



Library Creation Tutorial

Copyright Notice

Copyright in the whole and every part of this software and manual belongs to RS Components and may not be used, sold, transferred, copied or reproduced in whole or in part in any manner or in any media to any person, without the prior written consent of RS Components. If you use this manual you do so at your own risk and on the understanding that neither RS Components nor associated companies shall be liable for any loss or damage of any kind.

RS Components does not warrant that the software package will function properly in every hardware software environment.

Although RS Components has tested the software and reviewed the documentation, RS Components makes no warranty or representation, either express or implied, with respect to this software or documentation, their quality, performance, merchantability, or fitness for a particular purpose. This software and documentation are licensed 'as is', and you the licensee, by making use thereof, are assuming the entire risk as to their quality and performance.

In no event will RS Components be liable for direct, indirect, special, incidental, or consequential damage arising out of the use or inability to use the software or documentation, even if advised of the possibility of such damages.

RS Components reserves the right to alter, modify, correct and upgrade our software programs and publications without notice and without incurring liability.

DesignSpark is a Trademark of RS Components, Microsoft, Windows, Windows NT and Intellimouse are either registered trademarks or trademarks of Microsoft Corporation.

Eagle is the copyright of CadSoft

All other trademarks are acknowledged to their respective owners.

Copyright © RS Components. 1997-2011. All rights reserved. E&OE

Issue date: 18/10/11 iss 3

RS Components Ltd
International Management Centre
8050 Oxford Business Park North
Oxford
OX4 2HW
United Kingdom
Tel: +44 (0)1865 204000
Fax: +44 (0)1865 207400

Contents

CONTENTS	3
CHAPTER 1. LIBRARY CREATION	5
Introduction	5
Understanding DesignSpark Libraries	5
Schematic and PCB Symbol Libraries	7
Component Libraries	7
Library Manager Overview.....	8
CHAPTER 2. CREATING SCHEMATIC SYMBOLS	9
Getting Started.....	9
CHAPTER 3. CREATING PCB SYMBOLS (FOOTPRINTS)	19
Getting Started.....	19
Editing Footprints in the Library	24
Creating Surface Mounted Footprints	27
The Reference and Symbol Origins	27
CHAPTER 4. CREATING COMPONENTS	29
Using the Component Editor.....	29
Adding Logic Names	35
Adding Component Values.....	35

Chapter 1. Library Creation

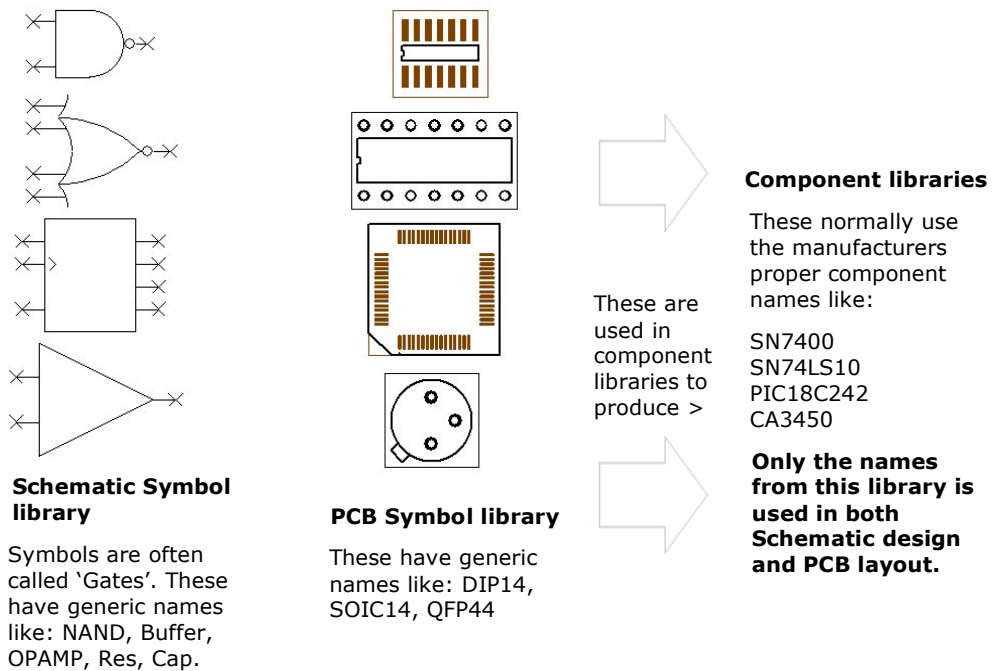
Introduction

DesignSpark is supplied with a set of libraries. These libraries can be used, modified and added to as required. You can mix and match new and existing library items together to create your own set of unique libraries.

The next few chapters concentrate on getting you started with creating your own libraries. After the last chapter on designing Components, there is more detail about how to use the Library Manager.

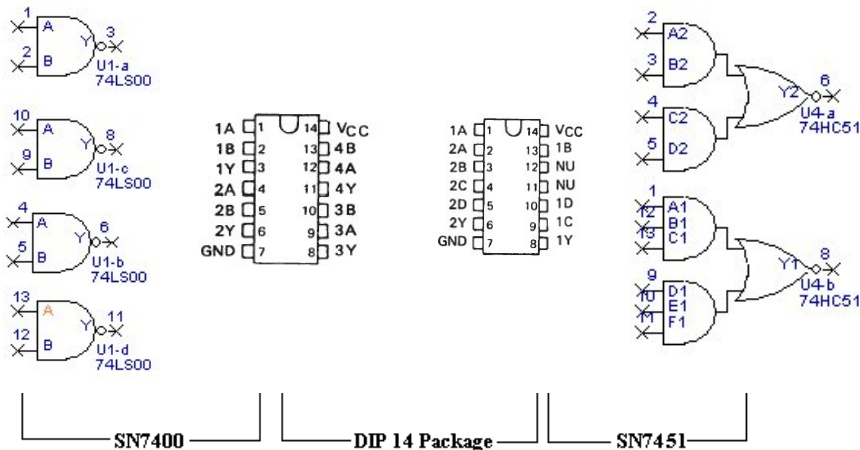
Understanding DesignSpark Libraries

DesignSpark libraries are easy to understand and use once your grasp the basic concepts. The libraries are made up of three individual libraries. There is a **Component library** made-up of elements from the **Schematic Symbol** library and the **PCB Symbol** (footprint) library.



The benefits of separate Schematic symbols, PCB symbols and Component libraries

In electronic design we have Schematics symbols and we have PCB symbols (or footprints). Commonly, integrated circuits and other components use the same physical footprint on the PCB, but have completely different technical specifications and internal digital or analogue circuitry. For example, take the standard 14-pin DIP footprint:



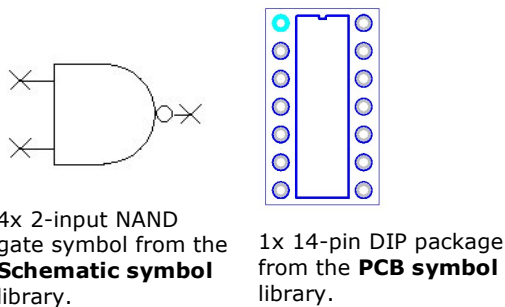
6 DesignSpark Library Creation Tutorial

The 14-pin DIP (Dual Inline Package) is used on 1,000s of different components. For efficiency, we only want to store the 14-pin DIP PCB footprint once and then use it repeatedly to build different Parts or components that specify a 14-pin DIP footprint.

By the same token we only want to store a schematic symbol once and use it repeatedly in various configurations to produce different components. Hence, we have a **PCB symbol library** to store our 14-pin DIP footprint plus any other footprints and a **Schematic symbol library** for our 2-input NAND gate plus any other Schematic symbols. These can then be used on their own or brought together (Schematic symbol & PCB symbol) in many different permutations to create a **Component Library**. We also have multiple Component libraries in DesignSpark to make the categorization of components clearer, these can be categorized by manufacturer, type, functionality etc.

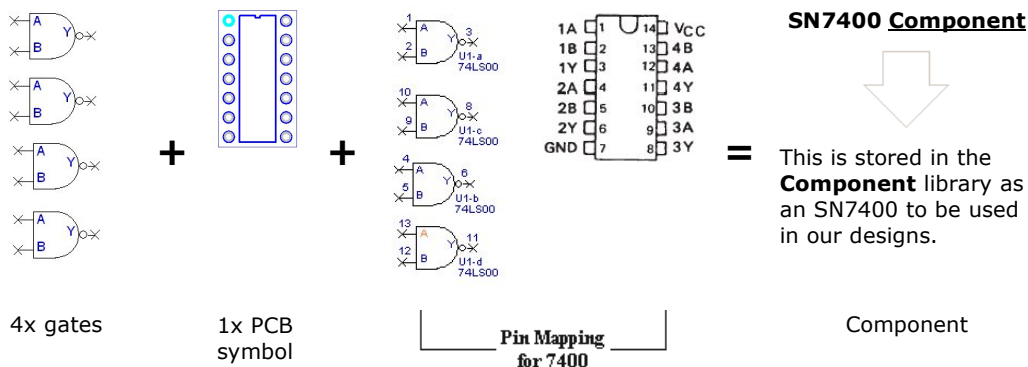
Key Point: In DesignSpark, it is only the component name taken from the **Component library** that is used in the Schematic design and PCB layout.

As an example, here's how we go about creating a simple 7400 component to use in the Schematic design and the PCB layout:



We can create a new 7400 component by selecting 4x 2-input NAND gates from the **Schematic symbol library** and a 14-pin DIP package from the **PCB symbol library**. By combining these in the component editor and inserting the correct pin mapping as per the databook specification, we produce an SN7400 **Component**. This is then stored with a unique component name (SN7400 in this example) in the **Component Library** ready to use in our designs.

In a working example, this is how it might look:



Question: Can I create a Schematic symbol in DesignSpark and then select it for use in the Schematic design without creating a PCB symbol?

You can create a new Schematic symbol and store it in the Schematic symbol library. But before you can use it in a design you have to create a **Component** for the symbol and store this in the **Component library** for use in your designs. Typically, this is how power and ground symbols would be defined for example.

Why is there this added step of creating a Component - why can't I just use the symbol immediately?

Working this way, any Schematic Symbols you produce are stored with an appropriate name in the Schematic symbol library. It can then be reused multiple times to create different components using the manufacturer's component name. Most components you produce will have the complete Schematic symbol and PCB symbol.

Is this the same for PCB symbols? Can I create a PCB symbol without a Schematic symbol?

This is exactly the same for PCB symbols as with Schematic symbols. Before you can use your new PCB symbol in a design you have to produce a component even though this component may not have a Schematic symbol associated with it. A PCB only symbol could be added to the design as a drawing blank or a test-pad probe point for example.

Once a PCB only or Schematic only component has been created, the missing opposite symbol type can be added to make it available to both design editors.

On the following pages is a tutorial that goes through the steps of creating Schematic symbols, PCB symbols and a component to tie them together.

Note: Standards for symbols vary from company to company. The tutorial is presented only as a means to understand the process of design and management and how to use the tools available.

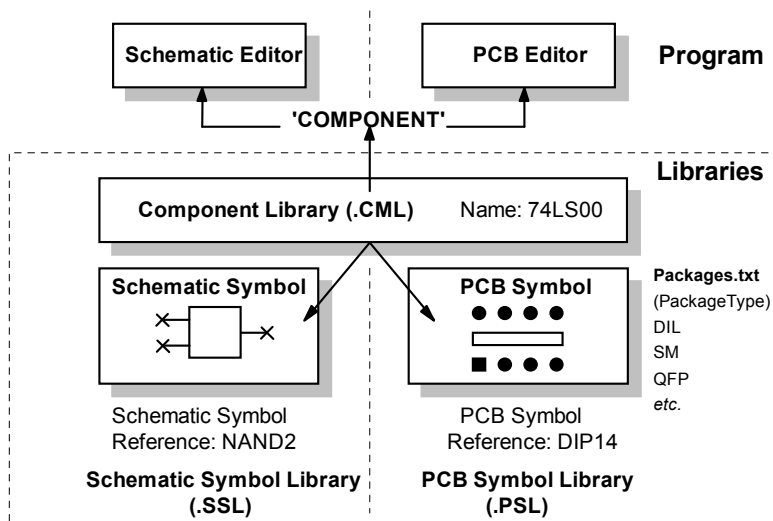
Schematic and PCB Symbol Libraries

Two types of symbol libraries exist, those for Schematic symbols with the extension .ssl, and those for PCB symbols with the extension .psl. There is no limit how many symbols may be kept in a given library.

Component Libraries

Component libraries can be identified with the file extension .cml. Each component in a library is aware of the symbol name and library name of its constituent symbols. The components can contain several Schematic symbol 'gates' in each component, and a number of PCB footprints, each with its own symbol. For example, a component might have assigned to it 3x NAND gates and a power gate for use in the Schematic design, a DIL and an SMD footprint for use in PCB.

The diagram below shows how the Component will access both the Schematic symbol library and the PCB symbol library as required in the relevant design editor.



Library Manager Overview

The Library Manager is used to create and manage all your libraries. This brief overview summarizes its operation and is discussed later in this document.

Use the **Symbol** and **Component** tabs to enter the correct library 'type' mode. For example, if you need to edit a PCB footprint, click on the PCB Symbol tab.

