Module 16: PCB Library Editor

Module 16: PCB Library Editor

16.1 PCB Library Editor 16-1
16.1.1 The PCB Library workspace 16-1
16.1.2 PCB Library Editor Panel 16-2
16.1.3 Creating a component using the Component Wizard 16-2
16.1.4 Creating a component using IPC footprint wizard...................................... 16-3
16.1.5 Creating a component using IPC Footprints batch generator...................... 16-3
16.1.6 Manually creating a component 16-4
16.1.7 Copying a component 16-5
16.1.8 Special strings in the Library Editor.. 16-5
16.1.9 Component Rule Check 16-5
16.1.10 Exercise – Creating the component footprint 16-6
16.1.11 Browsing footprint libraries.. 16-9
16.1.12 Creating footprints with an irregular pad shape 16-9
16.1.13 Managing components that include routing primitives in their footprint 16-11
16.1.14 Footprints with multiple pads connected to the same pin 16-12
16.1.15 Handling special solder mask requirements 16-12
16.1.16 Other footprint attributes.. 16-13

16.2 3D dimensional component detail 16-14
16.2.1 Adding height to your PCB footprint.. 16-14
16.2.2 Adding a 3D body to a footprint... 16-14
16.2.3 Manually placing 3D body objects.. 16-14
16.2.4 Using the 3D Body Manager 16-16

Module Seq = 16

16.1 PCB Library Editor

The PCB Library Editor is used to create and modify PCB component footprints and manage PCB component libraries. The PCB Library Editor also includes a **Component Wizard** that you can guide through the creation of most common PCB component types.

16.1.1 The PCB Library workspace

An existing PCB library (*.PcbLib) can be opened using the **File » Open** command, displaying the first footprint in that library, and a list of all footprints in the library in the **PCB Library** panel. Click on the required component in the **Components** list.

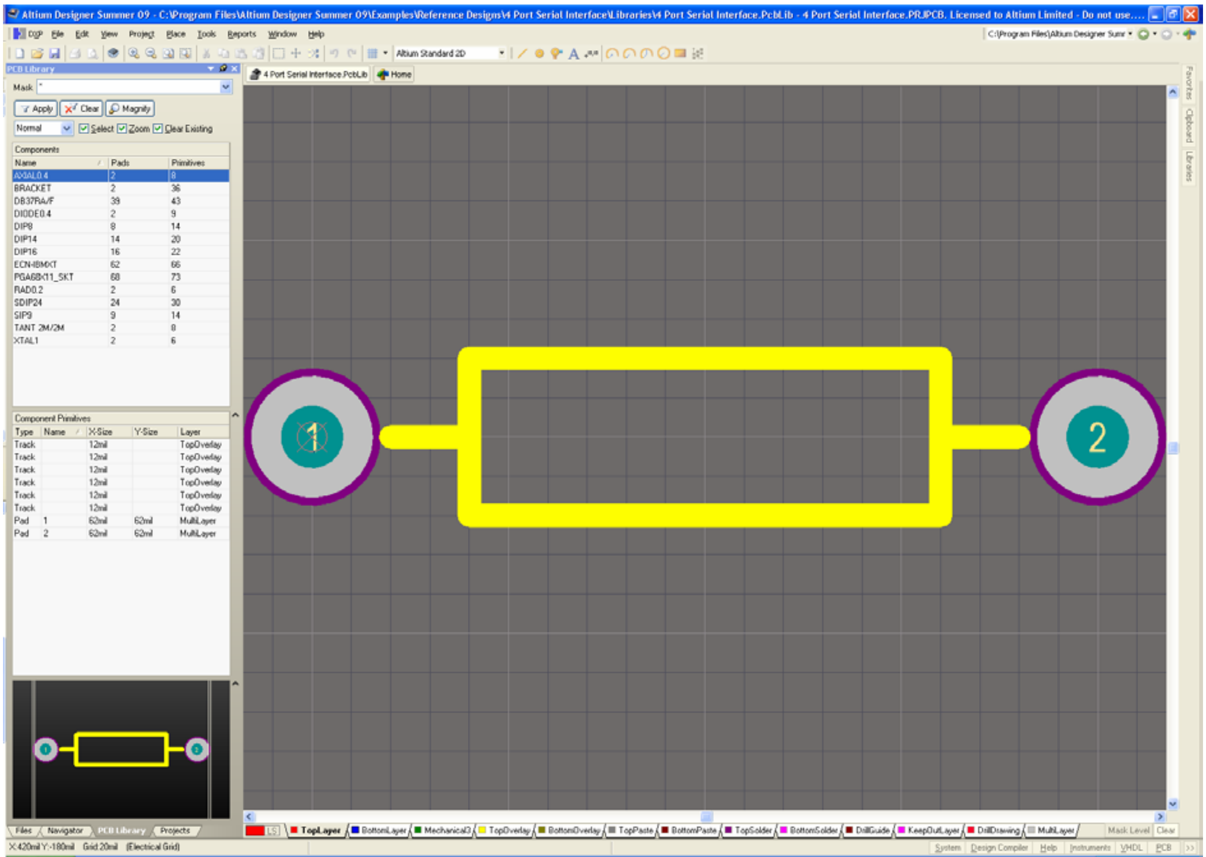

Figure 1. PCB Library Editor Workspace

The view commands, primitive objects, layers, selection and focus, grids and general editing functions are all identical to the PCB Editor.

Settings in the **Preferences** dialog and **Board Options** dialog also apply in the PCB Library Editor.

16.1.2 PCB Library Editor Panel

The **PCB Library** panel of the PCB Library Editor panel provides a number of features for working with PCB components. These include:

- The **Components** section of the panel lists all the components in the active library.
- Right-click in the **Components** section to display menu options to Create New Components, edit Component Properties, Copy or Paste selected components, or update the component footprints on open PCBs.
- Note that the copy/paste commands in the right-click menu can be used with multiple footprints selected, and support:
  - copying and pasting within a library,
  - copying and pasting from a PCB into a library,
  - copying and pasting between PCB libraries.
- The **Components Primitives** section lists the primitives that belong to the currently selected component. Click on a primitive in the list to highlight it in the design window.
- The way that the chosen primitive is highlighted depends on the options at the top of the panel:
  - Enabling **Mask** will result in only the primitive(s) you click on remaining at normal visibility, all others will be faded. Click the Clear button down the bottom right of the Workspace to remove the filter and restore the display.
  - Enabling **Select** will result in the primitive(s) you click on being selected, ideal if you need to edit them.
- Right-click in the **Component Primitives** section to control which types of primitives are listed in this section.

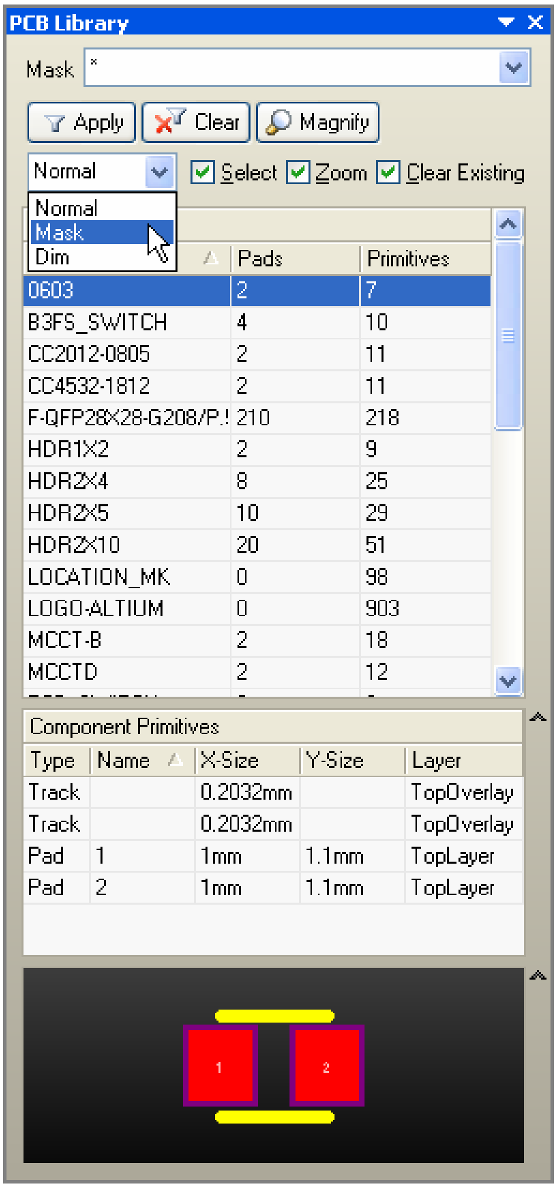

Figure 2. PCB Library panel

16.1.3 Creating a component using the Component Wizard

The PCB Library Editor includes a **Component Wizard**. This Wizard allows you to select from various package types, fill in appropriate information and it will then build the component footprint for you.

