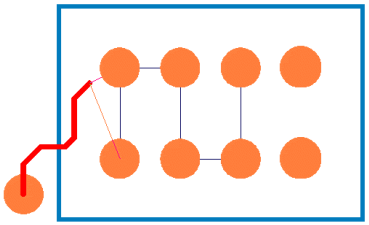

The picture above shows the partial connection of three bond pads to their respective die pads using wires (*this is discussed later in the section on the Chip Packaging Toolkit*).

### Using Wires for common connection points

Wires can also be used as an ‘internal’ connection in Footprints to create ‘common’ PCB pins.

To create this, you can use the **Insert Wire** option and make multiple wire connections to the pads you wish to make common.

When the component is added to the design, any connection which is made to any of the ‘common’ pins will automatically force an optimise to the nearest connective point. This is illustrated below, the track net connection is shown to the original ‘netlist’ point and the nearest point on the net.

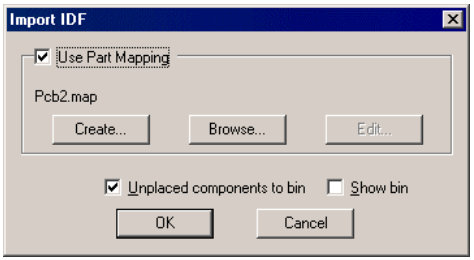


# Import IDF

IDF files in the V2.0 and 3.0 format can be imported into Pulsonix PCB. V4.0 files cannot be imported yet as this is a relatively new format. IDF import also includes import of component placement data.

IDF is a neutral data format that can be exported from 3D mechanical design systems. Pulsonix can import Board outlines and cutouts, Area, Components and Component Placement, Placement outlines, Placement keep in and out areas, all other area types (keep in and out) and Mounting Hole (V3.0 format only) data from this format. IDF data is generally output as a board file and a library file. Pulsonix only imports the board file.

To import the IDF format use the **Import Design Data** option from the **File** menu. The following dialog is displayed.



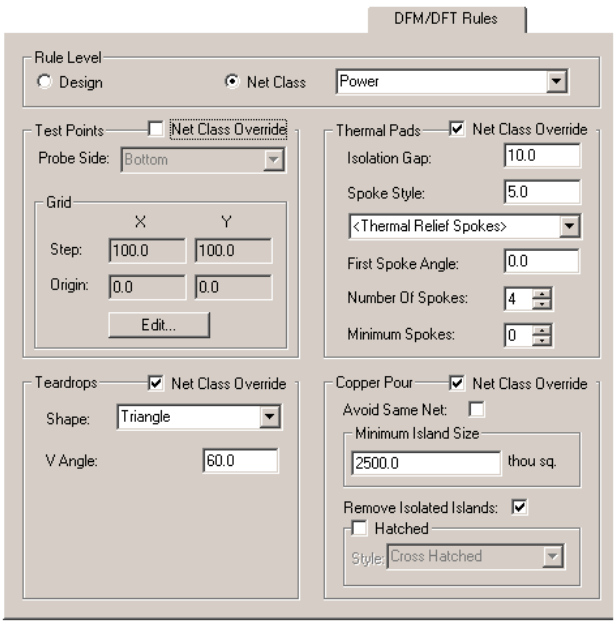

This allows you to choose a **Part Mapping** file or create a new one for mapping Component Name information to Parts which exist in your libraries.

Where Components are imported, Unplaced Components can be sent to the Component bin and the Bin shown.

# New Class Rules Spacings

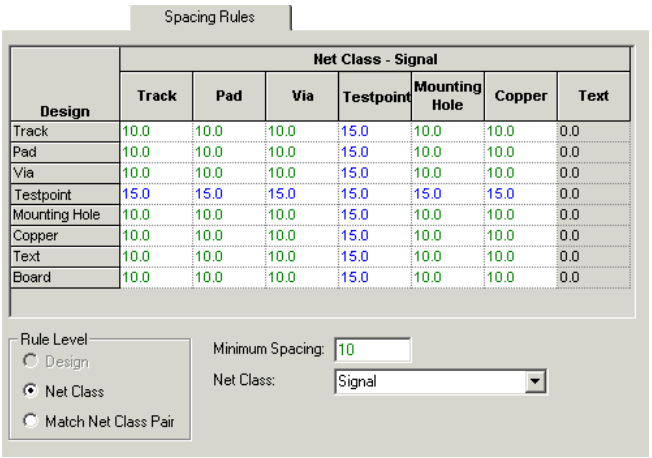
## Rule Level

The **Rule Level** specifies if the rules you are defining apply generally across the design, or if they apply only to nets using a specific Net Class. If you wish to define rules for a net class, select the net class box, then select the required net class from the list. Each of the four categories of rules can be overridden separately. You must check the **Net Class Override** box for the rule category you wish to override then change the values as required. Similarly, you can remove the override by unchecking the override box.




### Net Class Level Rules

The Net Class Level Rules define the minimum spacing between items on a net using the named Net Class (columns in the grid) and all other design items (rows in the grid). Select the required Net Class from the list.




*Note that these spacings are not reversible. For example, you can have a different spacing between Pads using the Net Class and Tracks on another net, and between Tracks using the Net Class and Pads on another net. When two explicitly assigned values could apply, the largest is used.*

To assign an explicit value, edit the appropriate cell in the grid. You can assign an explicit value less than the **Design Level** value if you wish.

