



Building an Integrated Library

Summary

Tutorial
TU0111 (v2.0) June 19, 2006

This tutorial looks at using, creating and modifying integrated libraries. Creating source Schematic and PCB libraries and Library Packages, adding models and compiling integrated libraries are investigated.

Integrated libraries combine schematic libraries with their related PCB footprints and/or SPICE and signal integrity models, all together in a non-editable form. All model information is copied into the integrated library from the model libraries or files and so all the component information is stored together, regardless of the location of the original source libraries. This makes integrated libraries truly portable.

Source libraries, including any number of schematic libraries and the related model libraries and files (PCB footprints, SPICE or signal integrity models) are added to a Library Package project file which is then compiled to generate an integrated library. To modify an integrated library, you must change the source library first and then recompile the integrated library.

Altium Designer comes with a set of source libraries and integrated libraries (.IntLib files) stored according to the manufacturer's name in the \Program Files\Altium Designer 6\Library folder. The schematic source libraries (.SchLib files) are included in these integrated libraries and can be extracted by opening the integrated libraries. PCB footprint models are located in the \Program Files\Altium Designer 6\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 \Program Files\Altium Designer 6\Library folder and signal integrity models are located in the Altium Designer 6\Library\SignalIntegrity folder.

Using Altium Designer integrated libraries

Using an integrated library is very similar to using schematic libraries to place components and add model names. The only difference is that all the information about the component and its related models has already been added to the schematic symbol for you. You can check the **Models** list of the *Component Properties* dialog of a component to see what model names have been included with the schematic symbol. Model names can be changed or added from PCB or other model libraries once you have placed a component in a schematic sheet.

When the schematic is transferred from the Schematic Editor to a blank PCB using the **Design » Update PCB** command, the Source Reference Links fields of the *Component* dialog for each PCB footprint are populated with source library pathnames so you can easily trace where the components and models originated from if you need to change them.

Building an Integrated Library

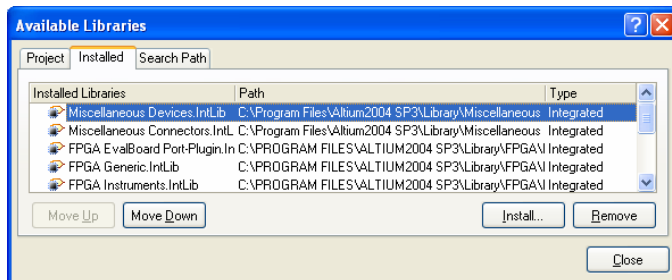
Note that you can still use schematic or PCB libraries (without being in an integrated library) by adding them to the Library list as usual.

Adding and removing libraries

All libraries must be added to the Library list in the **Libraries** panel for the component symbols to become available for placement in a schematic and footprints for the components to be available when creating the PCB.

To add integrated libraries to the Libraries list:

1. Click on the **Libraries** tab or select **View » Workspace Panels » System » Libraries**. The **Libraries** panel displays.
2. Click on the **Libraries** button at the top of the panel to open the *Available Libraries* dialog.



3. Click on the **Installed** tab and click **Install** to add libraries.
4. Browse to the library you require in the *Open* dialog and click **Open**. The library appears in the Installed Libraries list.
5. Click **Close** and the integrated library is added to the Libraries list in the **Libraries** panel. The library name appears in Libraries panel and is now the active library.
6. If a schematic document is open, you can select the component you wish to place from the Components list of the **Libraries** panel. Click **Place <component name>** to place it.

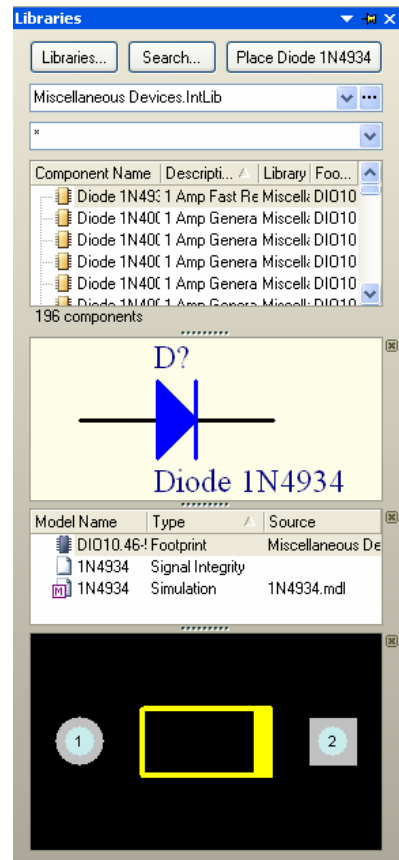
To remove a library from the Library list:

