cadence®

Allegro® EDM
Library Flow User Guide

Product Version 23.1
September 2023

Cadence Design Systems, Inc. (Cadence), 2655 Seely Ave., San Jose, CA 95134, USA.

Contents

Preface 5
About This Guide 5
Related Documentation 5
Related Tools and Flows 5
Typographic and Syntax Conventions 6

1
Library Flow Overview 7
Why Allegro EDM Library Flow? 7
Library Elements 8
Development States 9
Distribution States 10
Lifecycle States 11
Classifications and Metadata 11
Version Control 13
Storing Library Data 13
Working Sets 15
Launching Library Flow 16
Library Flow User Interface 17
Library Design Tasks 19
Walk Through Library Flow 20
Adding Metadata 22
Create and Verify Library Element 22
Checking in Library Element 22
Prereleasing a Library Element 23
Releasing a Library Element 24
Check Out Library Element 25
Communicating With Designers 25
What Next? 25

2
Creating Schematic Models 27
Prerequisites 27
Creating Sample Project 28
Creating Metadata 29
Creating and Verifying Schematic Model 33
Synchronize Models 36
Associating Schematic Classification 39
Check in Schematic Model 41
Summary 42

3
PCB Models 43

4
Working With Parts 49
Prerequisites for Creating a Part 50
Adding Metadata Information 50
Linking Models 52
Check in Parts 52

5
Dialog Box Help 55
Library Design Verification 55
New Schematic Model 56
Options 57

Preface

About This Guide

The *Allegro® EDM Library Flow User Guide* explains the library import methodology and the procedures and tasks you need to perform for successful library import.

Related Documentation

You can also refer to the following documents to know more about related tools and methodologies:

- For information on the new features, refer to *Allegro Pulse and Allegro EDM: What's New in Release.*
- To learn how to use Database Administrator, refer to *Allegro EDM Database Administrator User Guide.*
- To learn how to use Database Editor, refer to *Allegro EDM Database Editor User Guide.*
- To learn how to use Library Distribution, refer to *Allegro EDM Library Distribution User Guide*.
- To learn how to migrate non-Allegro EDM designs and libraries into Allegro EDM, refer to the document titled *Converting Non-Allegro EDM Designs to Allegro EDM-Compatible Designs*.

Related Tools and Flows

- For information on various PCB design working environments such as a team of designers working on a Design Entry HDL project, implementing FPGAs in designs, working with high-speed constraints, importing IFF files for radio-frequency designs, and reusing existing modules, refer to *Allegro PCB Design Flows*.
- To learn how to create and configure Design Entry HDL projects, refer to the *Allegro Project Manager User Guide*.

Typographic and Syntax Conventions

This list describes the syntax conventions used for this user guide:

`literal`	Nonitalic words indicate keywords that you must enter literally. These keywords represent command (function, routine) or option names.
`argument`	Words in italics indicate user-defined arguments for which you must substitute a name or a value.
|	Vertical bars (OR-bars) separate possible choices for a single argument. They take precedence over any other character.
`[ ]`	Brackets denote optional arguments. When used with OR-bars, they enclose a list of choices. You can choose one argument from the list.
`{ }`	Braces are used with OR-bars and enclose a list of choices. You must choose one argument from the list.

1

Library Flow Overview

Why Allegro EDM Library Flow?

The library flow is the library development and management solution of Allegro EDM. The flow integrates the Cadence PCB design tools with library and design data management and provides users with a mechanism to create, validate, manage, and distribute library parts. The parts created in the library flow can be used directly in Cadence PCB design tools, such as Design Entry HDL, PCB SI, and PCB Editor.

The component creation, validation and distribution mechanisms supported by the Allegro EDM library flow ensures that everyone in the design process is using known-good library data to build their designs. Besides managing and developing libraries, the library flow also provides features such as the creation of Preferred Parts List (PPL) and reports generation that help in increasing productivity and reducing the development time. A PPL is a list of approved parts in the company parts library. Using a PPL ensures that preferred components are being used. The ability to find preferred components means significant time savings when researching parts for a design.

