Altium.

Creating Library Components

Summary

Tutorial
TU0103 (v1.1) November 24, 2004

This tutorial covers the creation of schematic components and PCB footprints using the Schematic and PCB Library Editors. A glossary of terms used in this tutorial is included as an appendix.

This tutorial presumes you have a working understanding of the Schematic and PCB environments and are competent with placing and editing components. The example components and libraries used in this tutorial are available for you to look over in the `Altium2004\Examples\Tutorials\Creating Components` folder.

In this tutorial, we will cover the following topics:

- creating new libraries
- creating schematic components with single and multiple parts
- checking the components using Schematic Library Editor reports
- creating PCB component footprints manually and using the Component Wizard
- checking the components using PCB Library Editor reports
- creating an integrated library of the new components and models.

Creating schematic components

The Schematic Library Editor is provided to create and modify schematic components and manage component libraries. It is similar to the Schematic Editor and shares the same graphical objects, with the addition of the Place Pin tool. Schematic components can be made up of parts that are separately selectable and are associated with their nominated PCB footprint components, which are stored in the PCB libraries or integrated libraries, during synchronization.

Components can be created in the Schematic Editor and copied and pasted into an open schematic library to create a new component, or you can use the drawing tools in the Schematic Library Editor.
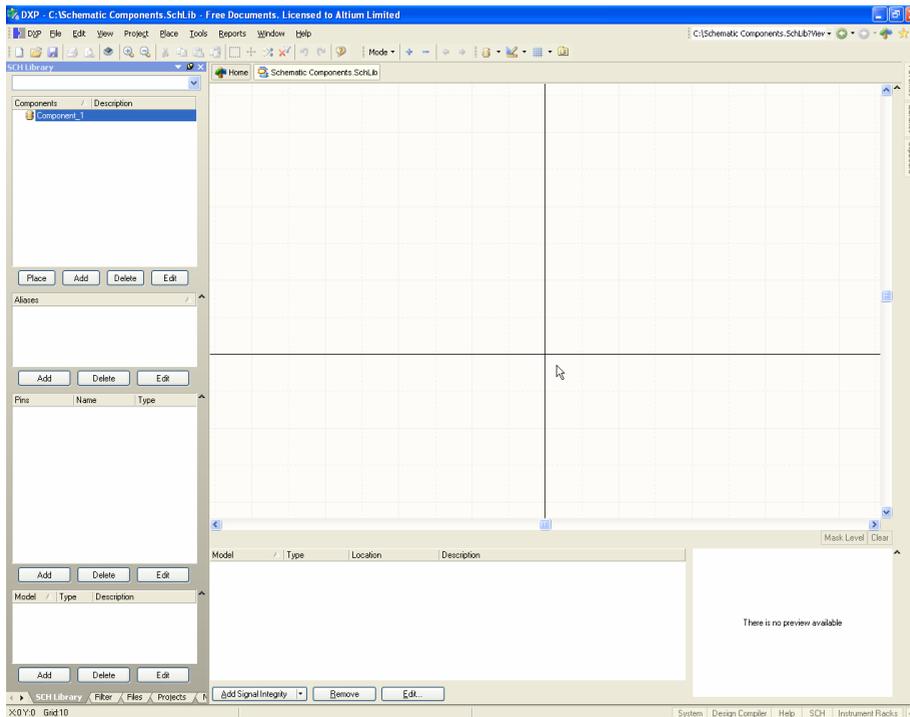
Schematic libraries

Schematic libraries (`.SchLib`) are included as the essential part of the integrated libraries (`.IntLib`) supplied in the `Altium2004\Library` folder. To create a schematic library out of an integrated library, open the integrated library and choose **Yes** to extract the source libraries that can then be opened for editing. For more information, refer to the *Building an Integrated Library* tutorial.

You can also create a schematic library of all the components that have been placed in the schematic documents of the active project using the **Design » Make Project Library** command.

Creating a new schematic library

Before we start creating components, let's make a new schematic library to store them in. To create a new schematic library, complete the following steps.

1. Select **File » New** » **Schematic Library**. A new library, named `Schlib1.SchLib`, is created and an empty component sheet, Component_1, displays in the design window.

2. Rename the library file to `Schematic Components.SchLib`, for example, by selecting **File » Save As**. Browse to the folder required, rename the file with a `.SchLib` extension and click **Save**.

3. Click on the **SCH Library** tab to open the SCH Library panel.

Creating a new schematic component

To create a new schematic component in an existing library, you would normally select **Tools » New Component**, but since a new library always contains one empty component sheet, we'll simply rename Component_1 to get started on creating our first component, an NPN transistor.

1. Select `Component_1` from the Components list in the SCH Library panel and select **Tools » Rename Component**. Type the new component name that uniquely identifies it, e.g. `TRANSISTOR NPN`, in the *Rename Component* dialog and click **OK**.

2. If necessary, relocate the origin of the sheet to the center of the design window
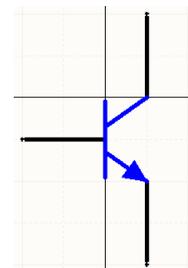
NPN component

by selecting **Edit » Jump » Origin** [shortcut J, O]. Check the Status line at the bottom left of thtdtcreen to confirm that you have the cursor at the origin. Components supplied by Altium are created around this point, marked with a crosshair through the center of the sheet. The reference point of a component is the point you will be "holding" the component by when you place it. For a schematic component, this point is the nearest electrical 'hotspot' to the origin, usually the electrical end of the nearest pin.

3. Set the Snap grid to 1 and the Visible grid to 10 in the *Library Editor Workspace* dialog by selecting **Tools » Document Options** [shortcut T,D].
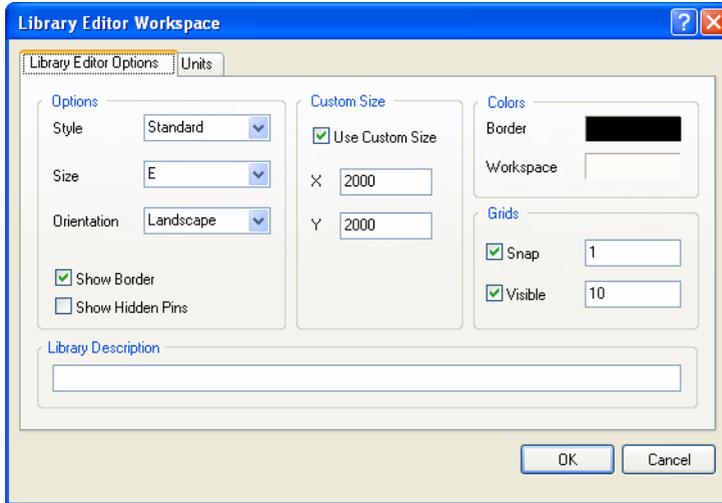
Press **G** to quickly toggle through and set the Snap Grid to 1, 5 or 10 units.

Click on the **Units** tab to set Imperial or Metric as the base Unit system for the library. We will use the **DXP Defaults** unit system (10mils) and will accept the other Library Editor default settings by clicking **OK**. If the grid is not visible, press **Page Up** to display it.

4. To create the NPN example above, we will first define the component body. Select **Place » Line** [shortcut P, L], or click on the **Place Line** toolbar button (Utilities toolbar). Press the **TAB** key to set the line properties in the *PolyLine* dialog and click **OK**. Click to start the vertical line at 0, -1 and finish at 0,-19. Right-click to place the line. Now create the other two lines from 0,-7 to 10, 0, and 0,-13 to 10, -20, using **Shift+SPACEBAR** to set the modes of these lines to Any Angle. Right-click or press **ESC** to end line placement mode.

Body of NPN

5. The arrow is created out of a closed polygon. Select **Place » Polygon** [shortcut P, Y] or click on the **Place Polygon** toolbar button (Utilities toolbar). Press the **TAB** key to set the polygon properties in the *Polygon* dialog. Set the Border Width to **Smallest,** enable **Draw Solid** and set the fill and border colors to the same color and click **OK**. Click to form the vertices of a triangle and right-click to end. Right-click or press **ESC** to end polygon placement mode.

6. Save the component [shortcut Ctrl+S].
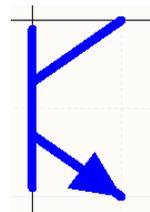
Adding pins to a schematic component

Component pins give a component its electrical properties and define connection points on the component. They also have graphical properties.

To place pins on a component in the Schematic Library Editor:

1. Select **Place » Pin** [shortcut P, P] or click on the ⟋ toolbar button. The pin appears floating on the cursor, held by the non-electrical end that goes against the component body.