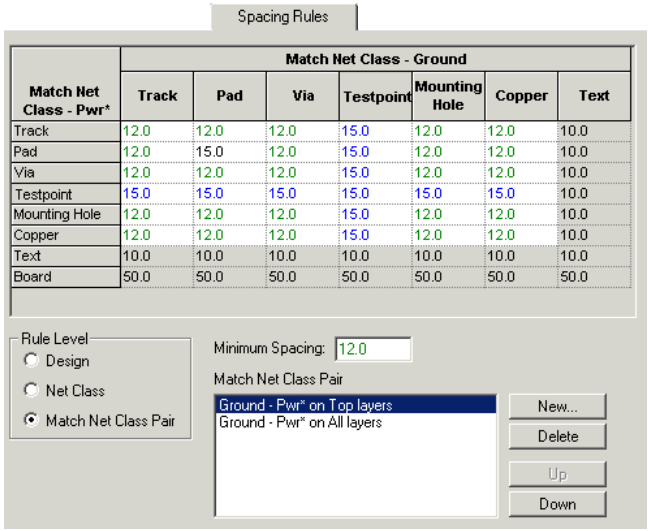
If no explicit value has been assigned to a spacing, the **Design Level** spacing is inherited. There is also a **Minimum Spacing**, which will be applied if the inherited value is smaller.

The grid is colour-coded to show what values apply. **Black** for inherited **Design Level**, **Green** for **Minimum Spacing**, and **Blue** for **an explicit value**. You can remove an explicit value by deleting it.

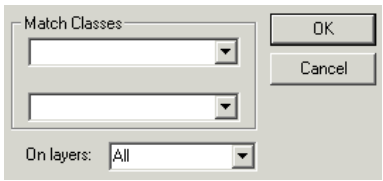
You can change a value back to the default by selecting the value in the grid and deleting it.

## Match Net Class Level Rules

These Rules are similar to the **Net Class Level Rules**, except they only apply between two Net Classes.




Firstly, you must define, which Net Classes the Rule applies to. To do this, press the **New** button. This gives a dialog into which you should select the two Net Classes, you can also use a wildcard string to match several Net Classes.



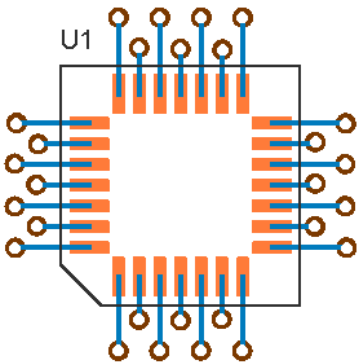

You must also specify if the rule applies to all electrical layers, or just the **Top**, **Bottom** or all **Inner** electrical layers.

For example, if you have net classes called *Ground*, *Power +5* & *Power +12*, you could define a rule pair between *Ground* and *Power**, so the rule would apply between *Ground* nets and all *Power* nets (but does not say anything about rules between *Power +5* & *Power +12*).

All the rule pairs you have defined are shown in the **Match Net Class Pair** list box. The rules are applied in the order they are shown in this box, so if more than one matches, only the first match will be used. You can use the **Up** and **Down** buttons to move the selected rule pair up and down the list. The **Delete** button deletes the selected rule pair.

# Track & Via Breakouts

Breakouts are used to create predefined or potential track patterns for use where the pads are restricted on a Footprint. Typically these patterns would be used where routing is very dense on a surface mounted device, and where 'human' intervention is the most sensible option. The breakout patterns are more suited to designs where automatic routing will take place. They often give the router more solutions to what may appear a 'fixed' option. For instance on a surface mounted QFP where all the pads are on one side of the design, by using breakouts and breakout vias, the pins then become accessible to all layers if the vias are defined as <Through-Hole>.




Breakouts can be used on SMD and through-hole pads.

In Pulsonix, breakouts are generated and stored in the Footprint definition. This means that the definition always knows about the breakout and can apply or reapply the original breakout pattern at any time.

## The process of using breakouts

The 'flow' for using breakouts is as follows:

- **Create the breakout pattern in the PCB Footprint and save this.**
- **In the PCB design the breakout guide will be displayed.**
- **Either edit a track/connection to the breakout which will be a legal finishing point and will be shown as node point on the net. The breakout automatically gets converted to a track, or;**
  **Select the breakout or Component and from the right mouse menu select** Hide Breakout **or** Convert Breakout To Track**.**

## Breakouts in the Footprint

The Footprint editor has the ability to create breakout patterns, with or without breakout vias. The method for creating breakouts is the same as if adding a track to the Footprint in the design.

The advantage of creating the breakout with the Footprint is that it only has to be done once. Whenever the Footprint is used it then automatically has these properties. Each Footprint instance for a Component can have the breakouts switched on or off, and they can automatically be converted to tracks.

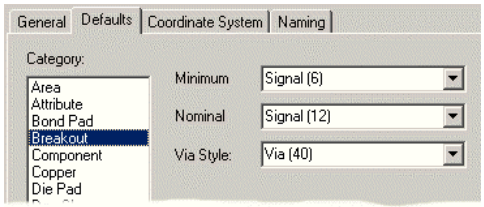
You have some specific design features for creating breakouts: **Insert Breakout**, and once the breakout has been added, **End On Breakout Via**.