The **Dir:** list shows the current library path

The **Library:** list shows the **current selected library**.

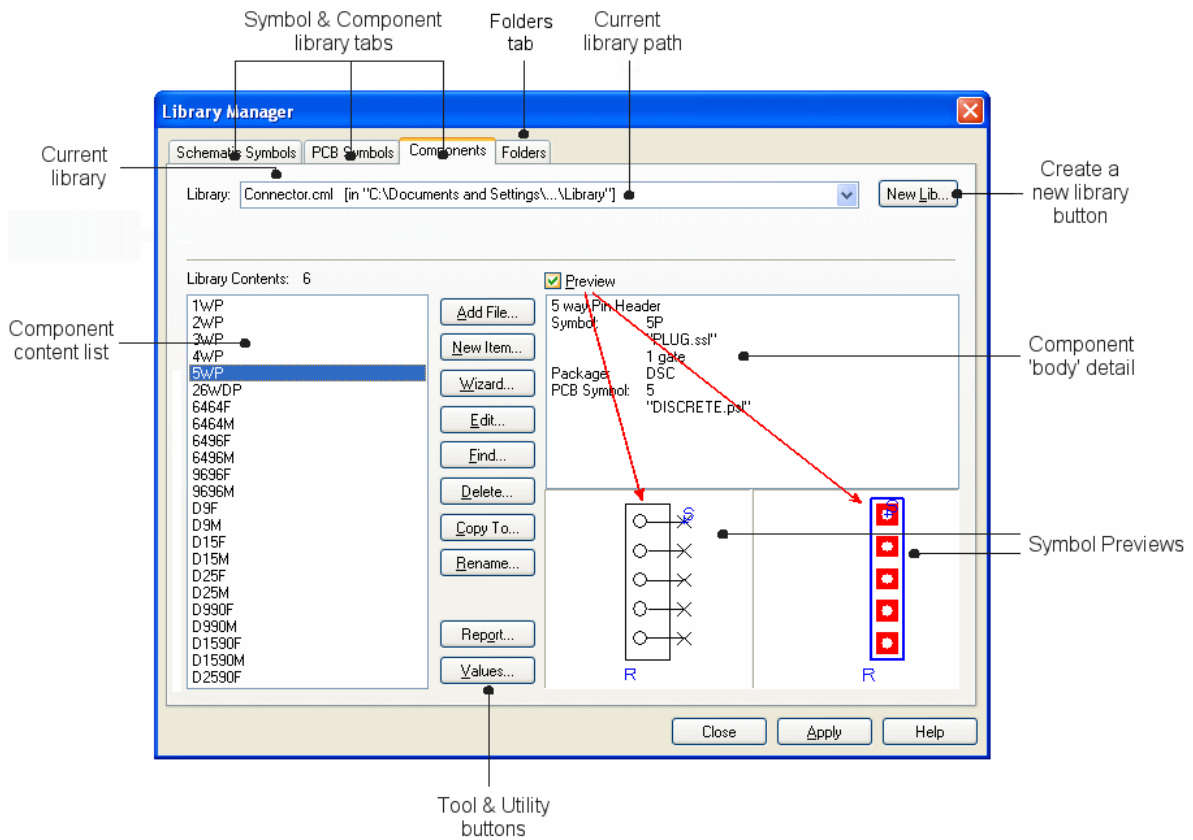
Any library items found in the selected library are displayed in the **Library contents list**. The example below shows the Component library.

When an item is selected from the Library Contents list, the contents are displayed in the **Preview** pane on the right hand side of the page:

If on the Components tab the contents are shown as detail (a detailed description of which symbols are used in the component, the libraries they are from, the gate/symbol relationships, the package/PCB symbol relationship). It also shows a preview of the symbols used for that component.

If on one of the Symbol tabs, the Preview button reveals a picture of the symbol selected.

The **Tools and Utility buttons** are used to perform actions on the library or use to manipulate them.

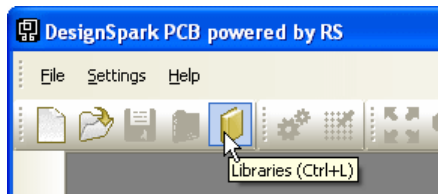


Chapter 2. Creating Schematic Symbols

Getting Started

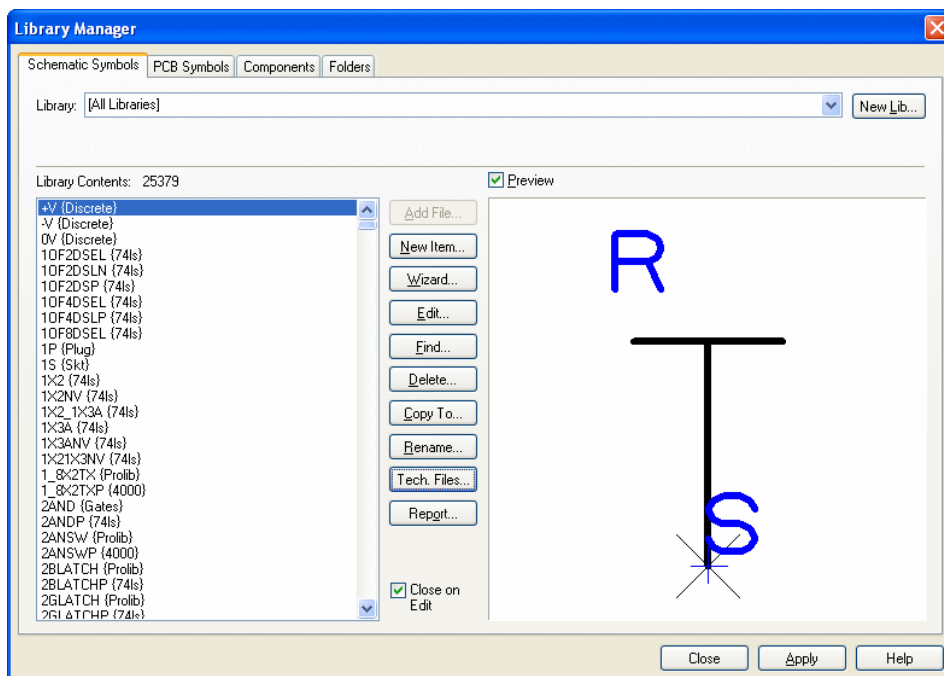
► To open the library manager

From the **File** toolbar, select the **Libraries** icon (shortcut **Ctrl+L**).



► To open the symbol editor

From the Library Manager dialog, select the **Schematic Symbols** tab.



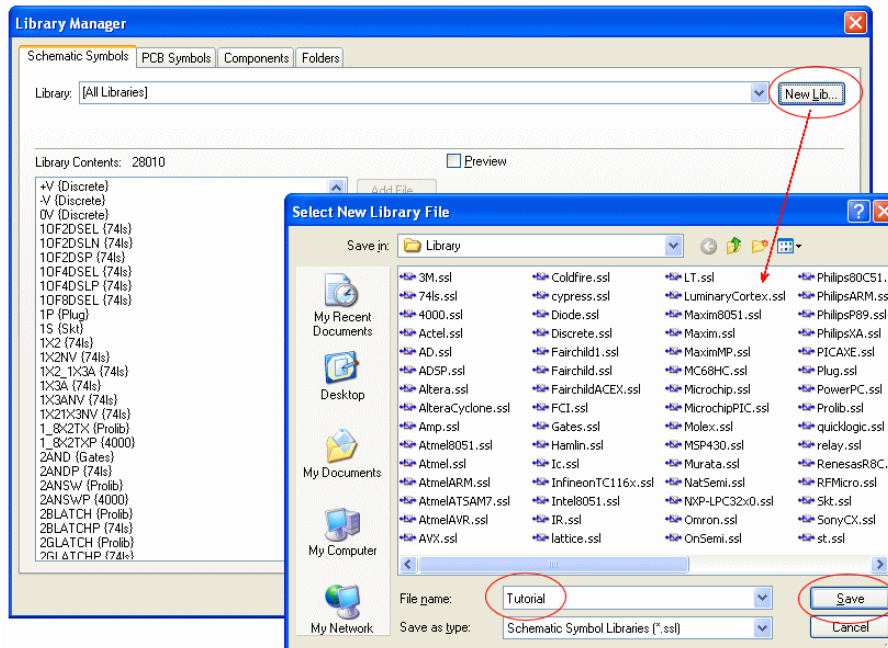
For this tutorial, we will create a new library for each symbol/component type. You could of course, add your new symbols to an existing library.

The Symbol Editor will take a set of default design settings, such as Styles, Colours, Grids, Units etc. These are taken from a selected Technology file. Once you become familiar with DesignSpark, you can customize these files using your own selections. If you click on the **Tech Files** button on this dialog, you will see how to choose this Technology file.

For now we will leave the defaults set to the ones supplied in the standard technology file.

10 DesignSpark Library Creation Tutorial

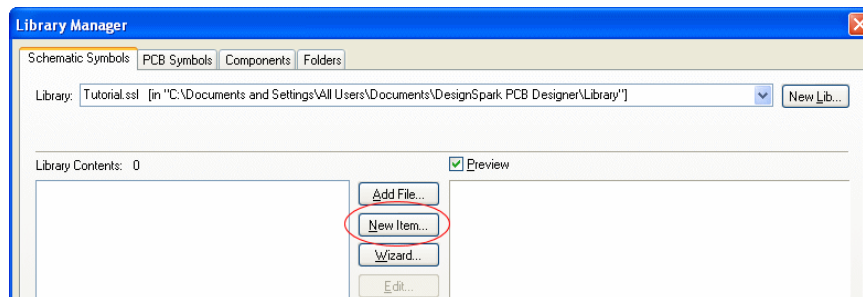
We will create a new library in which to save the new symbol into. Click the **New Lib** tab.



In the **File name:** box, type in the name **Tutorial**. This will be the name of your new library. Click **Save** to save this file name and exit the dialog.

The new Tutorial library is selected and is ready to have new symbols saved into it.

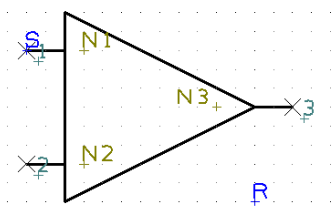
You need to create a new Schematic symbol so open the Symbol Editor using the **New Item** button.



The Symbol Editor opens. You will notice this looks very similar to the Schematic design editor used in the design tutorial. It is similar in its use and its interaction, the way in which items move and feel.

For this tutorial exercise, we are going to create an op-amp component. The component will contain two symbols, both different. It will use a op-amp gate plus a power gate, we will do this to demonstrate using multiple gates for one component.

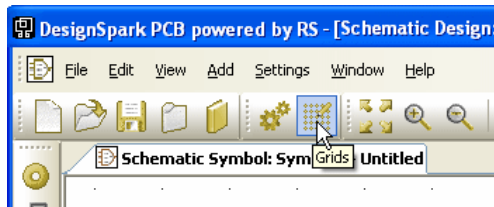
You will create the op-amp gate only, the power gate (PWR) already exists in the supplied libraries. We will create this gate:



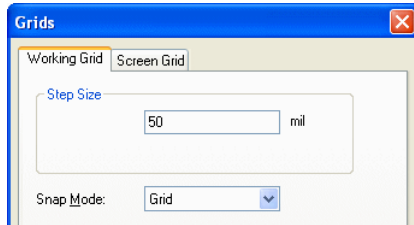
► Checking the screen grid dots

Firstly, zoom in (roll the mouse wheel or click the <Z> key) so you can see the **Screen Grid** displayed. The grid needs to be set to 50 mil so check this.

From the **File** toolbar select the **Grids** icon.



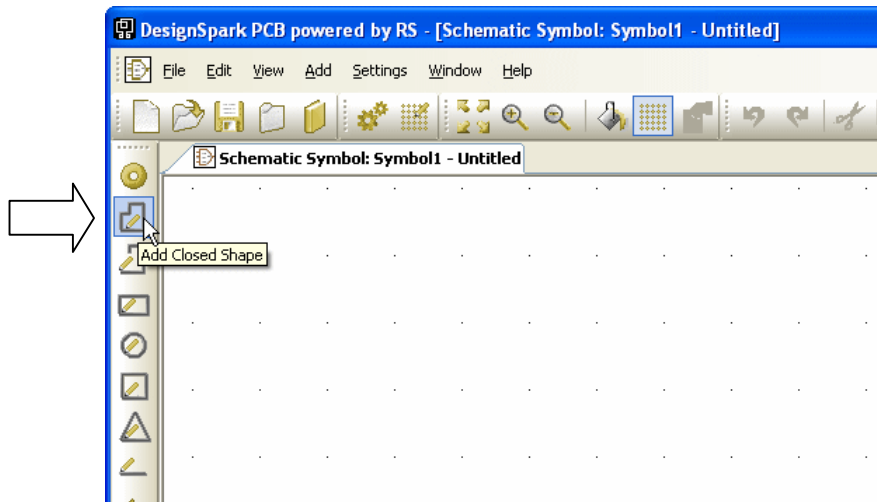
The **Grids** dialog is displayed.