To launch the Component Wizard, right-click on the Components section of the PCB Library Editor panel and select **Component Wizard**, or select the **Tools » New Blank Component**.

16.1.4 Creating a component using IPC footprint wizard

The IPC Footprint Wizard creates IPC-compliant component footprints. Rather than working from footprint dimensions, the IPC Footprint Wizard uses dimensional information from the component itself, in accordance with the algorithms released by the IPC.

Available through **Tools » IPC Footprint Wizard** menu when a PCB library is the active document, the new IPC Footprint Wizard creates IPC-compliant component footprints.

In accordance with the IPC standard it also supports three footprint variants, tailored to suit the board density. The wizard supports BGA, BQFP, CFP, CHIP, CQFP, DPAK, LCC, MELF, MOLDED, PLCC, PQFP, QFN, QFN-2ROW, SOIC, SOJ, SOP, SOT143/343, SOT223, SOT23, SOT89, and WIRE WOUND footprints.

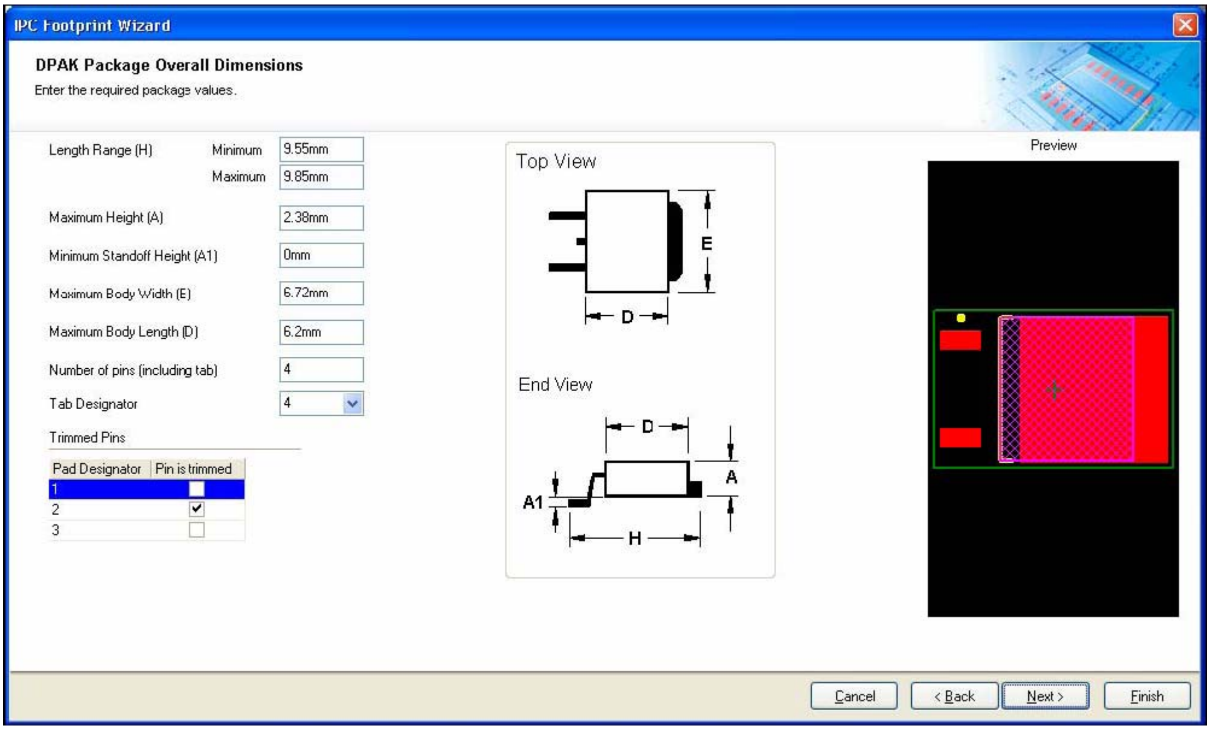

Figure 3. One of the new supported packages in the IPC footprint Wizard is the DPAK (Transistor Outline).

Some of the **IPC Footprint Wizard** features include:

- Overall packaging dimensions, pin information, heel spacing, solder fillets and tolerances can be entered and immediately viewed.
- Mechanical dimensions such as Courtyard, Assembly, and Component (3D) Body Information can be entered.
- Wizard is re-entrant and allows reviewing and making adjustments easy. Previews of the footprint are shown at every stage.
- The finish button can be pressed at any stage to generate the currently previewed footprint.

16.1.5 Creating a component using IPC Footprints batch generator

Available through **Tools » IPC Footprint Batch Generator** menu when a PCB library is the active document, the **IPC Footprint Batch Generator** makes it possible to quickly generate multiple footprints as well as multiple density levels for a single component from a package input file that contains datasheet package information.

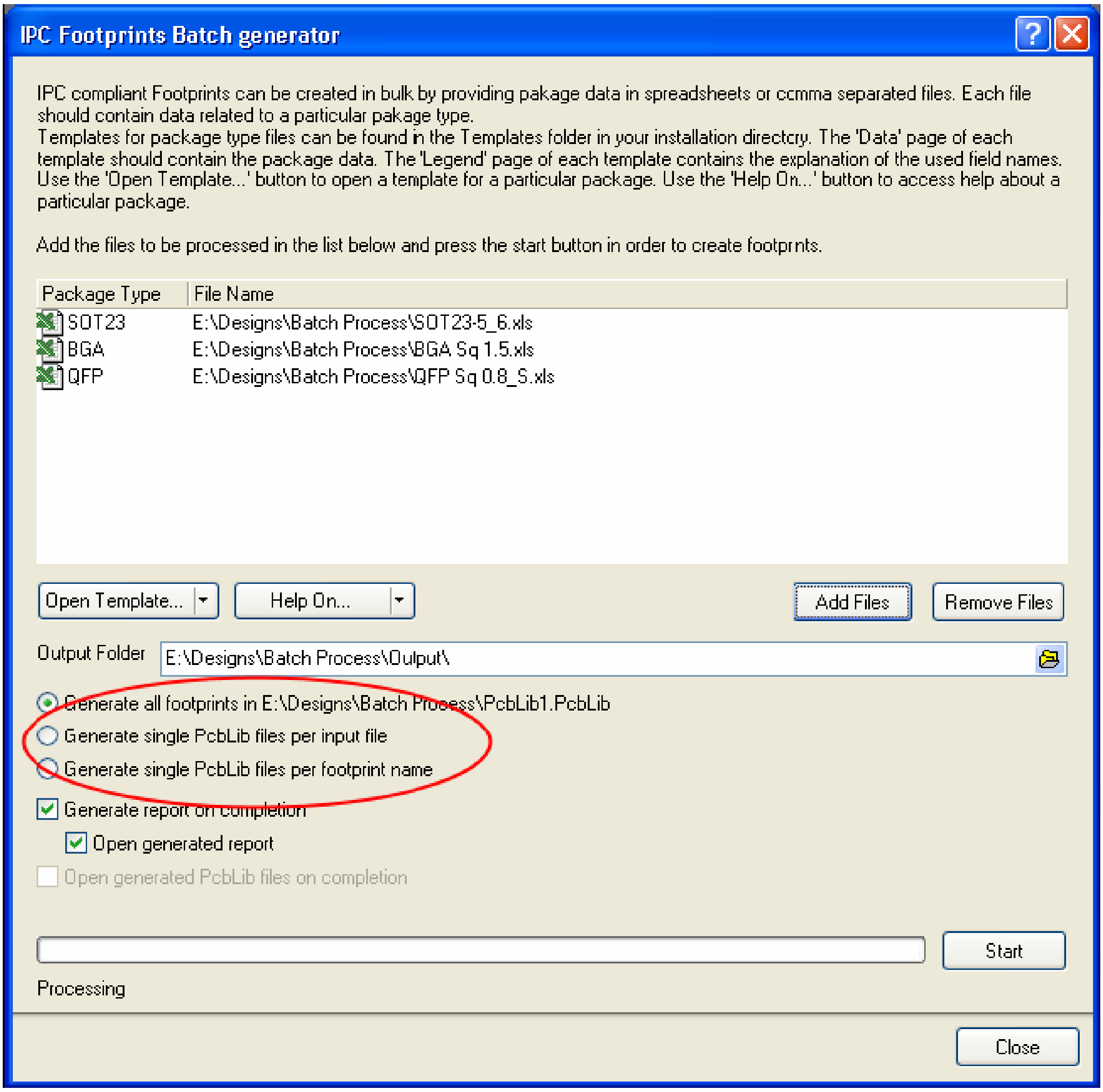

Figure 4. The IPC Footprint Batch Generator has options that either create all the footprints into the current open .PcbLib, or generate a single .PcbLib file based on either an input file or footprint name.