This book provides an overview of the Allegro EDM library flow. It also explains various concepts and terms used in the flow.

Library Elements

Library element refers to all objects that are in the Cadence libraries and that can be managed using the library flow. While working in the Allegro EDM library flow, a library element can be any one of the following:

- Schematic Models
- PCB Models
- Datasheets
- Parts

Schematic Models

Schematic models are logical parts or symbols used by a logic designer during the design capture phase. In a library, a schematic model has three views: `sym_1`, `chips`, and `entity`.

For more information, see Chapter 2, “Creating Schematic Models.”

PCB Models

PCB Models refers to the back-end models, such as shapes, padstacks, and footprints, which are required while designing a board. Library flow provides you with tools to create new shapes, padstacks, and footprints, and to maintain the existing ones.

For more information, see Chapter 3, “PCB Models.”

Datasheets

The Allegro EDM library flow provides support for adding part datasheets to the library database. A part datasheet provides technical details about a part and how to use the part in a design.

Parts

In the library flow, a part is a library element that is an electronic representation of an off-the-shelf part, with complete information about the logical data required to capture the design logic, physical data required to complete the physical layout, and also information required for procuring and the manufacturing the part.

A part can be instantiated in the logical design and has a physical representation as well. For more information, see Chapter 4, “Working With Parts.”

Development States

Depending on the stage in the development process, the elements in the Allegro EDM component database have different development states associated with them. Each development state indicates whether or not certain verification tasks have been performed on the library element. The checks that are run when the development states changes from one to another are listed in the Rules Manager.

In the Allegro EDM library flow, the development state of a library element is indicated using the Development Status attribute. The attribute values currently supported in the Allegro EDM library flow are as follows.

- Preliminary

  Library elements in the Preliminary state are in the initial phases of development. In this stage, the details of the library element are available only in the working area of a librarian and not in the Allegro EDM database.

- Checked-In & Verified

  Library elements are in the database after a set of verification steps have been performed on these. A part in checked-in state indicates that it is in the final states.

- Check-out

  Indicates that the library element was once checked into the database, but someone is currently working on a higher version.

- Pre Released

  Indicates a library element that is not complete and is still under development. This library element can be made available to a PCB designer, but can be replaced with the modified version at any time.

  For example, if the development status of a part is set to Pre Released, it might indicate that the footprint information is currently missing or that front-to-back checks have not been executed on the part.

- Released

  This development state indicates that the development of the library element is complete, all required verification tasks have been performed, and it is ready for use by designers.

- Deleted

This development status indicates that the library element in the database is not accessible to PCB designers. This state is usually used if you need to indicate that the part is an old part. Deleted library elements are not removed from the database.

A schematic model can only be marked deleted when all the latest versions of its linked part are also in the deleted state.

Distribution States

After a librarian creates a library element, the next step is to make this available to PCB designers so that the new element can be used for design creation. The process of making library elements available to the designers is called library distribution. The `lib_dist` utility is used for distributing libraries.

For detailed information about Library Distribution, refer to *Allegro EDM Library Distribution User Guide*.

The distribution status of a library element is indicated using the Distribution Status attribute. The possible values that can be assigned to this attribute are:

- Pending Distribution

  This distribution state indicates that the library element is not yet available to designers. By default, for library elements with the development state set to Preliminary, checkout, or check in, the Distribution status is set to Pending Distribution.

- Distributed

  Indicates that the library element is available to PCB designers. For example, parts whose distribution status is *Distributed*, are displayed in Part Information Manager to be used by PCB designers.

  Note: Parts whose distribution status is set to *Distributed* and whose development status is *Deleted*, are not available for use in designs. These parts are not displayed in Part Information Manager.

- Previously Distributed

  As the name suggests, library elements with distribution status set to *Previously Distributed* were available to the designers at some point, but are currently not available for use in designs. This status indicates that a higher version of the library element is available in the database.

  Note: Parts whose distribution status is *Previously Distributed* are not visible in Part Information Manager and are therefore not accessible to designers.