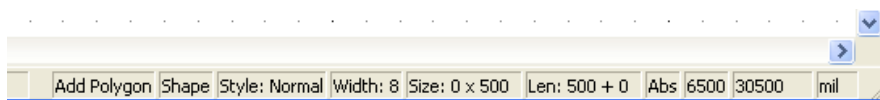
Ensure the grid **Step Size** is to **50** mil steps and click **OK**.

► Adding the symbol detail

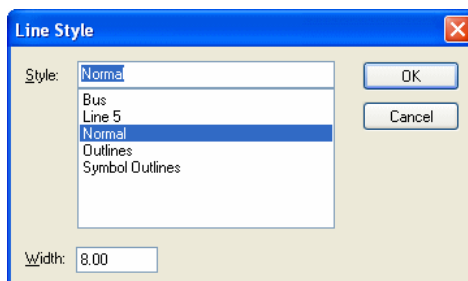
On the Schematic Symbol toolbar (the vertical toolbar on the left side of the screen) select **Add Closed Shape**.



The status bar at the bottom of the screen will show you the size being drawn.



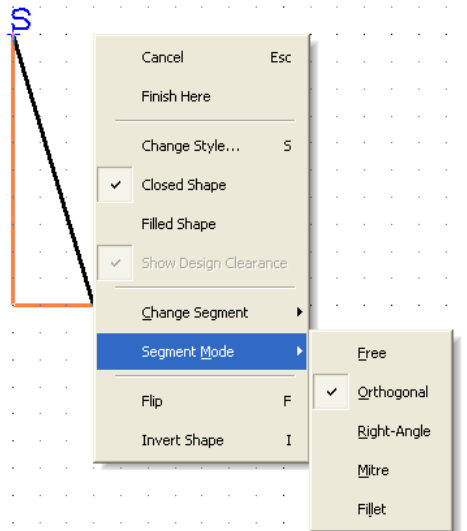
The **Style:** being used should be **Normal**. If the status bar isn't showing this, change the Style by pressing the <S> key. The **Line Style** dialog is displayed. Select Normal from the list and press **OK**.



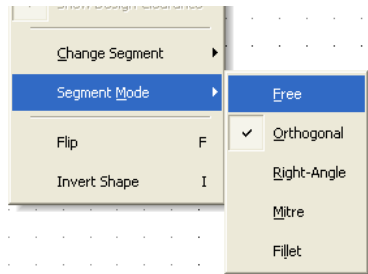
You are ready to start drawing the shape. Choose a starting point and create a shape 500 mil long moving down.

Notice the Symbol origin point '**S**' at the top left of the shape. We will move this later on to a more relevant location. This is always added to the original of the first symbol item added during creation.

In order to get the triangle shape, you will need to take the editing mode out of **Orthogonal** and switch it into **Free** hand editing. You could of course just simply use the Add Triangle option but this would be too easy! To change to Free hand mode, click once to add a corner, 500 mil down (10 grid points) from the start point and then right click to display the shortcut menu.

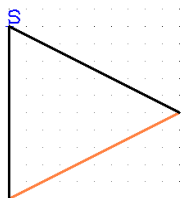


From the menu, select **Segment Mode>** and **Free**.



Now you can move the 'floating' second segment right 500 mil and up 250 mil to give you the point on the triangle. Because the grid is set to 50 mil steps you will be able to snap to 250 mils exactly.

Your shape should look like the picture below.

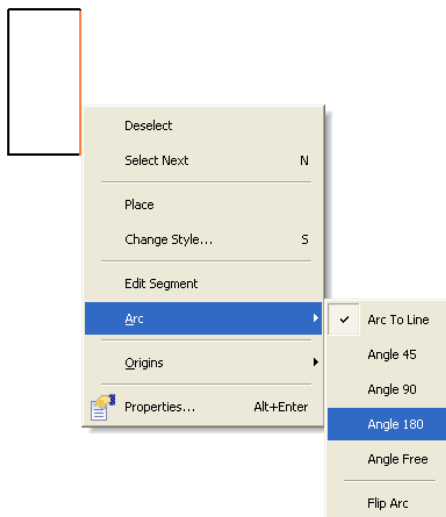


► To create an arc

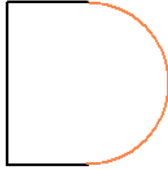
This example doesn't require an arc but if yours does in future, they are easy to create.

To add an arc, select a shape segment and right click. In the example below we have add an additional 'box' shape to demonstrate this on.

From the shortcut menu choose **Arc>** and then **Angle 180**. The segment becomes an arc.



The resulting arc looks like this:

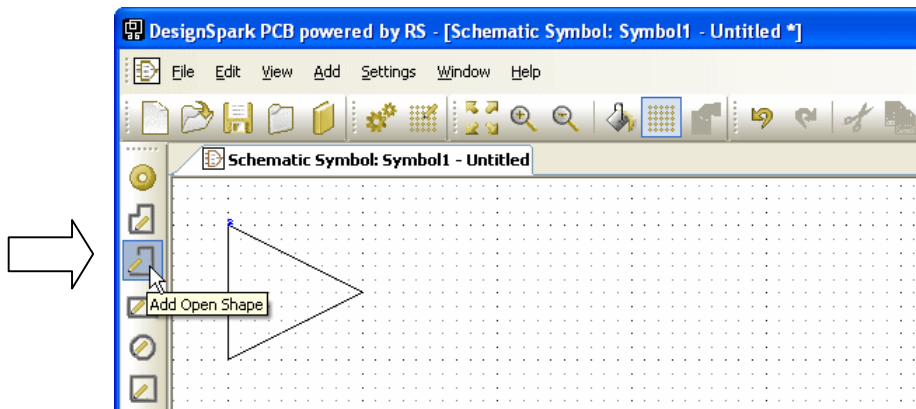


If your arc is the wrong way (faces inwards no out), right click on the arc shape again and choose **Arc>** from the shortcut menu and **Flip Arc**. This will swap it the other way round.

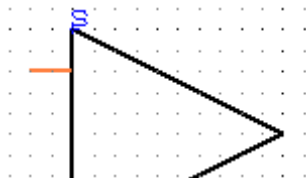
► Adding the symbol detail

You may need a bit more space around the existing shape to add some additional shape detail. Use the pan function <P> on the keyboard or press the middle wheel button, or use zoom out or zoom in.

On the **Schematic Symbol** toolbar, select **Add Open Shape**, you can also use the **Add Shape Single Line** option, they do the same thing.

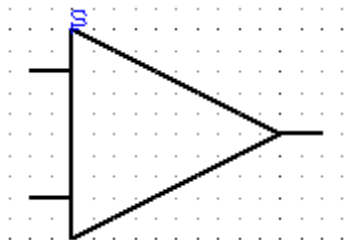


Select a start point 100 mils down from the symbol origin 'S' on the vertical segment (2 grid points), drag a line 100 mils (2 grid points) to the left. Like this:



Use a double-click to finish.

Add the additional lines required by selecting the line you have just inserted and clicking <Ctrl-C> to copy and <Ctrl-V> to paste. Do this twice. Of course you can always simply draw them in using the Add Shape Line tool as well.

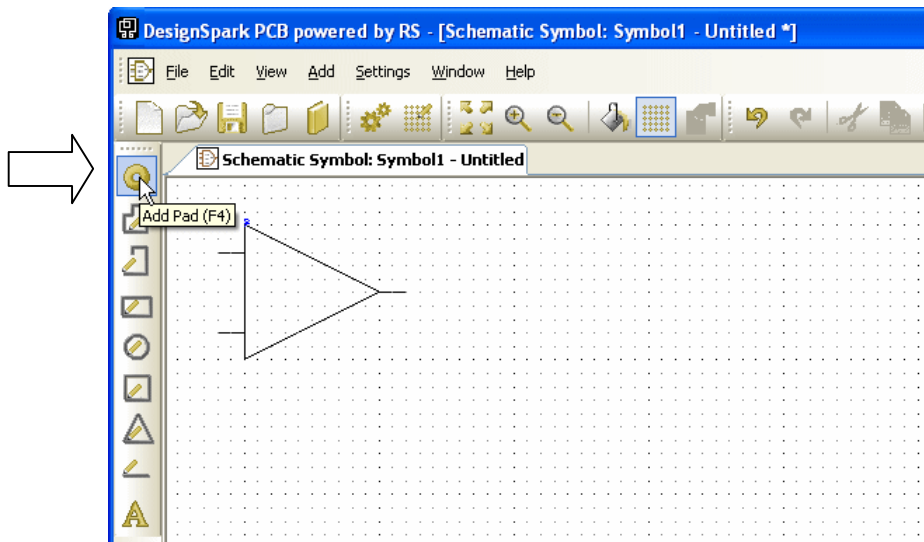


Position the two additional lines as shown above. At this stage DesignSpark does not recognize these lines as electrical connect points or pins, we are now going to insert these.

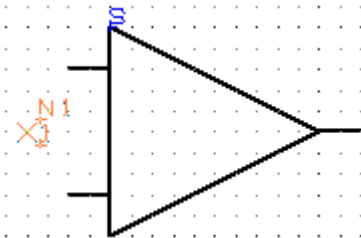
► Adding pins

Pins need to be added to the Symbol to give DesignSpark 'real' electrical points on which to make connections. This gives DesignSpark its unique true electrical connectivity at all times.

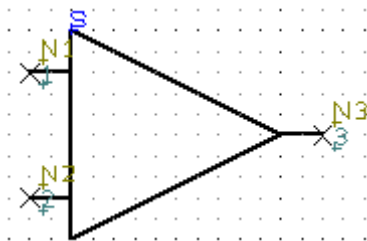
On the Schematic Symbol toolbar, select **Add Pad <F4>**.



A pin will be available on the end of your cursor shown in the highlight color.



Position each pin on the end of the lines so that it looks like the example below. As each pin is positioned, the next one in the sequence is available on the end of your cursor.



You will see that these are auto-numbered as you proceed. After the last pin has been added, click the <Esc> key to cancel.

These pins must be sequential but the numbering sequence on the Symbol doesn't matter, they are there for reference only. They must also be unique and with no gaps in the sequence.

Click the <A> key to position the symbol to fit the window.

► Information on the pins

Each Pin contains three pieces of information:

- the big cross marking the shape and location of the pin
- the Pin Number (**1, 2, 3 etc.**)
- the pin name position marker, **N1, N2, N3** etc. which marks the location of the pin name when used on the Component in the design

The **Pin name position marker (N3)** will only be populated once the Component is used in the Schematic design, until then you only see the name position marker.



You may also see the symbol origin 'S' character on Pin 1 if this pin was the first item to be added to the symbol (before the symbol shape).

The **Pin Number** is the automatically numbered. This can represent the actual terminal number if required but can be renumbered in the Component definition. The number is shown in the position it

will appear in when used in the design. This can be moved in the Symbol editor but cannot be moved once placed in the design.

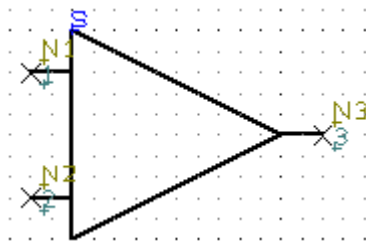
*Tip: When adding pins, if you have moved the sequence number and the location or alignment of the pad name text, if you wish to use this on subsequent terminals, select it first and then select **Add Pad**. This will take the style of the selected terminal and add the next one in exactly the same way.*

► Moving the symbol origin

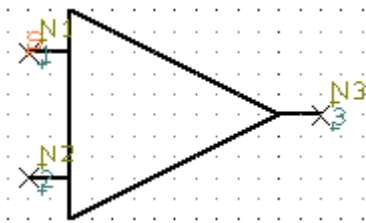
The Symbol origin is the Symbol's reference point used when moving it in the Schematic design (the small 'S' marker). For consistency we always try and place this in the same relative location, usually on Pin 1 of the symbol.

The symbol origin would have been added with the first design item. In our example, this was the symbol shape but it could have been the first pin added.

Select the origin on the symbol.



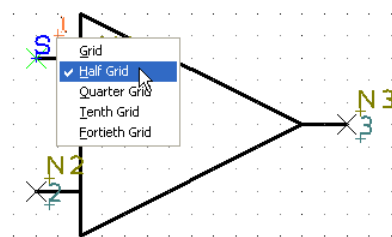
Select and drag the symbol origin to Pin 1 as shown below.



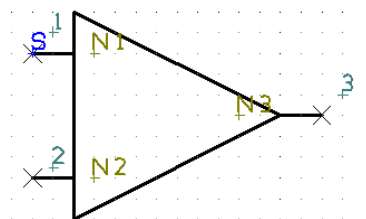
► Moving the Pin names

Select and drag the pin numbers (pins 1, 2 and 3) and pin names (pins N1, N2 and N3) and move them into the required location making sure only the name is selected and not the pin itself (the big X).

To make the positioning of pin names and numbers easier and more refined, during the move, use the shortcut keys <Ctrl-G> to display the snap mode. Change the snap to a **Half Grid** or **Quarter Grid**.



