

# ***Altium*** ***Designer***

---

## **Module 15: Schematic Library Editor**

---

## Module 15: Schematic Library Editor

<b>15.1 Introduction to Library Editing .....</b>	<b>15-1</b>
<b>15.2 Schematic Library Editor .....</b>	<b>15-2</b>
15.2.1 Schematic Library Editing Tools.....	15-3
15.2.2 Schematic Library Editor Terminology .....	15-3
15.2.3 Component Properties .....	15-5
15.2.4 Using supplier data.....	15-6
15.2.5 Exercise – Creating a new component symbol.....	15-10
15.2.6 Exercise – Completing the Sensor schematic .....	15-12
15.2.7 Creating a multi-part component.....	15-14

Software, documentation and related materials:

Copyright © 2009 Altium Limited.

All rights reserved. You are permitted to print this document provided that (1) the use of such is for personal use only and will not be copied or posted on any network computer or broadcast in any media, and (2) no modifications of the document is made. Unauthorized duplication, in whole or part, of this document by any means, mechanical or electronic, including translation into another language, except for brief excerpts in published reviews, is prohibited without the express written permission of Altium Limited. Unauthorized duplication of this work may also be prohibited by local statute. Violators may be subject to both criminal and civil penalties, including fines and/or imprisonment. Altium, Altium Designer, Board Insight, Design Explorer, DXP, LiveDesign, NanoBoard, NanoTalk, P-CAD, SimCode, Situs, TASKING, and Topological Autorouting and their respective logos are trademarks or registered trademarks of Altium Limited or its subsidiaries. All other registered or unregistered trademarks referenced herein are the property of their respective owners and no trademark rights to the same are claimed.

Module Seq = 15

---

## 15.1 Introduction to Library Editing

The Creating Library Components section explains the various types of libraries, and the fundamental way in which library documents are created and managed in Altium Designer.

The process of using the Schematic and PCB Library editors will be explored with hands-on exercises showing how to create single and multi-part packages in a schematic library, creating PCB footprints, attaching models, and working with separate library files. It will also introduce Integrated Libraries, a compiled, secure and portable form of library file.

This section will also discuss adding component parameters and provide hands-on exercises on importing pin information from vendor pin lists.

Figure 1. Outlines the workflow to follow when creating component libraries in Altium Designer.

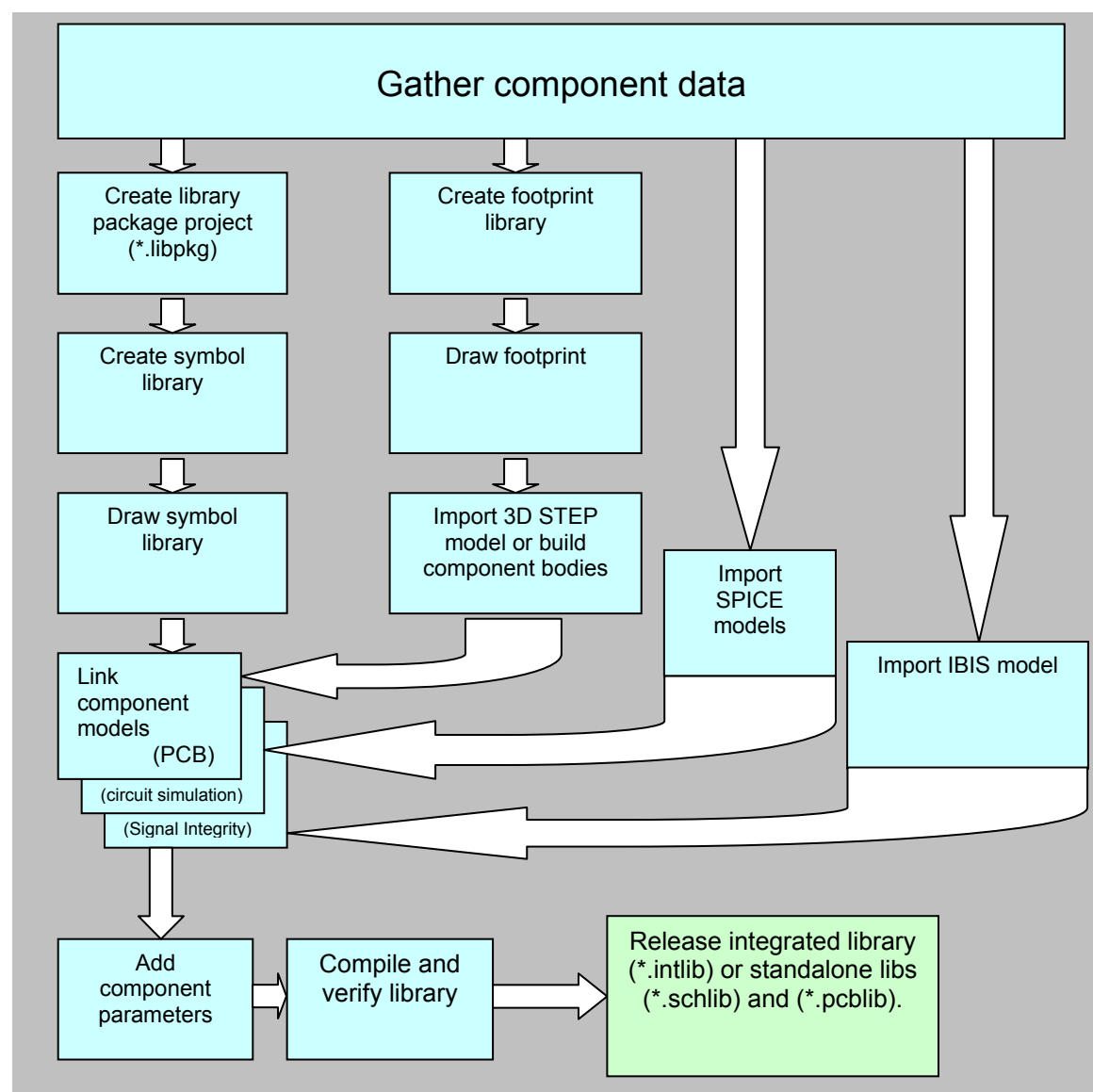


Figure 1. The Altium Designer component creation workflow.

## 15.2 Schematic Library Editor

This section covers how to use the Schematic Library Editor and how to create a new component. The Schematic Library Editor is used to:

- Create and modify schematic component symbols
- Attach models to the component
- Add parameters to the component
- Manage component libraries.