- Pending Delete

  Indicates that the models and parts have been marked as deleted but the library has not been distributed. The status of these models and parts will change to Deleted after the library is distributed.

- Pending Purge

  For models and parts that have been marked for permanent deletion from the database but the library has not yet been distributed.

Lifecycle States

The lifecycle states indicate the lifecycle of a library element from the designers' perspective. Depending on company policies and business needs, the lifecycle states of a library element can differ from company to company. To provide users with the flexibility of defining custom lifecycle states, states are added to the library element as attributes and are configurable.

Lifecycle states are defined by the administrator using the Database Administrator tool. With each lifecycle state, administrators can associate one of the following tasks.

- Add With Warning
- Do Not Add
- Do Not Show

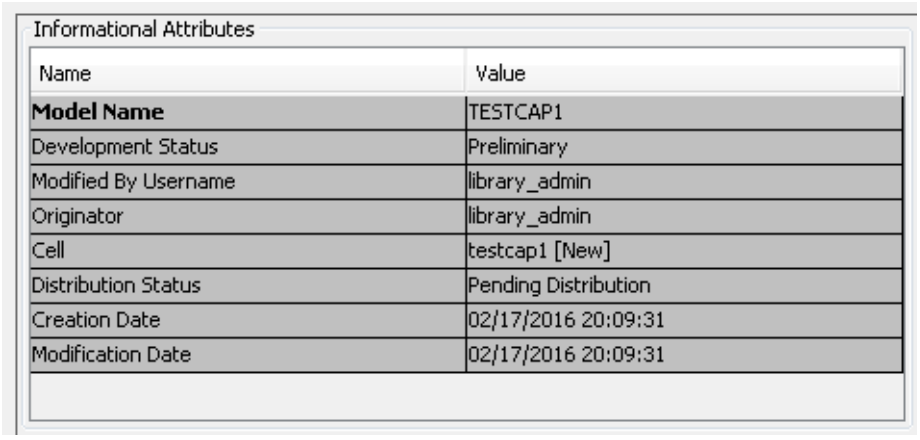
Once defined, lifecycle states can be controlled through a configuration file and can be managed either through the external system integration or through Database Editor.

Classifications and Metadata

To enable easy management and sharing of component information and design libraries, Allegro EDM uses the concept of metadata (data about data) and classification of design data.

All library elements in the Allegro EDM library database have metadata values. Using metadata allows Allegro EDM users to use database search features to find the required library element.

For example, when you create a new schematic model, the first step is to add the metadata for the schematic model. The following figure shows the metadata added for a schematic model named TESTCAP1.


Informational Attributes

Name	Value
Model Name	TESTCAP1
Development Status	Preliminary
Modified By Username	library_admin
Originator	library_admin
Cell	testcap1 [New]
Distribution Status	Pending Distribution
Creation Date	02/17/2016 20:09:31
Modification Date	02/17/2016 20:09:31

Classification is a group of parts that usually perform the same function and that have common characteristics. Classification involves categorizing library elements into logical groups in a way that indicates that the members of the group have a common set of properties. Properties defined on a classification are inherited by the components associated with the classification.

Allegro EDM support the following classification types:

- Part Classification
- Mechanical Part Classification
- Block Part Classification
- Model Classification