In the **Technology** file, breakouts use the **Track** Width and **Via Pad Style** but are in fact breakouts when added to the Footprint.

The breakouts are created on the normal electrical layers but once the Footprint is added to the design these layers act as the 'default' layer for the track once converted.

On the **Options** dialog there is an **Edit Breakout** tab, this is only available in the Footprint editor and is replaced with the equivalent dialog **Edit Track** in the PCB design editor.

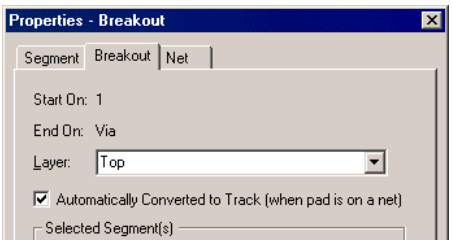
Default values for breakouts are taken from the **Design Settings** option, **Defaults** and **Breakouts**. The default values define the **Minimum** and **Nominal** breakout widths and the default **Via Style**.




**Layer defaults** for breakouts are taken from the **Track** defaults.

Once a Footprint contains a breakout it is saved in the normal way. If a breakout pattern needs to be changed, the Footprint can be edited and saved, then reloaded in the usual way. You cannot edit breakouts directly in the PCB design.

Using **Properties** on a breakout in a Footprint there is a switch you can use to stop it from being automatically converted to a track (**Automatically convert to Track (when pad is on a net)**). The only way to convert a breakout that has this flag cleared to a track is to select it directly in a PCB design and use the context menu **Convert Breakouts to Tracks** option. Selecting the whole component or net and using the same option will not convert it to a track unless its flag is set.




With nothing selected, you must use the context menu option **Convert Breakouts to Tracks**. This is the only way to convert breakouts with the switch mentioned above set on.

### ▶ To add breakouts to a Footprint

1. Start a new Footprint or edit an existing Footprint from the library.
2. Select the **Insert** menu and click on the **Insert Breakout** option, or select the Insert Breakout option from the PCB Footprint toolbar.
3. Draw in the breakout as if add a new track. Use the options from the right mouse menu to change the style and layer etc.
4. Use the options on the right mouse menu to finish the breakout. This can be with no via, with a via or with a Testpoint via.
5. Save the Footprint to the library and include in a Part definition as normal.

## Breakouts in the Design

Once a Footprint is used in a design, the breakout is a self-contained item, at this point it is not a track or via but a ‘guide’ to the final track pattern when connected.

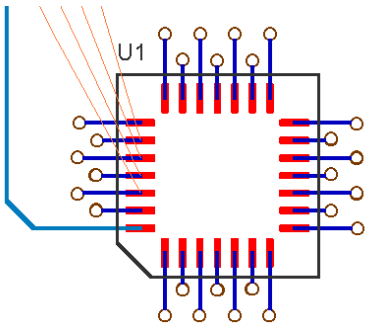
For breakouts to work the pads to which they are attached must be on a net. If a single (or multiple) breakout is required on an unconnected pad then the pad must have a net name. A single pin net name can be assigned in the Schematic design so that the Synchronise Designs option doesn’t rip out any single pin net in the PCB.

To differentiate breakouts from ‘normal’ vias or tracks they have their own colour slots in the **Colours** dialog. Under the **Vias** tab you have **Breakout** (vias) as a Layer Span, and under the **Track** tab, you have **Comp Breakouts**. By default in the supplied Technology files these items have their own colours so they are visible when tracks and vias are used in a design.

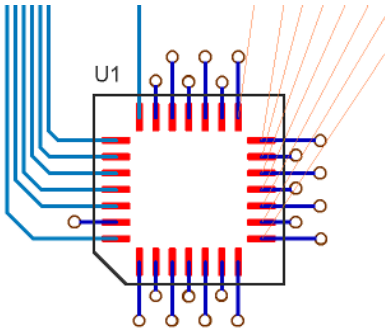
To use breakouts which exist on Footprints you have a number of options, all available on the right mouse menu for selected Footprint entities: on a selected Component(s), on individual pads on a Component which have breakouts and on the whole design (but the design must have breakouts on at

least one Component): **Hide Breakouts**, **Show Breakouts**, **Convert Breakouts To Tracks**. .

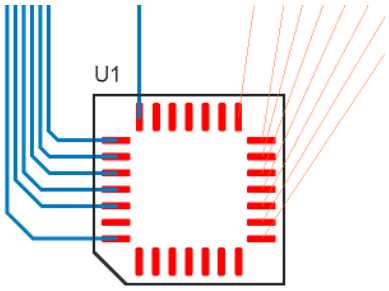
**Hide** and **Show Breakout** will allow the breakout to be hidden or shown on a Component. If the breakout has already been converted to a track it can still be shown even though it may be redundant.




The example above shows a track that has been connected to a breakout. The breakout has been converted to a track and the breakout via removed. Automatically the breakout is switched to 'hidden'. If you need to see the breakout pattern, for confirmation purposes of its original direction for example, you can then select the pad and from the right mouse menu select **Show Breakout**.




The example above shows one side of the IC routed to the pads without using the breakouts. One pad has no connection and still shows the breakout.