The Schematic Library Editor is very similar in operation to the Schematic Editor and shares the same graphical object types (but not the electrical objects). In addition, the Schematic Library Editor has one additional object, the Pin, which is used at points where wires connect to components.

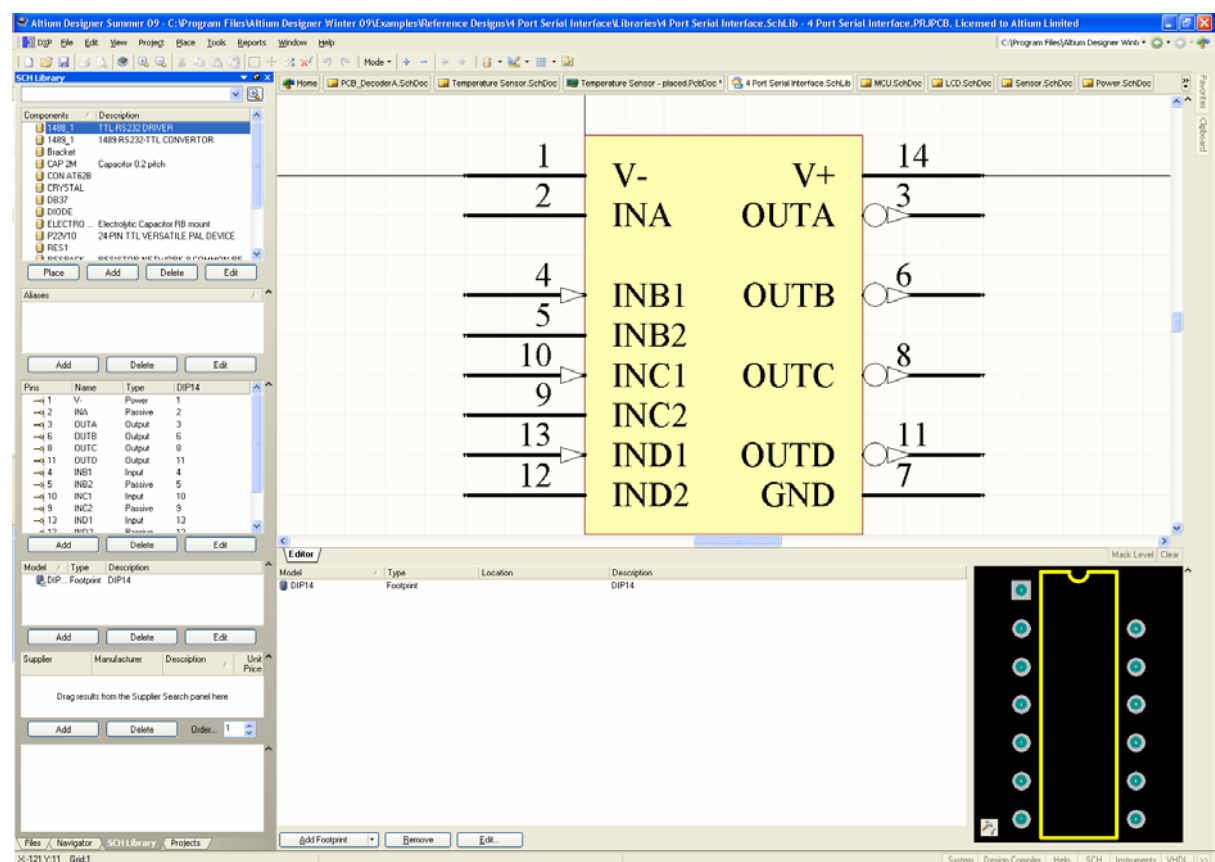


Figure 2. Schematic Library Editor Workspace

- **Schematic Libraries (\*.SchLib)** can be opened for editing using the **File » Open** menu command. Navigate to the folder that the required library is stored in and locate the library, e.g. C:\Program Files\Altium Designer Summer 09\Examples\Training\PCB Training\Temperature Sensor\Libraries\Temperature Sensor.SchLib and click on **Open**.
- **Integrated Libraries (\*.IntLib)** are compiled binary files, which cannot be edited. If you attempt to open an integrated library, it will be de-compiled, i.e. all the source libraries will be extracted and a new Library Package will be created. All the libraries supplied with the software are integrated libraries.

- **Database Libraries (\*.DBLib)** is a link to a database (ODBC or ADO based) where references are stored for symbol reference, model linking and parameter information. Each record in the database represents a component, storing all of the parameters, along with links to the models. The record can include links to inventory or other corporate component data. With this approach the schematic component is only used as a symbol, with the models (footprint, 3D Model and simulation model) stored in standard schematic library files, PCB library files, and so on. Components are placed from the database by installing a new DBLib document in the **Libraries** panel, with the DBLib document being configured to reference the component database.
- **Subversion Database Libraries (\*.SVNDBLib)** is similar to database libraries however all the linked symbol and footprint data is version controlled within a subversion database. Due to this, each symbol and footprint is stored separately in individual \*.schlib and \*.pcblib files within the SVN database.

### 15.2.1 Schematic Library Editing Tools

- The Schematic Library Editor has a right-click menu, a **Utilities** toolbar and a **Mode** toolbar, as shown in Figure 3.
- The Utilities toolbar includes a range of standard drawing tools, and a comprehensive set of IEEE symbols.

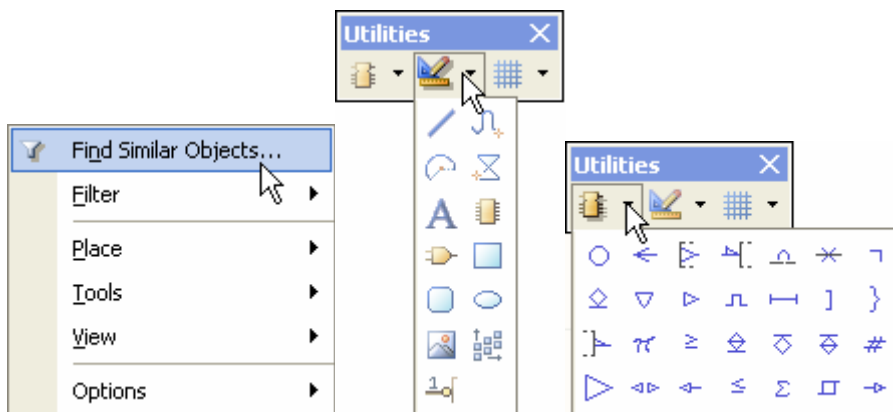


Figure 3. Library Editor Toolbars and right click command options

### 15.2.2 Schematic Library Editor Terminology

- **Object** — any individual item that can be placed in the Schematic Library Editor workspace, for example, a pin, line, arc, polygon, IEEE symbol etc.