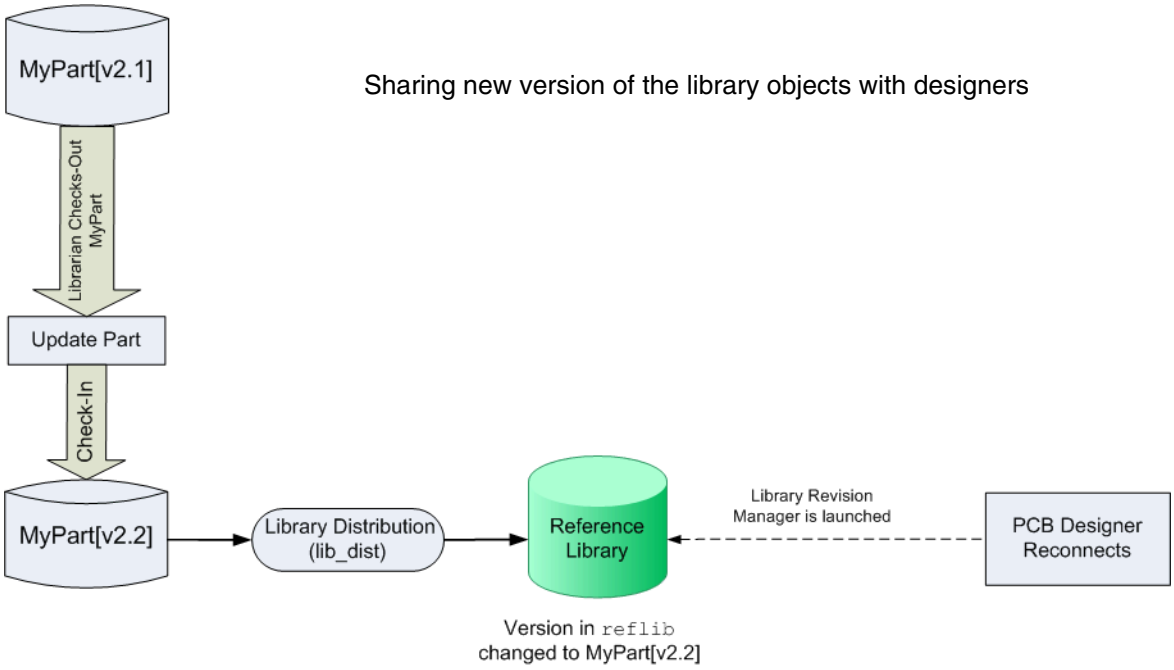
Classifications support inheritance. As a result, properties that are common for a category of library objects are defined in a classification. When an object is linked with the classification, these properties are inherited in the library object. The properties inherited from a classification are removed when you de-link the part or model with the classification. Allegro EDM classifications are hierarchical (parent and child relationship) and support versioning capabilities.

For more information on Allegro EDM classifications, refer to *Allegro EDM Database Editor User Guide*.

Version Control

The Allegro EDM library flow provides versioning support for all database objects.

When you create a new library element, a new revision is created. Every time you check in a library element, a revision of the library element is created. Any further changes made to this element are tracked through the version number. For example, you can have version 1, 2, or 3 of revision A. You can only check out the latest version of the library element in the database. A change in the version number indicates minor modifications to the library element.



Sharing new version of the library objects with designers

Storing Library Data

The library solution provided by Allegro EDM is unique in the way it stores information about library parts. The physical locations of the Allegro EDM libraries follow a multi-tier structure. The key areas required for the Allegro EDM library flow are: work area, integration area, and vault.

- **Work Area:** Represents the work-in-progress area for the librarians. This is a local area unique to each librarian where the library elements checked out by the librarian are stored till they go through the edit process. With this logic, library elements with development states set to Preliminary and Checked-out are in the work area of a librarian.

For a librarian, the work area is created in the folder specified by the `PCBDW_PROJECTS_DIR` environment variable.