Support for the **IPC Footprints Generator** includes:

- Package type blank template files can be opened within the dialog and are provided in the `\Templates` folder of the installation directory. Help for any of the package type templates is also available.
- Package input files need only to contain the information for one or more footprints of a single package type, and can be either an Excel or comma-delimited (CSV) file format.

16.1.6 Manually creating a component

Components are created in the PCB Library Editor using the same set of primitive objects available in the PCB Editor. In addition to PCB components, corner markers, phototool targets, mechanical definitions, etc. can be saved as components.

The typical sequence for manually creating a component footprint is:

- Open the desired library in the PCB Library Editor.

- Select the **Tools » New Blank Component** menu command. You will be presented with an empty component footprint workspace, called `PCBComponent_1`. Rename the component by double-clicking on the name in the Components list, select **Component Properties** and enter a new name in the **Component Properties** dialog. Component names can be up to 255 characters.
- Use tracks or other primitive objects to place the component outline on the Silkscreen layer.
- Place the pads according to the component requirements. Prior to placing the first pad, press the TAB key to define all the pad properties. Make sure you set the designator property correctly. Typically, the first pad you place is pin 1, so the set designator to '1' for the first pad. The designator automatically increments.

> **Note:** The 0,0 coordinate is the point where the component is 'held' during placement. Always confirm that it is set to a suitable location. Select **Edit » Set Reference** to change the location.

16.1.7 Copying a component

There is often the requirement to copy components, either from one library to another or within the same library. For a single component you can achieve this using the **Edit » Copy Component** command. This command copies the current component, ready for pasting back into a PCB library.

You can also copy/paste multiple component footprints using the commands in the **PCB Library** panel's right-click menu. Select the required component footprints using CTRL+click in the list, then right click and choose **Copy**, then right-click again and choose **Paste X Components** (where X is the number of component footprints you selected).

16.1.8 Special strings in the Library Editor

There are two special strings that are active in the Library Editor. These are provided to allow you to control the positioning of the designator (`.Designator`) and the comment (`.Comment`).

Place these in the PCB Library Editor workspace at the location relative to the component where you would like the designator or comment to be placed.

When you use these, you can hide the default designator and comment that are added when the component is placed in the PCB file.

16.1.9 Component Rule Check

The **Reports » Component Rule Check** command allows you to check either the current component or the whole library for any of the objects selected in the *Component Rule Check* dialog.

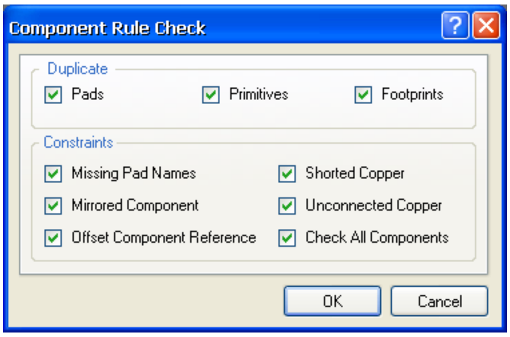

Figure 5. Component Rule Check dialog

The results of the component rule check are displayed in a text document.

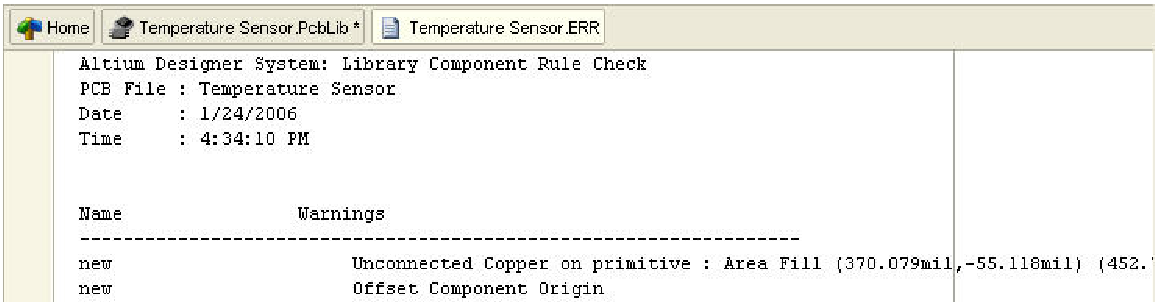

Figure 6. Library Component Rule Check report

16.1.10 Exercise – Creating the component footprint

In this exercise, we will create a new component footprint called `SOIC8`, to use with the Temperature Sensor component you just created, the TCN75.

1. The first thing we need is a footprint library. For this training session we will create the footprint in a project library. If it is not open, open the footprint library `\Program Files\Altium Designer Summer 09\Examples\Training\PCB Training\Temperature Sensor\Libraries\Temperature Sensor.PcbLib`
2. Before creating the footprint, we will make this footprint library part of the Temperature Sensor project. If it is not already open, re-open the project created during the *Environment and Editor Basics* training session, `\Program Files\Altium Designer Summer 09\Examples\Training\PCB Training\Temperature Sensor\Temperature Sensor.PrjPcb.`
3. To add the library to the project, click and hold on the `Temperature Sensor.PcbLib` in the **Projects** panel, then drag and drop it onto the project filename, `Temperature Sensor.PrjPcb`. It will disappear from the Free Documents, instead appearing under the *Libraries* folder icon in the project structure.
4. Right-click on the project name and select **Save Project**.
5. To create the new SOIC8 footprint we will use the IPC Footprint Wizard, select **Tools » IPC Footprint Wizard** to run the Wizard.
6. In the list of pattern types, select Small Outline Integrated Package (SOIC), as shown below in Figure 7.

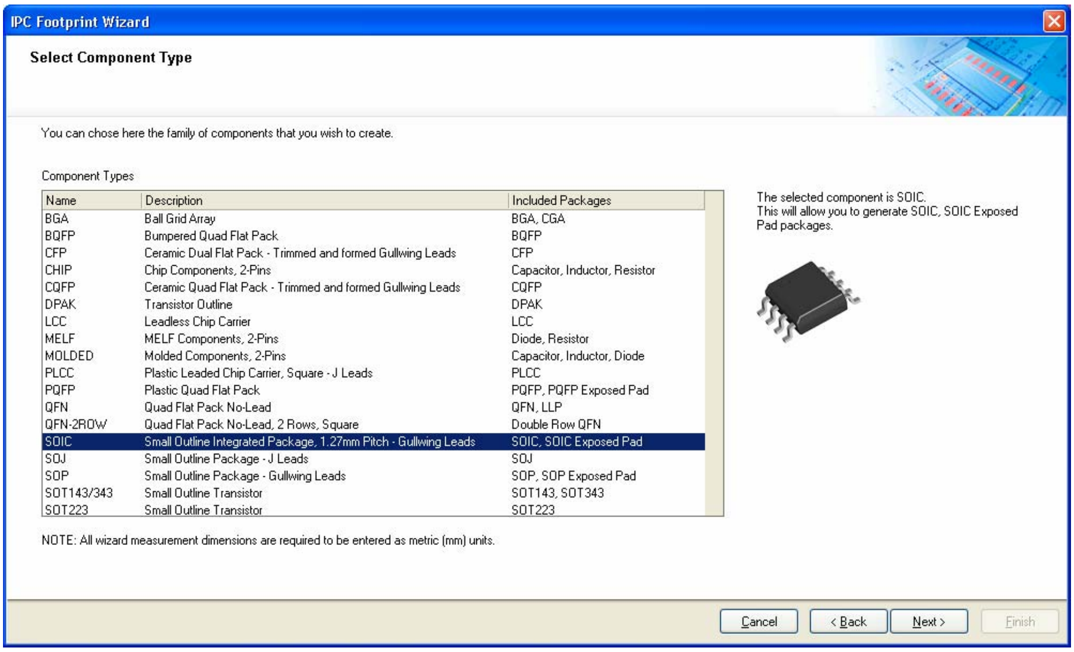

Figure 7. Choose the footprint type in the IPC Footprint Wizard

7. Refer to information in Figure 8 for dimensions. Note that it will be created with 8 round-ended pads at this point if we finish the wizard.