Position the pins to look like this:

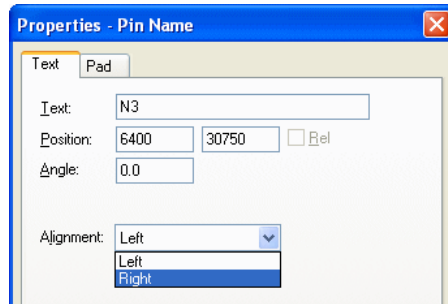


► Changing the text alignment

The pin names are written from left to right by default, this is a left text alignment. In our example, if the pin name for N3 is populated in the Component and then displayed, this will overwrite the

symbol shape outline and will look cluttered. Ideally, this should be written from the right back into the symbol body.

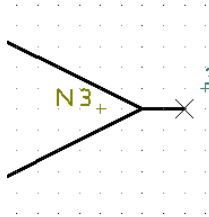
To do this, select the **N3** pin name position marker and from the shortcut menu choose **Properties**, the following dialog is displayed:



You can change the text alignment that will be used once the symbol is placed in the design and if a pin name has been used, you can opt to align it Left or Right, that is the direction it will read from.

For our example, select the **Right** alignment and click **OK** on the dialog.

The symbol will update to reflect this change. The pin name direction is now left most of the pin maker indicating it will be justified from right to left.

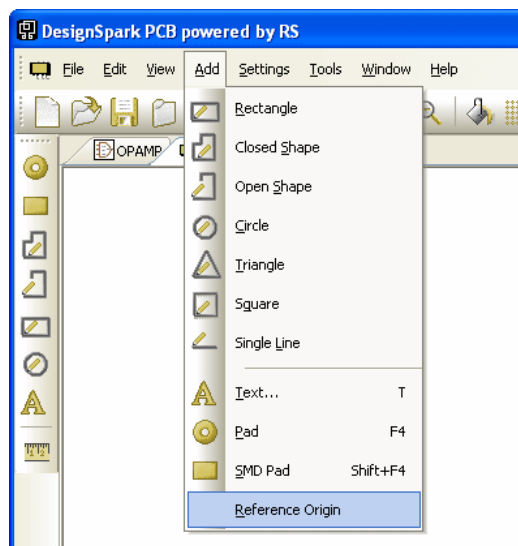


► Adding the reference origin

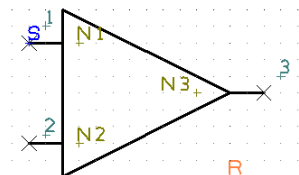
Without a **Reference Origin**, the component name (U1, IC1 etc.) used in the Schematic design will automatically be placed on the Symbol Origin. By adding a specific Reference Origin you can position the Component name where you like.

By default, the DesignSpark libraries have the Reference Origin placed at the bottom right side of the Symbol shape.

To add the Reference Origin, select **Reference Origin** from the **Add** menu.



Place the 'R' marker on the Symbol.

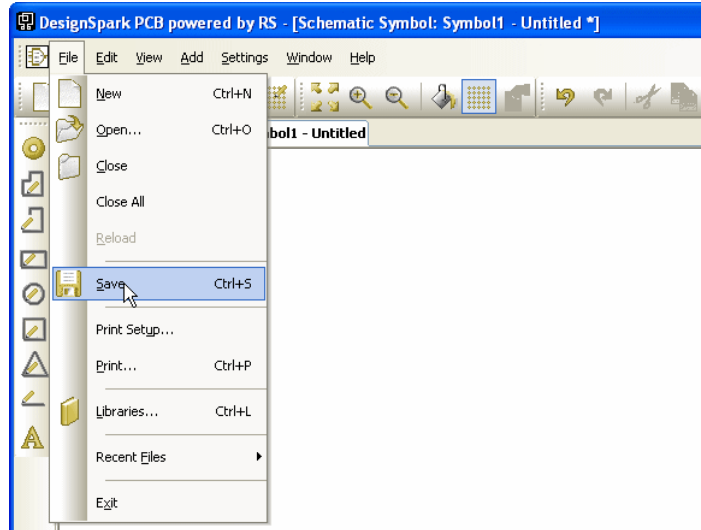


When the Symbol is used with the component in the design, the Reference Origin is also used for positioning the Component Name, Package Name and PCB Symbol Name if these items are selected as displayed.

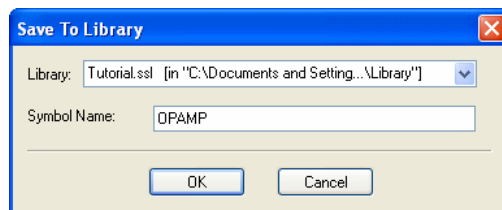
You are now ready to save the symbol into the library.

► Saving the symbol

From the **File** menu, select **Save**.



The **Save** dialog opens.



Choose the **Tutorial.ssl** library and give the symbol the name **OPAMP**, click **OK**.

► Viewing the symbol in the library

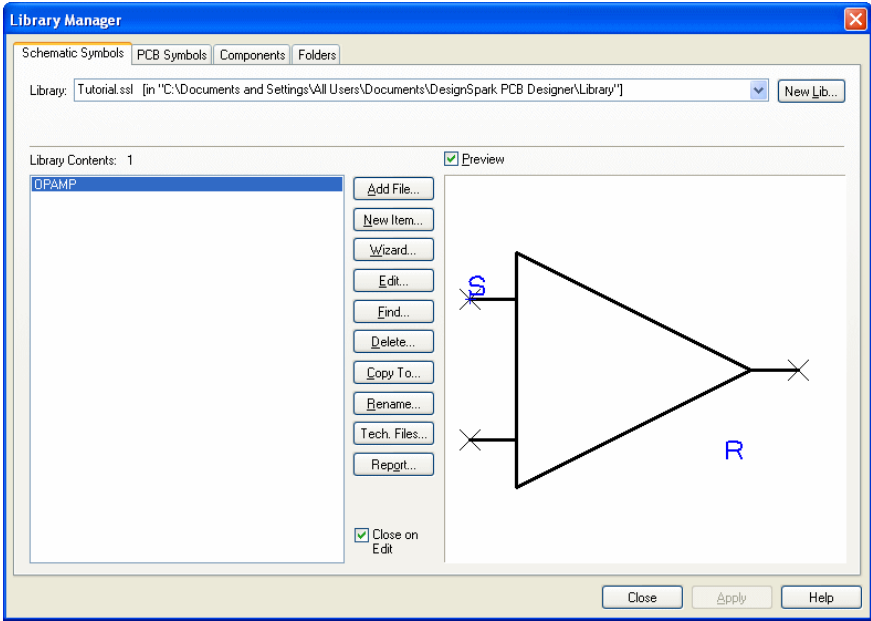
Now that the symbol has been created and saved, let's view it in the library manager.

From the **File** toolbar, select **Libraries**, you can also use the <Ctrl-L> shortcut key.

The Libraries dialog opens. Select the Schematic Symbols tab if it isn't already selected and select the **Tutorial.ssl** library from the drop down list.

Ensure the **Preview** box is checked.

From the **Library Contents:** list, select the **OPAMP** symbol that you've created, it should look like the one below:



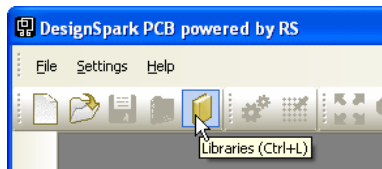
Once you are satisfied with this, move onto the next section to create a PCB Symbol (footprint).

Chapter 3. Creating PCB Symbols (Footprints)

Getting Started

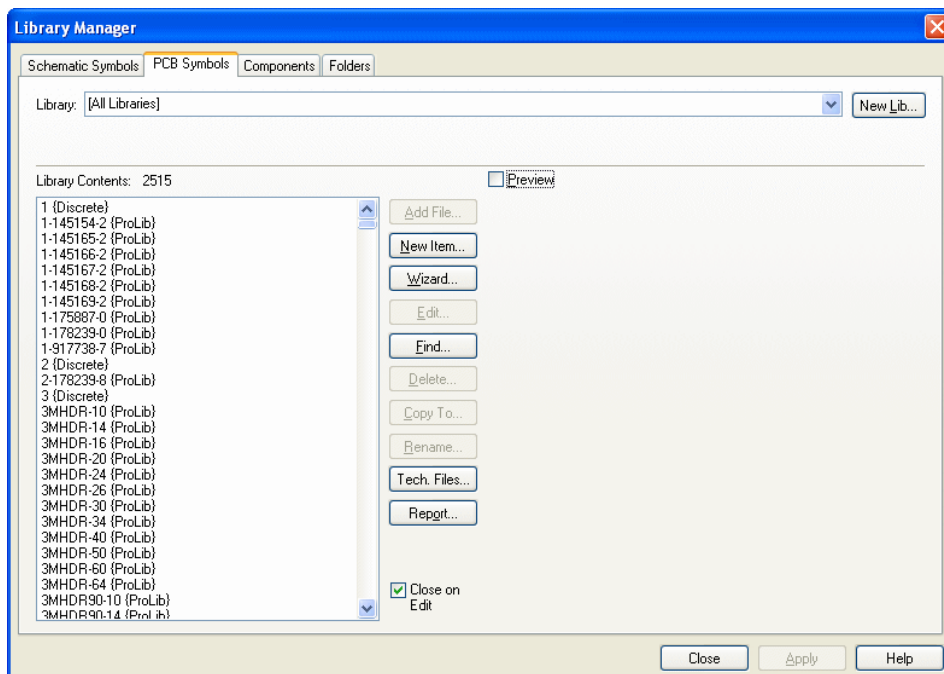
► Start the Library Manager

From the **File** toolbar, select the **Libraries** icon (shortcut **Ctrl-L**).



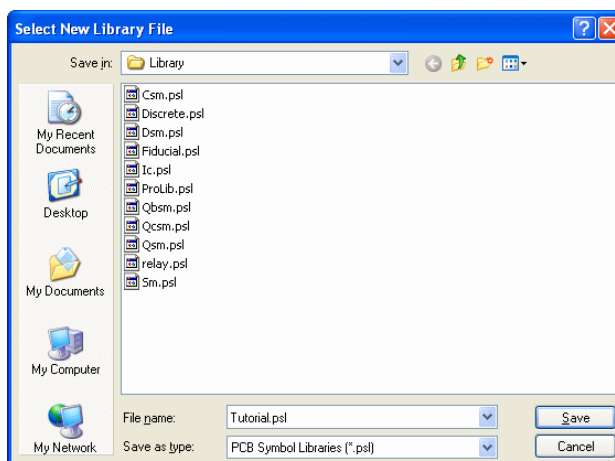
► To open the symbol editor

From the **Library** manager dialog select the **PCB Symbols** tab.



For this tutorial, we will create a new library to add your new PCB symbol to.

Click the **New Lib** button.

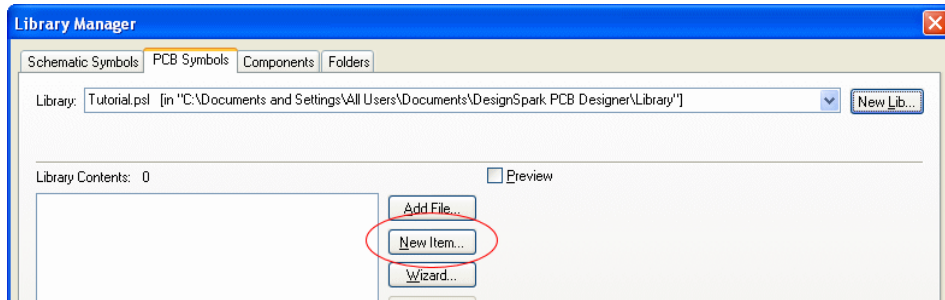


In the **File name** box, type **Tutorial** as your new PCB symbol library. Click **Save** to accept this. The library manager is now ready to start entering new PCB symbols.

► **To start the PCB symbol wizard**

For this tutorial, we will use the PCB footprint wizard. For footprints that need calculating dimensions for, this is the easiest way to create them. If you wish, you can also use the graphical method as demonstrated in the Schematic Symbol editor.

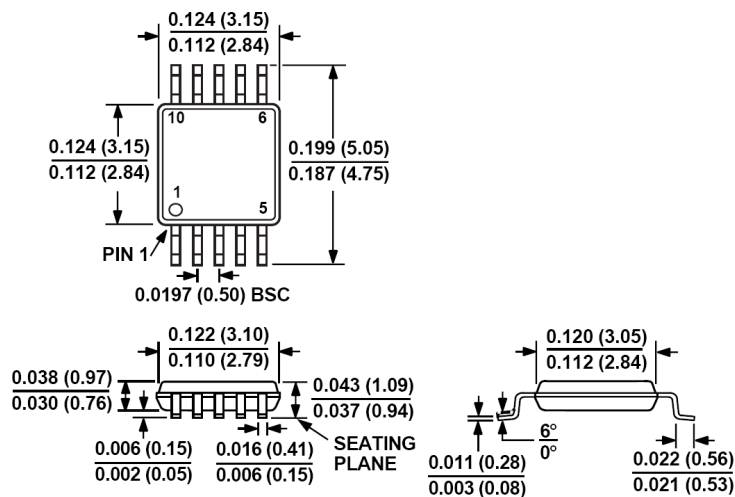
From the **Library Manager**, click the **Wizard** button. Ensure that you are on the **PCB Symbols** page.



► **The example footprint**

Using the manufacturers datasheet, you may be lucky enough to be presented with footprint pattern dimensions. Typically, the datasheet will just reference a footprint name which you must then source the dimensions for. The one we will create is shown below, be aware that this is a very small package:

10-Lead μ SOIC (RM-10)



Copyright © acknowledged to Analog Devices Inc.