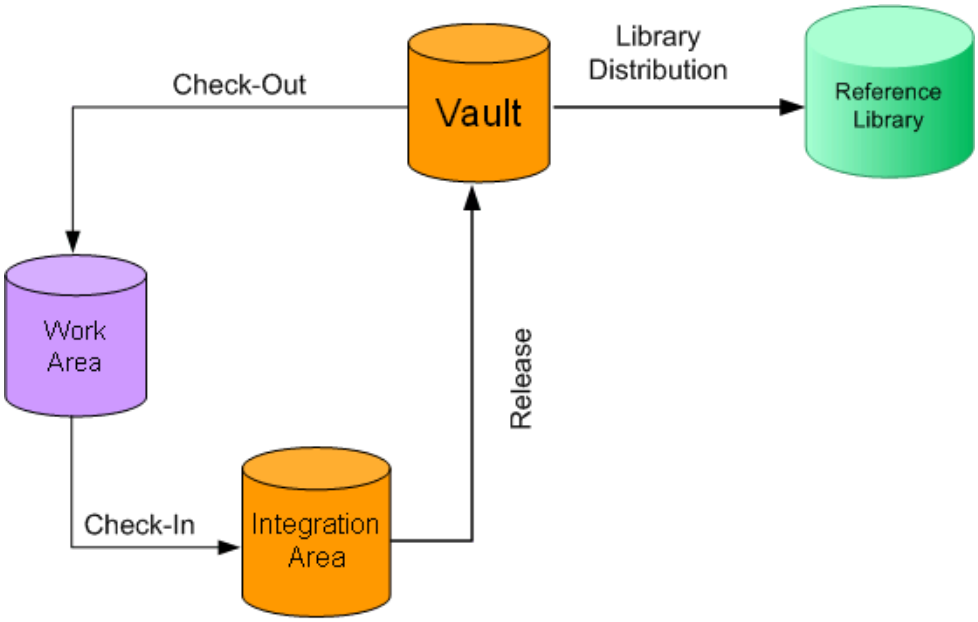
The modifications made to a library element in the work area of one librarian are not visible to any other librarian till these modified elements are checked into the database.

- **Integration Area:** An intermediate area where the library elements checked in by a librarian are stored until they pass through the verification process, and their development state changes to Released. The integration area stores all the library components available to a design site. This area is used during library synchronization and the library authoring process.
- **Vault Area:** Once the library elements in the integration area are verified and released, the archive files are moved to the vault. A vault is a hierarchical file folder where all the archive files of the library elements are stored with entries of these models available in the component database. The model files are stored in the vault in a compressed format (as a tar ball).

For a site, the Integration and Vault areas use the `PCBDW_LIB` environment variable. To know how to set this environment variable, refer to *Allegro Pulse Configuration Guide*.

Note: The data in the vault is in compressed format and cannot be used directly by design tools. To make this data available to PCB designers, run the library distribution utility. After distribution, the compressed data from the vault is uncompressed and stored in the `reflib` directory, also referred to as the reference library. The contents of the `reflib` folder are used as a reference library by the PCB design tools. Part Information Manager reads the contents of the reference library and displays it to PCB designers.

The following figure shows the relationship between different work areas used in the Allegro EDM library flow.



Working Sets

The file structure discussed in the Storing Library Data section describes how physical data associated with a library element is stored during different development states. Besides the vault, integration area, and work area, Allegro EDM also uses the concept of working sets.

Working sets are individual workspaces in the database that are available to a librarian. A working set is a virtual place to keep all your work-in-progress and non-released data. Unlike the work area that contains physical data, working sets are librarian-level workspaces that contain metadata of the work-in-progress library element.

Working sets enable librarians to work concurrently in a collaborative environment. Librarians can share these workspaces among them. This makes it possible to share data between librarians who work on different parts of a component (symbol, footprint, and so on). Working sets also support sharing of library creation tasks between librarians.

For more information on working sets, see the section on *Using Working Sets* in *Allegro EDM Database Editor User Guide*.

Launching Library Flow

To launch the library flow, create a library project.

1. From Flow Manager Welcome Page, choose *File – New Project*.

   The Allegro EDM Project Wizard appears.