1. Click on the **Libraries** button at the top of the **Libraries** panel to open the *Available Libraries* dialog. Click on the **Installed** tab.
2. Select the library you want to remove. Hold down the **Shift** or **Ctrl** key to multiple select libraries. Click on **Remove**.
3. The library pathname disappears from the Installed Libraries list. Click **Close**. The library is no longer available in the Library panel. Simply add it back in when required.

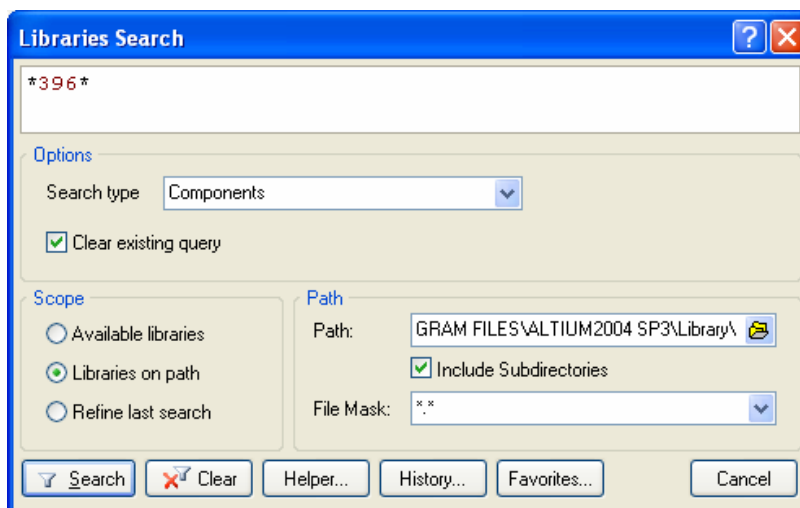
Finding a component in integrated libraries

If you do not know where the component you wish to use is located, use the Libraries Search facility.

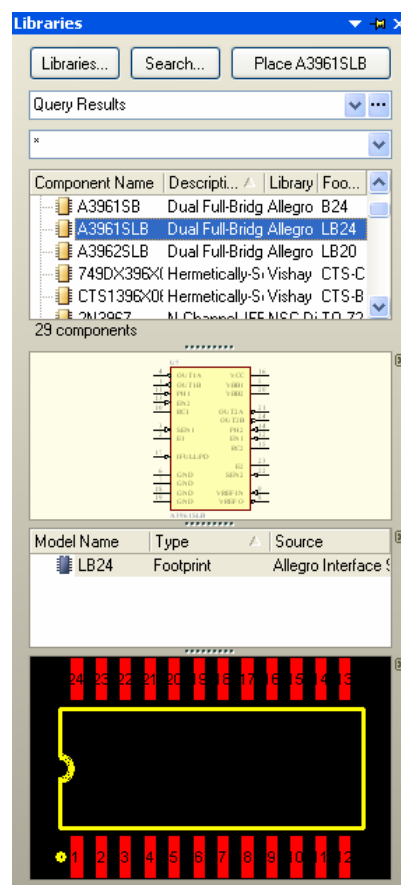
1. Click on the **Libraries** panel tab and the **Libraries** panel displays.



- Click on the **Search** button at the top of the **Libraries** panel to open the *Libraries Search* dialog.



- In the search text field at the top of the *Libraries Search* dialog, type in the name of the component you wish to search for. The * symbol is a wildcard used to take into account the different prefixes and suffixes used by different manufacturers, e.g. *396* will find all components with this string in its name. The system will interpret your search text as a query which is visible the next time you enter this dialog, e.g. *396* becomes (Name like '*396*') or (Description like '*396*'). Click the **Helper** button for more information about writing queries or refer to the Query Language reference in the Help system.
- Select a Search type from the Search Type drop-down list, e.g. Components, to find all the component libraries that match your query.
- Select a Scope for searching in installed libraries or libraries on the search path you nominate by clicking on the folder icon in the Path field. Make sure **Include Subdirectories** is selected if you are searching through the libraries that reside in directories below the nominated pathname.
- Click the **Search** button to begin the search. The Query Results are displayed in the **Libraries** panel as the search takes place.
- Click on the component you require in the Components list of the Libraries panel to select it and to display its model names and graphical representations.