2. Before placing the pin, press the **TAB** key during placement to edit the pin's properties. The *Pin Properties* dialog displays. If you define the pin attributes before you place it, the settings you define will become the defaults and the pin numbers and any numeric pin names will auto-increment when you place them.
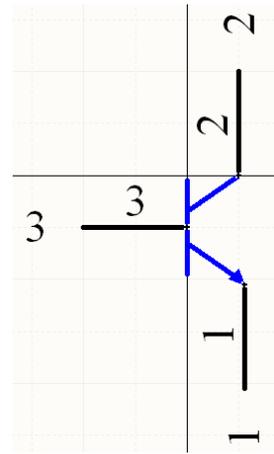
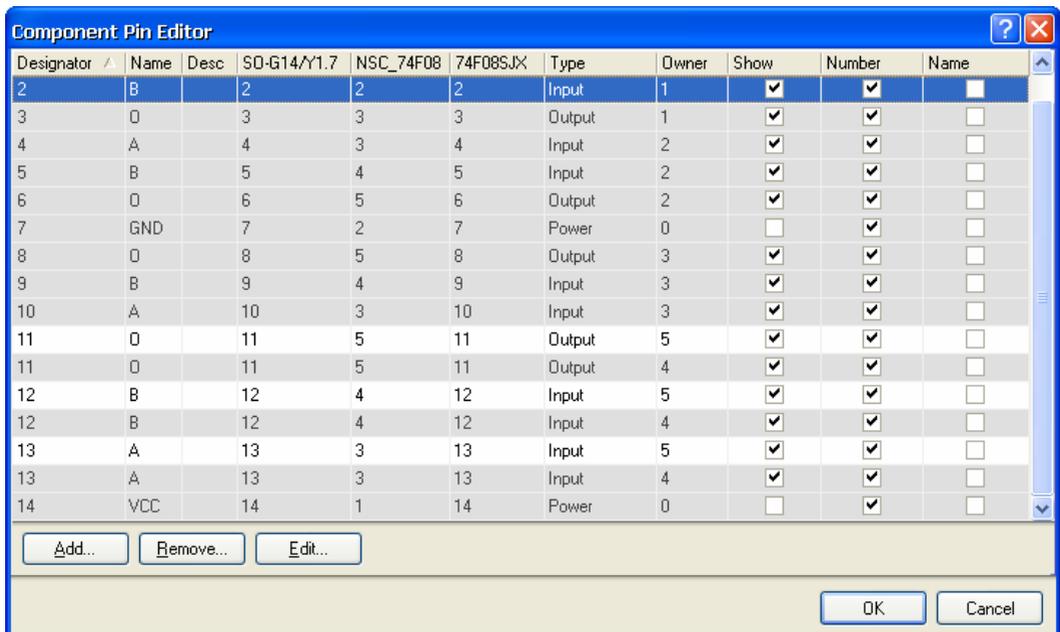*NPN symbol with Pin Name and Number visible*
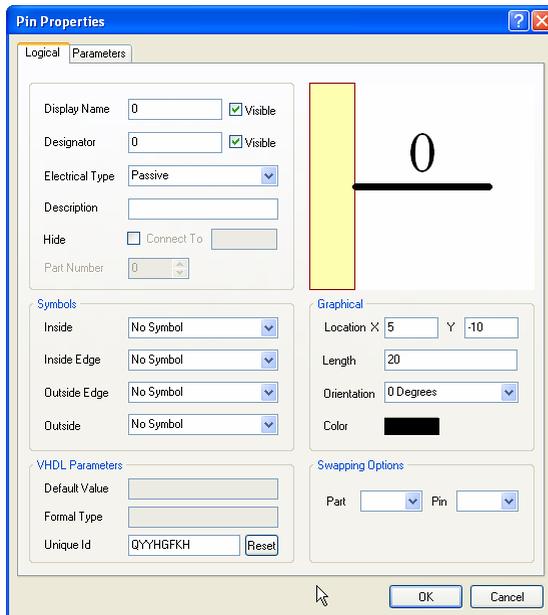
If you wish to alter the distance (in hundredths of an inch) between the pin name or number and the body of the component, select **Tools » Schematic Preferences** and change the **Pin Margin** options in the **Schematic — General** page.

3. In the *Pin Properties* dialog, type in a pin name in the Display Name field, e.g. 1, and a unique pin number, e.g. 1, in the Designator field. Click on the **Visible** checkboxes if you want the pin name and designator visible when you place the component on a schematic sheet.

4. Set the **Electrical Type** to represent the type of electrical connection the pin makes by choosing an option from the drop-down list. This type can be used by the Electrical Rules Checker when compiling a project or analyzing a schematic document to detect electrical wiring errors in a schematic sheet. In this component example, all pins are a Passive electrical type.

5. Set the length of this pin in hundredths of an inch in the Length field. All pins in this component are set to 20 and click **OK**.

6. Press the **SPACEBAR** to rotate the pin in 90º increments while it is floating on the cursor. Remember that only one end of a pin is electrical and you must place the pins with this end out from the component body. The non-electrical end of the pin has the pin name next to it.

7. Continue to add the pins required to finish the component, making sure the pin names, numbers, symbols and electrical types are correct as shown on the NPN symbol above.

8. You have now finished drawing your component, so select **File » Save** [shortcut Ctrl+S].

Notes on adding pins

- To set pin properties after placing the pin, double-click on the pin, or double-click on the pin in the Pins list in the SCH Library panel to open the *Pin Properties* dialog.

- Use a backslash (\) after a letter to define an overscored letter in a pin name, e.g. M\C\L\R\/VPP will display as $\overline{\text{MCLR}}/\overline{\text{VPP}}$.

- If you wish to hide the power and ground pins on a component, click on the **Hide** checkbox. When they are hidden, these pins will be connected to power and ground nets as specified in the Connect To field, e.g. the pin for VCC will connect to net VCC when placed.

- To view hidden pins, select **View » Show Hidden Pins**. Any hidden pins will be displayed in the design window. The display name and default designator will display as well.

- You can edit pin properties directly in the *Component Pin Editor* dialog, without having to edit each pin through its corresponding *Pin Properties* dialog. Click on **Edit Pins** in the *Library Component Properties* dialog (double-click on a component name in the SCH Library panel to open) to display the *Component Pin Editor* dialog.

Designator	Name	Desc	SO-G14/Y1.7	NSC_74F08	74F08SJX	Type	Owner	Show	Number	Name
2	B		2	2	2	Input	1	✔	✔	☐
3	O		3	3	3	Output	1	✔	✔	☐
4	A		4	3	4	Input	2	✔	✔	☐
5	B		5	4	5	Input	2	✔	✔	☐
6	O		6	5	6	Output	2	✔	✔	☐
7	GND		7	2	7	Power	0	☐	✔	☐
8	O		8	5	8	Output	3	✔	✔	☐
9	B		9	4	9	Input	3	✔	✔	☐
10	A		10	3	10	Input	3	✔	✔	☐
11	O		11	5	11	Output	5	✔	✔	☐
11	O		11	5	11	Output	4	✔	✔	☐
12	B		12	4	12	Input	5	✔	✔	☐
12	B		12	4	12	Input	4	✔	✔	☐
13	A		13	3	13	Input	5	✔	✔	☐
13	A		13	3	13	Input	4	✔	✔	☐
14	VCC		14	1	14	Power	0	☐	✔	☐

- For a multi-part component, the relevant pins for the selected part will be highlighted with a white background in the *Component Pin Editor* dialog. All pins of other parts are grayed. You are,

however, still able to edit the pins of these non-selected parts. Select a pin and click **Edit** to display the *Pin Properties* dialog for that pin.

Setting the schematic component's properties

Each component has properties associated with it such as the default designator, the PCB footprint and/or other models and parameters of the component. You can also set up the various part fields and library fields that are displayed when the component properties are edited from a schematic sheet. To set the component's properties:

1. Select the component in the Components list of the SCH Library panel and click on the **Edit** button, or simply double-click on the component name. The *Library Component Properties* dialog displays.
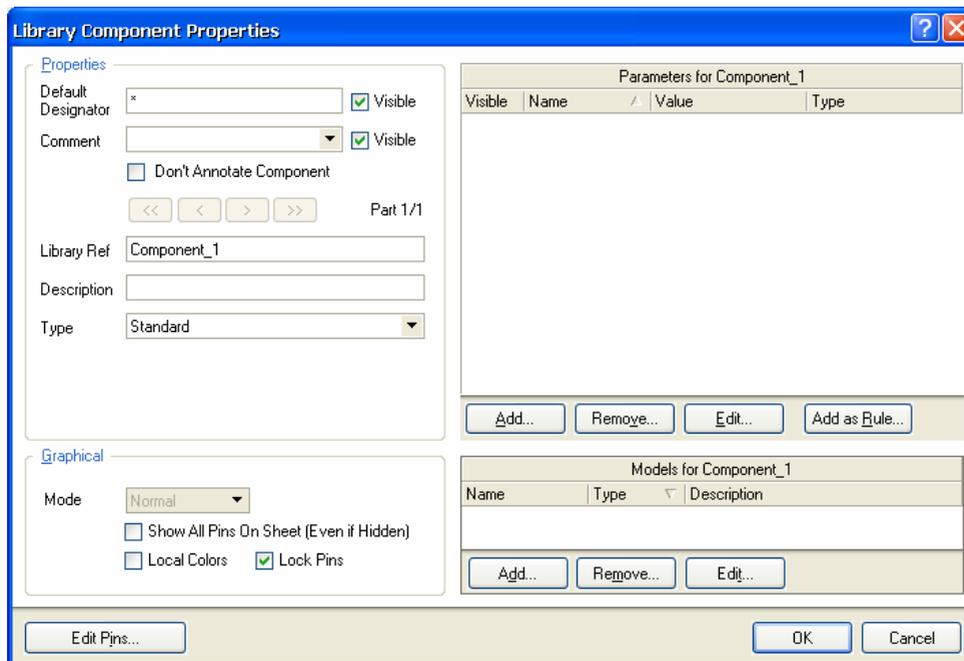
2. Type in the default Designator, e.g. `Q?` and a comment that will display when the component is placed on a schematic sheet, e.g. `NPN`. Using the question mark will allow the designator number to auto-increment on placement, e.g. Q1, Q2. Make sure the **Visible** boxes are checked.

3. Leave the other fields at their default values while we add models and parameters, as required.

Adding models to the schematic components

You can add any number of PCB footprint models to be associated with a schematic component, as well as model files that are used for simulation and signal integrity analyses. You can then select the appropriate models from the *Component Properties* dialog when you place the component in a schematic document.

There are several different ways to add models to a component. You may wish to download a vendor's model file from the web or add models from the existing Altium libraries. PCB footprint models are located in the `C:\Program Files\Altium2004\Library\Pcb` folder in the form of PCB libraries (`.pcblib` files). SPICE models used for circuit simulation (`.ckt` and `.mdl` files) are located within the integrated libraries in the `C:\Program Files\Altium2004\Library` folder. You can also use the XSpice Model Wizard to create Spice models to add to the component (**Tools » XSpice Model Wizard**).

The Model Manager allows you to organize your component models in one go, for example to add the same model to multiple, selected components. To access the Model Manager, right-click on a component name in the SCH Library panel and select **Model Manager**. Alternatively, select **Tools » Model Manager**.

Alternatively, you can add models from the SCH Library panel by clicking on the **Add** button for a selected component, or from the Models View in the Schematic Library Editor (click on the ⌄ arrow located at the lower right side of the library workspace to show the models list).

Search locations for model files

When a model is added to the component in the Schematic Library Editor, the link from the component to the model information is resolved by the following valid search locations:

1. Libraries that are included in the current library package project are searched first.

2. PCB libraries (but not integrated libraries) that are available from the currently Installed Libraries list are searched next. Note that the list of libraries can be ordered.