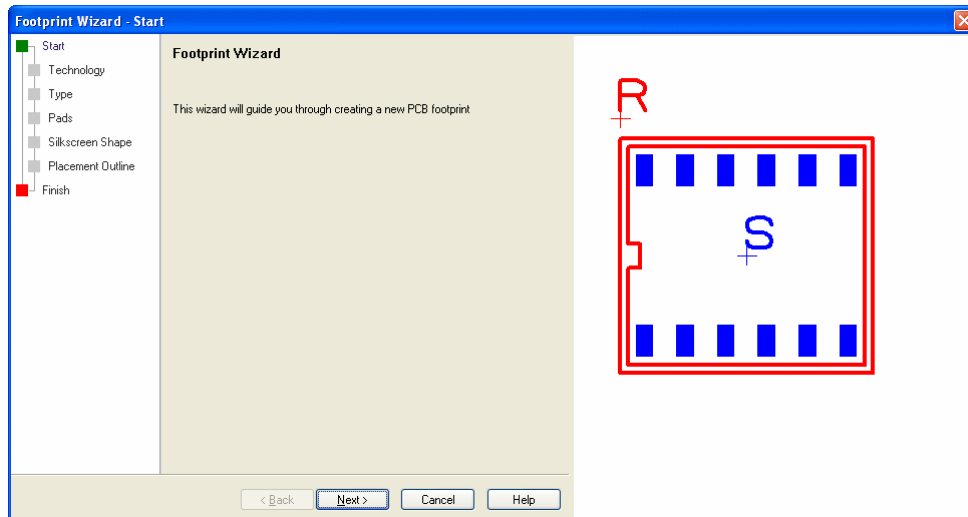
► **Using the footprint wizard**

Click the **Wizard** button.

For the purpose of clarity in this tutorial we have changed the colours of the items and the background so they contrast better when printed or viewed as a PDF file. If you prefer, you can change the colours to be the same using the Colors dialog.

From our example footprint shown above, the other aspect you'll notice is how exact the range of dimensions is. This exactness also gives you a level of ambiguity as you need to know how the footprint will be manufactured before being able to complete it.

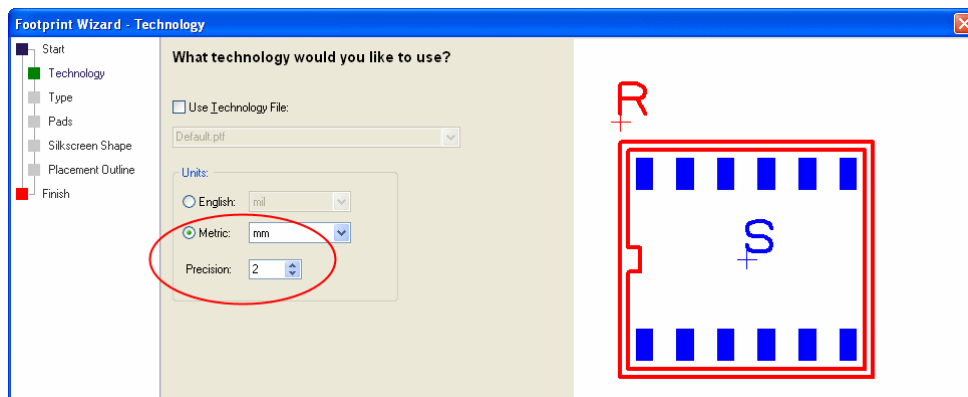
The first page you are presented with is **Start**.



As you will notice, this is the start of the footprint creation process.

Click the **Next>** button to move forward and start the new symbol. Across the bottom of the page are always located the **<Back**, **Next>**, **Cancel** and **Help** buttons. You can use these to navigate each page, or you can simply click the mouse on the navigation bar at the top left side of the wizard.

The next page is used to choose a possible **Technology** file and the **Units** to be used.



No technology file will be used yet but we will change the **Units** to **Metric** and a **Precision** of **2**. Our footprint is very small and will require a pin pitch of 0.5mm.

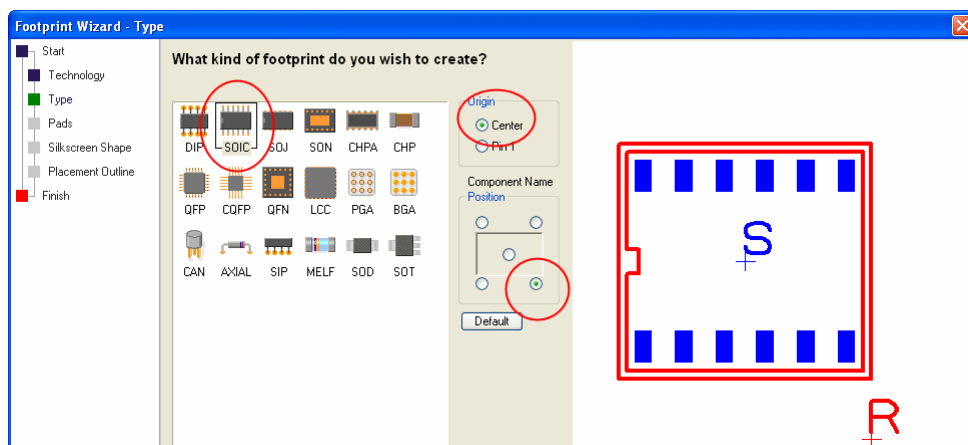
The preview pane will display a symbol, this is purely a graphical representation of the current shape and will be changed shortly. We've also changed the background color from black to white to make it more readable in this PDF file. Yours will be black still.

Click **Next>** to move on.

On the type page, select the **SOIC** shape. Your preview will now change to represent an SOIC device.

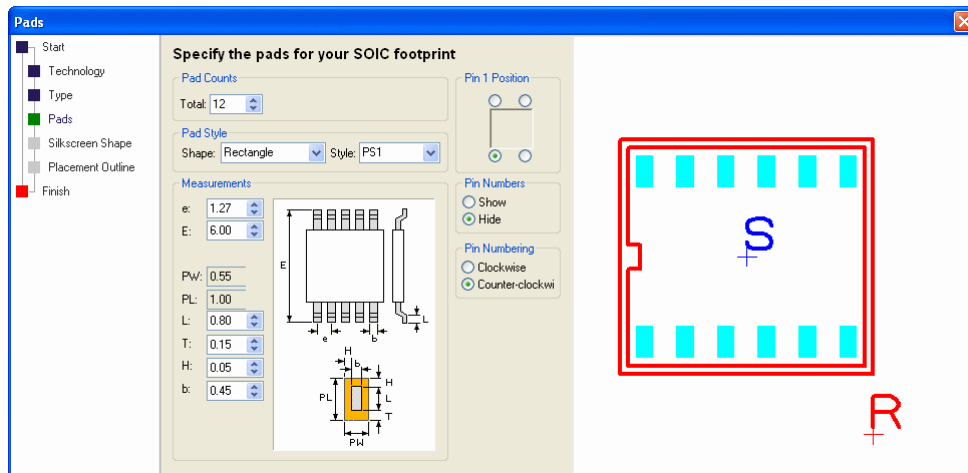
On **Origin**, select **Center**. For a surface mount (SM) component, this is normal.

For **Component Name** position, click the bottom right radio button. You'll see the preview update.



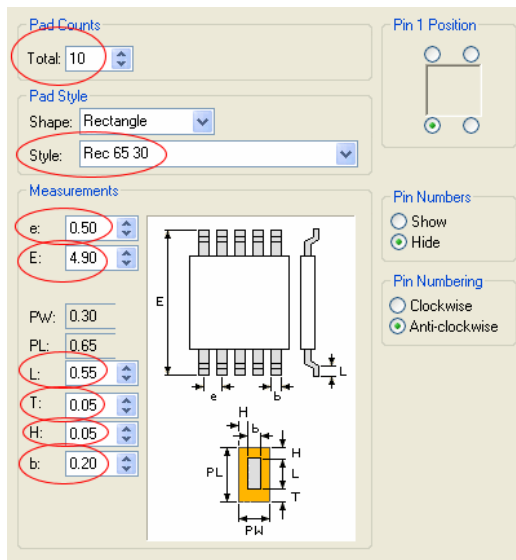
Click **Next>** to move on.

The **Pads** page is used to define the overall footprint layout. This is probably the most important page of the wizard. The default page setup is shown below, we need to change the parameters defined here.



The **Pad Count** should be set to a **Total** of **10**, use the spinner or type in the number.

You must type in the name of the pad **style** that will get saved once you've edited the dimensions. Use **Rec 65 30** for this tutorial example.

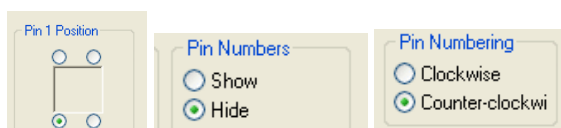


In Measurements, you need to enter the dimensions to be used. Generally speaking, these will be located on the manufacturer's datasheet or in another central resource, such as the IPC Council standard.

We've highlighted the correct dimensions for our footprint; all are in mm (as defined on the Technology page).

e	0.50
E	4.90
PW	Automatically calculated by the wizard
PL	Automatically calculated by the wizard
L	0.55
T	0.05
H	0.05
b	0.20

You must now define the position for **Pin 1** to be the bottom left corner, to **Hide Pin numbers** and the **Pin Numbering** to be **Counter-clockwise**.



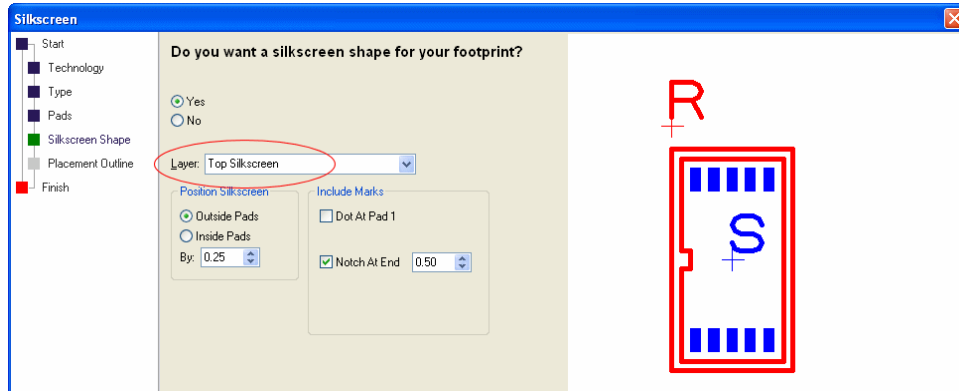
Click **Next>** to move on.

You should now define the **Silkscreen** outline.

This dialog is setup by default to assume you will always require a Silkscreen outline. If you do not wish to have one, check the **No** radio button. Leave it checked as **Yes**.

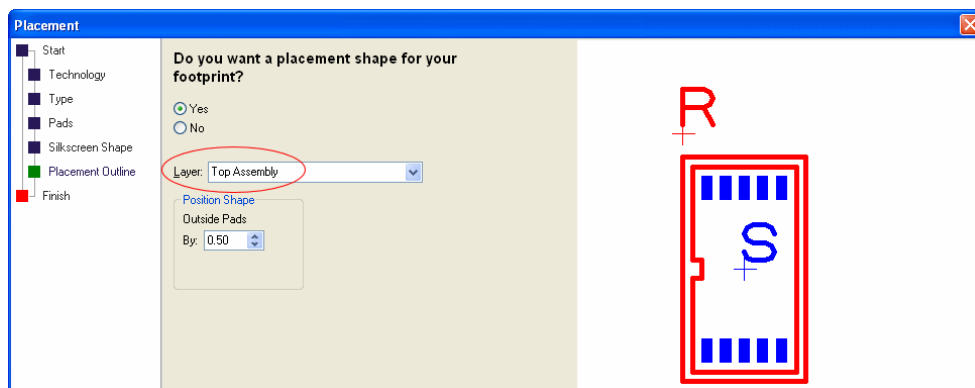
Select **Top Silkscreen** in the **Layer:** drop down box.

The only parameter to change here is the **Notch At End** value, change this to **0.50**



Click **Next>** to move on.

We will use the placement shape and add it to the **Top Assembly** layer. Select this layer from the drop down list.



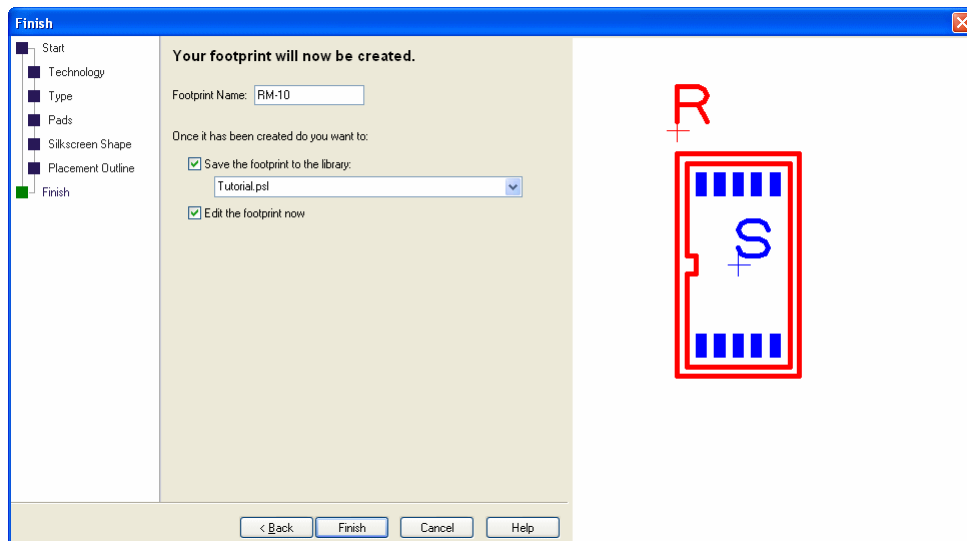
Click **Next>** to move on.