**Note:** The IEEE range of symbols can be resized during placement. Press the + and - keys to enlarge and shrink the symbols as you place them.

- **Part** — a collection of graphical objects that represent one part of a multi-part component (e.g. one inverter in a 7404), or a library component in the case of a generic or singly packaged device (e.g. a resistor or an 80486 microprocessor).
- **Part Zero** — this is a special non-visible part available only in multi-part components. Pins added to part zero are automatically added to every part of the component when the component is placed on a schematic. To add a pin to part zero place it on any part, edit it, and set the Part Number attribute in the *Pin Properties* dialog to Zero.
- **Component** — either a single part (e.g. a resistor) or a set of parts that are packaged together (e.g. a 74HCT32).

- **Aliases** – refer to the naming system when a library component has multiple names that share a common component description and graphical image. For example, 74LS04 and 74ACT04 could be aliases of a 7404. Sharing graphical information makes the library more compact. When using database libraries, the use of aliases has become obsolete.
- **Hidden Pins** – these are pins that exist on the component, but do not need to be displayed. Typically, this is done for power pins, which can then be automatically connected to the net specified in the *Pin Properties* dialog. This net does not need to be present on the schematic; one will be created, connecting all hidden pins with the same Connect To net name. The pins will NOT automatically connect if they are visible on the schematic sheet (i.e. un-hidden). Hidden pins can be shown on the schematic sheet by selecting the **Show All Pins** option in the *Component Properties* dialog.
- **Mode** – a component can have up to 255 different display modes. This can be used for things like IEEE component representations, alternate pin arrangements for op-amps, and so on. Use the options in the **Tools » Mode** submenu or the Mode toolbar to add a new mode to a component. The displayed component mode can be changed on the schematic sheet.

### 15.2.2.1 Schematic Library Editor Panel

The Schematic Library Editor panel provides a number of features for working with Schematic components.

- To display the panel, select **SCH » SCH Library** via the panel display buttons at the bottom right of the workspace, when a library is open for editing.
- The buttons below each region of the panel apply to the selected entry in that region.

#### Components section

- Lists all the components in the active library.
- Double-click on a component to open its **Library Component Properties** dialog.
- Use the buttons and the options in the right-click menu to manage the library.

#### Aliases section

- This allows you to add alternate names to a component that share the same graphics and description.

#### Pins section

This section lists the pins in the current component.

- Edit individual pins by double-clicking.
- Select **View » Show Hidden Pins** to display all those pins that are defined as hidden. This does not change the actual pin hidden/unhidden status; rather it only displays the hidden pins in the Library Editor.
- When placing multiple pins with incrementing name/designator, press the **TAB** key after selecting **Place » Pin** from the menus to define the starting value. By default, both the pin number and name will increment.
- Increment behavior can be controlled using the **Auto-Increment during Placement** options in the **Preferences** dialog (the primary value is the pin

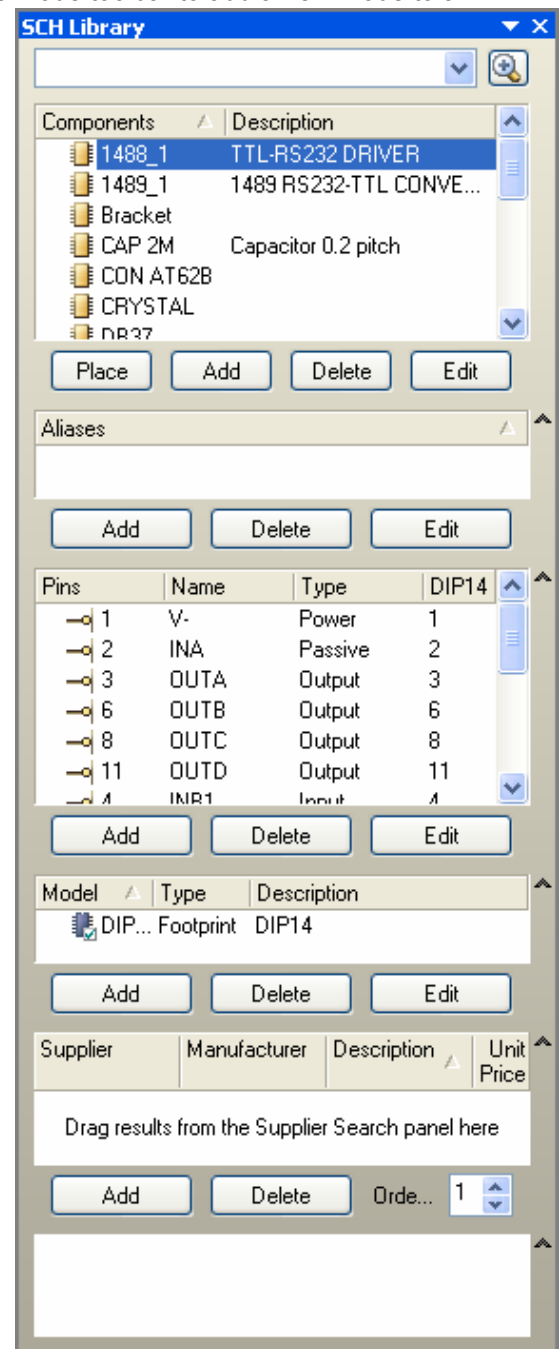


Figure 4. Schematic Library panel

number). Enter a negative sign to decrement a value. Enter an alpha value to increment alphabetically. A single alpha followed by numbers increments the leading alpha. If there are multiple alphas, the last character is incremented/decremented.

- The entire set of pins for the current component can also be viewed and edited in the **List** panel. To filter the component to only show pins, right-click in the graphical area and select **Filter » Examples » Pins** from the floating context menu. If the List panel is not currently visible press **Shift+F12** to display it. Note that you can edit multiple pin properties in the **List panel**, and can also copy and paste to and from a spreadsheet using the **Smart Grid** commands in the List panel right-click menu.

**Note:** If you find that the component pans out of view too often, disable auto pan in the **Schematic – Graphical Editing** page of the *Preferences* dialog. Alternatively, use the **V, F** shortcuts to bring the component back into view.