2. From the *Type of Project* drop-down list in the *Project Initialization* page, select `library` and click *Next*.

3. In the *Get Project Information* page, specify the required details and click *Next*.

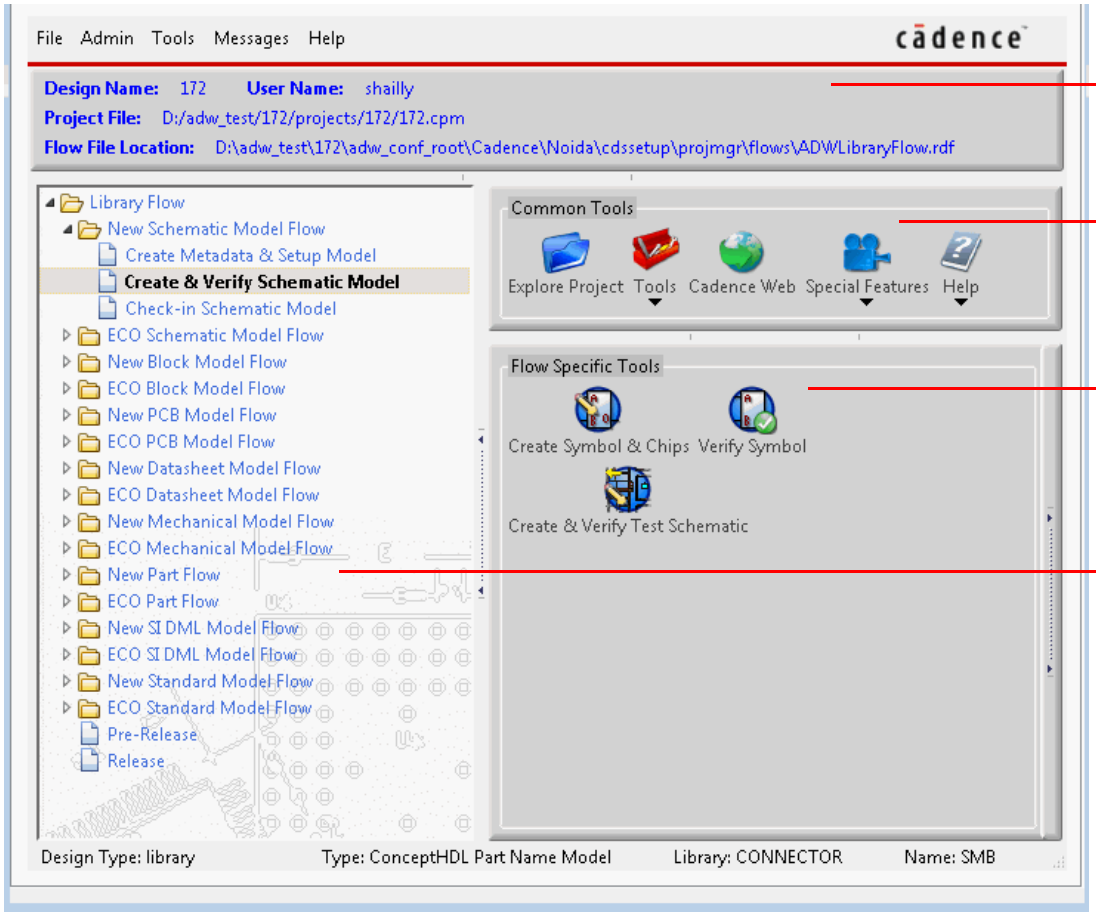
4. Click *Finish*.

   The Library Flow user interface appears.

To know more about managing projects, refer to *Allegro EDM Flow Manager User Guide*.

Library Flow User Interface

The following figure displays the user interface for the Allegro EDM library flow.



Information Pane

A non-editable pane that displays the information about the current project.

Flow Tree View

Lists the sequence of tasks to be performed. This is a hierarchical view in which the first-level nodes indicate all the tasks that can be performed using the Allegro EDM library flow. The second-level nodes indicate the steps to be performed to complete the task.

For example, to complete the task of creating a New Schematic Model Flow, you need to first create metadata for the model, then create and verify schematic model, and finally check in the schematic model.

Note: The tree structure displayed in this frame is governed by the flow file (*.rdf) used by the project. The name and location of the flow file is specified by the Flow File Location field in the Information Pane.

Common Tools Pane

The buttons displayed in the Common Tools Pane are used to launch tools and utilities that can be used by the librarians at any time, regardless of the current step in the library flow. These buttons in this pane are customizable.