In the **Footprint Name:** box, type **RM-10**.

Check the **Save the footprint to the library** box.

Select your **Tutorial.psl** library from the drop down list.

Leave the **Edit the footprint now** check box unchecked.



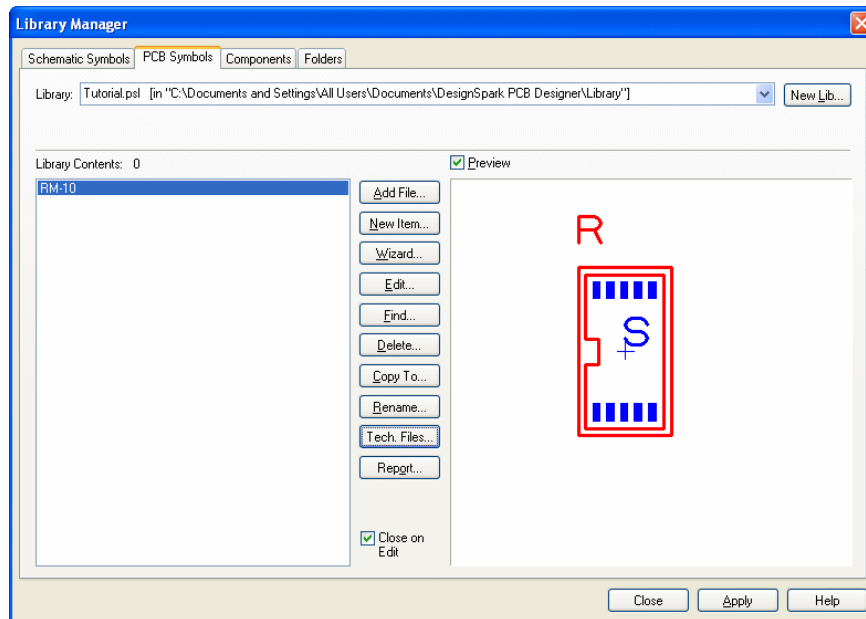
Click **Finish** to complete the footprint creation.

► Viewing the symbol in the library

Now that the symbol has been created and saved, let's view it in the library manager.

From the **Libraries** button on the File toolbar, select **Libraries**, shortcut key <F11>.

Select the newly created footprint from the contents list (RM-10). Check the **Preview** box if not already selected so you can view the footprint. You can also edit it from here as well.



The footprint is now ready to be used in a component.

Editing Footprints in the Library

The following information is additional to using the footprint wizard. At this stage you can skip this section and jump to the Design Components section if you wish to proceed with the tutorial.

If you edited your footprint or wish to edit it, you can add or change the entities displayed.

► Adding mounting holes to your component

If your symbol requires additional pads to be added for use as mounting holes you should use the **Add Pad** option from the **Add** menu.

Although these pads will be sequenced along with the symbol pads, it does make it much easier if they exist outside the main pad sequence as they can then be simply ignored. You can have a different number of pads to terminals between the Schematic symbol and PCB symbol.

When mapping terminals and pads you can simply ignore the extra pads added for your mounting holes.

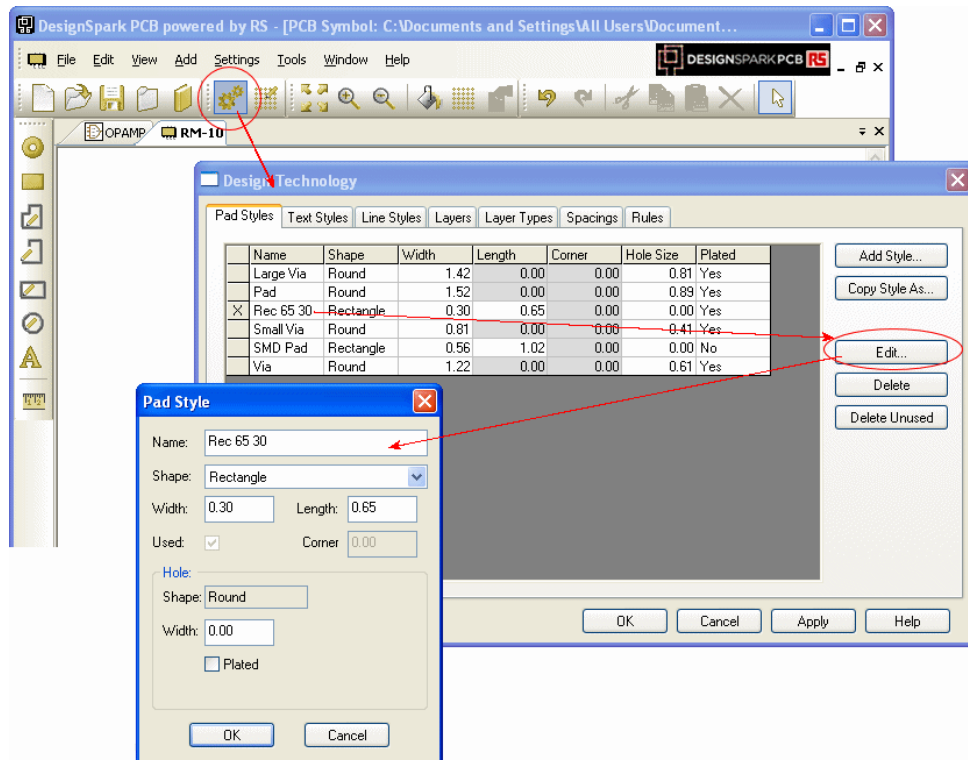
► Changing Pad Styles

Pad styles can be changed on pads at any time. They can be changed globally on all pads of a particular style or they can be changed locally on each pad.

Global pad styles

To change a pad style for all pads in the footprint which use a named style, select the **Styles** button from the **File** toolbar.

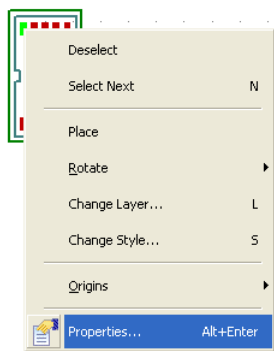
Click the **Pads** tab. Select the pad style to change in the grid, in our example **Rec 65 30**.
Click the **Edit** button. Change the properties of that style and click **OK** to accept them.



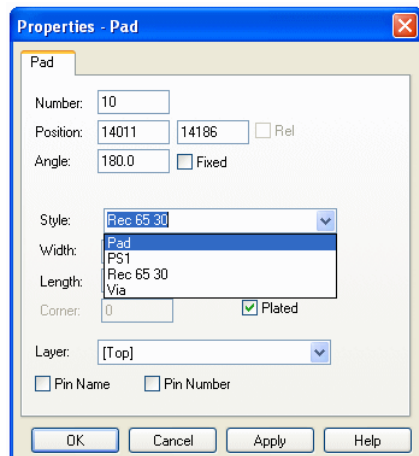
When back in the footprint editor, all pads using this style will now be changed.

Local pad styles

To change one pad to use a different pad style, select the pad on the footprint. Right click and select **Properties** from the context menu.

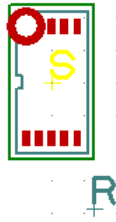


From the **Pad Properties** page, select a new style from the drop down list.



Click **OK** to accept this new style.

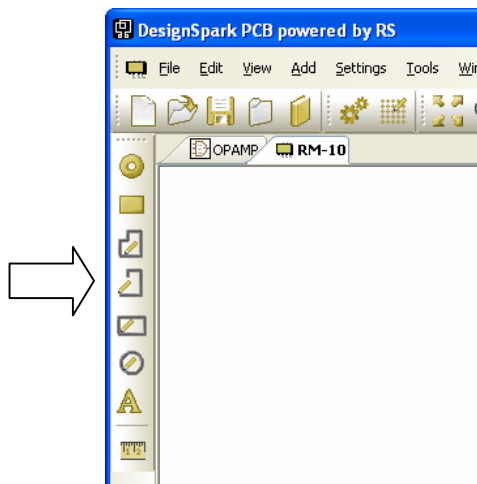
The footprint will update to reflect this new pad style. The example we've given below is a bit extreme as it will cause DRC errors if used but it clearly demonstrates the different pad style. You may require this facility to add a square pad on the first pad of an IC to denote pad 1 for example, this is useful where silkscreen is not being added to a board.



► Adding extra shapes

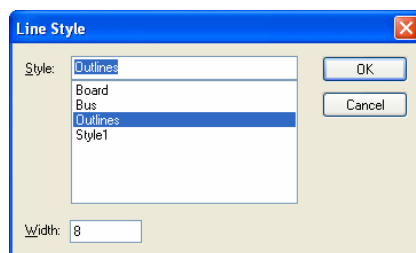
Shapes can be used to define any number of design items, such as the Silkscreen outline, the placement outline and any assembly outlines that may be required.

Use the Shapes provided on the **PCB Symbol** toolbar.



These shapes can also be added to specific layers within the design or layers swapped after addition (see below on changing layers of items).

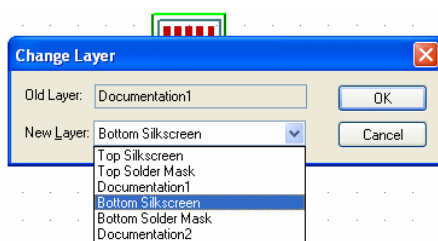
As with all items, the shape style can also be changed. To do this, select the shape and use the shortcut key **<S>** (**Change Style**) to modify it. This option is also available on the context menu if you right click on the selected shape.



► Change the layer of items

To change the layer of a shape or pad use the **Change Layer** option. This is located on the context menu or by using the shortcut key **<L>** for a selected item.

Layer changes for shapes



The **Old Layer** is displayed for reference. The **New Layer** can be selected from the drop down list. Click **OK** to accept the layer change.

Creating Surface Mounted Footprints

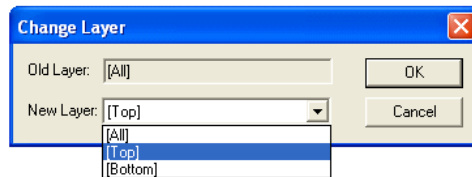
Surface Mounted pads of the PCB Symbol are easy to create. The only consideration needs to be for the side that the pads reside on. For normal pads, this would be the **[All]** layer. This means that the pad occupies all of the layers in the design. Whereas a surface mounted (SM) pad would only occupy the outer layers **[Top]** or **[Bottom]**.

All surface mounted pads are defined on the Top layer. When used in the PCB design, the component is flipped to the other side of the board (Bottom).

► Changing the layer of a pad

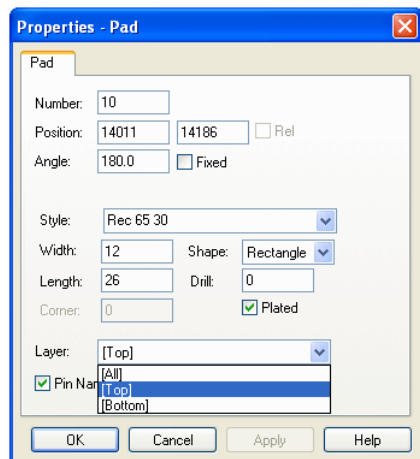
During the addition or once the pad has been added, the layer can be changed. You may also find that you need to change the pad style as well.

Select the pad(s) and right click. From the menu, select **Change Layer <L>**.



Select the **New Layer** from the drop down list. The **Old Layer:** box will display the current layer used.

An alternative method is to use the **Pad Properties**, also on the shortcut menu.



The **Layer:** drop down list will contain legal layers for which to swap your pad to.

You should be aware that there may also be a **Pad Style** change required, especially to note is a drill hole and plated status of the current pad style.

Click **OK** to accept the change.

The Reference and Symbol Origins

► Moving the Reference origin

The reference origin is used to position the Component Name when the symbol is used in the design. If using the Footprint Wizard, this is automatically assigned and positioned using controls presented.