### 15.2.3 Component Properties

The **Component Properties** dialog is where you add model and parameter information to the component symbol. Double-click on a component name in the **Sch Library panel** to display the dialog.

**Library Component Properties**

**Properties**

Default Designator:  ☒ Visible ☐ Locked

Comment:  ☒ Visible

Description:   ☒ Locked

Type:

**Library Link**

Physical Component:

**Graphical**

Mode:  ☒ Lock Pins

☐ Show All Pins On Sheet (Even if Hidden)

☐ Local Colors

**Parameters for EP1K10FC256-2**

Visible	Name	Value	Type
<input type="checkbox"/>	Code_IPC	PBGA 17x17 FE256	STRING
<input type="checkbox"/>	Code_JEDEC	MS-034-AAF-1	STRING
<input type="checkbox"/>	ComponentLink1Descriptor	Manufacturer Link	STRING
<input type="checkbox"/>	ComponentLink1URL	http://www.altera.com/products/dev	STRING
<input type="checkbox"/>	ComponentLink2Descriptor	Datasheet	STRING
<input type="checkbox"/>	ComponentLink2URL	http://www.altera.com/literature/ds/	STRING
<input type="checkbox"/>	DatasheetDocument	May-2003	STRING
<input type="checkbox"/>	LatestRevisionDate	19-Mar-2004	STRING
<input type="checkbox"/>	LatestRevisionNote	Alternate schematic symbols removed	STRING
<input type="checkbox"/>	PackageDocument	ver13.0, May-2005	STRING
<input type="checkbox"/>	PackageReference	FBGA256	STRING
<input type="checkbox"/>	Published	17-Jul-2002	STRING
<input type="checkbox"/>	Publisher	Altium Limited	STRING

**Models for EP1K10FC256-2**

Name	Type	Description
EP1K10-FBGA256	PCB3D	256-Pin FineLine Ball-Grid Array (FBGA) - Option 1
FBGA256	Footprint	256-Pin FineLine Ball-Grid Array (FBGA) - Option 1

**Buttons:** Add..., Remove..., Edit..., Add as Rule..., Edit Pins..., OK, Cancel


Figure 5. Component Properties dialog

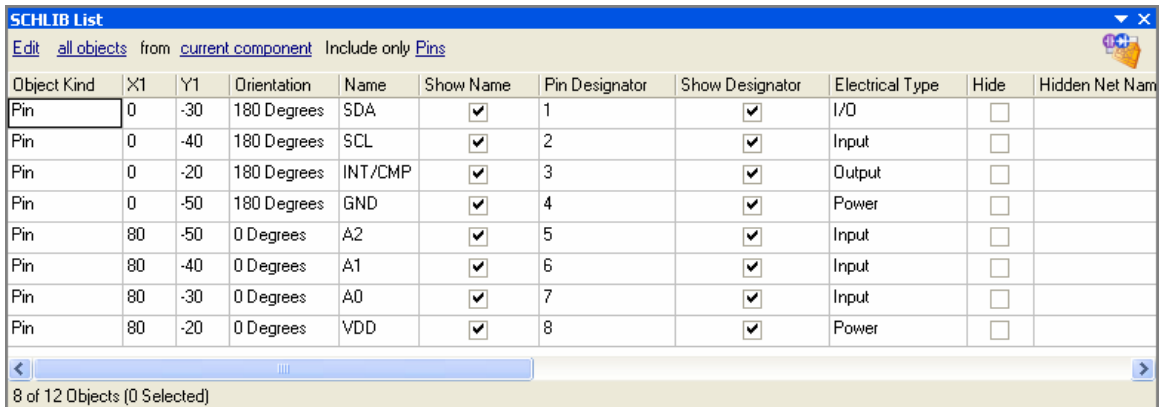
Information that would typically be defined for a component includes:

- Default Designator** – defines the prefix string to be used with the component designator.
- Comment** – description of the component. For a component whose definition is fixed, such as a 74HC32, this standard descriptive string would be entered. For a discrete component whose value can change, such as a resistor, the value would be entered. Note that this field supports indirection, which allows you to display the value of any of this component's parameters. Indirection is enabled by entering an equals sign, then the parameter name (spaces are not

supported). If this field is left blank, the component library reference will be entered as the comment when the component is placed, allowing you to define the comment after it has been placed on the schematic.

- **Description** – meaningful description that can be used for searching and in the BOM.
- **Type** – alternate component types are provided for special circumstances. Graphical components do not get synchronized or included in the BOM. Mechanical types only get synchronized if they exist on both the schematic and the PCB and do get included in the BOM. Net Tie components are used for shorting two or more nets on the PCB.
- **Parameters** – any number of parameters can be added either in the Library Editor, or on the schematic sheet. Parameters can be linked to a company database; add a database link document to the project to do this.
- **Models** – various component models can be added, including footprint, simulation, signal integrity, and so on.
- **Lock Pins** – if this option is enabled, you will not be able to edit pins, only the component as a whole entity, when the component is placed on a schematic. Disable this option if you wish to edit the pins and click on the **Edit Pins** button.

**Note:** Use the What's This Help  for more information about options in the dialog.



Object Kind	X1	Y1	Orientation	Name	Show Name	Pin Designator	Show Designator	Electrical Type	Hide	Hidden Net Nam
Pin	0	-30	180 Degrees	SDA	<input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>	I/O	<input type="checkbox"/>	
Pin	0	-40	180 Degrees	SCL	<input checked="" type="checkbox"/>	2	<input checked="" type="checkbox"/>	Input	<input type="checkbox"/>	
Pin	0	-20	180 Degrees	INT /CMP	<input checked="" type="checkbox"/>	3	<input checked="" type="checkbox"/>	Output	<input type="checkbox"/>	
Pin	0	-50	180 Degrees	GND	<input checked="" type="checkbox"/>	4	<input checked="" type="checkbox"/>	Power	<input type="checkbox"/>	
Pin	80	-50	0 Degrees	A2	<input checked="" type="checkbox"/>	5	<input checked="" type="checkbox"/>	Input	<input type="checkbox"/>	
Pin	80	-40	0 Degrees	A1	<input checked="" type="checkbox"/>	6	<input checked="" type="checkbox"/>	Input	<input type="checkbox"/>	
Pin	80	-30	0 Degrees	A0	<input checked="" type="checkbox"/>	7	<input checked="" type="checkbox"/>	Input	<input type="checkbox"/>	
Pin	80	-20	0 Degrees	VDD	<input checked="" type="checkbox"/>	8	<input checked="" type="checkbox"/>	Power	<input type="checkbox"/>	