3. Finally, any model libraries that are located down the Project search paths are searched. These paths are defined in the *Options for Project* dialog (**Project » Project Options**). Note that libraries that are down the search path cannot be browsed to locate a model, however, the compiler does include them when searching for a model.

Please refer to the *Components, Models and Library Concepts* article for more information about the way models are searched for in the Schematic Library Editor and the Schematic Editor.

In this tutorial, we will use the first method of linking the components and its model files, i.e. by adding all necessary model files to the library package project, along with the schematic library, before compiling the library package into an integrated library.

Adding footprint models to a schematic component

First, we will add a footprint that the schematic component will use when synchronized to a PCB document. The footprint required for our schematic component is named `BCY-W3`.

Note that when linking a PCB footprint model to a schematic component in the Schematic Library Editor, the model must exist in a PCB library, not an integrated library.

1. Click on **Add** in the Models section of the *Library Component Properties* dialog. The *Add New Model* dialog displays.
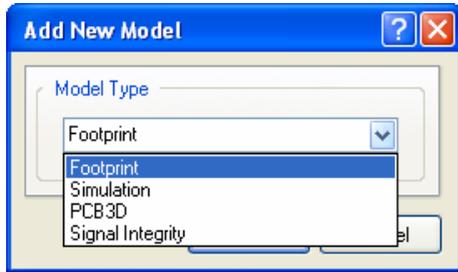
2. Select **Footprint** from the Model Type drop-down list. Click **OK**. The *PCB Model* dialog displays.

3. Click on the **Browse** button to find a model that already exists from the *Browse Libraries* dialog (or simply type in the name of a model that you are going to create later using the PCB Library Editor).

4. When in the *Browse Libraries* dialog, click on **Find**. The *Libraries Search* dialog displays.

5. Select the scope of **Libraries on Path** and browse to the `\Altium2004\Library\Pcb` folder by clicking on the **Browse Folder** 📁 button next to the Path field. Make sure **Include Subdirectories** is selected in the *Libraries Search* dialog.

6. In the query field at the top of the dialog, type '`BCY-W3`' and click on **Search**.
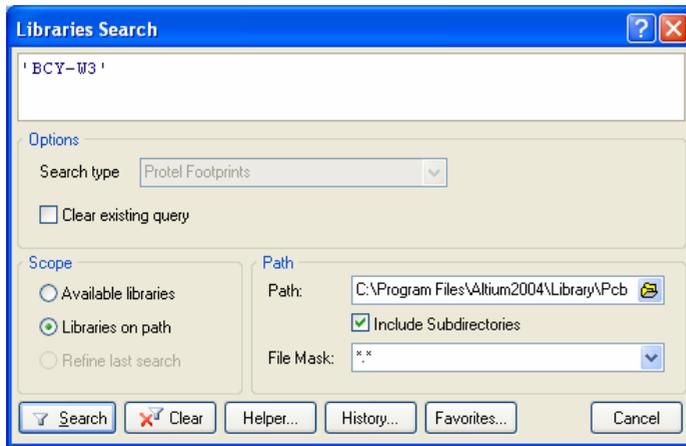
7. You should get one result from a PCB library (`Cylinder with Flat Index.PcbLib`) displayed in the *Browse Libraries* dialog. Select `BCY-W3` from this library in the *Browse Libraries* dialog and click **OK** to return to the *PCB Model* dialog.

8. Since this is the first time you have used this library, you will be asked to confirm the installation of this library which will make it available for use. Click on **Yes**.
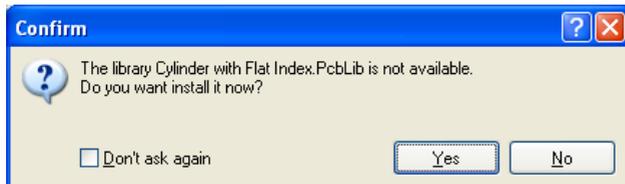
The *PCB Model* dialog is updated with the footprint model information.
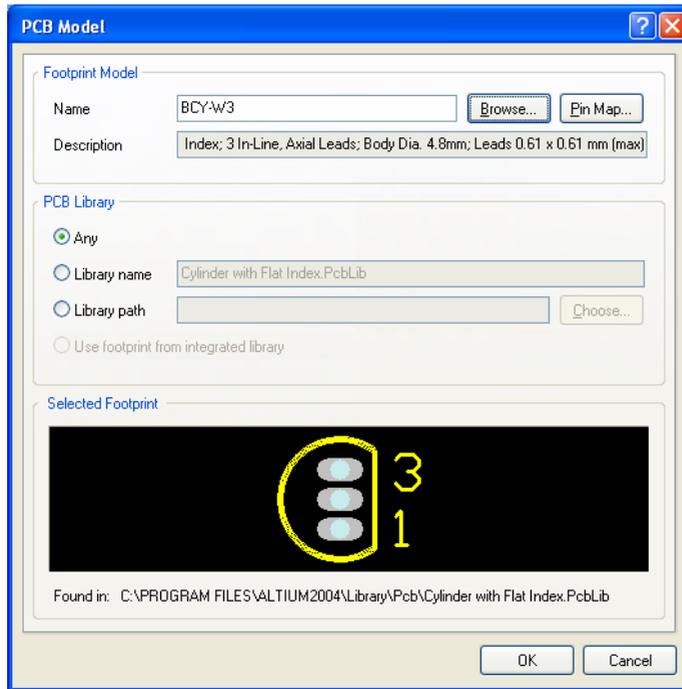
8. Click **OK** to add the model. The model name is included in the Models list of the *Library Component Properties* dialog.


Adding Circuit Simulation models

SPICE models used for circuit simulation (`.ckt` and `.mdl` files) are located within the integrated libraries in the `C:\Program Files\Altium2004\Library` folder. Add these models if you wish to run analyses on your design.

It is recommended that if you use these simulation models in your library components, open the `.Intlib` file that contains the required models (**File » Open** and confirm you want to extract the source libraries). Then copy the model files from the output folder (generated when you opened the integrated library) into the folder that contains your source libraries.

1. Click on **Add** in the Models section of the *Library Component Properties* dialog. The *Add New Model* dialog displays. Select **Simulation** from the Model Type drop-down list and click **OK**. The *SIM Model – General / Generic Editor* dialog displays.
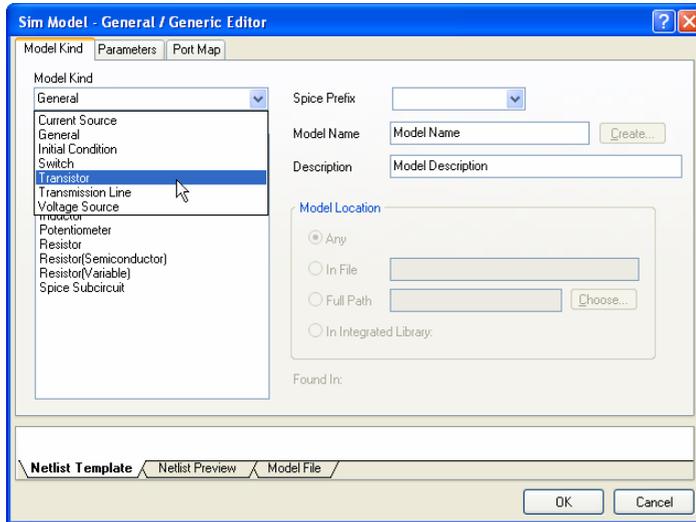
2. For this example, choose a Model Kind of **Transistor** from the Model Kind drop-down list. The *Sim Model – Transistor/BJT* dialog displays.
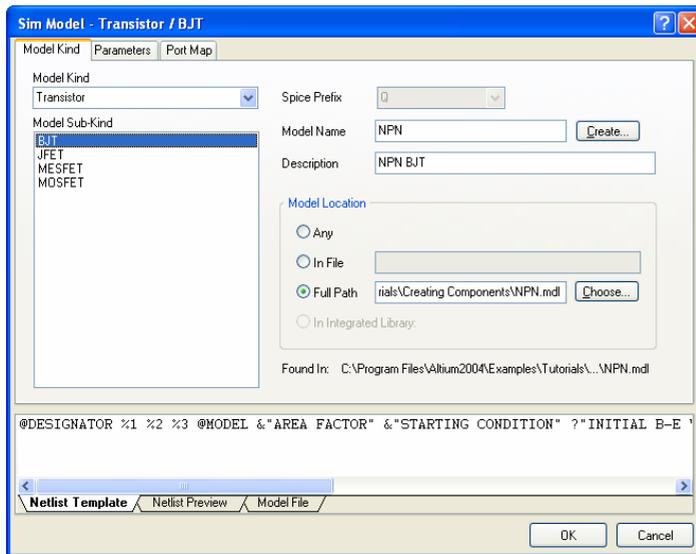

> The Model Name is the crucial link to the SIM model file, so make sure it is a valid model name.

3. Make sure **BJT** is selected as the Model Sub-Kind. Enter a valid Model Name, e.g. NPN (for model file NPN.mdl), and a description, e.g. NPN BJT. Set the Model Location to **Full Path** and click on **Choose** to select the model filename NPN.mdl located in C:\Program Files\Altium2004\ Examples\Tutorials\Creating Components and click **Open**.

4. If you do not have a model file (.mdl), you could click on the **Create** button to run the Spice Model Wizard that will step you through the creation of a simulation model file for your component.

5. Click **OK** to return to the *Library Component Properties* dialog where the model NPN has been added to the Models list.

Adding Signal Integrity models

The Signal Integrity simulator uses pin models rather than component models. To configure a component for signal integrity simulation, you either set the Type and Technology options, which will use default pin models, or import an IBIS model.

1. To add a Signal Integrity model, click on **Add** in the Models List section of the *Library Component Properties* dialog. The *Add New Model* dialog displays.

2. Select **Signal Integrity** from the Model Type drop-down list and click **OK**. The *Signal Integrity Model* dialog displays.
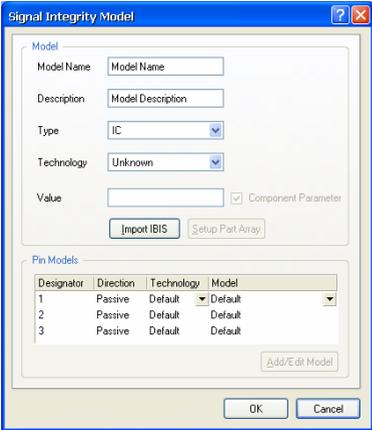
3. If you wish to import an IBIS file, click on the **Import IBIS** button and navigate to the required `.ibs` file. For this example however, type in the Model Name and Description of `NPN` (for example) and select a Type of **BJT**. Click **OK** to return to the *Library Component Properties* dialog where the model is added to the Models list.