The example above shows the whole Component with the breakouts hidden. Connections can still be seen to the Component, as these must still be routed.

If converted to a track or if the breakout is left unrouted, you can select **Remove Redundant Breakout Tracks** to remove them.

If a track is completely unrouted thus also removing the breakout path, the breakout can be 'regenerated' by right clicking on the pad and selecting **Show Breakouts**.

While editing a connection that is attached to a pad with a visible breakout, if a breakout is at the end of the connection you have picked, an initial track will be created from the breakout.

If a breakout is at the other end of the connection, the connection will be re-drawn to end at the end of the breakout. Using **Complete As Track** option in this situation will use the breakout path for its completed track.

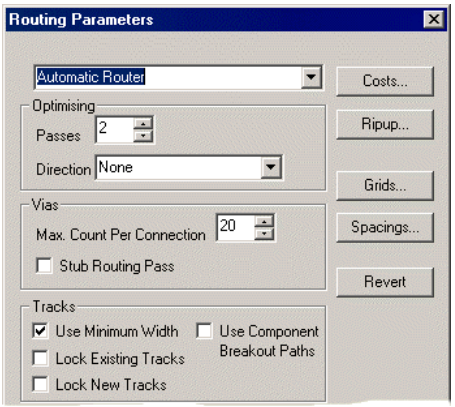
Also, double click in select mode on a visible breakout in a PCB design will add a new track that uses the breakout path from the selected point back to its pad. You will then be in **Insert Track** mode to complete the track.

## Plotting breakouts

You cannot output breakouts directly through the CAM Plot option, they must be converted to tracks or vias.

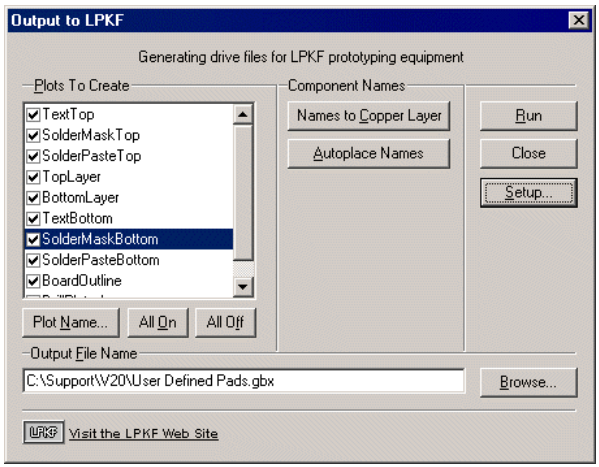
## Breakouts in the Autorouter

Autorouter now has a switch **Use Component Breakout Paths**. If checked, all breakouts marked as auto convertible (see above) on pads on the nets to be routed will be converted to locked tracks prior to routing.

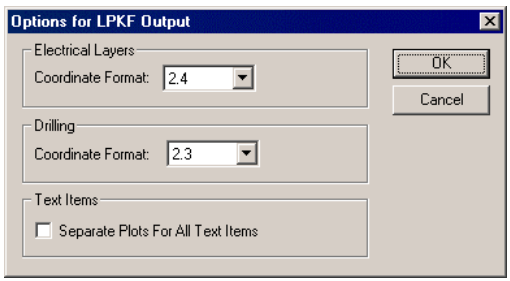


## LPKF Output Changes

Some minor changes to the LPKF Output dialog have been added for improved usability. The dialog now has a button for editing the output layer names (**Plot Names**), this also works by double clicking on the name in the list.




You can also output ‘profiled’ text. There is an option on the Setup page for creating **separate plots for all text items**.




Text items can be plotted in two different ways. To plot them as normal items on each plot along with other items from that layer, leave the checkbox clear. Depending on the text you have on your board, this can produce an 'untidy' result:


To counteract this, you can plot text items separately by setting the checkbox. This will result in two additional layers being created in the output file for each plot that contains text, and text items being omitted from the other layers.


Once processed using the appropriate steps in LPKF CircuitCAM, the resulting plot would then look something like this:


# PCB Specific Drawing Changes

## Draw in Layer Order Option

There is a new check box on the Options dialog under Display named **Draw in Layer Order**. Like the other options within the box, it only applies if **Fast Redraw** is also checked. When checked, ALL items will be drawn in the correct top-down layer order as defined in the **Technology** and **Layers** dialog. This includes layer specific items within Components such as Copper.

Unchecking the option will leave Fast Redraw as it used to be, without the full layer ordered drawing, although it does provide a limited layer ordered drawing of Tracks only based on the Default Track layers. This old method provides a more item-type based order of drawing, e.g. all copper, then all pads, then all tracks.

## Pick Top to Bottom

**Pick Top to Bottom** is removed as an option and is combined with **Reverse Layer Order** (see below), so Graphics and Picking now work together.

## Reverse View Option

The **Reverse View** command is available for a PCB design or Profile from the **View** menu. It can also be assigned as a shortcut key. The toolbar button latches to indicate a currently reversed design view. It is a toggle option that visually flips the entire PCB design allowing it to be viewed from the bottom side.

If **Fast Redraw** and **Drawing Layer Order** are being used, the layer order is also flipped so that layers appear in the correct bottom to top order.

Full interactive capability is maintained when the view is reversed and the Windows Print option will output the same reversed view seen on screen.