Building an Integrated Library

- Click on the **Place <component name>** button to place the component. Alternatively, just double-click on the component name in the Components list. If you choose a component that resides in a library that is not currently installed, you will be asked to confirm the installation of that library before you can place the component on your schematic. Click on **Yes** to install the library and the component appears 'floating' on the cursor.
- Press **TAB** to display the *Components Properties* dialog while placing the symbol to set the designator.

Component Properties

Properties

Designator: U? ☒ Visible

Comment: A3961SLB ☒ Visible

☐ Don't Annotate Component

Library Ref: A3961SLB

Library: Interface Stepper Motor Controller Driver.IntLib

Description: Dual Full-Bridge PWM Motor Driver

Unique Id: AUCMLLIG

Sub-Design:

Type: Standard

Graphical

Location X: 440 Y: 720

Orientation: 0 Degrees ☐ Mirrored

Mode: Normal

☐ Show All Pins On Sheet (Even if Hidden)

☐ Local Colors ☒ Lock Pins

Parameters for U?-A3961SLB

Visible	Name	Value	Type
<input type="checkbox"/>	Class I	Interface	STRING
<input type="checkbox"/>	Class II	Stepper Motor Controller/Drive	STRING
<input type="checkbox"/>	Datasheet	Integrated and Discrete Semic	STRING
<input type="checkbox"/>	Manufacturer	Allegro	STRING
<input type="checkbox"/>	Package Information	Latest Revision: 1995	STRING
<input type="checkbox"/>	Package Reference	LB24	STRING
<input type="checkbox"/>	Published	13-Oct-2000	STRING
<input type="checkbox"/>	Publisher	Altium Limited	STRING
<input type="checkbox"/>	Revision	July-2002: Re-released for DX4	STRING

Models for U?-A3961SLB

Name	Type	Description
S0-G24	Footprint	

- Check the **Models** list to check that all the required model information, e.g. a footprint model, is already added from the integrated library.
- Click **OK** and then click to place the component symbol on the schematic sheet. Right-click or press **ESC** to end component placement mode.

Creating integrated libraries

There are three ways to create an integrated library:

- by adding existing schematic and PCB or model libraries to a Library Package, or
- from open schematic or PCB documents using the Make Integrated Library command.
- from an existing Database Library or SVN Database Library, using the *Offline Integrated Library Maker Wizard*.

Each process is detailed in the following sections.

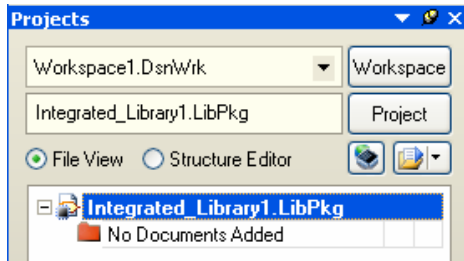
Creating an integrated library using a Library Package

A Library Package is created first with at least all the schematic libraries added and pathnames can be set to the models libraries. Using the Project commands, the Library Package is then compiled to create the integrated library. Any errors generated during the compiling of the integrated library are displayed in the Messages panel for analysis.

Creating the source Library Package

The source of an integrated library is an integrated Library Package. First, we will create a new Library Package, then add schematic libraries to it and then compile it into an integrated library.

1. Select **File » New » Project » Integrated Library**. Alternatively, click on **Blank Project (Library Package)** in the **New** section of the **Files** panel.
2. The **Projects** panel displays with an empty Library Package file named `Integrated_Library1.LibPkg`. There are no source libraries (schematic or PCB libraries) added to the Library Package at this stage.



3. Rename the new Library Package using the **File » Save Project As** command and save it (with a `.LibPkg` extension) to your chosen location. The pathname to the Library Package file is added to the Output Path field in the **Options** tab of the *Options for Integrated Library* dialog (**Project » Project Options**). When the integrated Library Package is compiled, the resulting integrated library file (`.IntLib`) will be saved to an output folder named `Project Outputs for Integrated_Libraryname` which is generated in the same folder as the Library Package file.

Creating a schematic library

Before you can add any schematic libraries to the Library Package, you need to create some! You can create a schematic library out of the components that have been already placed on schematic documents in a project using the **Design » Create Schematic Library** command which is available in the Schematic Editor.