For more information about adding and editing Signal Integrity models, refer to the *Performing Signal Integrity Analyses* tutorial.

Adding component parameters

Parameters are a means of defining additional information about the component. Parameters, such as strings that identify data like the component manufacturer or the date, can be added to the document. A string parameter could also be added for the component's value, where applicable (e.g. 100K for a resistor).

Parameters can be set to display as special strings when components are placed on a schematic sheet. Other parameters are required as values for simulation or can be used as PCB rules created in the Schematic Editor. To add a parameter to a schematic component:

1. Click **Add** in the Parameters list section of the *Library Component Properties* dialog to display the *Parameter Properties* dialog.
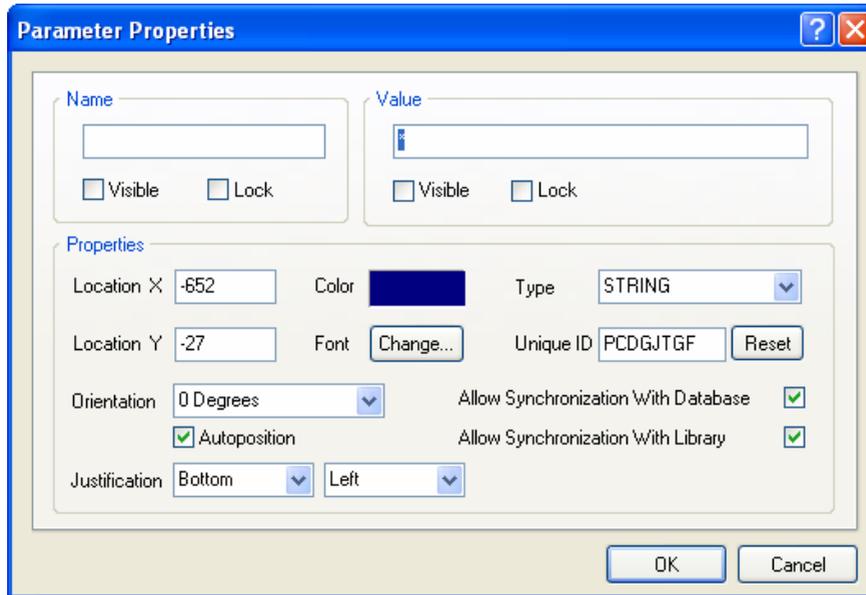
2. Enter the name of the parameter and a value. Make sure **String** is selected as the parameter type if you require a text string and the value's **Visible** box is ticked if you want the value to display when the component is placed on a schematic sheet. Click **OK**. The parameter is added to the Parameters list in the *Library Component Properties* dialog.

Indirection strings

Using 'indirection' strings, you can set up a parameter field for a component that will display on the schematic and can also be used by DXP when running a circuit simulation, for example. Any added component parameter can be used as an indirection string. The parameter name has a prefix of '= ' attached to it when it is used as an indirection string.

Value parameter

The Value parameter can be used for any general information about the component, but discrete components, such as resistors and capacitors, use it when simulating.

We can set the component's Comment, using indirection strings, to read the value from an added Value parameter that is then mapped to the Comment information in the PCB Editor. Rather than enter the value twice (i.e. in the parameter named Value and then again in the Comment field), DXP supports 'indirection' which will replace the contents of the Comments field with the Value parameter's string.

1. Click **Add** in the Parameters list section of the *Library Component Properties* dialog to display the *Parameter Properties* dialog.
2. Enter the name `Value` and a value of `100k`. This value will be used when you run a circuit simulation on a schematic sheet with this component placed on it. Make sure **String** is selected as the parameter type and the value's **Visible** box is ticked. Set up any font, color or orientation

options and click **OK** to add the new parameter to the Parameters list in the *Library Component Properties* dialog.

3. In the **Properties** section of the *Library Component Properties* dialog, click on the **Comment** field and select the **=Value** string from the drop-down list and turn **Visible** off.
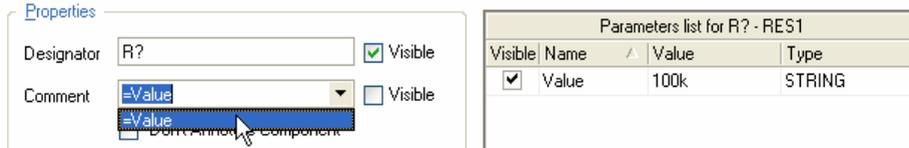

4. Save the component sheet and its properties in the library by selecting **File » Save** [shortcut Ctrl+S].

5. When viewing special strings in the Schematic Editor, make sure the **Convert Special Strings** option is enabled in the **Schematic – Graphical Editing** page of the *Preferences* dialog (**Tools » Schematic Preferences**). If the Comment does not display in the PCB document after updating the PCB from the schematic document, make sure that the Comment is unhidden in the footprint's *Components* dialog.

Creating a new schematic component with multiple parts

In this next part of the tutorial, we will create a new component with four parts, a Quad 2-IN AND gate, named 74F08SJX. We will also create an alternate mode view using an IEEE symbol as an example.

1. Select **Tools » New Component** [shortcut T, C] when in the Schematic Library Editor. The *New Component Name* dialog displays.
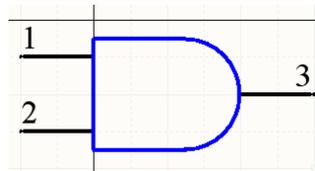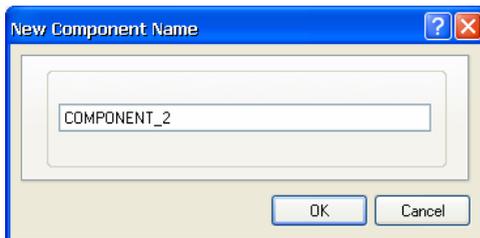
Part A of component 74F08SJX

2. Type in the name of the new component, e.g. `74F08SJX`, and click **OK**. The new component name displays in the Components list in the SCH Library panel and an empty component sheet displays with a crosshair through the center (origin) of the sheet.

3. Now we will create the first part of the new component as shown above, including its pins, as detailed in the following sections. The first part will be used as the basis for the other parts as only the pin numbers will change in this example.

Creating the body of the component

The body of this component is constructed from a multi-segment line and a circular arc. Make sure the component sheet origin is in the center of the workspace by selecting **Edit » Jump » Origin** [shortcut J, O]. Also, make sure the grid is visible [shortcut Page Up].

Placing lines

1. Select **Place » Line** [shortcut P, L] or click on the ╱ toolbar button. The cursor changes to a crosshair and you are now in multi-segment line placement mode.

2. Press the **TAB** key to set the line's properties. Set the line width to Small in the *Polyline* dialog.

3. Left-click or press **ENTER** to anchor the starting point for the line at 25, -5. Check the X, Y co-ordinates in the Status bar at the bottom left corner of the Design Explorer. Then position the mouse and left-click to anchor a series of vertex points that define the segments of the line (at 0,-5; 0,-35 and 25,-35).

4. When you have finished drawing the line, right-click or press **ESC**. Right-click or press **ESC** to exit line placement mode. Save the component.
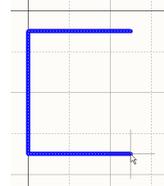
Drawing an arc

Placing an arc is a four-step process that sets the center point, radius, start point and end point of the arc. You can press **Enter** instead of left-click to place the arc.
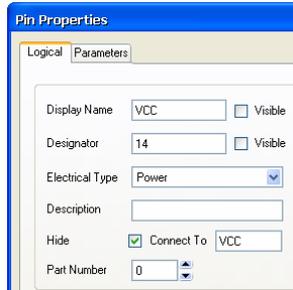
1. Select **Place » Arc (Center)** [shortcut P, A]. The last arc drawn appears on the cursor and you are now in arc placement mode.

2. Press the **TAB** key to set the arc's properties. The *Arc* dialog displays. Set the radius to 15mils and the line width to 10mils.
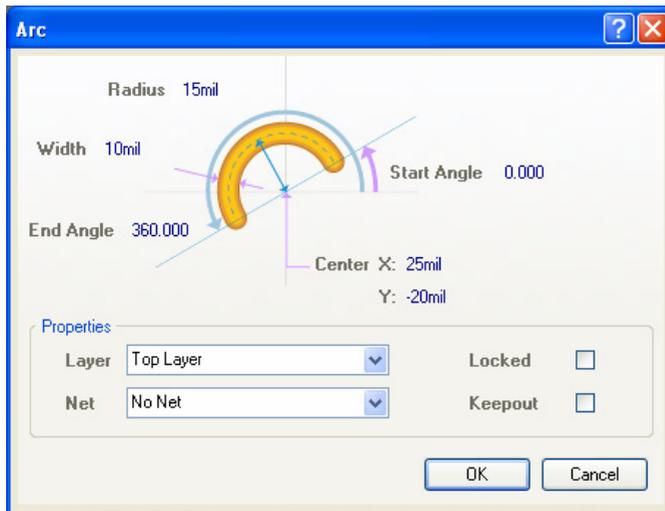
3. Position the cursor and left-click to anchor the center point of the arc (25, -20). The cursor will then jump to the current default arc radius (15mils) which we have already set in the *Arc* dialog.

4. Click to set the radius. The cursor will then jump to the start point of the arc.

5. Move the cursor to adjust the start point, then left-click to anchor it. The cursor will then jump to the end point of the arc. Move the cursor to change the position of the end point, then left-click to anchor it and complete the arc.

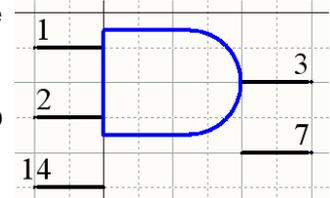6. Right-click, or press **ESC**, to exit arc placement mode.