8 of 12 Objects (0 Selected)

Figure 6. Use the List panel to view or edit component pins

## 15.2.4 Using supplier data

Creating a live link between an Altium Designer component and a Supplier Item has always been a simple process and importing information from an item is equally so. Simply select the parameter(s), data sheet link(s), pricing information, or stock information that you wish to import, in the detailed information section of the **Supplier Search** panel, then drag and drop:

- Anywhere within the main editing area of the Schematic Library Editor – ensuring the source Schematic Library document is the active document in the main design window and the recipient component is focused. You can also add information to a component by dropping onto the component's name (in the **Components** region of the **SCH Library** panel).
- Onto the required component record, in the **Table Browser** tab of the relevant Database Library file or SVN Database Library file – ensuring the DbLib or SVNDbLib file is the active document in the main design window.
- Onto the schematic symbol of a placed component – ensuring the source schematic document is the active document in the main design window.

For parameters, pricing and stock information, the import will proceed in accordance with any defined parameter import options.

The following sections take a look at the import of these different pieces of information.



### 15.2.4.1 Importing Parameters

Consider, as an example, the case where we have a component defined in a Schematic Library, which represents an N-channel MOSFET (a 2N7002). We have already linked a Supplier Item to this component and now need to import the parameters.

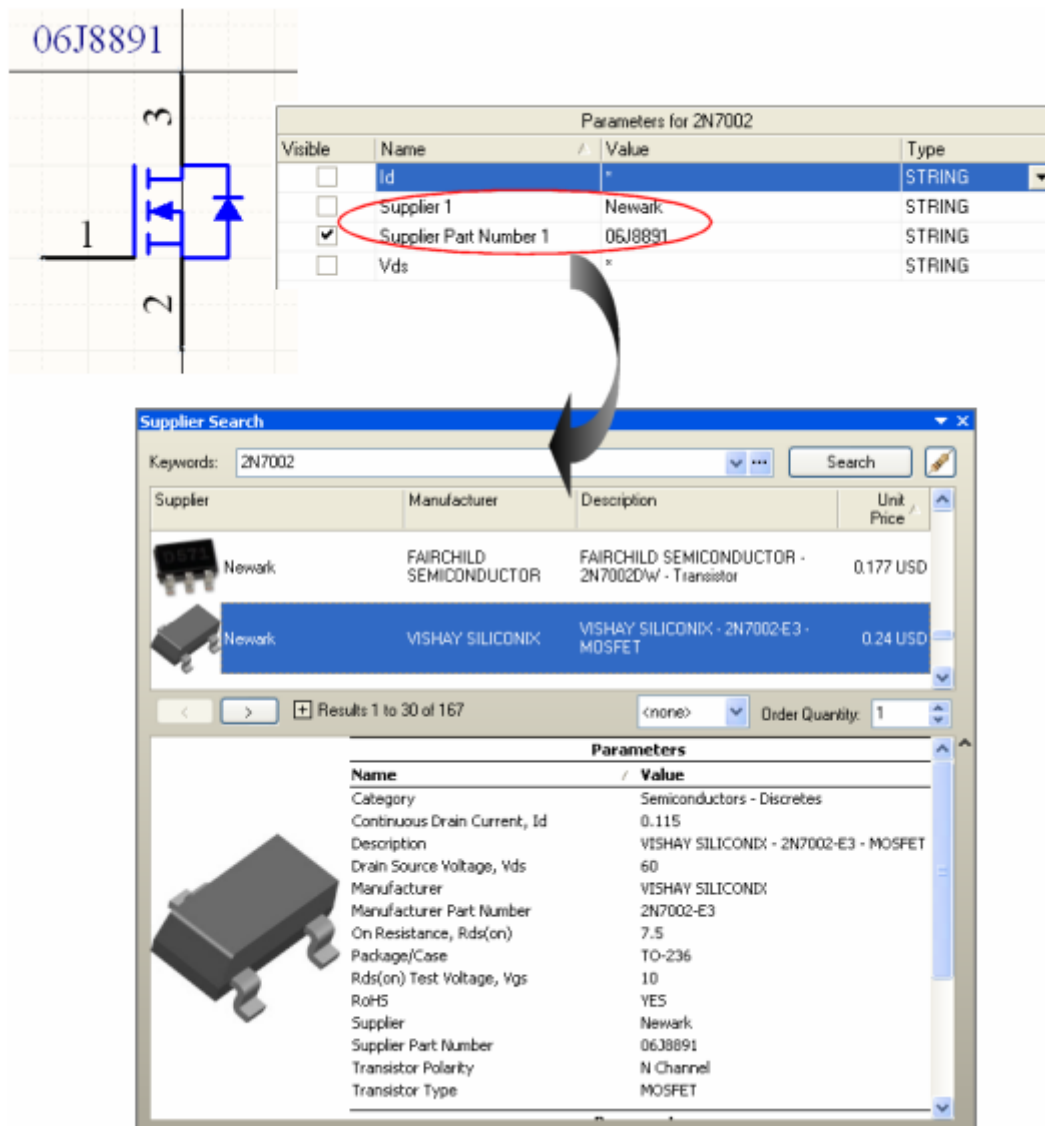


Figure 7. The example library component and the linked Supplier Item.

For the purposes of demonstrating the feature, a couple of parameters have already been defined for the component in the library – the Continuous Drain Current (Id) and the Drain Source Voltage (Vds). The Supplier Item names these parameters differently, so let's set up some parameter name mapping to ensure the data comes in to our existing parameters in these two cases.

It's a simple case of adding the two parameters – named as per the Supplier Data area – and using our naming for the **Imported Parameter Name** fields.

**Parameter Import Options**

Add any parameter names here that you would like renamed on import or made visible. Parameters imported along with supplier links or for new components can also be excluded or suffixed.

Supplier Parameter Name	Imported Parameter Name	Visible	Exclude	Suffix
Category	Category	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Continuous Drain Current, Id	Id	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Description	Description	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Drain Source Voltage, Vds	Vds	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Manufacturer	Manufacturer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Manufacturer Part Number	Manufacturer Part Number	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Pricing	Pricing	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
RoHS	RoHS	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Stock	Stock	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Supplier	Supplier	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Supplier Part Number	Supplier Part Number	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

**Add** **Remove**

Figure 8. Define parameter name mappings to ensure the existing parameters for the component get used, rather than adding new parameters for this same data.