This is a view option only and is not preserved when the design is closed.

There is also a separate **Reverse Layer Order** command which is available to be assigned to a shortcut key or to a toolbar. This toggle command will, if Fast Redraw and Drawing Layer Order are being used, reverse the repaint layer order from top to bottom to bottom to top without mirroring the design as well.

*Note that the separate option to display the current routing layer **on top** whilst adding or editing tracks remains unchanged.*

# Minor Changes

## Layer Change For Doc Symbols

You can now change the layer of a **Documentation Symbol** using **Properties** provided all the Doc Symbol's items are on the same layer to start with.

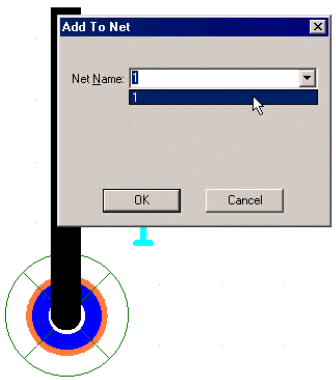
## Associate pads to copper

You can now associate pads to copper (or other pads) in the Footprint editor so that DRC errors don't occur; previously, this was not possible.

The pad and copper association is done in the Footprint editor using the **Add To Net** dialog on both the pad and the copper.

The pad must be selected first and using the right mouse menu option, **Add To Net**. The pad number will be displayed by default, this is the most logical choice for the name but is user defined, click **OK** to add it.

Now select the copper to be associated and right click. Again, select **Add To Net** from the menu. From the drop down list select the pad number or the relevant name to make the association. If using breakouts on any other pads these will also have net names or pad numbers so you must choose carefully.

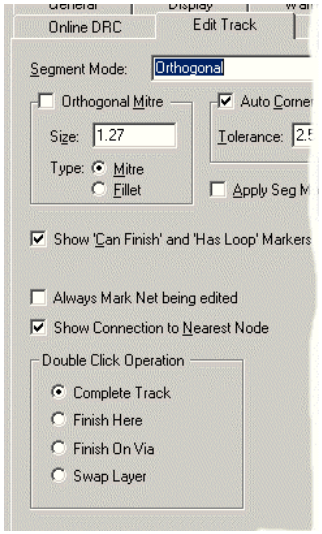


## Finish On Via Command

There is a new **Finish On Via** command. This will drop a via at the track end and finish without selecting any other option. It is used for tracks and breakouts to put down stub routes. It can also be put on double click using the **Options** dialog and **Edit Track** page.

## Edit Track in Options

From the **Options** dialog and **Edit Track** page, you can now assign the **Double Click Operation**. There are now **Swap Layer** and **Finish On Via** as a choice.




**Always Mark Net being edited** is used when you wish to highlight the net being **inserted** or **edited** using the **Mark Net** colour.

*Note: the Finish On Via mode has also been added as a right mouse option for routing stub routes.*

## Highlight Track Using Stripe

In the Options dialog and Display there is a new option **Highlight Track Using Stripe**. This draws the track using the **highlight** colour as centreline stripe or solid colour.

## Adding PCB Track changes

In **Select** mode and in **Insert Track** mode, starting the track on a connection which is attached to a via will now start a track from the via on the opposite side of the board.

In **Insert track** mode starting a track on a via will no longer move it. It will add a track to the via on the opposite layer.

Double click on a via in **select** mode will still go into **Insert Track** with the via moving on the cursor.

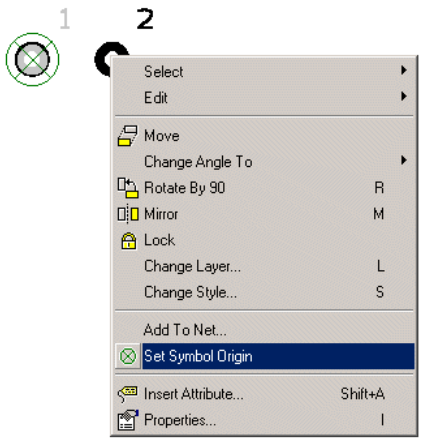
## Unplated Pads, Vias, etc and Optimise

**Optimise** now assumes that unplated pads do not connect items on different layers. Therefore a connection would be added between tracks on different layers connected to an unplated pad (unless they are connected by some other route).

*Note that the Properties dialog does not currently show the plated state of a pad style.*

## Set Symbol Origin Command

There is a new command within the Symbol editors for Setting the Symbol Origin. To use this, select the item chosen to be the origin and right click. From the menu select **Set Symbol Origin**.




## Drill Table

You can now specify all the length unit parameters (precision etc), including an optional unit string, for drill tables. The drill size tolerance and step can also be specified.

### Length Units

The units used for the drill table are defined using this set of controls. The unit **Name** that is displayed on any generated drill table is shown. This defaults to the same as the unit name, but can be any string. An empty string means that the unit name is not displayed. The precision gives the number of decimal places shown for each drill size. Note that this does not change how the drills are grouped together. You can also control if trailing zeros are displayed, and the rounding indicator.

### Drill Sizes

Similar drill sizes (as specified by the **Pad Style**), may be grouped together by specifying a **Tolerance** and **Step**. The **Tolerance** gives a value by which each size is rounded (up or down). The **Step** gives a value by which each size is rounded up. The Tolerance is applied first. A value of 0 means no rounding is performed. If you change the **Tolerance** or **Step** values, you should press the **Reset** button to apply them to any existing drills.