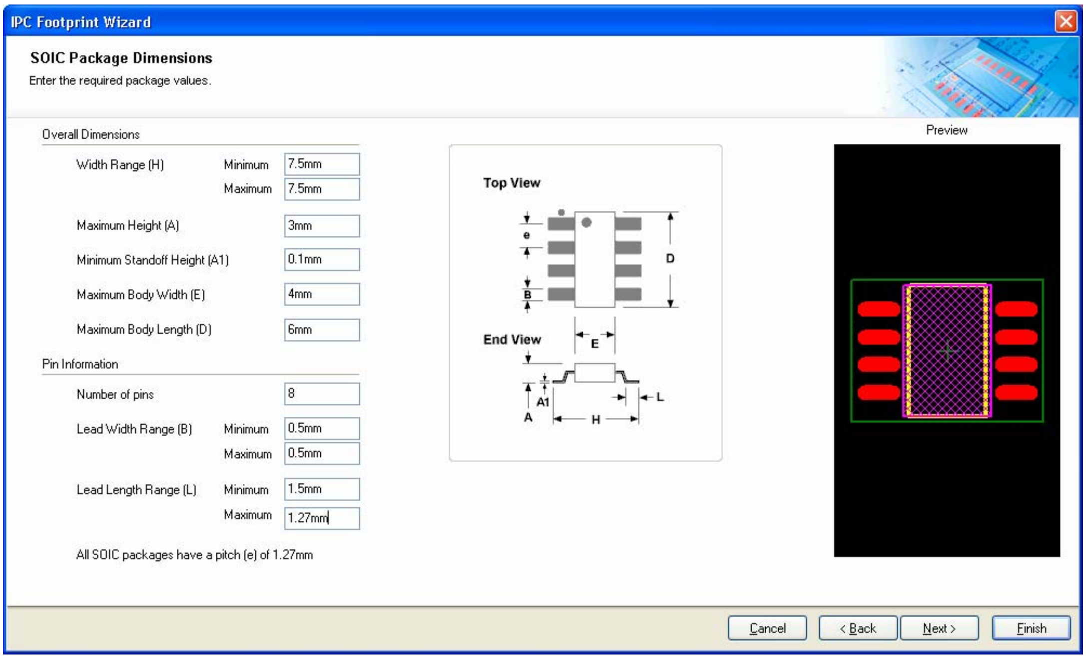

Figure 8.SOIC package dimensions

8. Step through the Thermal Pad dimensions, Heel spacing, Solder fillets, Component Tolerances and IPC Tolerances.
9. In the SOIC Footprint Dimensions page change the pad shape to Rectangular.
10. In the SOIC Silkscreen Dimensions page change the silkscreen line width to 0.1mm.
11. Step through the Courtyard, assembly and component board information page.
12. In the footprint Description page change the name to SOIC8 and leave the description as is.

13. In the Footprint Destination page, set to current PcbLib file and set it to `\Program Files\Altium Designer Summer 09\Examples\Training\PCB Training\Temperature Sensor\Libraries\Temperature Sensor.PcbLib`

14. Click on finish to end the wizard and create the new component as per Figure 9.

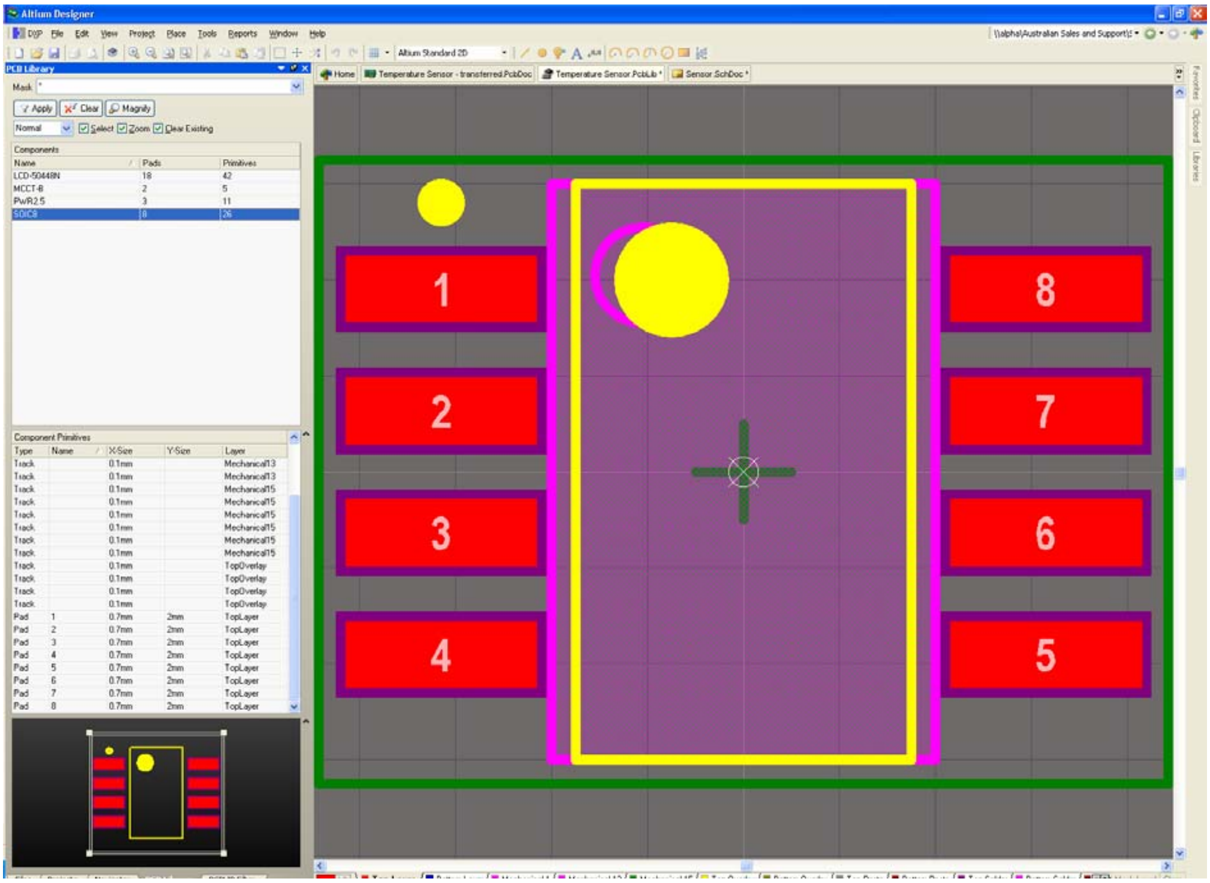

Figure 9. Finished footprint created in the temperature sensor.pcblib.

15. Save the library, and the project.

16.1.11 Browsing footprint libraries

PCB libraries are accessed through the same panel as schematic libraries – the **Libraries** panel.

- Enable the footprint display mode by clicking the button at the top of the panel and enabling the **Footprints** checkbox.
- Select a library name in the drop down list to choose it and display all the footprints in that library. This can be either an integrated library or a footprint library.
- Footprint libraries that are in the active project, currently installed or found down the search path are available in the panel.
- Click the **Libraries** button at the top panel to install a footprint library.
- Library search paths are defined in the **Search Path** tab of the *Options for Project* dialog.
- To Search for a footprint, first enable the **Footprints** mode, then click the **Search** button.
- Click on a footprint name in the list to display that footprint in the MiniViewer.
- Click on the **Place** button to place the chosen footprint in the workspace, or double-click on the footprint name.