Now the options are set, we need to just select the parameters we want to import, in the detailed **Parameters** section for the Supplier Item. As this Supplier Item is already linked to the component, we don't need to import the *Supplier* or *Supplier Part Number* parameters. Let's import all others. Once selected, just drag the selection onto the component's name – in the **Components** region of the **SCH Library** panel – or within the main editing area for the component itself. That's all there is to it – a quick check in the properties dialog for the component shows that the parameters have been imported, and the two existing parameters have been used to receive data as required!

Visible	Name	Value	Type
<input type="checkbox"/>	Category	Semiconductors - Discretes	STRING
<input type="checkbox"/>	Description	VISHAY SILICONIX - 2N7002-E3 - MOSFET	STRING
<input type="checkbox"/>	Id	0.115	STRING
<input type="checkbox"/>	Manufacturer	VISHAY SILICONIX	STRING
<input type="checkbox"/>	Manufacturer Part Number	2N7002-E3	STRING
<input type="checkbox"/>	On Resistance, Rds(on)	7.5	STRING
<input type="checkbox"/>	Package/Case	TO-236	STRING
<input type="checkbox"/>	Rds(on) Test Voltage, Vgs	10	STRING
<input type="checkbox"/>	RoHS	YES	STRING
<input type="checkbox"/>	Supplier 1	Newark	STRING
<input checked="" type="checkbox"/>	Supplier Part Number 1	06J8891	STRING
<input type="checkbox"/>	Transistor Polarity	N Channel	STRING
<input type="checkbox"/>	Transistor Type	MOSFET	STRING
<input type="checkbox"/>	Vds	60	STRING

Figure 9. Parameter import in action!

#### Notes:

Parameters can be imported without setting up a prior live link to the Supplier Item, however importing the *Supplier* and *Supplier Part Number* parameters in this manual, drag and drop fashion, will not create a live link to the item.

Another way to initiate the import of parameters is to select the required parameter(s) in the detailed Parameters section for the Supplier Item, right click and choose the command from the context menu that appears. If a Schematic Library, Database Library or SVN Database Library is active, the command will appear in the format *Add Parameters To ComponentName*, where *ComponentName* is the currently focused component/component record in the library. If a Schematic document is active, the command will appear in the format *Add Parameters To Part*. Simply click on each placed component to which you want to add the parameters using the cross hair cursor. Right-click or press Esc to exit.

All importable data for a Supplier Item can be imported, along with a new supplier link to that item, in one step. Simply right-click on the Supplier Item entry in the Supplier Search panel and choose the *Add Supplier Link And Parameters To* command (see [Adding a Supplier Link and Parameters Simultaneously](#)).

#### 15.2.4.2 Importing Data Sheet Links

The process for importing a data sheet link from a Supplier Item is the same as that for importing parameters. The only difference is that there are no options to set up beforehand – just click and drag the data sheet link from the detailed **Documents** section for the Supplier Item onto the target component/component record in the active library (SchLib, DbLib, SVNDbLib), or component on the active schematic sheet as required.

The process will add a *ComponentLink* parameter pair, targeting the data sheet. By default, the same value will be entered for both the *ComponentLinknDescription* and *ComponentLinknURL* parameters – the URL for the data sheet. Access the link from the usual **References** sub-menu for the component.