For example, the following table shows how the drill sizes 25.4, 26.0, 26.4, 26.9, would be distributed for the given values.

| **Tolerance** | **Step** | **Resulting drill size** | |
|---|---|---|---|
| | | **(Original sizes)** | |
| 1.0 | 0.0 | 26.0 | 27.0 |
| | | (25.4, 26.0, 26.4) | (26.9) |
| 0.0 | 2.0 | 26.0 | 28.0 |
| | | (25,4, 26.0) | (26.4, 26.9) |
| 1.0 | 2.0 | 26.0 | 28.0 |
| | | (25.4, 26.0, 26.4) | (26.9) |
| 1.0 | 5.0 | 25.0 | 30.0 |
| | | (21.0, 24.4, 25.2) | (26.9) |

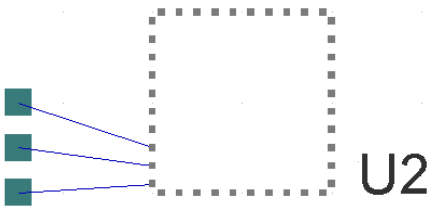
## Drill Size Table Report

There is now a new PCB report called **Drill Size Table Report**. This can be run from the **Reports** option, or from the **CAM/Plot Drill Sizes** dialog where a **Report** button can now be found.

# Chapter 5. Chip Packaging Toolkit

## Chip Packaging Toolkit (Cost Option)

The chip Packaging Toolkit is a new cost option for Pulsonix. It enables the support for single and multiple Chip-On-Board (COB) technology. Chip-On-Board is the use of bare die directly onto the PCB substrate rather than using the traditional packaging of the device.




The Chip Packaging Toolkit is licensed using a license file entry, it enables you to:

- Add bond pads using **Insert Bond Pad** and Die pads using **Insert Die Pad.**
- For the Die and Bond pads to be connected together using a Wire so that they have the same net name.
- Automatically **rotate** the bond pads during positioning around the shape or during interactive moving of the bond pads.
- Use the **Properties** dialog to convert from a ‘normal’ pad to Bond or Die Pads.
- To place bond pads around a user defined shape for specific positioning.
- To move floating bond pads of Footprints in the PCB design editor (normal pads are in fixed positions, die pads are also fixed) independently of the die Component body.
- Use the Library Generator toolkit (where available [an additional cost option]) to write commands for importing Bond pads, Dies pads and connecting Wires.

Where no license is available for this toolkit but the design has Chip-On-Board design items then they can be seen but not manipulated.

Parts created which use PCB Footprints specific to the Chip Packaging technology do not affect the Schematic design editor. These Parts will use normal Schematic Symbols but the translation of the design to the PCB design editor will then utilise the Chip Packaging features.

## What makes the Chip Packaging Toolkit?

In simple terms, the Chip Packaging Toolkit is constructed of three design items, bond pads and die pads (specific to this toolkit) and Wires which used generally across the Pulsonix PCB design editor.

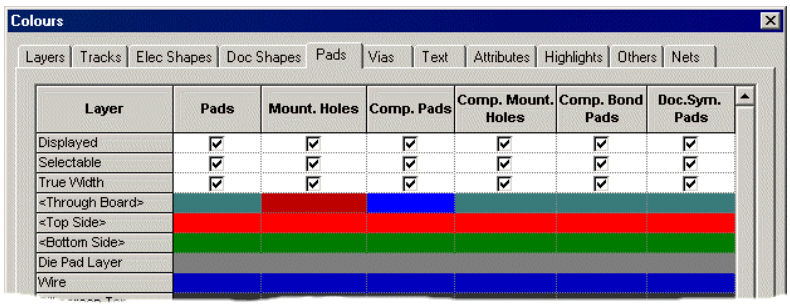
**Bond pads** are specially defined pads. They use normal pad styles and electrical layers but because of the licensing and because they are bond pads, can be moved independently of other pads in the same Component in the PCB design. Bond pads will share the same pad number as the die pad if they are connected using a wire.

**Die pads** are also specially defined pads. They use normal pads as well (albeit very small ones usually). They appear on a **Wire Top** or **Wire Bottom** layer which has a special **Layer Class** of **Wire**. This means that a Wire layer can sit 'over' the normal electrical layers and be connected to as a pseudo electrical layer. You need to connect to it but not plot its contents with the other electical items. Die pads can only be defined on one of the two outer layers and not available on inner layers. The die pads sit on the same layer type as the Wires. Die pads are not normally plotted.

**Wires** connect the die pads to the bond pads. Wires are not the same as connections in the design. Wires appear on the wire layer along with the die pads. When a wire connects the die pad to the bond pad, they take the same net name. Wires are not normally plotted.

## Specific Chip Packaging Toolkit items

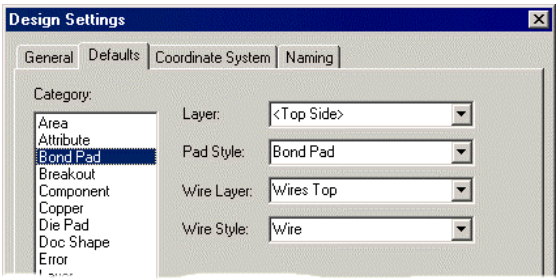
The **Colours** dialog shows items specific for **Component Bond Pads** and **Wires**. Wires used in the Chip Packaging Toolkit are displayed on the **Tracks** page as **Comp Wire Links**.