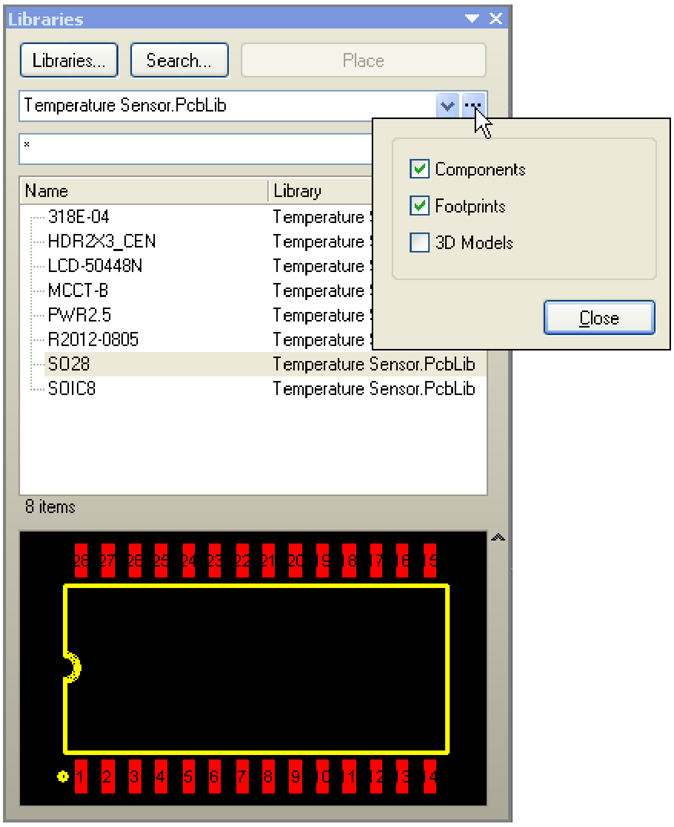

Figure 10. Libraries panel

16.1.12 Creating footprints with an irregular pad shape

There will be situations where you need to create a footprint with pads that have an irregular shape. This can be done using any of the design objects available in the library editor, but there is an important factor that you must keep in mind.

The software automatically creates solder and paste masks based on the shape of pad objects, if you use pad objects to build up an irregular shape then the matching irregular mask shape will be generated correctly. If you build the irregular shape from other objects, such as lines (tracks), fills, regions or arcs, then you will also need to define any required solder or paste masks by placing suitably enlarged or contracted objects on the solder mask and paste mask layers.

Figure 11 and Figure 12 shows versions of an SOT-89 footprint created by different designers. Figure 11 uses 2 pads to create the large irregular shaped pad in the center, Figure 12 uses a pad and a line (track). Figure 12 would need to have the solder and paste masks defined manually.

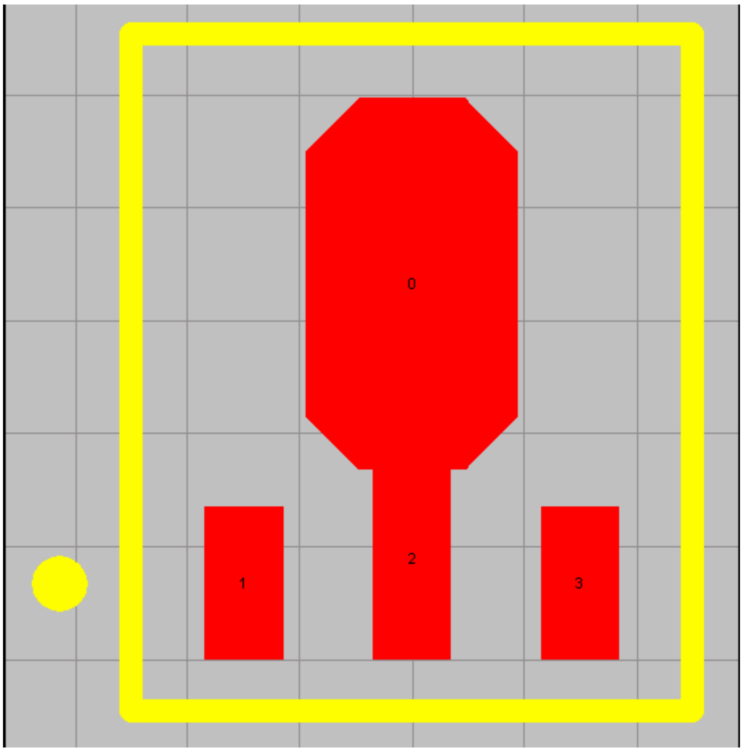

Figure 11. Using pads to create irregular shapes

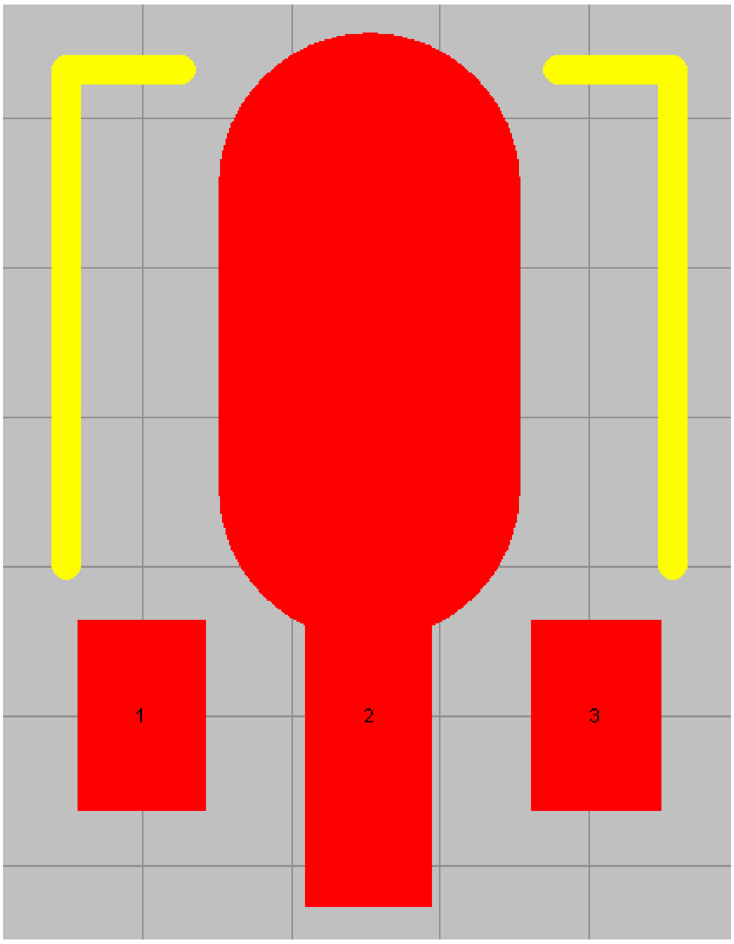

Figure 12. Using a track to create irregular shapes

16.1.13 Managing components that include routing primitives in their footprint

When you transfer a design, the footprint specified in each component is extracted from the available libraries and placed on the board. Then each pad in the footprint has its net property set to the name of the net connected to that component pin in the schematic. If the footprint includes copper primitives touching the pads, these primitives will not be assigned the net name automatically, and will create a design rule violation. In this case, you will need to perform an update process to assign the net name.

The PCB editor includes a comprehensive net management tool, to launch it select **Design » Netlist » Configure Physical Nets** from the PCB editor menus. Figure 13 shows the **Configure Physical Nets** dialog being used to update the extra primitives detected in the switch footprint shown in Figure 14. Click the **Menu** button for a menu of options, and click the **New Net Name** region to select the net to assign to the unassigned primitives.