If a schematic document is not part of a project, you can still create a schematic library from it when it is open. The only difference is that the generated schematic library will not be added to a project and will display as a free document in the Projects panel when created.

Alternatively, you can create a schematic library from scratch using the **File » New » Library » Schematic Library** command. Then create your own components using the Schematic Library Editor, or copy in components from other open schematic libraries using the **Tools » Copy Component** command. See [Decompiling an integrated library](#) later in this tutorial for more information about extracting a schematic library from an existing integrated library.

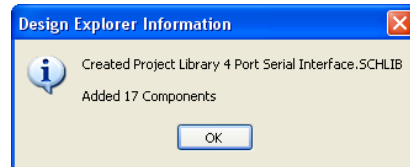


For more information about creating components and footprints, refer to the [Creating Library Components](#) tutorial in the *Library Management* book (online).

Making a schematic project library

To create a schematic library from components in all schematic documents in a project:

1. Open the documents in the project by right-clicking on the project filename in the **Projects** panel and selecting **Open Project Documents**.
2. With the schematic documents that contains all the components you want to add to the new schematic library already active, select **Design » Make Project Library** in the Schematic Editor. Click **OK** to confirm.
3. The new schematic library will open in the Schematic Library Editor when it is created. All the components in the open schematic files are copied to the new schematic library, named `Project_name.SCHLIB`, stored in the same folder as the project file (`Project_name.PRJPCB`). The filename will appear in the **Projects** panel in the `Libraries\Schematic Library Documents` folder.
4. Save or rename the new schematic library using **File » Save As** and close it.



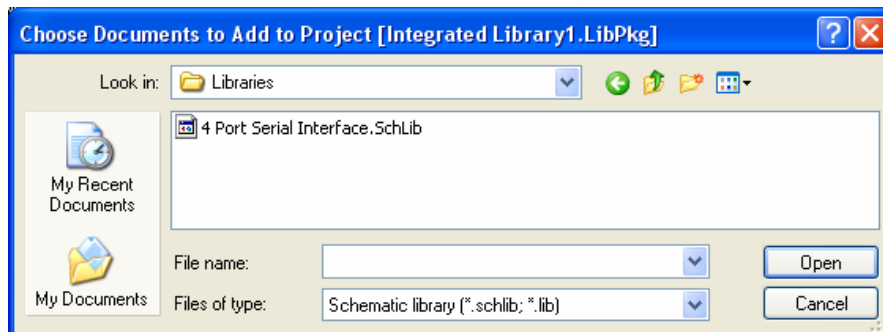
Creating a PCB library

PCB libraries are supplied with Altium Designer and are stored in the default location of `\Program Files\Altium Designer 6\Library\PCB`. However, you can create your own PCB library of footprints from an open PCB file, in a similar manner to creating a schematic library.

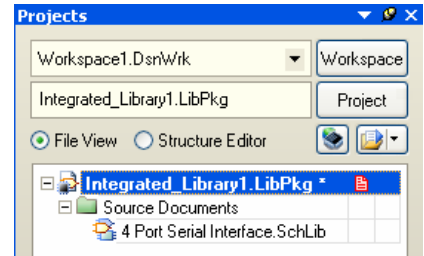
1. With the PCB document that contains all the footprints you want to add to the new PCB library already active, select **Design » Make PCB Library**.
2. The new PCB library will open in the PCB Library Editor when it is created. All the footprints in the open PCB document will be copied to the new PCB library named `PCBfilename.PcbLib`, which is stored in the same folder as the source PCB document. The filename will appear in the **Projects** panel as a free document.
3. Rename the new PCB library using **File » Save As** and close it.

Adding source libraries to the Library Package

1. Add in the source libraries to the Library Package by selecting **Project » Add Existing to Project** or right-click on the selected `.LibPkg` file and select **Add Existing to Project**. The *Choose Documents to Add to Project [Integrated_Libraryname.LibPkg]* dialog displays.



- Browse to find the schematic libraries (.schLib) that you want to add to your Library Package. The schematic components store all the information needed to find related models in their *Component Properties* dialogs, so these are the most essential elements to be included in an integrated library.
- Click **Open** and the added libraries are listed as Source Documents in the **Projects** panel.