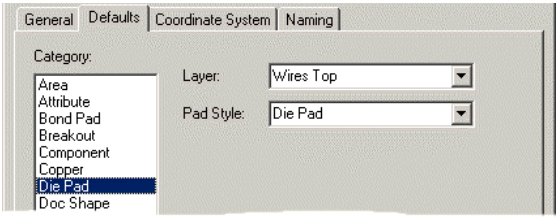
Footprint Editor

## Design Settings Defaults for Chip Packaging Features

The Chip Packaging **Default** settings are defined in the **Design Settings** dialog. There are three items, **Bond Pads**, **Die Pads** and **Wires**. Except for Wires, these don't appear in the PCB design editor.




**Bond Pad** defines the default **Layer** that the bond pad will be added to, its **Pad Style**, **Wire Layer** and **Wire Style**. The **Wire Layer** and **Wire Style** are used when using **Insert Bond Pad**. This option will request that the die pad is selected from which to connect to using the Wire (which has a Wire Layer and Style).




**Die Pads** defines the **Layer** for the die pad and its **Pad Style**.

## Creating the footprint

There are two main features for the creation of the die 'footprint' and the bond pad 'footprint'. You have the **Insert Die Pad** and **Insert Bond Pad** options. The two options provide you with a mechanism for adding pads directly to the footprint with pre-assigned characteristics (taken from the Design Settings options). Die pads have same pad number as bond pads and are connected with a Wire.

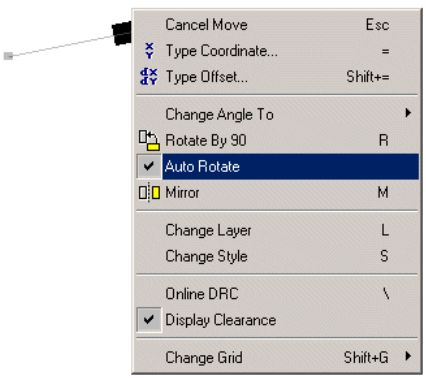
**Insert Die Pad** is used to add die pads to a Wire layer. These pads sit 'above' or 'below' the top or bottom electrical layers but cannot be connected to, only with a Wire for connectively with the bond pads.

Once in the **Insert Die Pad** mode, you place the first pad (the die pad) which is automatically numbered, you are then given another pad (the bond pad) to place. This is shown automatically connected with a Wire. If you decide to

only place die pads without their respective bond pads then the Escape (ESC) key can be clicked. With the first ESC key click you will leave the Insert Bond Pad mode, then be given the next die pad in the sequence. If you wish to exit completely you must click the ESC key again.

You can 'convert' normal pads to die pad using the **Properties** dialog. To do this you must already have a suitable **Wire layer** defined (and Wire Layer Class), and a suitable **pad style**, although the pad is usually only very small and nothing else characteristic which makes it a die pad style. When the pad is changed from <Top>, <Bottom> or <Through-hole> to Wire Top (Wire or Bottom), it is then changed to be a Die Pad. This is really only used if an existing footprint is to be used as a die.

While placing the bond pads you can elect to automatically rotate rectangular pads to be in-line with the Wire. This mode is switched on and off from the right mouse menu available during Move.




The two pictures below illustrate the effects of auto-rotation on bond pads:

With rotation applied

Without rotation applied

**Insert Bond Pad** can be used once die pads have already been added to the footprint. On selection of this mode, you are requested to select the die pad to 'pair' with the bond pad. Once selected, a Wire is added to the die pad and you can then place the bond pad.

Again, the bond pads can be auto-rotated using the option on the right mouse menu.

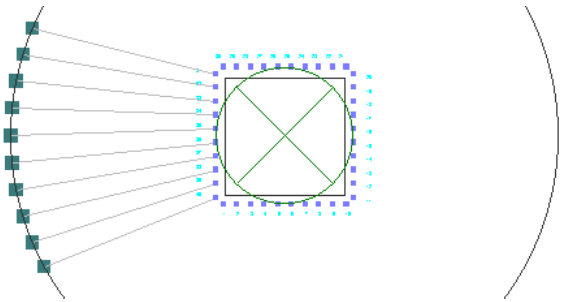
Bond pads cannot be 'converted' from any other type of pad (normal or die pads), they must be added as bond pads.

The Wire between the die and bond pads (wires are different to normal connections) can be defined **Insulated** or un-insulated. This property is used for the **Cross** check when using the **Design Rules Checking** option. If wires

are insulated and cross, no error is reported, if they are not insulated and cross, then an error is flagged.

### Place bond pads around shapes

To aid the placement of bond pads you can use the **Place Bond Pads** option. By default this is available on the PCB Footprint toolbar but is available as a command to be added as a shortcut key.



This option is used by selecting the bond pads to place, then selecting the shape on which to place them around. The pads automatically snap to the shape but may need the placement grid to be corrected to fall exactly on the shape.