Flow Specific Tools Pane

Displays the buttons that launch point tools required to perform the tasks specific to the design step selected in the Flow Tree View. If this pane has more than one button, the sequence in which they are displayed is also an indication of the sequence in which each sub-task should be performed.

Library Design Tasks

The tasks that are performed by a librarian can be classified as administration and library development tasks.

Administration Tasks

Tasks that have to be performed before library development can start are referred to as database administration tasks. These include creating lifecycle states, managing classifications, creating new libraries, and specifying verification checks to be performed before promoting a part from one development state to another. These tasks can only be performed by librarians with administrator rights.

The Database Administrator and Database Editor tools are used to perform the library administrator tasks.

To launch Database Administrator from the library flow, use one of the following methods:

- Choose *Tools – Database Administration*.
- Choose *Tools – Allegro EDM System Console*. In the console window, type `dbadmin` and press Enter.

To launch Database Editor from the library flow, choose *Tools – Allegro EDM System Console*. In the console window, type `dbeditor` and press Enter.

Library Development Tasks

Core library development tasks include creating the following: schematic models, footprints, and parts. While these tasks are performed using multiple point tools, the Allegro EDM library flow user interface can be used as a cockpit for launching all the relevant tools. The Flow Specific Tools Pane is used to launch the tools required for a particular design task.

For example, irrespective of the type of library element created, the metadata information must be added for all library elements. To add metadata for an object, the Database Editor tool is used. This tool is launched when you select the *Create Metadata & Setup Model* button in the Flow Specific Tools pane.

For information on using Database Administrator, refer to *Allegro EDM Database Administrator User Guide*.

For information on using Database Editor, refer to *Allegro EDM Database Editor User Guide*.

For information on using other point tools such as Part Developer, Design Entry HDL, and Allegro PCB Editor, see the respective product documentation.

Walk Through Library Flow

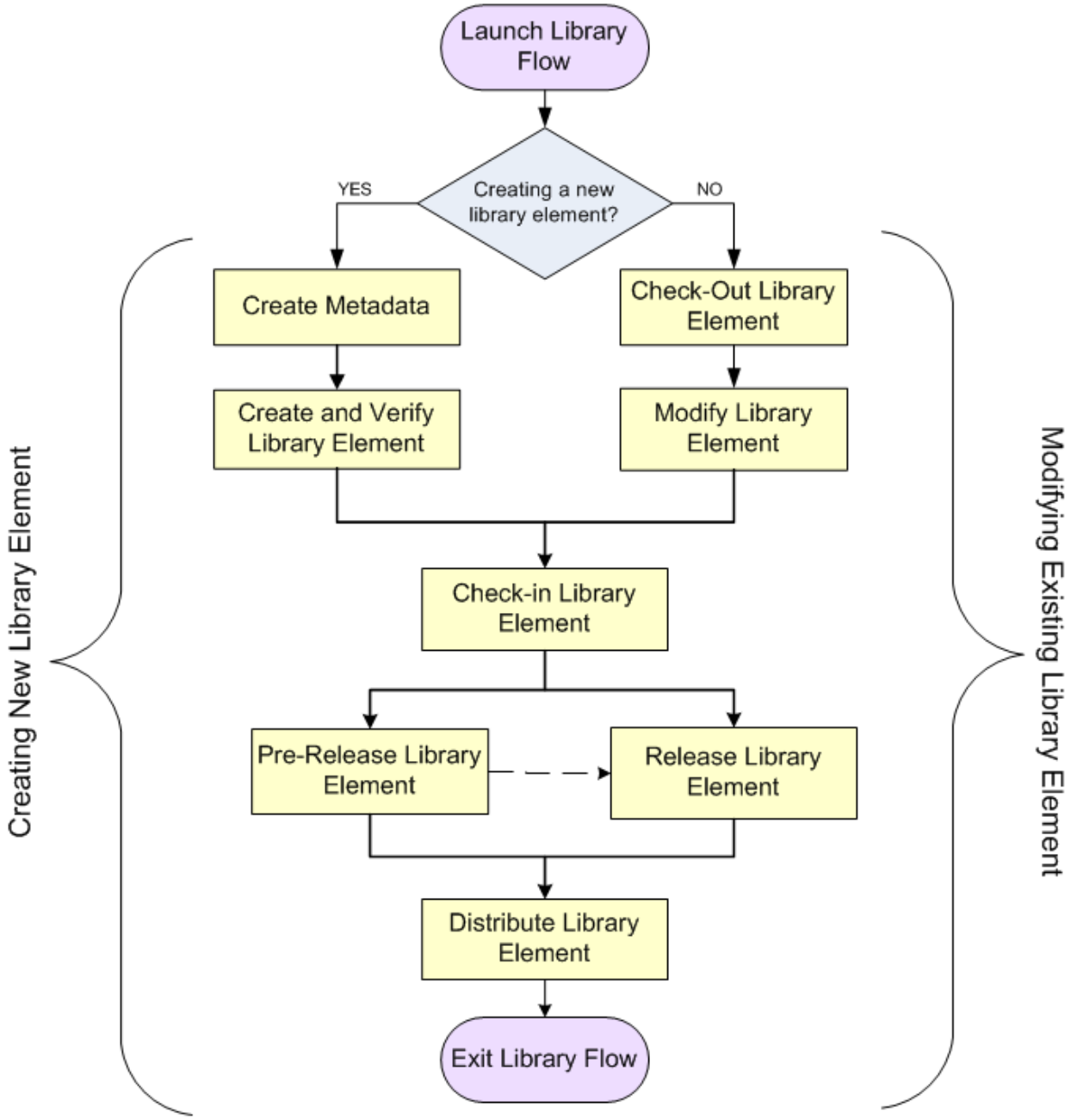
Using the Allegro EDM library flow, you can do one of the following tasks.