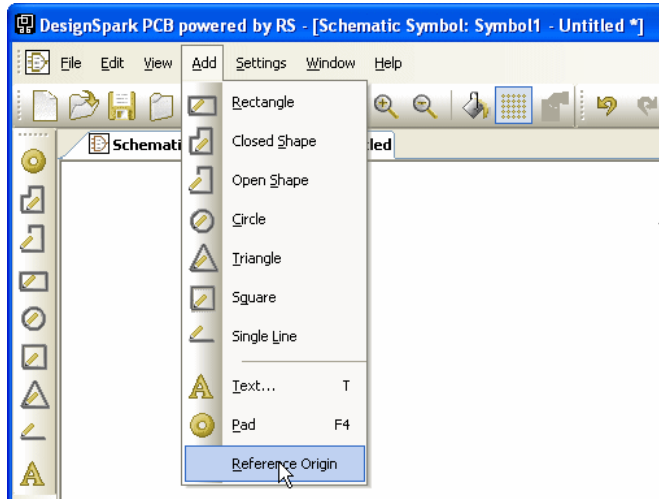
Once in the editor, you may reposition this origin as you like.

To move, simply pick and drag the name origin marker (**R**) to the position required.

If one isn't defined (because you created the footprint using the editor and not the wizard), the name will be located on the Symbol origin (**S**).

You can add this interactively as well.

From the **Add** menu select **Reference Origin**.



Position the reference origin at the lower right corner of the symbol, this can be anywhere.

► Moving the Symbol origin

The symbol origin is used when positioning the component in the design. If using the Footprint Wizard, this is automatically assigned and positioned using controls presented.

Once in the editor, you may reposition this origin as you like.

To move, simply pick and drag the name origin marker (**S**) to the position required.

By default, on through-hole components, this is on pad 1. On surface mounted devices, this will be the component center.

► Saving the symbol after editing

Once any changes have been made, you must save the symbol to the library.

From the **File** menu, select **Save To Library**.

Confirm the **Library:** name to save into and the **Symbol Name:**, click **OK** to accept these.

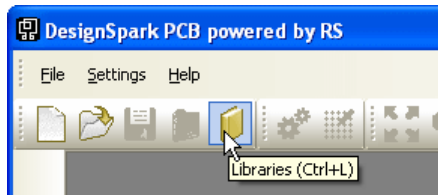
Chapter 4. Creating Components

Using the Component Editor

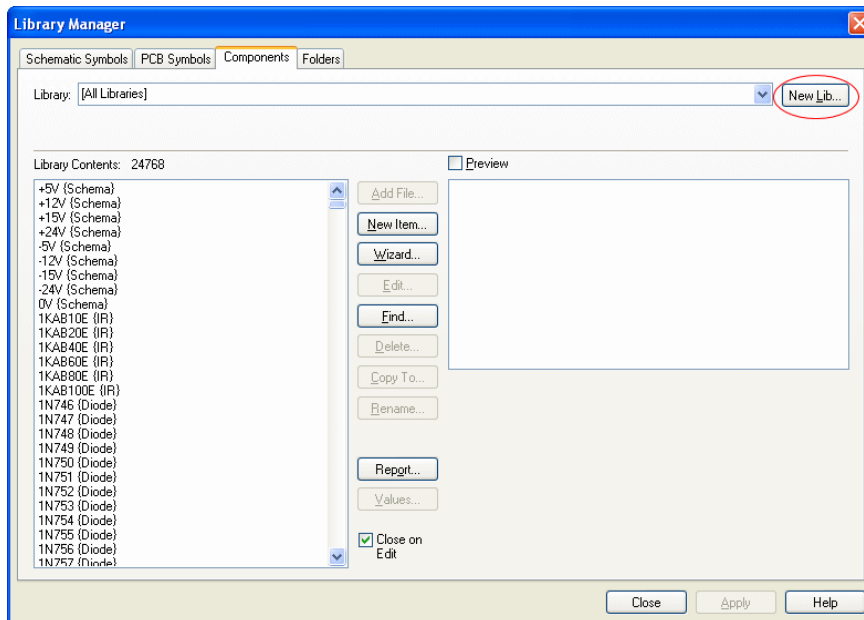
This tutorial uses the Schematic Symbol and PCB Symbol (Footprint) created in the previous tutorials, but equally applies to creating a component using other Schematic Symbols and PCB Symbols.

► To open the Component editor

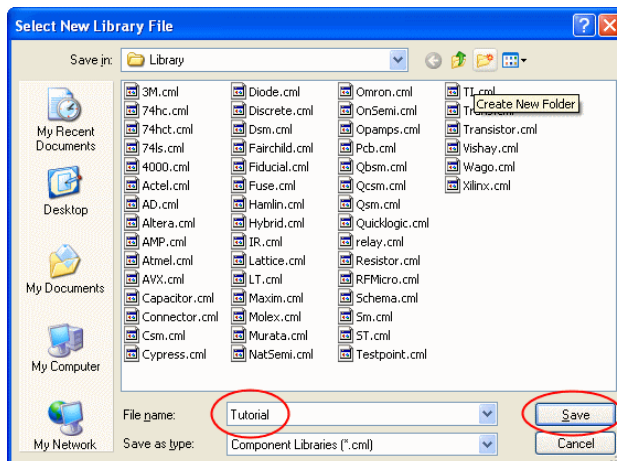
On the **File** toolbar click **Libraries <F11>** or **<Ctrl-L>**.



From the **Library** manager select the **Components** tab. For this tutorial, we will create a new library to add your new component to.

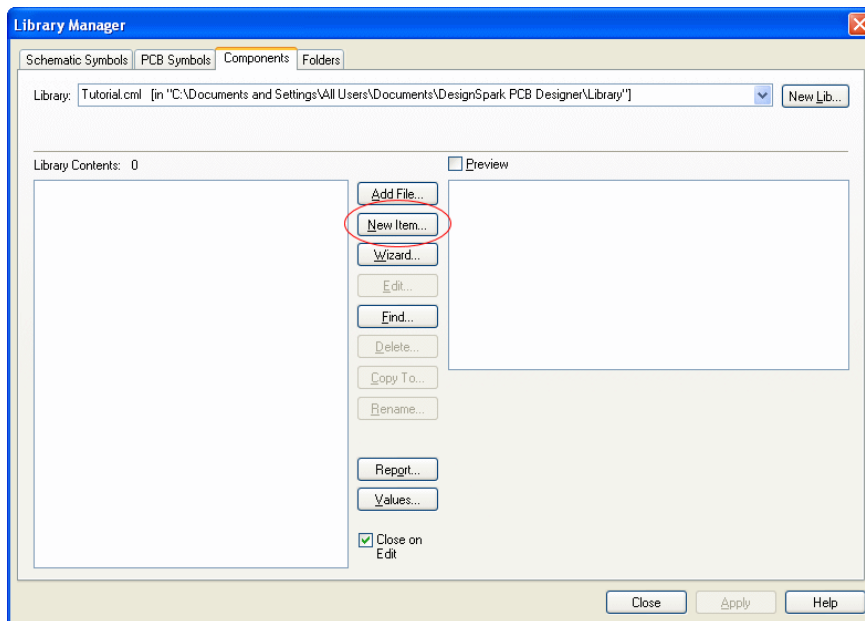


Click the **New Lib** button.



In the **File name** box, type **Tutorial** as your new component library.

Click **Save** to accept this. The library manager is now ready to start entering new components.



We need to create a new component so click on **New Item**.

The **New Component** dialog opens.

There are 8 pieces of information required:

In the **Component:** box type the component name **AD8592RM**

In the **Package:** box, type **SOIC**, this identifies the package type, multiple package types can be used for selecting alternative footprints. This doesn't appear in the list until you type it in.

In the **Default Reference:** box type **U**. This will be the component name used in the actual design (U1, IC1, R1, C1 etc.)

For **Schematic Symbol**, ensure that the library is set to **Tutorial.psl**.

Select your Schematic Symbol previously created **OPAMP** from the list.

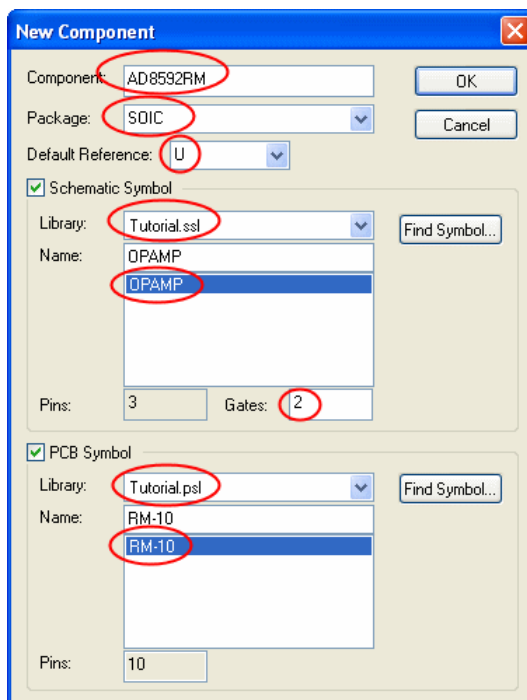
The dialog will automatically show **Pins:** as **3** (3 pins on the gate created) and **Gates:** as **1**. Change the **Gates:** to **2**. This will add 2x gates to the component.

For PCB Symbol, ensure that the library is set to **Tutorial.psl**.

Select your PCB Symbol previously created **RM-10**.

Ensure that the Pins box shows 10 to verify that your PCB Symbol is correct.

Your dialog will be complete and should look like the one below.

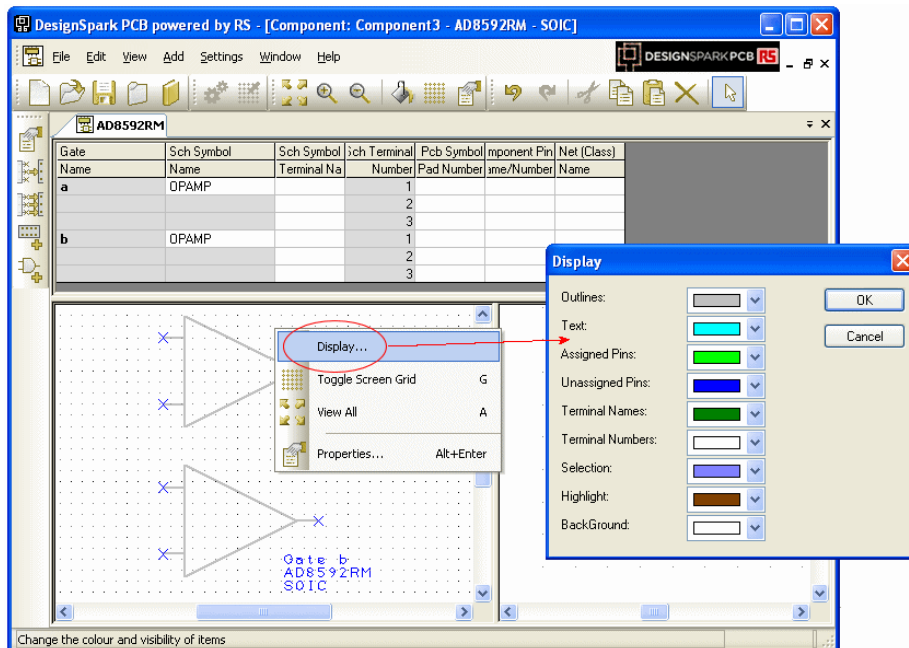


Once it does, click **OK** to continue.

The **Component Editor** opens showing the items added from the **New Component** dialog.

We've changed the colors of the preview images to make them easier to see in this document. Yours will still have the black background.

If you wish to change the colors, right click in the Symbol (or footprint) **Preview** window and select **Display** from the context menu. From the **Display** dialog you can change the colors as required. Click **OK** to accept the choice.



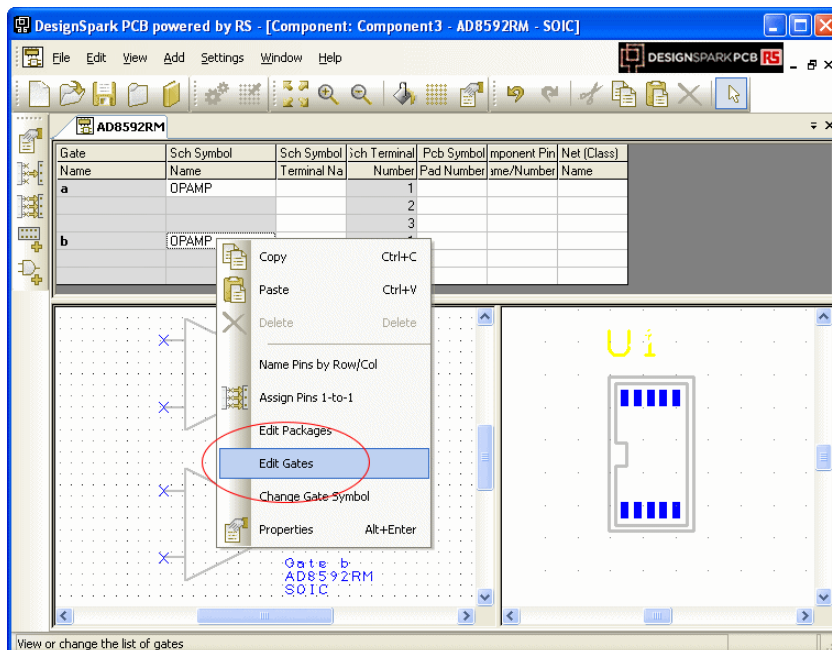
So far we've given the Component a **Name**, selected the name **default reference** name stem to be used and selected the Schematic and PCB Symbols to be used.

We now need to add more detail to the component to give it more intelligence.