Adding pins

Add the pins to the first part, using the same technique described in the section *Adding pins to a schematic component* earlier in this tutorial. The pins 1 and 2 are Input electrical types and 3 is Output.

Part A with power pins unhidden

The power pins are hidden pins, i.e. GND (pin number 7) and VCC (pin number 14). They will remain constant for all parts and so only have to be set once as belonging to Part zero (Part 0).

Part zero is simply a place to store pins that are common to all parts in a component and the pins in this part will be added to each part when it is placed in the schematic. In the **Logical** tab of the *Pin Properties* dialog for these power pins, make sure they are set to Part 0 in the Part Number field, the Electrical Type is 'Power', the **Hide** box is checked and the power pin connects to the correct net name, e.g. VCC (pin number 14) connects to VCC as entered in the Connect To field.

Creating the new parts

1. Select the component using **Edit » Select » All** [shortcut Ctrl+A].

2. Select **Edit » Copy** [shortcut Ctrl+C]. The cursor will change to a crosshair. Click on origin or the top left corner of the component body to set the copy reference point (that is where the cursor will be 'holding' the selection when you paste it) and copy the selected objects to the clipboard.

3. Select **Tools » New Part**. A blank component sheet displays. The Part counter in the SCH Library panel will be updated to include Part A and Part B if you click on the + to the right of the component name in the Components list in the SCH Library panel.

4. Select **Edit » Paste** [shortcut Ctrl+V]. The cursor appears with an outline of the component part attached at its reference point. Move the copied part until it is positioned the same as the original part. Click to paste the copied part.

5. Update the pin information in the new part, Part B, by double-clicking on each pin and changing the pin name and number in the *Pin Properties* dialog.
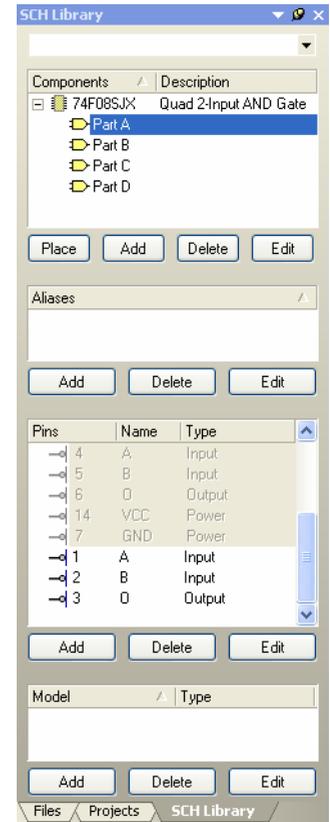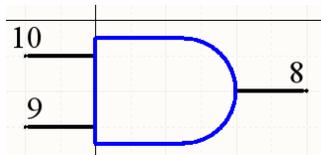
Alternatively, copy and paste to create all the new parts and then use the *Component Edit Pins* dialog (accessed by clicking **Edit Pins** in the *Library Component Properties* dialog) to update all the pin names and numbers in one go.
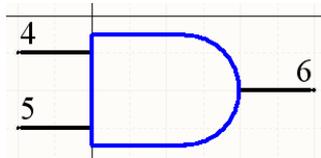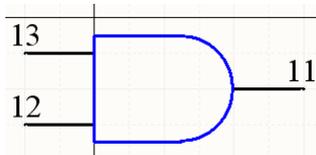
Part B

6. Repeat steps 3 to 5 above to create the remaining two parts, Part C and D. Save the library.

*Part C*   *Part D*

Creating an alternate view mode for a part

You can add up to 255 alternate view modes to a component part. These view modes can contain any different graphical representation of the component, such as DeMorgan or IEEE symbols. A library of IEEE symbols are available from the Sch Lib IEEE toolbar (**View » Toolbars » Utilities**), or select **Place » IEEE Symbols**.

If any alternate views of a part have been added, they are displayed in the Schematic Library Editor by selecting an alternate mode from the **Mode** button drop-down list. When the component is placed on the schematic sheet, the view mode is selected from the Mode drop-down list in the Graphical section of the *Component Properties* dialog.

To add an alternate view mode, with the component part displayed in the design window of the Schematic Library Editor:

1. Click on the ✚ button or select **Tools » Mode » Add.** A blank sheet for Alternate 1 displays.
2. Place the appropriate IEEE symbol, for example, for that part and save the library.
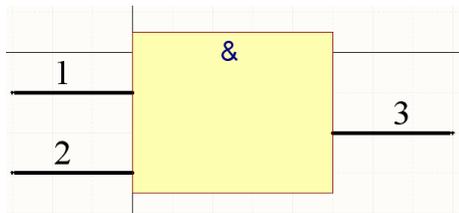
Alternate 1 view mode of IEEE symbol

Setting the component's properties

1. Set the component's properties by clicking on the **Edit** button in the SCH Library panel when the component is selected in the Components list. Fill in the *Library Component Properties* dialog specifying the default designator as U?, the description as `Quad 2-Input AND Gate` and add the footprint name `DIP14` to the Models list. We will create a DIP14 footprint using the PCB Component Wizard later in this tutorial.

2. Save the component in the library by selecting **File » Save** [shortcut Ctrl+S].

Adding components from other libraries

You can also add in components to your schematic library from other open schematic libraries and then edit their properties as required. If the component is part of an integrated library, you will have to open the `.IntLib` file (**File » Open**) and choose **Yes** to extract the source libraries. Then open the generated source library (`.Schlib`) from the Projects panel.

1. Select the component that you wish to copy in the Components list of the SCH Library panel so it displays in the design window.
2. Select **Tools » Copy Component** to copy a component from the current library document to any other open library document. The *Destination Library* dialog displays listing all currently open schematic library documents.
3. Select the document to which you want to copy the component. Click **OK** and a copy of the component will be placed in the destination library where you can edit it, if necessary.

Copying multiple components

You can copy single or multiple library components within a schematic library or to another open schematic library using the SCH Library panel.

1. Select the component(s) in the Components list of the SCH Library panel using the standard Windows selection keys (Click, Shift+Click and Ctrl+Click). Right-click and select **Copy**.
2. Change to the destination library, right-click in the Components list of the SCH Library panel and choose **Paste** to add them to the list.

Checking components using Schematic Library reports

To check that the new components have been created correctly, there are three reports that can be generated when a schematic library is open. All reports are created in ASCII text format.

Make sure the library file is saved before the reports are generated. Close the report file to return to the Schematic Library Editor.
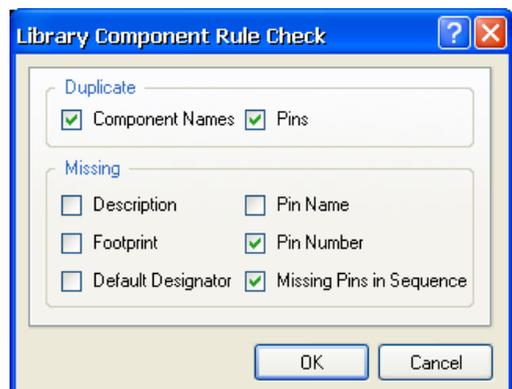
Component Report

To create a report that lists all the information available for the active component:

1. Select **Reports » Component** [shortcut R, C].
2. The report '`libraryname.cmp`' displays in the Text Editor and includes the number of parts with the pin details for each part in the component.

Library Report

To create a report that lists each component in the library and its description:

1. Select **Reports » Library** [shortcut R, L].
2. A report named `libraryname.rep` displays in the Text Editor.

Component Rule Checker

The Component Rule Checker tests for errors such as duplicates and missing pins.

1. Select **Reports » Component Rule Check** [shortcut R, R]. The *Library Component Rule Check* dialog displays.
2. Set the attributes you wish to check for. Click **OK**. A report named `libraryname.err` displays in the Text Editor, listing any components that violate the rule check.
3. Make any adjustments necessary to the library and rerun the report.
4. Save and close the Schematic Library.

Creating PCB component footprints

This part of the tutorial covers the following topics:

- creating a new PCB library
- using the Component Wizard to create a footprint for a schematic component
- manually creating unusual footprints in a PCB library
- using routing primitives within a footprint.

Footprints can be created in the PCB Editor and copied into a PCB Library, copied between PCB libraries, or created from scratch using the PCB Library Editor's PCB Component Wizard or drawing tools. If you had a PCB design with all the footprints already placed, you could use the **Design » Make PCB Library** command in the PCB Editor to generate a PCB library that includes those footprints only.

DXP also includes comprehensive libraries of predefined through-hole and SMD component footprints for use in designing PCBs. The footprint libraries (`.PcbLib` files) supplied are stored in the `\Altium2004\Library\Pcb` folder in your DXP installation directory.

In this part of the tutorial, we will be creating new footprints to illustrate the procedures required only. Please check the specifications for a particular footprint using the manufacturer's data books.