Adding models to the Library Package

Now that you have schematic symbols in the library package, the next step is to link the required models to each symbol. This could include a PCB footprint, a simulation model, a signal integrity model, and a 3D model.

Altium Designer has a standard system for making models available, regardless of whether you are building an integrated library package, or working on a schematic design. There are three ways of making models available in Altium Designer:

- installing the library/model in the Installed Library list
- adding the library/model to the project
- defining a search path to the model.

There are advantages to each, so choose the method that best suits your work practices. Different models work better with different approaches too, for example, you might not want to see a large number of simulation models listed in the Projects panel when you open an integrated Library Package, but might like to see the PCB footprint libraries. In this case, you would define a search path to the folder where the simulation models are stored, and add the PCB footprint library to the integrated Library Package.

Installing the library/model in the Installed library list

Library or model files added to the Installed Libraries list in the **Installed** tab of the **Libraries** panel will be available for all projects and remain in the list until removed.

The following types of library files are supported:

- Integrated Libraries (*.IntLib)
- Schematic Libraries (*.SchLib)
- Database Libraries (*.DBLib)
- SVN Database Libraries (*.SVNDBLib)
- Footprint Libraries (*.PcbLib)
- Sim Model Files (*.Mdl)
- Sim Subcircuit Files (*.Ckt)
- PCB3D Model Libraries (*.PCB3DLib).

See the section [Adding and removing libraries](#) for more information about installing libraries.

Adding models as source libraries to the Library Package

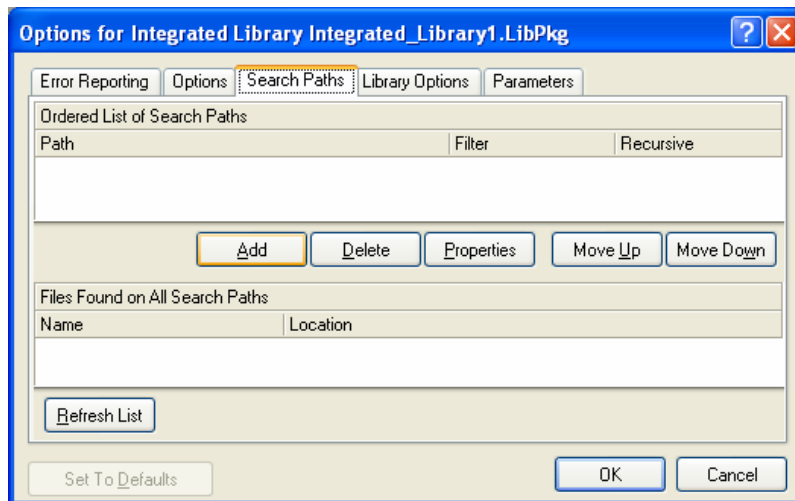
Add model libraries, e.g. PCB libraries, to the Library Package in the same way as the schematic libraries are added.


1. Select **Project » Add Existing to Project**, or right-click on the selected `.LibPkg` file and select **Add Existing to Project**.
2. Browse to find the model libraries that you want to add to your Library Package.
3. Click **Open** and the added libraries are listed as Source Documents in the **Projects** panel.

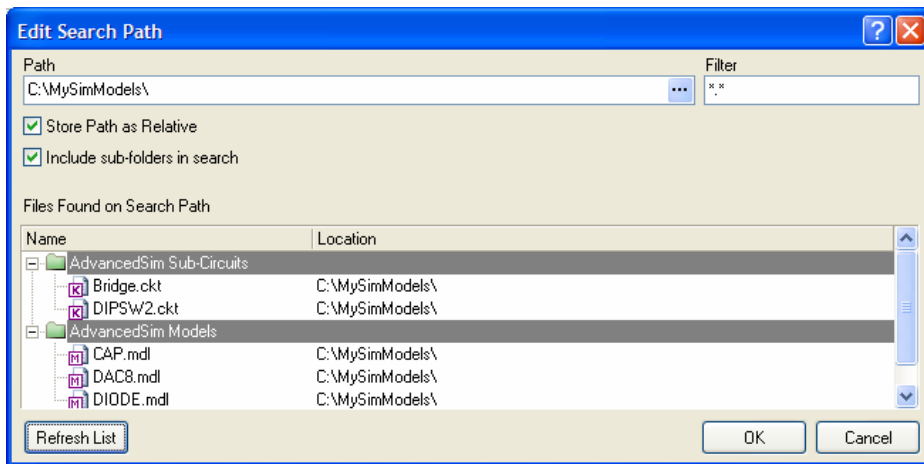
Setting the pathname to model libraries and files

Alternatively, if the PCB footprints libraries, SPICE models or signal integrity models are not added to the Library Package, the schematic symbols in the integrated library will refer to them using the pathname set up in the *Options for Integrated Library* dialog and stored in the Library Package project file (`.LibPkg`).