► Adding an extra gate

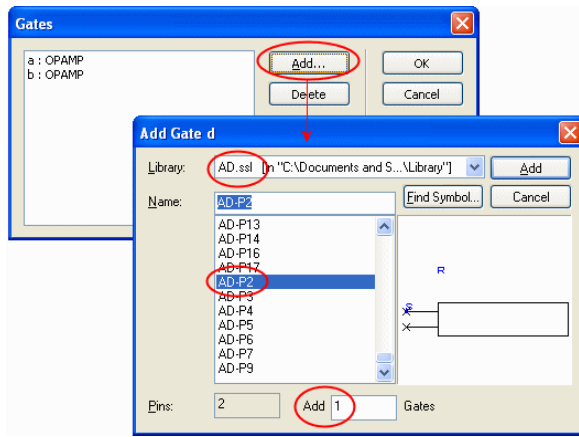
We need to add a power gate to the component. There are various methods in DesignSpark to adding power and ground to the component, for now this is one of the easiest.

Right click anywhere in the component grid. From the list, select **Edit Gates**.

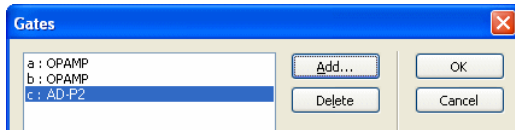


From the **Gates** dialog, select the **Add** button. Notice your existing 2x OPAMP gates are already shown in the list as gates a & b.

From the **Library**: list select the **AD.ssl** library.



Scroll down the **Name:** list and select the **AD-P2** gate symbol. Ensure that you only have **1** selected in the **Add 1 Gates** box. Click the **Add** button.



The Power gate will appear at the bottom of the list. Click **OK** to add it to the component.

Gate Name	Sch Symbol Name	Sch Symbol Terminal Name	Sch Terminal Number	Pcb Symbol Pad Number	Component Pin Name/Number	Net (Class) Name
a	OPAMP		1			
			2			
			3			
b	OPAMP		1			
			2			
			3			
c	AD-P2		1			
			2			

The power gate will now appear at the bottom (as the last gate) in the component grid and in the Schematic Symbol preview.

Now we need to add more detail to the component and to 'map' the pins used in the Schematic Symbol to those that will actually be used in the PCB Symbol.

► **Mapping pins to pads – why?**

For a Component to work, it needs to know how the pins defined in the Schematic Symbols relate to the pads used in the PCB Symbol. This mapping is presented in a 'grid' format in the component editor.

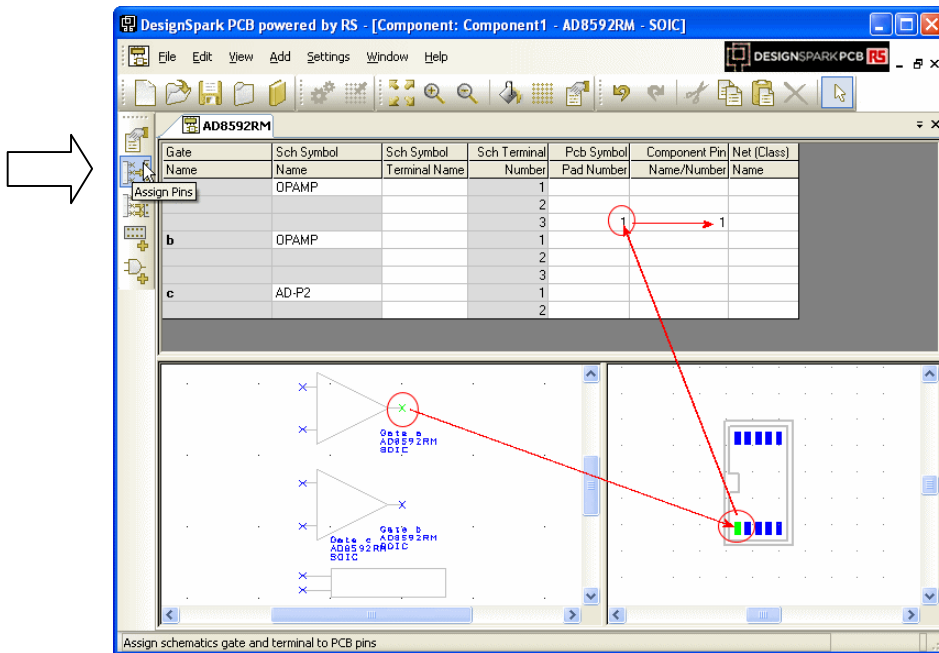
You can map the pin relationships manually or automatically by using the tools provided. On complex devices like a 256 pin BGA, the automatic method may be more preferable than the manual method which may be better for resistors for example.

You can use three methods: you can directly type the pad numbers into the respective grid cells or you can use the **Assign Pins** modes that let you click on the Schematic Symbol pin first and then the equivalent PCB pad. You also have the Assign Pins 1:1 which automatically assigns pins on a 1 to 1 basis, Schematic to PCB symbol.

► **To map pins to pads**

Generally speaking, you will use the **Assign Pins** method most times to do this task. This enables you to manually click on the Schematic symbol pin and then the corresponding PCB pin.

Click on the **Assign Pins** button on the Component toolbar.



The natural pin mapping of the Schematic Symbol and the PCB Symbol during creation will not be the same and will not necessarily be the correct sequence for mapping. PCB pins generally drive the mapping. If you consider the number sequence of the Schematic Symbol during creation, you could add them or number them to follow more closely the PCB pin sequence. It doesn't matter though that you don't or haven't for this example.

Select **Pin 3** on the **Schematic Symbol**.

Note, you can zoom in/out using the shortcut keys <Z>, <U> and <A> in the Symbol windows as well as rolling the mouse wheel.

Select **Pad 1** on the **PCB Symbol**.

The component grid will display the number **1** for **Sch Terminal Number 3** on gate **a**, it will also display a number **1** in the corresponding **Component Pin Name/Number** cell.

Once assigned, a number 1 will be shown on the Schematic Symbol output pin (pin 3). The color of the Schematic Symbol and PCB Symbol pins will also change color to reflect this. A different color will be shown for assigned and unassigned pins, and the currently selected pin.

The picture above shows how this has been completed for the first gate a and the second gate b is now ready to be assigned.

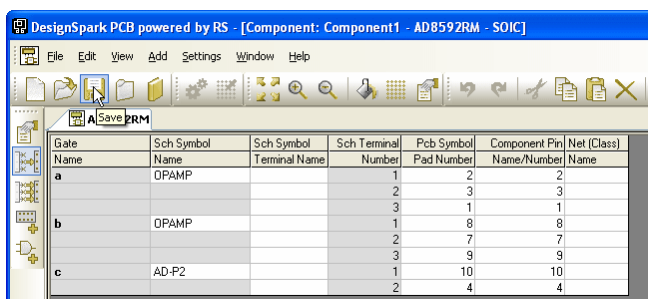
You should complete all pin to pad mapping so that looks like this:

Gate Name	Sch Symbol Name	Sch Symbol Terminal Name	Sch Terminal Number	Pcb Symbol Pad Number	Component Pin Name/Number	Net (Class) Name
a	OPAMP		1	2	2	
			2	3	3	
			3	1	1	
b	OPAMP		1	8	8	
			2	7	7	
			3	9	9	
c	AD-P2		1	10	10	
			2	4	4	

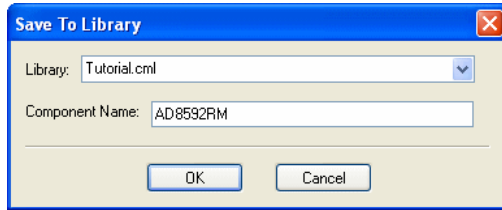
With the assignment of all pins between the Schematic and PCB symbols, this would be considered a minimum configuration to be able to save the component to the library and then use it in a design. Other information can be added and is discussed below after saving the component.

► Saving your Component

From the **File** toolbar, select **Save <Ctrl-S>**



The **Save To Library** dialog opens.



Choose the **Tutorial.cml** library from the list if not already selected.

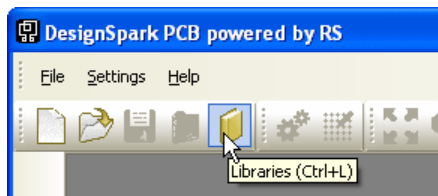
You entered the Component editor through the New Item dialog and have already given the component a name. This name will be displayed.

Click **OK** to confirm the library name and component name. You have now saved the Component in the Library.

The component will still be visible in the editor. We will edit it some more so keep it open for now.

► Viewing the Component in the library

From the **File** toolbar, select **Libraries**, <F11> or <Ctrl-L>

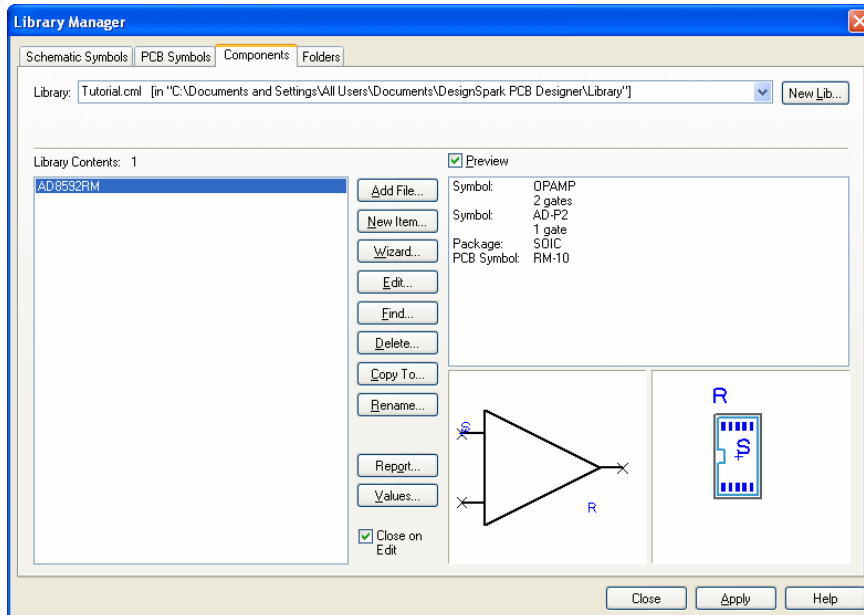


The **Libraries** dialog opens. Select the Components tab if it isn't already selected.

Make sure the **Preview** box is checked.

Select the **AD8592RM** Component that you created, it should look like the one below.

A quick check to whether you have created is correctly is the existence of both the Schematic and PCB Symbols in their respective Preview windows.



At this stage, you would add this component to a Schematic or PCB design and it would work.

Adding Logic Names

The Component has more value if Logic Names are shown. Optionally, you can still choose to switch these off if you need to.

► Adding Logic Names to your Component

To add or edit Logic Names to a Component in the editor (using our existing example which should still be open), type in the names required into the **Sch Symbol Terminal Names** field.

AD8592RM						
Gate	Sch Symbol	Sch Symbol	Sch Terminal	Pcb Symbol	Component Pin	Net (Class)
Name	Name	Terminal Name	Number	Pad Number	Name/Number	Name
a	OPAMP	-IN	1	2	2	
		+IN	2	3	3	
		OUT	3	1	1	
b	OPAMP	-IN	1	8	8	
		+IN	2	7	7	
		OUT	3	9	9	
c	AD-P2	V+	1	10	10	
		V-	2	4	4	

These names will be displayed on the Schematic Symbol in the design once the symbol has been saved.

Adding Component Values

Although your basic Component has been saved it can be enhanced further by the use of **Values**. Values are an important property of Components. They are the means by which, for instance, the value, rating, perhaps type or tolerance of a component (such as a resistor or capacitor) can be stored with the component, and displayed in a Schematic or PCB design. These values are also used during manufacture and we recommend using predefined value names shown below.

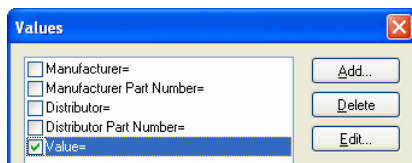
A further, and most important application is that values are the means by which part number and value information can be stored for inclusion in Parts lists.

Values can be added to a component in the design, or to a component in the library. Adding while in the component editor and storing it into the library means those values will be used each time that component is used. This means less editing and more consistency of values and value names across your designs.

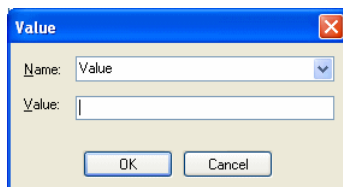
► Adding Values to your Component

To add or edit values to a component in the editor (using our existing example which should still be active, if not select it in the list in the Libraries dialog and click **Edit**), from the **Edit** menu select **Values**.

Click **Add** to create a new value.



From the **Add** dialog you can type the **Name** and **Value**: You do not have to add a value at this stage, the name can be left 'empty' to be fulfilled at a later stage or during design.



Click **OK** to confirm the selection.

The Value name is simply a 'label'. For Parts lists, it is important that consistent names are used.

We have allocated some value names which when used by you, will be passed through automatically into our system:

Manufacture, Manufacturer Part Number, Distributor, Distributor Part Number

Again, once these values have been added to a component, save the component to the library. This concludes the tutorial for the Libraries and Library Manager.