Creating a new PCB Library

To create a new PCB library:

1. Select **File » New » PCB Library**. A new PCB library document, called `PcbLib1.PcbLib`, is created and an empty component sheet called `PCBComponent_1` displays in the design window.
2. Rename the new PCB library document to `PCB Footprints.PcbLib`, for example, by selecting **File » Save As**.
3. Open the PCB Library Editor panel by clicking on the **PCB Library** tab.
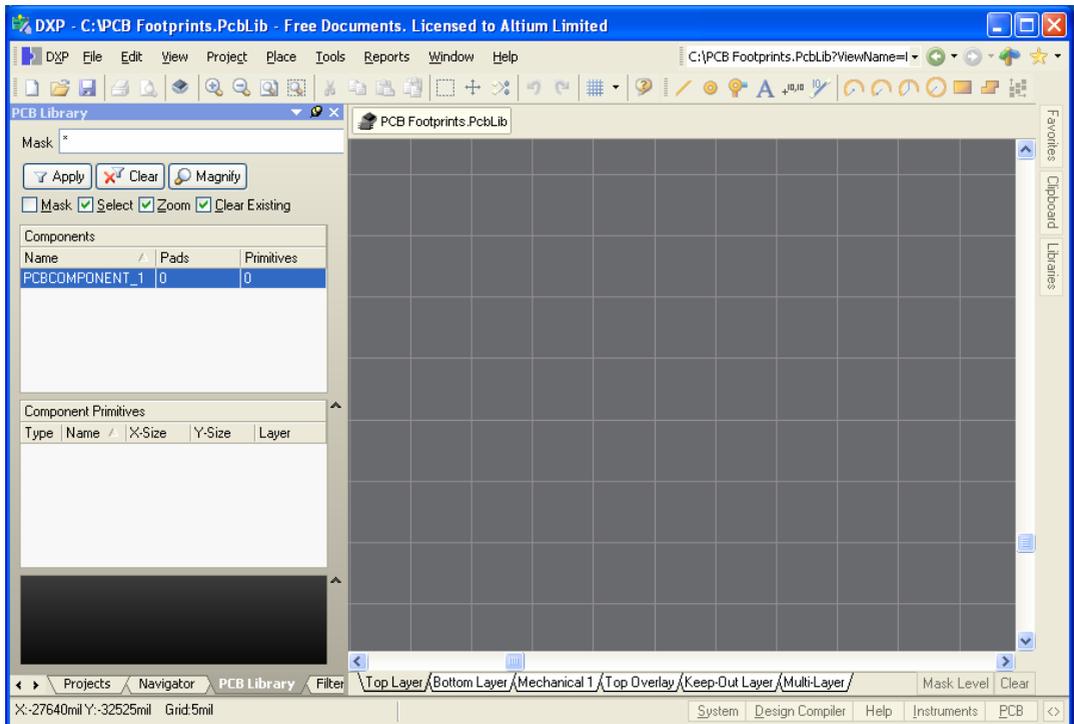
4. You can now add, remove or edit the footprint components in the new PCB library using the PCB Library Editor commands.

Using the PCB Component Wizard

The PCB Library Editor includes a PCB Component Wizard that will build a component footprint based on your input to a series of questions. We will use the wizard to create a footprint for a DIP14.

To create our new component footprint, DIP14, using the Component Wizard:

1. Select the **Tools » New Component** [shortcut T, C]. The PCB Component Wizard will automatically start. Click on **Next >** to progress through the Wizard.
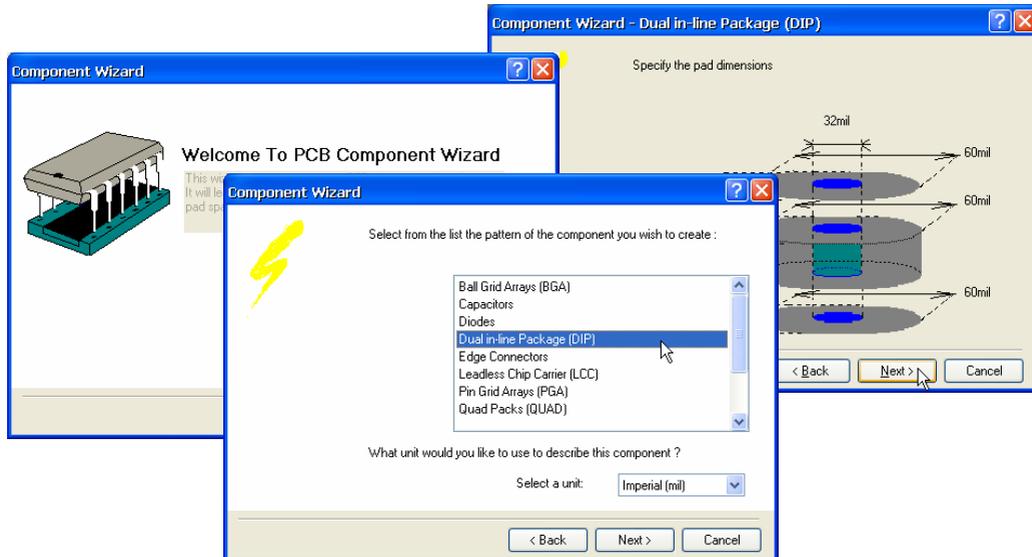
2. Answer the questions asked by selecting from the options available. To create our DIP14, select Dual in-line Package (DIP) as the pattern, Imperial units, 60mil round pads with a 32mil hole (select and type over the dimensions), a distance between pads of 300mil (horizontal) and 100mil (vertical) and then accept the rest of the defaults until you need to specify the number of pads required. Type in `14` from the number of pads required.

3. Click **Next >** until you come to the final page of the Wizard and click **Finish**. The filename of new footprint, `DIP14`, will appear in the Components list in the PCB Library Editor panel and the new footprint will display in the design window. You can then further modify the component to suit requirements.

4. Save the library with its new footprint component by selecting **File » Save** [shortcut Ctrl+S].
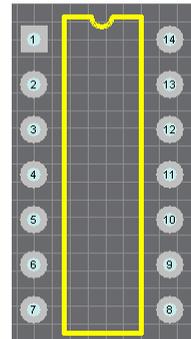
Manually creating component footprints

Footprints are created and modified in the PCB Library Editor using the same set of tools and design objects available in the PCB Editor. Anything can be saved as a PCB footprint, including corner markers, phototool targets and mechanical definitions.

To create the component footprint, we will place tracks and arcs for the outline and then place pads to form the component pin connections. These design objects can be placed on any layer, however the outline is normally created on the Top Overlay (silkscreen) layer and the pads on signal layers. When you place the footprint on a PCB document as a component, all objects that make up the footprint will be assigned to their defined layers.

To manually create a component footprint:

1. Select the **Tools » New Component** [shortcut T, C]. The Component Wizard will automatically start.

2. Press **Cancel** to exit the Wizard and manually create a component. An empty component footprint workspace displays, called `PCBComponent_2`.

3. To rename this default footprint, select `PCBComponent_1 - Duplicate` from the list in the PCB Library Editor panel and double-click. Type in the new footprint name in the *PCB Library Component* dialog.

4. It is recommended that you build the component footprint around the workspace 0, 0 reference point, often indicated by an origin marker. Select **Edit » Jump » Reference** [shortcut J, R] to position the cursor at the workspace 0, 0 coordinate.

> If the origin marker is not displayed, select **Tools » Preferences** [shortcut T, P], click on the **Display** tab and click on the **Origin Marker** box.

The reference point is the point you will be 'holding' the component by when you place it. Typically, the reference point is either the center of pad 1 or the geometric center of the component. The reference point can be set to either of these at any time using the **Edit » Set Reference** submenu options.

Placing pads on a new footprint

One of the most important procedures in creating a new component footprint is placing the pads that will be used to solder the component to the PCB. These must be placed in exactly the right positions to correspond to the pins on the physical device.

To place the pads on the footprint:

1. Click on the **Top Layer** tab at the bottom of the design window before you start placing pads.

2. Select **Place » Pad** [shortcut P, P] or click the 🔵 button on the toolbar. A pad will appear floating on the cursor. Before placing the first pad, press the **TAB** key to define the pad properties. The *Pad* dialog displays.
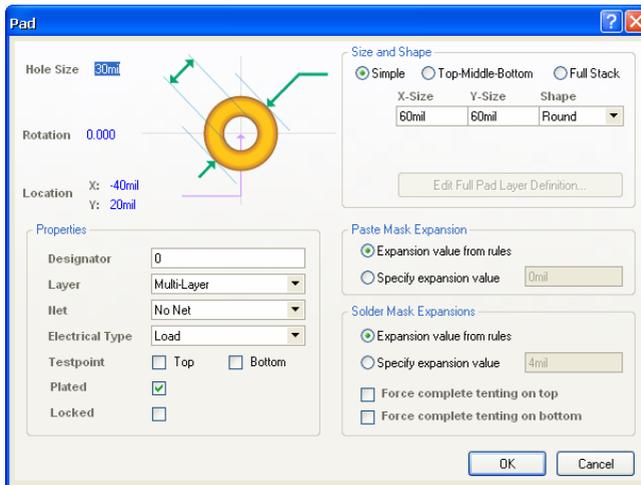
3. Change the size and shape as required and set the Designator to 1 (to correspond with the component pin number). Click **OK**.

4. Position the cursor and click, or press **ENTER**, to place the first pad centered on the origin 0, 0.

5. Before placing the next pad, press **TAB** to make any changes. Note that the pad designator increments automatically.

6. Right-click, or press **ESC**, to exit pad placement mode.

7. Save the footprint by selecting **File » Save** [shortcut Ctrl+S].

## Pad Designators and Paste Arrays

Pads can be labeled with a designator (usually representing a component pin number) of up to four alphanumeric characters and no spaces. The designator can be left blank if desired. If the designator begins or ends with a number, the number will auto-increment when placing a series of pads sequentially. To achieve alpha increments, e.g. 1A, 1B, or numeric increments other than 1, use the Paste Array feature.

By setting the designator of the pad prior to copying it to the clipboard and setting the Text Increment field in the *Paste Array* dialog, the following types of pad designator sequences can be placed:

- numeric (1, 3, 5)
- alphabetic (A, B, C)
- combination of alpha and numeric (A1 A2, or 1A 1B, or A1 B1 or 1A 2A, etc).

To increment numerically, set the Text Increment field to the amount you wish to increment by. To increment alphabetically, set the Text Increment field to the letter in the alphabet that represents the number of letters you wish to skip. For example, if the initial pad had a designator of 1A, set the Text Increment field to A, (first letter of the alphabet), to increment designators by 1. Set the Text Increment field to C (the third letter of the alphabet) and the designators will become 1A, 1D (three letters after A), 1G etc.

1. Create the initial pad with the designator required, e.g. 1A. Copy this pad to the clipboard using **Edit » Copy** [shortcut Ctrl+C]. Click on the pad center to define the copy reference point.

2. Select **Edit » Paste Special** [shortcut E, A]. The *Paste Special* dialog displays. Select **Paste on current layer** and **Keep net name**.

3. Click on the **Paste Array** button to display the *Setup Paste Array* dialog.

4. As an example, set the Item Count to 5, the Text Increment to C, select a Linear Array Type and the appropriate array spacing for the copied pad and click **OK**.

5. Click to place the array. Check that the pad designators have incremented as expected.
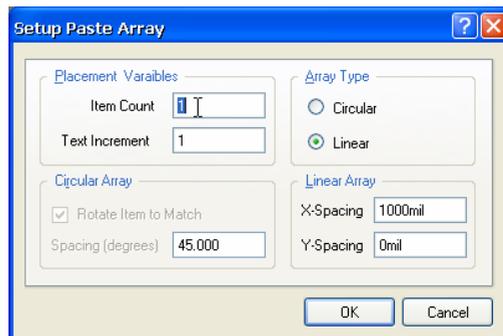
Drawing an outline of a new footprint

We will create the footprint outline on the Top Overlay layer so that it can be included in a silkscreen mask during manufacture of the PCB. The outline is a manufacturing guide only. It is the placement of the pads that is crucial.