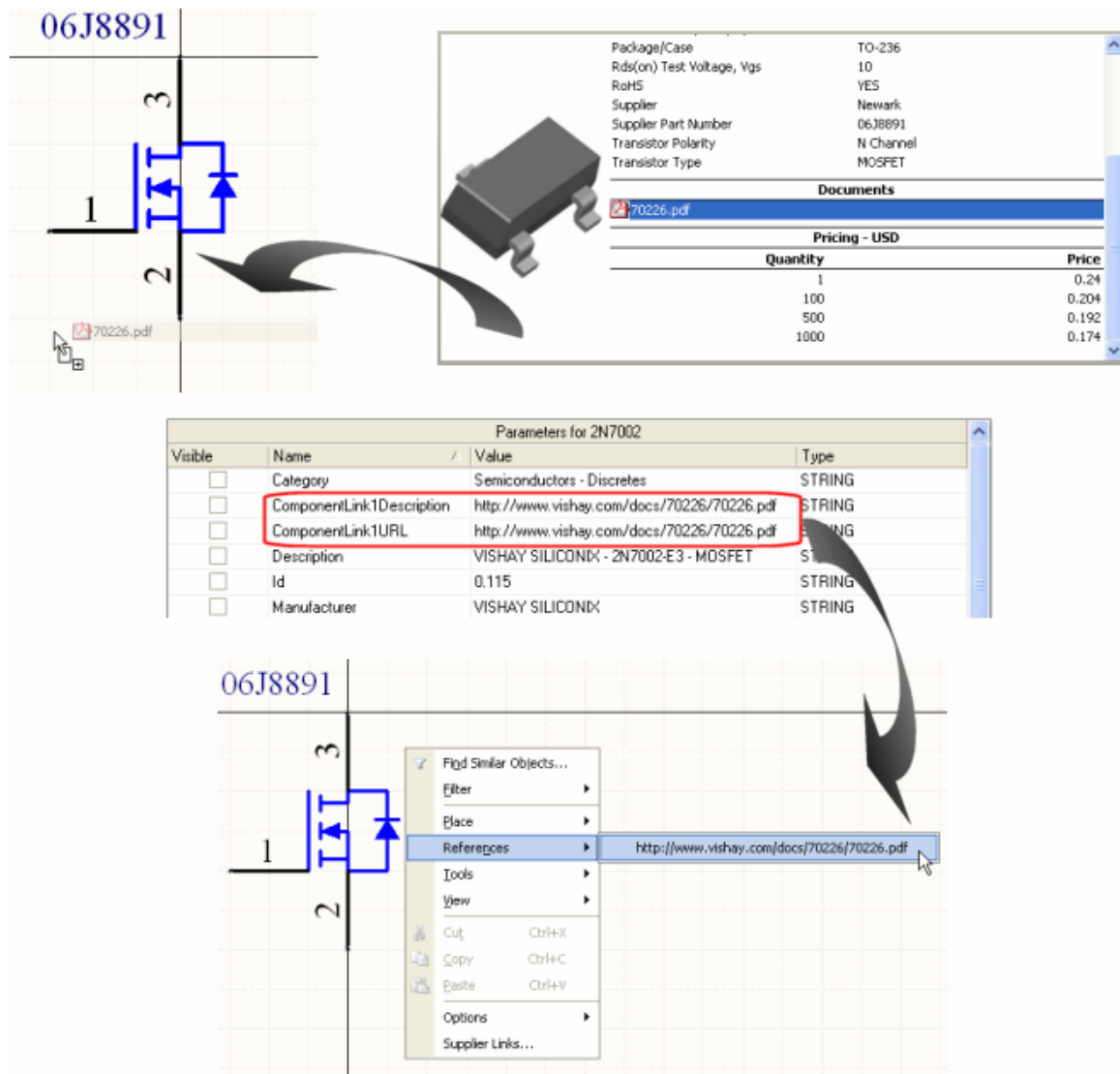


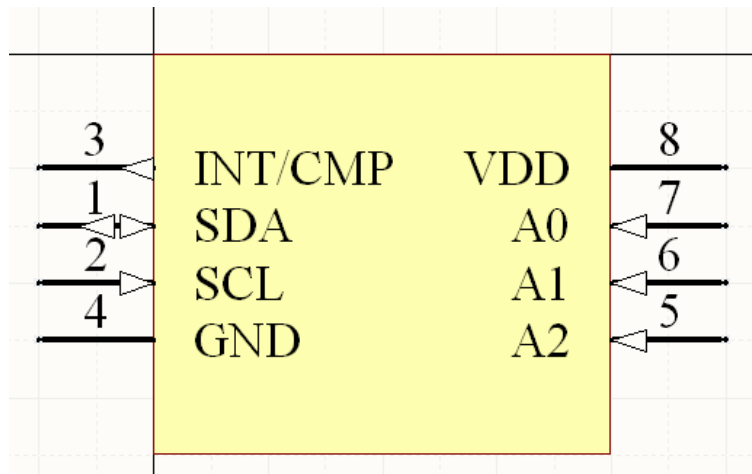
Figure 10. Add a link to a data sheet for a Supplier Item with a drag of the mouse!

It can be a good idea to change the value for the `ComponentLinkDescription` parameter to a shorter, more meaningful entry, for better display in the **References** sub-menu.

### 15.2.5 Exercise – Creating a new component symbol

1. We will now create a component – a serial temperature sensor. If it is not open, open the schematic library \Program Files\Altium Designer Summer 09\Training\PCB Training\Temperature Sensor\Libraries\Temperature Sensor.SchLib
2. Before creating the component, make sure that this library is in the Temperature Sensor project. If it is not already open, re-open the project created during *Module 4 - Schematic Capture* 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.SchLib` 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 a new component Select **Tools » New Component** to create a new component.
6. Enter TCN75 in the *New Component Name* dialog.
7. When the blank sheet appears, zoom in (PAGE UP) until you can see the grid. Components generally have the top left of the component body located at co-ordinates 0, 0 (indicated by the two darker grid lines).
8. Check that the Snap Grid and Visible Grid are set to 10 (**Tools » Document Options**).



**Note:** If the yellow rectangle covers the pin names, use the **Edit » Move » Send to Back** command to move it behind the pins.

Figure 11. Microchip TCN75 serial temperature sensor

9. Create the graphical representation for the component as shown in Figure 11. The component body is a **Rectangle**, placed at the origin in the center of the sheet. The origin is indicated by the two darker lines that form a crosshair, zoom in/out to show the crosshair and the gridlines. Start placing the rectangle at the origin, the body is 80 units wide by 70 units high, you can use the coordinates shown on the **Status bar** to guide you.
10. Place the pins for the part. It is important to orient pins so that the 'hot' end is away from the component body. When placing pins, the cursor will be on the 'hot' end of the pin. Press SPACEBAR to rotate the pin or **X** or **Y** to flip it.
11. **Before** placing the first pin, press TAB to edit the pin properties. The *Pin Properties* dialog will open. For each pin, set the Pin Name, Pin Number, Electrical Type as per the table, and set the Pin Length to 20.

Pin Number	Pin Name	Electrical Type
1	SDA	IO
2	SCL	Input
3	INT/CMP	Output
4	GND	Power
5	A2	Input
6	A1	Input
7	A0	Input
8	VDD	Power

**Note:** you can use the auto-increment/decrement feature when placing pins 5, 6 and 7.

**Note:** As well as using the pin properties dialog to edit the pin names, you can also use the List panel to edit the pin properties after they have been placed, as shown in Figure 6.

12. When you have completed drawing the component, set the
  - Designator to U?
  - Comment to TCN75

- Description to Serial temperature sensor

13. Save the library. You can not place a new component from a library until it has been saved.

**Note:** At the moment this component is really just a symbol, it has no models or parameters – as a minimum it needs a footprint. You will create the footprint for this component in the next section and then come back to the schematic library editor to link it to the symbol.

## 15.2.6 Exercise – Completing the Sensor schematic

At this stage in the training, the structure of your Temperature Sensor project should look like Figure 12. However, the `Sensor.SchDoc` is incomplete. To complete it:

1. Add a new schematic to the project and call it **Sensor.schdoc** if you haven't done so already.
2. Add the Ports, Power Ports and Wiring to finish the schematic, as show in Figure 13.
3. Since the project structure has been modified, you should **Compile** the project and resolve any errors.

**Hint:** use the **Design » Synchronize Sheet Entries and Ports** command to resolve sheet entry-to-port mismatches.

4. Save the Sensor schematic sheet.

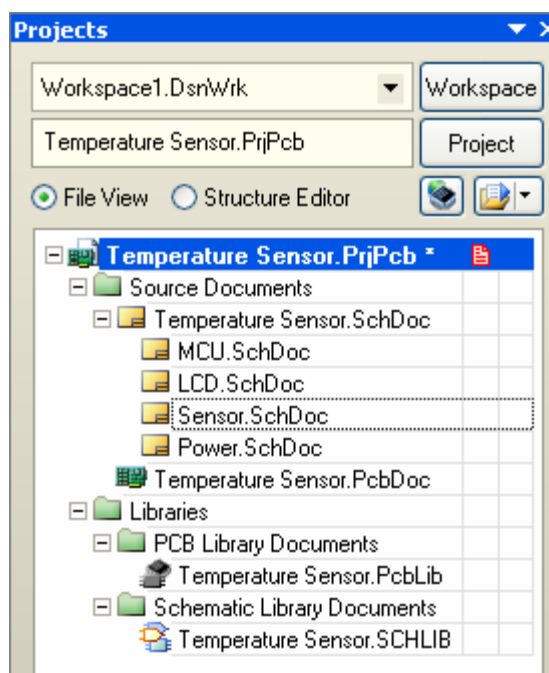


Figure 12. Project structure after completing the Sensor schematic.