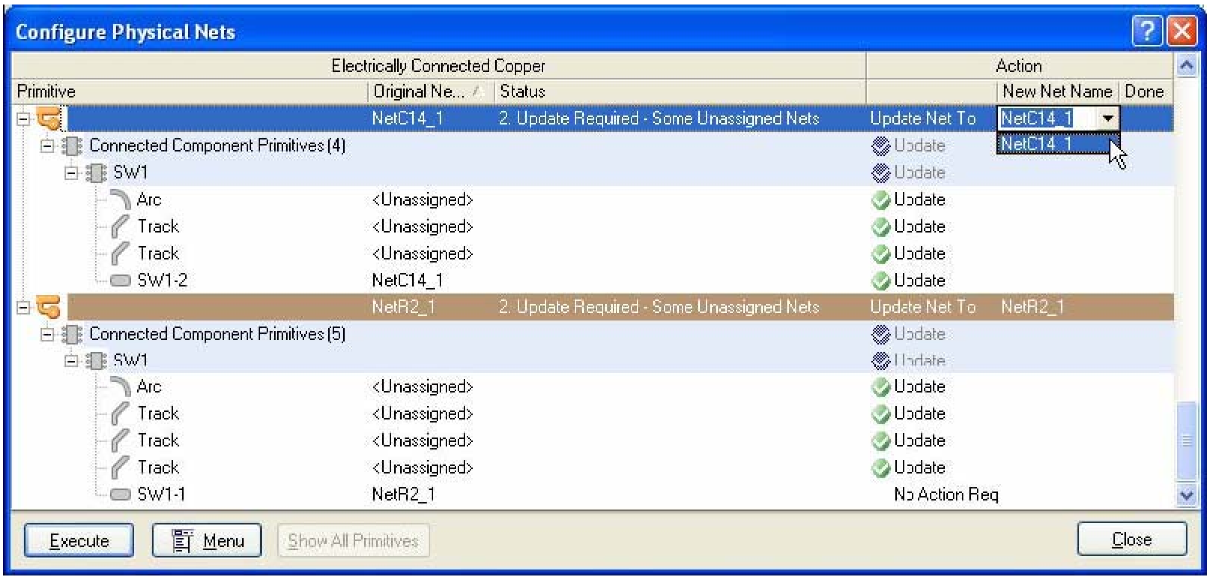

Figure 13. Update the net name on unnamed footprint primitives in the Configure Physical Nets dialog.

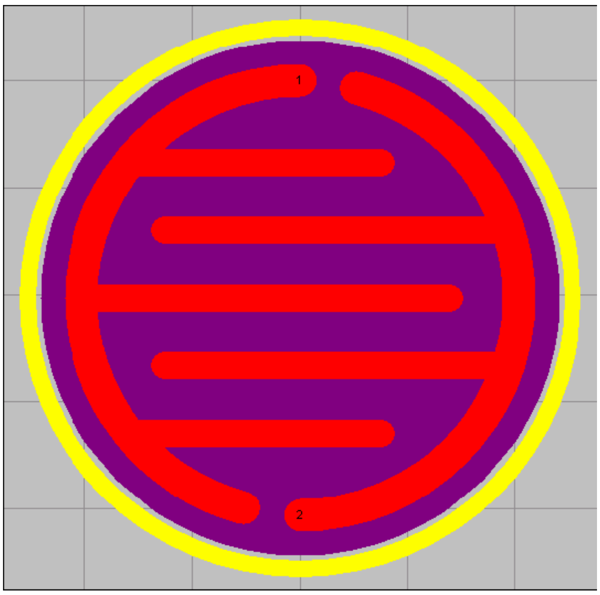

Figure 14. Printed push button footprint, designed by placing pads, lines and arcs.

16.1.14 Footprints with multiple pads connected to the same pin

The footprint shown in Figure 15, a TO-3 transistor, has multiple pads that connected to the same logical schematic component pin. For this component both of the 2 mounting hole pads have 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 will show the connection lines going to both pads in the PCB Editor, i.e. they are on the same net, as shown in Figure 15. Both of these can be routed.

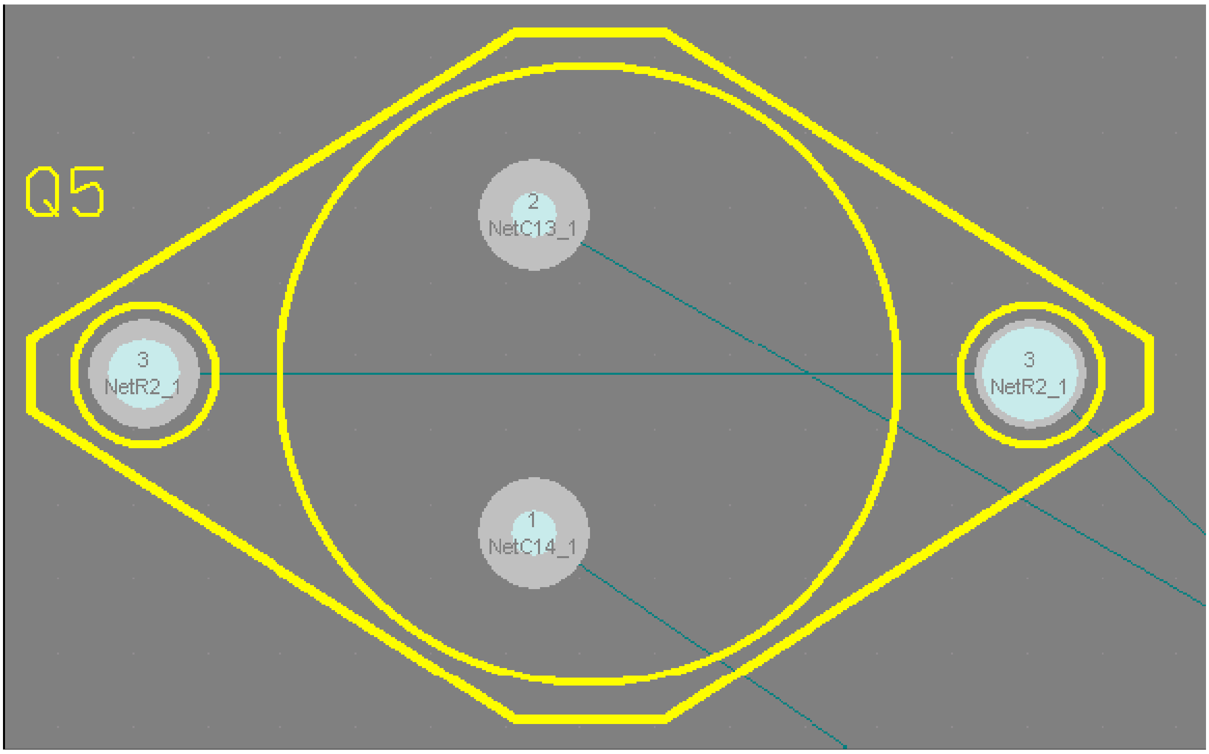

Figure 15. TO-3 footprint showing two pads with a designator of 3, on the same net

16.1.15 Handling special solder mask requirements

The footprint shown in Figure 14 is the contact set for a push button switch, which is implemented directly in the copper on the surface layer of the PCB.

A rubber switchpad overlay is placed on top of the PCB, with a small captive carbon button that contacts both sets of fingers in the footprint when the button is pressed, creating connectivity.

For this to happen both sets of fingers must not be covered by the solder mask. The circular solder mask opening has been achieved by placing an arc whose width is equal to or greater than the arc radius, resulting in the solid circle shown behind the 2 sets of fingers.

Each set of copper fingers has been defined by an arc, horizontal lines, and a pad. The pads are required to define the points of connectivity.

Note that manually placed solder mask definitions are automatically transferred when the component is placed on the bottom of the board.

16.1.16 Other footprint attributes

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.

When you edit a pad you will see the settings for the solder mask and paste mask expansions. While these settings are included to give you localized control of the expansion requirements of a pad, you will not normally need them. Generally it is easier to control the paste mask and solder mask requirements by defining the appropriate design rules in the PCB editor. Using rules you define one rule to set the expansion for all components on the board, then if required you can add other rules that target any specific situations – such as all instances of a specific footprint type used on the board, or a specific pad on a specific component, and so on.

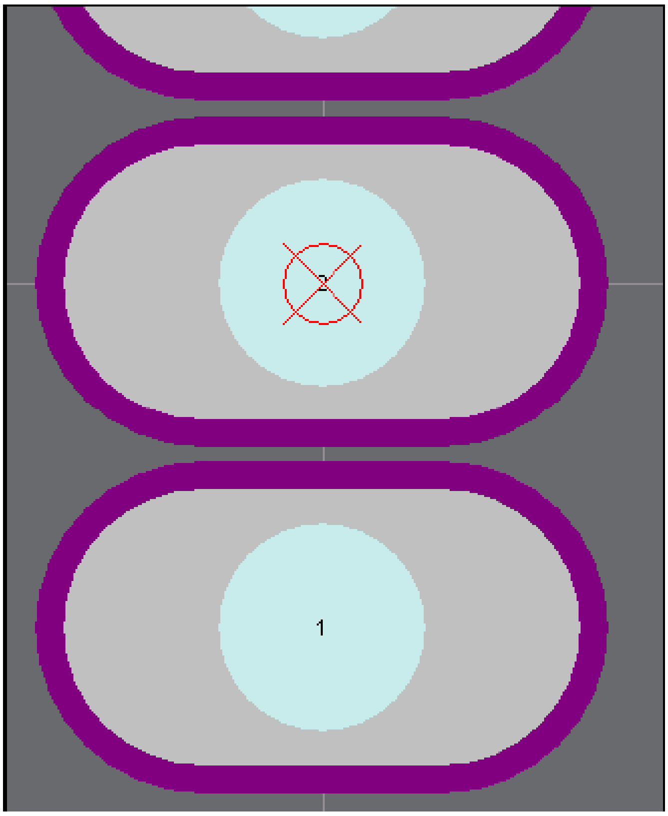

Figure 16. Pads with the solder mask displayed

16.2 3D dimensional component detail

Given the density and complexity of today's electronic products, today's PCB designer must consider more than the horizontal component clearance requirements, they must also consider height restrictions and component-under-component placement options. There is also the need to transfer the final PCB to a mechanical CAD tool, where a virtual product assembly can verify the complete packaging of the product being developed. Altium Designer includes a number of features to cater for these different situations.