1. Click on the **Top Overlay** layer tab at the bottom of the design window

> Always create surface mount footprints on the Top Layer. They can be flipped to the Bottom Layer during placement using the L shortcut key.

before you start placing lines (tracks).

2. Check the manufacturer's specifications for the footprint. Press **Q** to toggle the coordinates from imperial (mils) to metric (mm). Check the coordinates in the Status bar at the bottom of the screen to see which measurement mode you are in (mils or mm). Also, set the grids by selecting **Tools » Library Options**.

3. Use the Line tool to create the component outline on the Top Overlay layer. Select **Place » Line** [shortcut P, L], or click on the ╱ button.

4. Click to define a start point of the tracks that will define the top of the footprint.

5. Press **TAB** to define the line width (0.2mm) and check the layer in the *Line Constraints* dialog.

> If you make a mistake during track placement, press BACKSPACE to remove the last track segment.

6. Click to create the outline tracks and right-click to end this series of connected track segments.

7. To exit track placement mode, right-click or press **ESC**.

Placing Designator and Comment special strings

The special strings, .Designator and .Comment, can be added to the footprint layer in the PCB Library Editor if you require control over their layer, location and text attributes prior to placing the component on a PCB sheet. These will be in addition to the standard designator and comment, which can be hidden, if required, when placing the component in a PCB document by selecting the **Hide** option in the Designator and Comments sections of the *Component* dialog.

Typically, these special strings are placed on a mechanical layer which is then included in the assembly drawing. Display the required Mechanical layer by selecting **Tools » Layers & Colors**. Click on **Enable** and then **Show** next to the Mechanical layer name in the *Board Layers and Colors* dialog.

1. Click on the **Mechanical** layer tab at the bottom of the design window to activate this layer. The tab will be highlighted and all new text will be placed on this layer.

2. Select **Place » String** [shortcut P, S] or click on the **Place String** button **T**.

3. Press the **TAB** key to type in the text string and define its properties, e.g. font, size and layer, before you place the text. The *String* dialog displays. Select **.Designator** from the **Text** drop-down list. Set the text height to 60mil and the width to 10mil and click **OK**.

4. Now we can place the text string. Position the text string in the required location and click.

5. Place the .Comment special string using the same procedure.

6. Right-click, or press **ESC**, to exit text placement mode.

If the text is not displaying when the footprint is placed in the PCB document, make sure that the **Convert Special Strings** option is selected in the **Protel PCB – Display** page of the *Preferences* dialog (**Tools » Preferences**) in the PCB Editor.

Adding height to your PCB footprint

To add a height attribute to your footprint, double-click on the footprint in the Components list in the PCB Library panel to display the *PCB Library Components* dialog. Type the recommended height for the component in the Height field and click **OK**.

Creating a footprint with an irregular pad shape

You can create irregular pad shapes by joining pad shapes together, as in the first example SOT89, or by adding primitives to a pad that will become connected to the pad's net when placed in a PCB document. This section of the tutorial looks at creating the surface mount footprint SOT89, how to include some routing primitives in a component footprint and also how to create footprints with multiple connection points on same pin.

The manufacturer's specifications for this SOT89 footprint are in metric, summarized below.

| Dimensions (millimeters)	
A	5.00
B	6.00
C	1.35
D	0.70
E	1.90
F	3.30
G	2.30
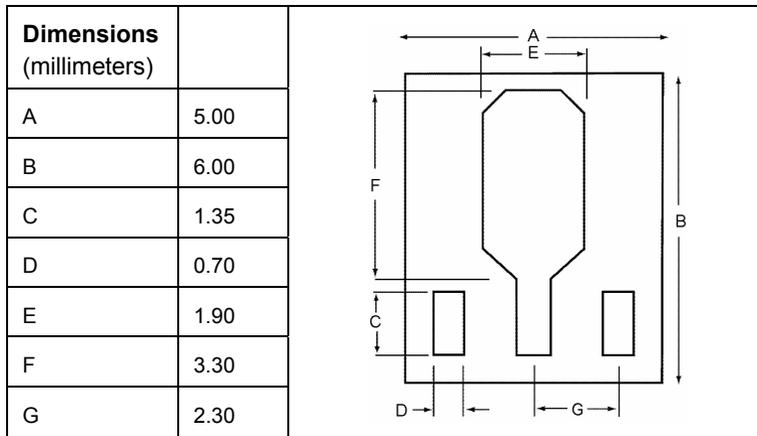
1. Press **Q** to toggle the coordinates from imperial (mils) to metric (mm) if necessary. Check the coordinates in the Status bar at the bottom of the DXP window to see which measurement mode you are in (mils or mm).

2. Make sure you have set the grid to metric, by changing the visible and snap grids by selecting **Tools » Library Options** [shortcut T,O]. Set up the grids to `10`mm and the Snap grid to `1`mm and click **OK**. Use the **PageUp** key or **Ctrl+mouse wheel** to zoom in and see the grids.

Placing the pads

When creating the component footprint SOT89, we will use the reference point on Pin 1 as the origin of this footprint, i.e. the center of pin number one and therefore corresponding pad 1, will be at coordinates 0,0.

1. To place the pads on the Top Layer of the footprint, select **Place » Pad** [shortcut P, P] or click the ⊙ button. Press the **TAB** key to define the pad properties in the *Pad* dialog. Make sure the layer is set to **Top Layer**, the Designator is set to `1` (to correspond with the component pin number) and the hole size is `0`mil. Click **OK**.

2. Position the cursor and click to place the three pads. The designators will automatically update. Right-click, or press **ESC**, to exit pad placement mode. Modify pad 2 to elongate it so it will join to Pad 0 when placed.
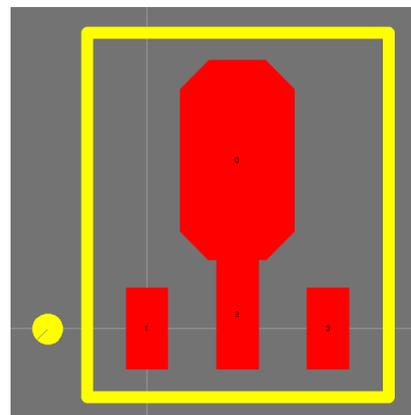
SOT89 footprint

3. Finally place pad 0. Set the pad size and shape in the *Pad* dialog by clicking on **Simple** and select **Octagonal** from the Shape drop-down list.

Drawing the component outline

1. To create the component outline on the Top Overlay layer, click on the **Top Overlay** layer tab at the bottom of the design window. Select **Place » Line** [shortcut P, L], or click the ✏ button.

2. Click to place the first corner of the outline box. Press the **TAB** key to display the *Line Constraints* dialog, set the Width, check the layer is Top Overlay, and click **OK**. Click to define the corners of the box and finish the outline by clicking back on the starting point. Right-click, or press **ESC**, to exit line placement mode.

3. A requirement for this footprint is an indicator near pin 1; in this case, a circle is placed near pad 1 on the Top Overlay. A chamfered edge to the footprint could also be used. To place the circle, select **Place » Full Circle** [shortcut P, U] or click the ⊙ button. Click to start the circle at its center. Then drag the crosshair and click to set a circle with a radius of 5mil. Right-click, or press **ESC**, to exit circle placement mode. Double-click on the circle to change its Width to 10mil in the *Arc* dialog, thereby creating a filled circle.

Checking solder and paste masks

Solder and paste masks are created automatically at each pad site on the Solder Mask and Paste Mask layers respectively. The shape that is created on the mask layer is the pad shape, expanded or contracted by the amount specified by the Solder Mask and Paste Mask design rules set in the PCB Editor, or specified in the *Pad* dialog.

Displaying the masks

Check the solder and/or paste masks have been automatically created in the correct layers in the PCB Library Editor. In this example, we will turn on the Top Solder layer.

1. To make these layers visible, select **Tools » Layers & Colors** [shortcut L] and click on the **Show** boxes next to the Mask Layers in the *Board Layers & Colors* dialog and click **OK**.

2. Now click on the layer tab, e.g. **Top Solder**, at the bottom of the design window to view the top solder mask. Use the shortcut toggle keys **Shift+S** to view the layers in single layer mode.

Setting mask expansions by design rules

If you wish to use design rules to set the mask expansions:

1. Select **Expansion value from rules** in the Paste Mask Expansion and/or Solder Mask Expansion sections of the *Pad* dialog.

2. Set up these rules by selecting **Design » Rules** from the PCB Editor menus and check or revise the Mask category design rules in the *PCB Rules and Constraints Editor* dialog. These rules will be obeyed when the footprint is placed in the PCB.

> If tracks, or any other objects, are used as part of a footprint, solder and paste masks must be defined manually using tracks, etc. on the Mask layers.

Specifying mask expansions

To overwrite the expansion design rules and specify a mask expansion:

1. Select **Specify expansion value** in the Paste Mask Expansion and/or Solder Mask Expansion sections of the *Pad* dialog.

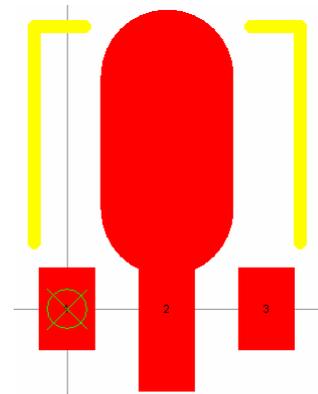2. Type the required value(s) and click **OK**. Save the footprint.

Including routing primitives in component footprints

Library component footprints can also include routing primitives, such as tracks and arcs placed on signal layers. The following example of footprint SOT89 includes a primitive object (a wider track connected to pad 2), as well as rectangular pads, which would be part of the net connectivity. This is simply an alternative way to creating the octagonal pad used for its irregular pad shape that we set up in the previous section of this tutorial.