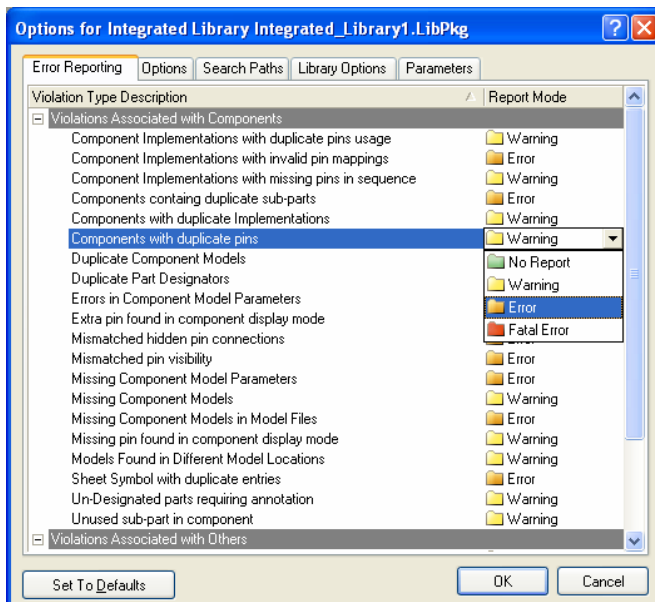
1. Set up the pathname to the PCB libraries you want used by the schematic symbols in the integrated library by selecting **Project » Project Options**, or right-click on the Library Package filename in the **Projects** panel and select **Project Options**. Click on the **Search Paths** tab of the *Options for Integrated Library* dialog.



2. 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.
3. Browse to the folders required in the *Edit Search Path* dialog by clicking on the  button and locating the required model libraries and clicking **OK**. In the example below, we have added in the pathname to the folder `C:\MySimModels` where some SIM models (`*.mdl`) and sub-circuits (`*.ckt`) have been saved.
4. Click **Refresh List** to view the files found on the search path and then click **OK** to close the dialog.



5. Click on **Refresh List** in the **Search Paths** tab in the *Options for Integrated Library* dialog to confirm that the models are located correctly.
6. While you have the *Options for Integrated Library* dialog open, click on the **Error Reporting** tab to see what type of errors and warnings could be generated when the integrated library is compiled.



7. You can change the severity of the violation by clicking on the Report Mode next to the required violation type and selecting another mode from the dropdown list. Click **OK** to save the project options and close the dialog.

Compiling the integrated library

Once you have added the libraries and set any pathnames required, compile them to create the integrated library.

1. Select **Project » Compile Integrated Library** or right-click on the selected Library Package (.LibPkg) file and select **Compile Integrated Library**.
2. The source libraries and model files are compiled into an integrated library. The compiler checks for any violations, such as missing models or duplicate pins, that have been set in the **Error Checking** tab of the *Options for Integrated Library* dialog (**Project » Project Options**). Any errors or warnings found during compilation are displayed in the **Messages** panel. Click on the **System** button at the bottom of the Altium Designer window and select **Messages** to view errors or warnings, or choose **View » Workspace Panels » System » Messages**.
3. Fix any inconsistencies in the individual source libraries at this point and recompile the integrated library. See *Modifying an integrated library* for more information.
4. A new `Integrated_Libraryname.IntLib` is generated, saved in the output folder nominated in the **Options** tab of the *Options for Integrated Library* dialog. The integrated library is automatically added to the current Libraries list in the **Libraries** panel, ready to use.

Creating an integrated library from schematics or PCBs

You can also create an Integrated Library from all the schematics in a project by selecting the **Design » Create Integrated Library** command in the Schematic Editor. An Integrated Library (named `Project_name.IntLib`) will be generated (compiled), added to the `Libraries\Compiled Libraries` folder in the **Projects** panel and installed in the **Libraries** panel.

You can also access the **Design » Create Integrated Library** command from the PCB Editor.