16.2.1 Adding height to your PCB footprint

At the simplest level you can add a height attribute to your footprint. To do this double-click on the footprint in the Components list in the **PCB Library** panel to display the **PCB Library Components** dialog and enter the recommended height for the component in the **Height** field.

Height Constraint design rules can then be defined during board design (select **Design » Rules** in the PCB Editor), typically testing for maximum component height in a class of components, or within a room definition.

16.2.2 Adding a 3D body to a footprint

For more detailed height requirements, you can add a 3D body object to the footprint. A 3D Body is a polygonal shaped object that can be added to a footprint on any enabled mechanical layer. One or more 3D body objects can be added to define the physical size and shape of a component in both the horizontal and vertical planes. The 3D body objects can then be used by the component clearance design rule check to test for component collisions, and they can also be used by the 3D visualization engine when it renders a 3D representation of the board (**View » Board in 3D** in the PCB Editor).

16.2.3 Manually placing 3D body objects

3D body objects can be placed manually in the PCB Library Editor (**Place » 3D body**). They can also be added automatically to footprints in the PCB Library Editor (and to placed footprints in the PCB Editor) using the **Component Body Manager** dialog (**Tools » Manage 3D Bodies for Components on Board…**). Note that can only be placed on a mechanical layer, the current mechanical layer is used if one is active when the command is selected, otherwise the 1st available mechanical layer will be selected in the *3D Body* dialog. The steps involved in setting this up for an example footprint like a DIP14 would include:

1. Select the footprint you wish to add a 3D body to using the PCB library panel.
2. Confirm that a mechanical layer is enabled and is the current layer.
3. Select **Place » Place 3D body** [shortcut P, B]. The *3D body* dialog will open, as shown in Figure 17.

> **Note:** 3D body objects can be created from shapes (extruded rectangular, cylindrical or spherical), or from an imported STEP model. You can also use a combination of both, if needed.

4. Enter overall height and stand off height information, and close the dialog.
5. Click to define the vertices. Note that the placement process for en extruded 3D body is the standard polygonal object placement process, use **Shift+Spacebar** to cycle the corner style, and **Spacebar** to toggle the current placement corner.

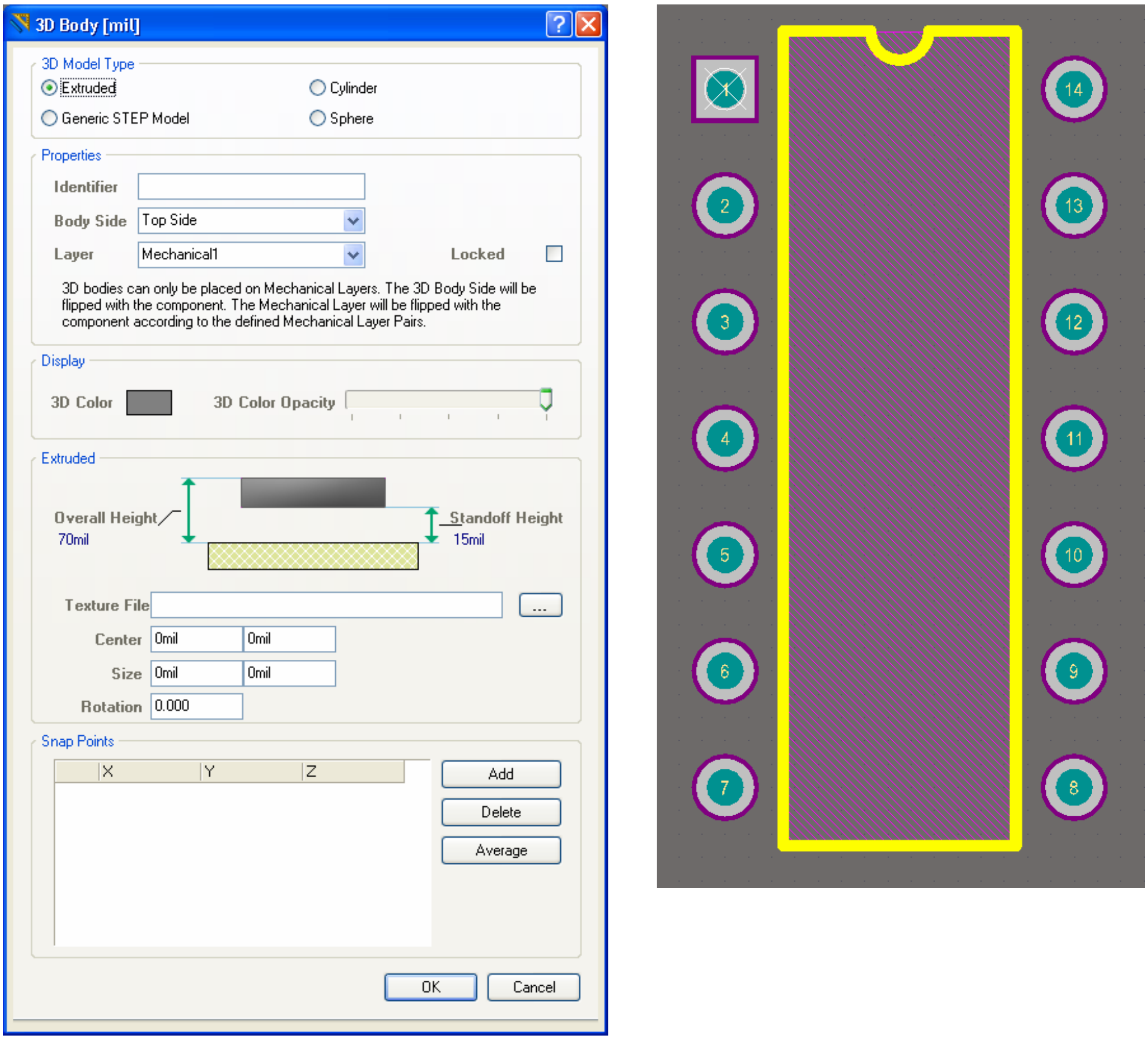

Figure 17. Define the 3D body overall height and standoff height. The Body after it has been placed on the DIP14.

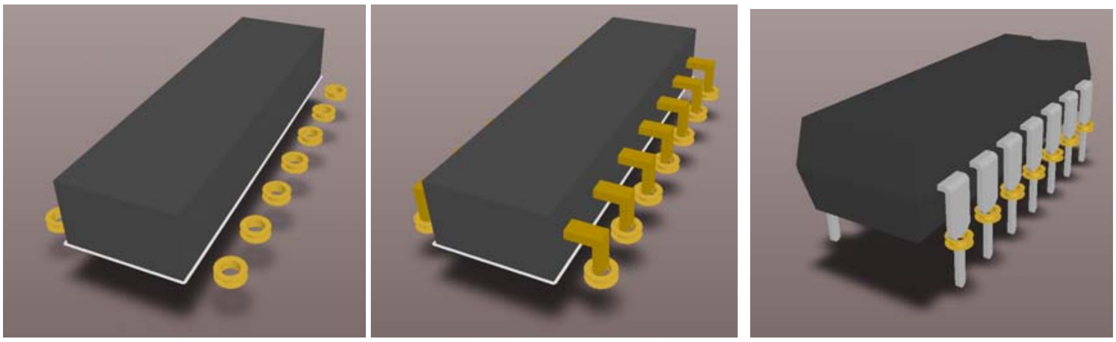

Figure 18. DIP14 footprint with a single 3D body, then with multiple 3D bodies added to define the pins, then with an imported STEP model for a DIP14.