If you manually place the footprint on the board, only pads will inherit a net name. Any other primitives on a signal layer, e.g. arcs and fills used to create routing within a footprint, will show as a DRC error. If it does not, you could force the online DRC by moving the component.

Net names can also be applied to routing primitives that are built into placed components in a PCB document at any time. To assign a net name to the primitive in the placed footprint in a PCB document:

*SOT89 with primitive (track) placed on a signal layer to create an irregular pad shape*

1. Select **Design » Netlist » Update Free Primitives from Component Pads** from the PCB Editor menu.

2. The net name of the routing primitives are resynchronized to the net name on the pads they connect to, i.e. this command will connect a primitive to the same net as its adjoining pad.

Footprints with multiple connection points on same pin

The footprint below, a TO-3 transistor, has multiple connection points on same pin. You will notice that there are two pads with the same designator of '3'.
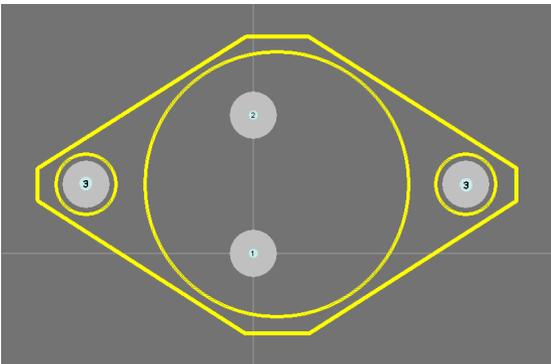

When the **Design » Update PCB** command is used in the Schematic Editor to transfer design information to the PCB, the resulting synchronization should show the connection lines going to both pads in the PCB Editor, i.e. they are on the same net, as shown below.
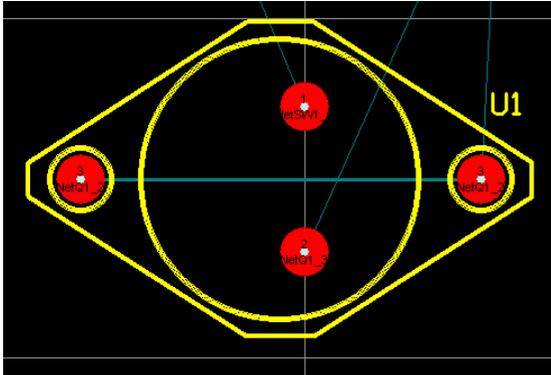
Drawing a footprint with a solder mask

The following footprint, named LCR1_KC1, is for a push button switch. It requires a solder mask to be included along with the footprint outline (Top Overlay) and the tracks and pads on the signal layer (Top Layer).

1. To create the component outline on the Top Overlay layer, click on the **Top Overlay** layer tab at the bottom of the design window. Select **Place » Full Circle** [shortcut P, U] or click the ⊘ button on the toolbar. Click on 0, -80 mil to start the circle at its center, then drag the crosshair to 100, -80 and click to set a circle with a radius of 100mil. Right-click, or press **ESC**, to exit circle placement mode.

2. Next, create the solder mask on the Top Solder layer. To make this layer visible, select **Tools » Layers & Colors** [shortcut L] and click on the **Show** box next to Top Solder in the Mask Layers section of the *Board Layers* dialog.
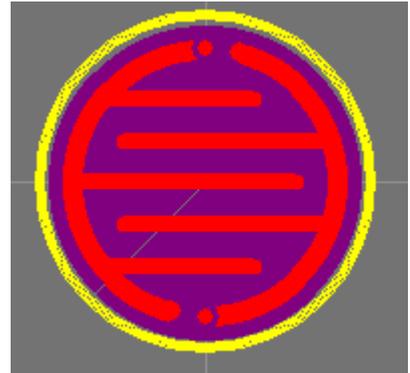
LCR1_KC1 footprint

   Now click on the **Top Solder** layer tab at the bottom of the design window and draw a circle as shown in the step above with its center on 0, -80 mil, a radius of 45 mil and width of 100mil (to fill in the circle). Right-click to exit circle placement mode.

3. Click on the **Top Layer** tab at the bottom of the design window and use tracks and arcs to create the copper connections on the Top Layer. To exit track placement mode, right-click or press **ESC**.

4. To place the pads on the Top Layer of the footprint, select **Place » Pad** [shortcut P, P] or click the ◉ button. Press the **TAB** key to define the pad properties in the *Pad* dialog. Set the pad size and shape by clicking on **Simple**, type in `10`mil for both the X and Y size and select **Round** from the Shape drop-down list. Make sure the layer is set to Top Layer, the Designator is set to 1 (to correspond with the component pin number) and the hole size is 0 mil. Click **OK**.

5. Position the cursor and click to place the first pad centered on the origin 0,0 and place the second pad centered on 0, -160mil. The designators will automatically update. Right-click, or press **ESC**, to exit pad placement mode.

6. Save the footprint by selecting **File » Save** [shortcut Ctrl+S].

Adding footprints from other sources

You can add existing footprints to your PCB library. The copied footprint can then be renamed and modified to match the specifications required.

If you want to add existing footprints to your PCB library, you can:

- select placed footprint(s) in a PCB document and copy (**Edit » Copy**) and paste them into an open PCB library using **Edit » Paste Component**, or

- Select **Edit » Copy Component** when the footprint to be copied is active in the PCB Library Editor, change to the open PCB destination library and select **Edit » Paste Component**. The footprint is added as a new component in the Components list in the PCB Library panel and displays in the design window.

Validating component footprints

As in the Schematic Library Editor, there are a series of reports that you can run to check the footprints have been created correctly and identify which components are in the current PCB library. To validate all components in the current PCB library, we will run the Component Rule Check report. The Component Rule Checker tests for duplicate primitives, missing pad designators, floating copper and inappropriate component reference.

1. Save your library file before running any of these reports.

2. Select **Reports » Component Rule Check** [shortcut R, R] to display the *Component Rule Check* dialog.
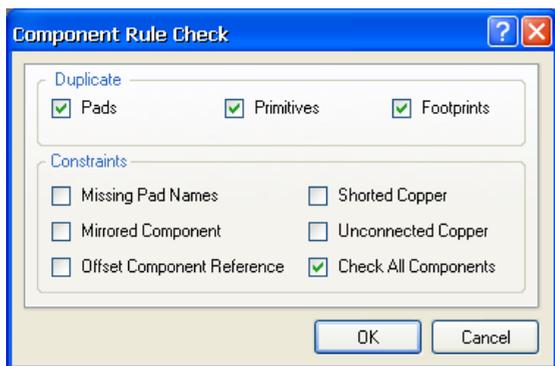
3. Check all the boxes available and click **OK**. A report, named `PCBlibraryfilename.err`, is generated and opens in the Text Editor. Any errors will be noted.

4. Close the report to return to the PCB Library Editor.

Creating an integrated library

Now we have created a schematic library with some schematic components and a PCB library with related footprint models, we can add these to a library package and then compile these libraries together into an integrated library. The components will then always be stored with their models. Note

that simulation model files should be copied into the same folder as the source libraries before compiling. Note that you can now also use the **Design » Make Integrated Library** command to create the source libraries and an integrated library from an open project.

The steps to create an integrated library are detailed in the tutorial *Building an Integrated Library*.

1. Create a source Library Package by selecting **File » New » Integrated Library**. The **Projects** panel displays with an empty Library Package file named `Integrated_Library1.LibPkg`. Rename the new Library Package using **File » Save Project As**.

2. Add the source libraries to the Library Package by selecting **Project » Add Existing to Project**. Browse to find the schematic libraries (`.schlib`) and model libraries (PCB footprint libraries (`.pcblib`), Protel 99 SE libraries (`.lib`), SPICE models or Signal Integrity models that you want to add to your Library Package. Click **Open** and the added libraries are listed as Source Libraries in the **Projects** panel.

   If you do not want to add in the model libraries and files, you can set a pathname to where they reside on your hard disk by right-clicking on the Library Package name in the **Projects** panel and selecting **Project Options**. Add in the pathnames to the location of the footprints and models required by clicking on **Add** in the **Ordered List of Search Paths** section of the **Search Paths** tab of the *Options for Project* dialog.

3. Compile the source libraries and model files in the library package into an integrated library by selecting **Project » Compile Integrated Library**. Any errors or warnings found during compilation are displayed in the **Messages** panel (**View » Workspace Panels » System » Messages**). Double-click on an error in the Messages panel to view more information and jump to the component. Fix any inconsistencies in the individual source libraries at this point and recompile the integrated library.

4. A new `Integrated Libraryname.INTLIB` is generated, saved in the output folder nominated in the **Options** tab of the *Project Options* dialog and displays in the **Library** panel, ready to use. The integrated library is automatically added to the current Libraries list in the **Libraries** panel.

Glossary

The following definitions are used in this tutorial.

Component
: A component is a physical device that is placed on the board, e.g. the integrated circuit or resistor. Within these components, there may be either a single part or a set of parts that are packaged together.

Designators
: Unique identifiers that are used to tell one component from another in a PCB. They can alpha, numeric, or a combination of both. Pads also have unique designators that correspond to the component pin numbers.

Footprint
: A footprint defines (or models) the space occupied by the component on the PCB. The footprint model of a component is stored in a PCB library. A footprint may contain pads for connecting to the pins of a device and a physical outline of the package created from track and/or arc segments on the silkscreen (overlay) layer. Device mounting features may also be included.

	Footprints in the PCB library have no designator or comment. They become components when placed on a PCB sheet where the designators and comments are allocated.
Hidden pins	These are pins that exist on the component but do not need to be displayed. Typically, they are power pins and can automatically be connected to a specified net.
Library	A Schematic Library is a set of components and its parts stored on individual sheets. A PCB Library contains the component footprints. Each library type has its own Editor. Integrated libraries combine schematic libraries with their related models and cannot be edited directly by the Library Editors.
Object	Any individual item that can be placed in the Library Editor workspace.
Pads	Pad objects are normally used in a footprint to create connection pads for component pins.
Part	A collection of graphical objects represents one part of a multi-device component. Parts are stored in separate sheets within components in the schematic component libraries.
Pins	Component pins give a component its electrical properties and define connection points on the component.

Revision History

Date	Version No.	Revision
09-Dec-2003	1.0	New product release
24-Nov-2004	1.1	Updated for DXP 2004 SP2.