Creating an Integrated Library from a Database Library

Altium Designer provides the facility to compile an integrated library directly from a database library – either a non-version-controlled Database Library (DBLib), or a version-controlled SVN Database Library (SVNDBLib). In this way, your CAD Librarians can still use database/version-controlled libraries, while your designers use regularly regenerated integrated libraries, working in an 'offline' fashion as it were.

Conversion is performed using the *Offline Integrated Library Maker Wizard*. This Wizard is accessed from either the active DBLib or SVNDBLib document using the **Tools » Offline Integrated Library Maker** command.

The process of conversion to an integrated library is carried out on a per-database-table basis. You have full control over which tables in the database – linked to your database library – are considered in this process. A separate integrated library will be generated for each included table.



For more information, refer to the [Database Library Migration Tools](#) application note.



For more information on database libraries, refer to the [Using Components Directly from Your Company Database](#) and [Working with Version-Controlled Database Libraries](#) application notes.

Modifying an integrated library

The integrated libraries are used to place components and cannot be edited directly. To make changes to an integrated library, make modifications in the source libraries first and then recompile the integrated library to include the changes. To modify an integrated library:

1. Open the required integrated library's Library Package file (*.LIBPKG). Select **File » Open** and browse to the Library Package file, e.g. *Integrated_Library1.LibPkg*, in the *Choose Document to Open* dialog and click **Open**.
2. Open the source library file you want to change. e.g. *libraryname.schlib*, by double-clicking on the library name in the Source Documents list in the **Projects** panel. The library opens in the Schematic Library Editor.

If you wish to modify a footprint, you would have to add in the required PCB library before you could edit the models. To do this, you could right-click on the .LIBPKG filename in the **Projects** panel and select **Add Existing to Project**, or alternatively, click on the **Libraries** button in the **Libraries** panel, select the required library in the **Project** tab and click on **Add Library**. You could also use **File » Open** to open a model file directly.



For more information about creating components and footprints, refer to the [Creating Library Components](#) tutorial in the *Library Management* book (online).

3. Make changes as required, save the modified libraries and close them.
4. Recompile the integrated library by selecting **Project » Compile Integrated Library** (or right-click on the .LIBPKG filename in the **Projects** panel and choose **Compile Integrated Library**). The integrated library is recompiled and any errors are listed in the **Messages** panel. The modified integrated library is added to the **Libraries** panel and is ready to use.

Decompiling an integrated library

Although integrated libraries cannot be edited directly, they can be decompiled back into their constituent source symbol and model libraries. To do this:

1. Open the integrated library (.IntLib) that contains the source library you need to modify. Select **File » Open**, browse to the integrated library in the *Choose Document to Open* dialog and click **Open**.
2. Confirm that you do wish to open the integrated library to extract the source libraries and not just install the library. Click on **Extract Sources**. The source schematic and model libraries are generated and saved in a new folder named *Integrated_libraryname*, which is created in the folder storing the integrated library.

A Library Package (*integrated_libraryname.LibPkg*) is also created and the source schematic libraries are extracted and listed in the **Projects** panel. PCB libraries (.PcbLib) are generated as well and stored in the new Library Package folder but are not automatically added to the **Projects** panel. The pathname in the **Search Paths** tab of the *Options for Integrated Library* dialog (**Project » Project Options**) indicates where the schematic components will search for when the footprints and model files are required.

3. Make necessary changes to the source libraries and save them by selecting **File » Save** and then close them.

Building an Integrated Library

4. Select the Library Package file (.LIBPKG) in the **Projects** panel and select **Project » Compile Integrated Library**. The integrated library is recompiled and any errors are listed in the **Messages** panel. The modified integrated library is added to the Libraries panel and is ready to use.
5. Close the Library Package and save it to the same folder as the source libraries.

Revision History

Date	Version No.	Revision
9-Dec-2003	1.0	New product release
13-May-2005	1.1	Updated dialogs and commands for Altium Designer SP3. Libraries Search modifications. New Make Integrated Library command included.
12-Dec-2005	1.2	Path references updated for Altium Designer 6
19-Jun-2006	2.0	Updated for Altium Designer 6.3 – section added with information on creating an integrated library from a database library.

Software, hardware, documentation and related materials:

Copyright © 2006 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, CAMtastic, CircuitStudio, Design Explorer, DXP, LiveDesign, NanoBoard, NanoTalk, Nexar, nVisage, P-CAD, Protel, 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.