16.2.4 Using the 3D Body Manager

Using the 3D Body Manager, 3D body objects can be automatically created based on the bounding rectangles or closed polygonal outlines of primitives that already exist in the footprint. The 3D Body Manager can be used for the current footprint, or across the entire PCB library.

In Figure 19 you can see the **3D Body Manager** used to define a 3D body for the transistor footprint, SOIC8. Using this approach is much easier than attempting to define the shape manually, because of the curved edge of the transistor body.

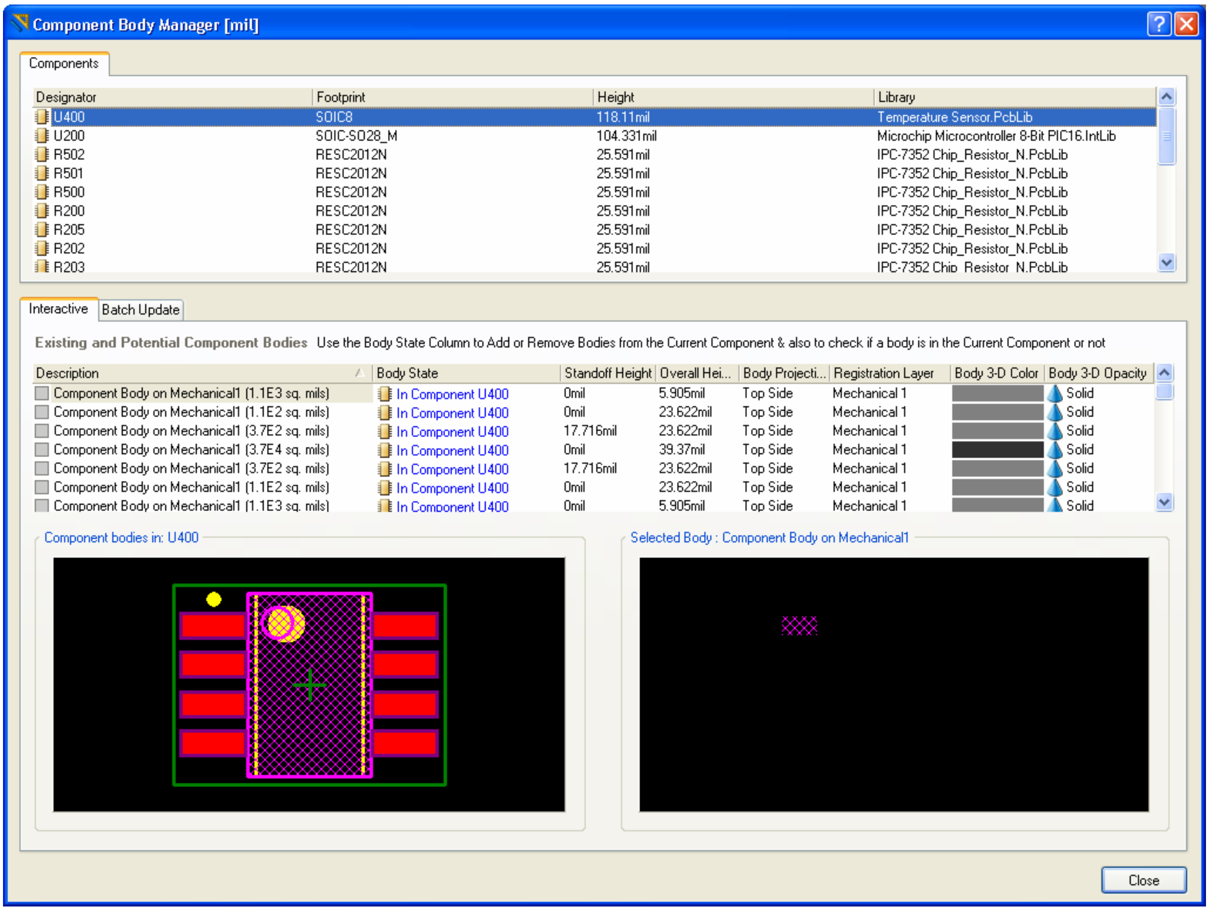

Figure 19. Use the 3D Body Manager to quickly create body objects based on existing primitives.

To create a shape that follows the outline defined on the component overlay click the second option that appears in the list, **Polygonal shape created from primitives on TopOverlay**. For this line in the dialog, click on the **Action** button **Add to component_name**), set the **Registration Layer** to the mechanical layer that the body object should be placed on (mechanical layer 3 has been renamed BodyTop in this example), and set the **Overall Height** to a suitable value, for example `200mil`, as shown in Figure 19.

> **Note:** Multiple 3D body shapes can be added to build up a 3D body using the dialog in 19. To do this, simply pick and choose which bodies you want from the action column to add. You can also mix this with manually placed bodies to build up a complex 3D model like for example the 16x2 character screen on the Altium NanoBoard, or the Spirit Level board.

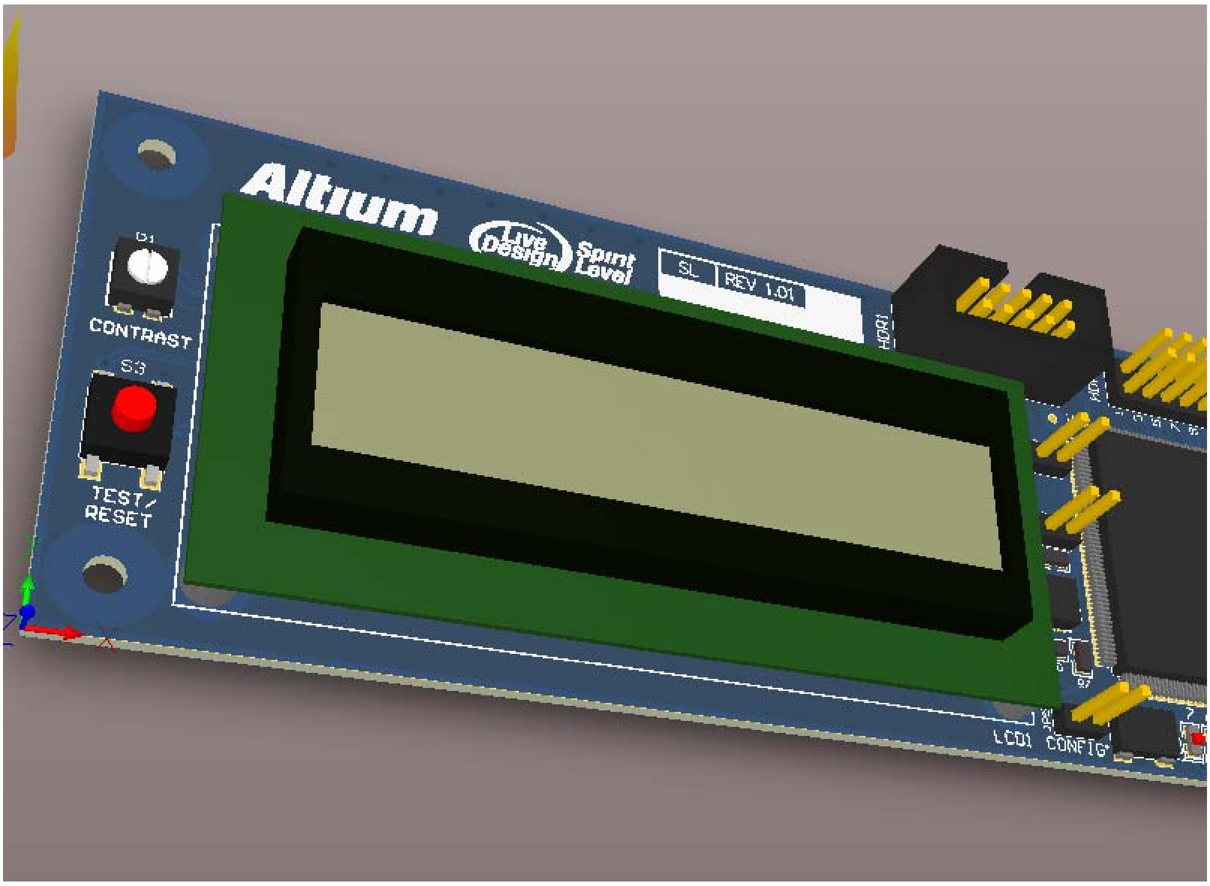

Figure 20. End result in the 3D view of a complex 3D body created in the PCB library editor.