*Note: Breakouts can also be used in footprints which have die and bond pads.*

## Chip Packaging Features in PCB Design Editor

Once the footprint has been saved into the PCB Footprint library (using **Save To Library**), a Part is created in the normal way. Once die pads and bond pads have been used in a footprint there is nothing more special about them.

The Parts containing the footprints with die pads and bond pads are added to the design using **Insert Component**.

Once in the design, and provided you have the chip Packaging Toolkit license option, the Components act differently. Bond pads have the ability to move or 'float' independently of the die pads or normal pads of the Component (normal pads are in fixed positions, die pads are also fixed).

When attaching connectivity in the design, you cannot connect to a die pad directly, only the bond pad. This is the same rule for both connections and tracks. However, the die and bond pads do have the same pad number.

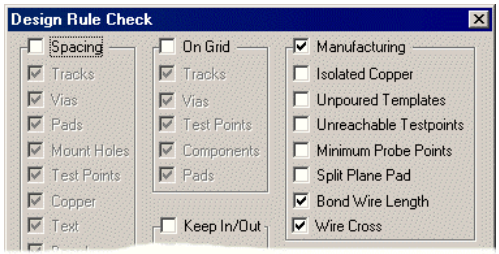
The option, **Reset Bond Pads** is available on the right hand mouse menu, this works on the whole Component or individually selected bond pads. This will allow the position of any selected bond pads to be restored to its original position as defined in the footprint.

The **Place Bond Pads** option (to place bond pads around a shape) also works in the PCB Design editor but is only available as a command by default.

As with the footprint editor, finger pad shapes are supported with automatic rotation offsetting on radial shapes. Again this is available on the right mouse menu while moving the bond pads.

## DRC in the Chip Packaging Toolkit

The Pulsonix Chip Packaging Toolkit is provided with a set of design rules, which can be checked using the **on-line DRC** option and **Design Rules Checking** option.

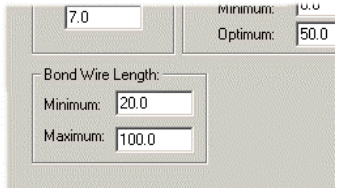



The rules available are:

- the ability to define/check **Minimum** and **Maximum lengths** of the bond **wires** from the die pads

- the ability to define the bond wire **cross-overs**, whether they are allowed or not through the support of insulated or un-insulated (bond) wires.

The **Minimum** and **Maximum** lengths of the bond wires is defined in the **Technology File** under the **Spacing Rules** tab.




The **Wire Cross** check uses the **Insulated** property of a Wire. The Wire between the die and bond pads (wires are different to normal connections) can be defined **Insulated** or un-insulated. If wires are insulated and cross, no error is reported, if they are not insulated and cross, then an error is flagged.

You also have the ability to plot Wires if required (and also connections). This may be required for check plots but will rarely be used for manufacturing plots.

Within the **Technology file** under **Layers** there are entries for **Layer Class** definitions for bond pad plots.

The standard supplied libraries include a small selection of bare die Footprints and their associated Parts.

## Report Maker features

The lists below are ancillary features which aid the production and reporting of the Chip Packaging Toolkit items but are included within the standard Pulsonix PCB system:

- Wire report output **wire.rff.** This reports the X and Y position of the pads attached to each end of the Wire and the Net Name of the Wire.

The **Report Maker** can output bond pad positions (**is bond pad**) and die pad positions (**is die pad**).

In addition to the Wires report, the Report Maker can also create a report for manual and automatic wire machines using the standard commands available and the format scripts.

# Appendix A Default Files

This section details new default files supplied or changes made to existing files.

## Technology Files

### Schematic Technology Files (*.stf)

- **Multi Instance** SCM default technology file for use with multi-instanced hierarchy

## Format Files

New format files can be output using the new **Report Maker** option or from the **Reports** mechanism both from the **Output** menu.

New format files include:

- **JTN Netlist.rff** JTAG format netlist for boundary scan technology
- **IPC-356.rff** Output formatter for the IPC356 test format
- **Wire.rff** Output formatter for reporting the start and end positions of wire.

## Colour Files

Colour files have been supplied as examples. These can be modified and saved as required.

- ***.pcl** PCB Design colour files
- ***.scl** Schematic Design colour files

## Block Files

Block files have been supplied as examples. These are only examples, you can create and save your own as required.

- **Half Adder.blk** Example block file
- **Full Adder.blk** Example block file

# Appendix B Libraries

## Changes to the Supplied Libraries

Some of the standard libraries have been modified with additional pin types for use with the Electrical Rules Checking.

These are:

| | |
|---|---|
| 74 | 74 series TTL logic Parts library |
| Resistor | Resistor Parts library |
| Capacitor | Capacitor Parts library |
| Generic | Generic Parts library |

Pulsonix now supports Associated Parts. The libraries have been updated to supply some real examples, such as Bolts, Heatsinks, Nuts etc.

| | |
|---|---|
| Hardware.pal | Hardware Parts library |

Jumpers and wire links have been added to Pulsonix, some Jumper Parts have been added to the Parts library.

| | |
|---|---|
| Hardware.pal | Hardware Parts library |

A selection of bare dies have been added to the Pulsonix library as examples for the Chip Packaging toolkit.

| | |
|---|---|
| Dies.pal | Chip Packaging Parts library |

# Index