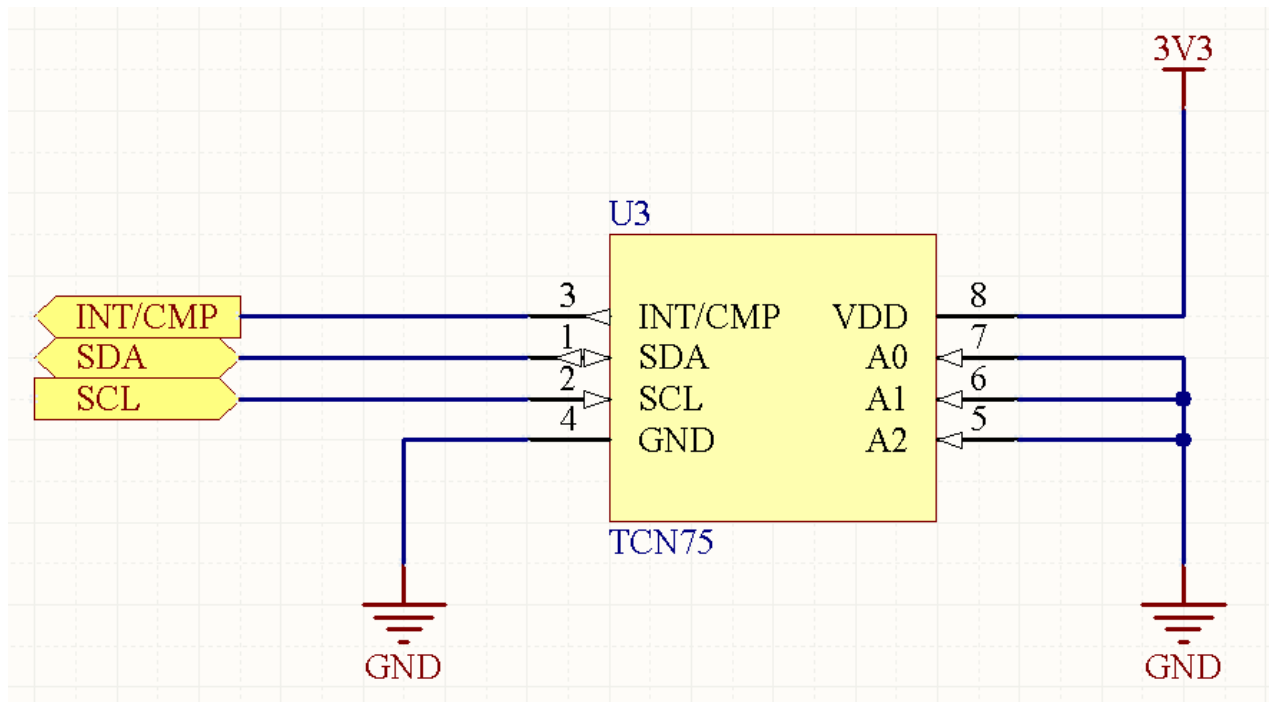
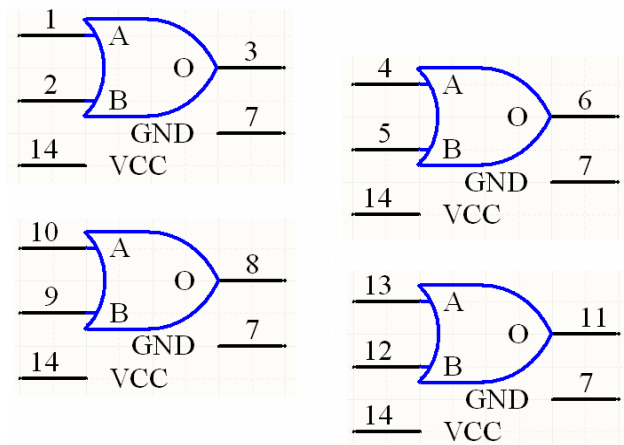
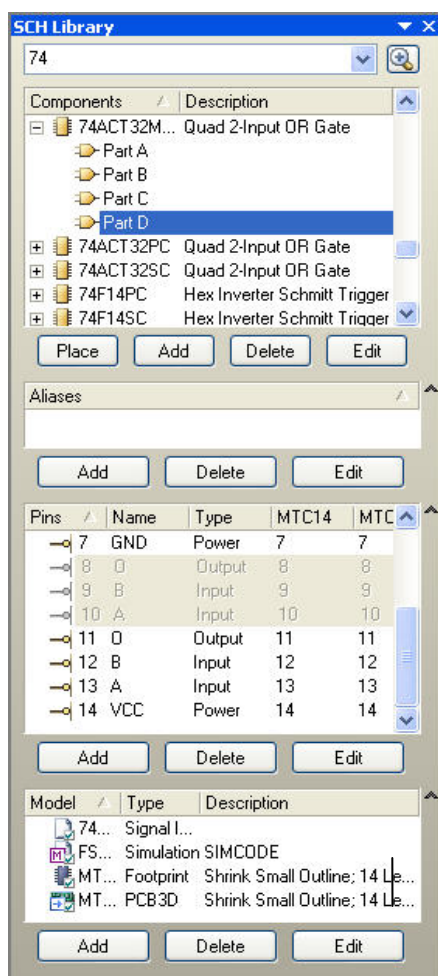


Figure 13. Wired sensor schematic (Sensor.SchDoc).

## 15.2.7 Creating a multi-part component

To create a multi-part component:

- First create one part, select all, then copy the part to the clipboard using the **Edit » Copy** menu command.
- Select **Tools » New Part** to add a new part sheet under the same component name.
- Paste the part onto the sheet and update the pin information. Note that the Part field in the panel will now show 2/2, meaning the second of two parts.
- Finally, add hidden pins (typically power pins) to any of the parts. Edit them, enable the **Hide** attribute and set their **Part Number** to zero. If they are to automatically connect to a net, enter the net name in the **Connect To** field.



The 4 parts of a multipart 74ACT32 component. Note the power pins on each part (hidden pins have been displayed), these exist once, on part zero (a non-visible part). Hidden pins must have the net that they connect to specified in the Pin Properties dialog.

Figure 14. Creating a multi-part component.