- Create New Library Elements
- Modify Existing Library Elements

Both the tasks can be performed on all the library elements.

Figure 1-1 on page 21 provides a top-level view of the design flow tasks to be performed while creating a new library element or while modifying an existing library element. Depending on the library element to be created, some extra sub-tasks may have to be performed.

Figure 1-1 Library Design Tasks



Adding Metadata

Before you can create a new library element using the Allegro EDM library flow, you need to add metadata information for the library element in the database. Metadata creation involves the creation of the object name and its mandatory relations and values for properties. This is a must as it helps to search the database.

For example, a part can have "COST", regulatory compliance attributes such as RoHS, WEEE, ELV as searchable attributes in addition to the properties which constitute the PTF such as PART_NUMBER, VALUE, DESC, and TOLERANCE.

Steps to Add Metadata

1. In Flow Manager, select *New Schematic Model Flow*.
2. Select the first task: *Create Metadata & Setup Model*.
3. In the *Flow Specific Tools* pane, click the *Create Schematic Metadata & Setup Model* button.

   Allegro EDM Database Editor is launched and the dialog box appears.
4. Enter the required values and click *OK*.

Create and Verify Library Element

In this step, you perform the tasks required to create the library element using the core design tools from Cadence. Depending on the library element created, different verification tools are used.

For example, to create a schematic model, you launch Part Developer (PCB Librarian), which is the tool used for creating and modifying symbols, and packages. To verify the schematic model, Rules Checker is run in batch mode, and the symbol view of the new schematic model is used to create a dummy schematic in Design Entry HDL.

Similarly, for creating padstacks, shapes or footprints, Allegro PCB Editor is used. For verifying footprints, PCB Editor rules are run. To verify a new part, the complete front-to-back flow is run for the part.

Checking in Library Element

When you check in a new library element, a copy of the data files is created in the database.

Note: Based on the command options specified by you in Database Editor, the local copies of the generated files are either removed or retained.

By default, on checking in a library element, files are moved from the work area (`flatlib`) to the Integration Area (`..\pcbdw_lib\exchange\transmit` folder), indicating that the information about the library element is now available in the database.

When a library element is checked in, a series of verification steps are run. The check-in process fails if there are any errors reported during verification. For example, when you check in a new version of the schematic model, Allegro EDM runs a check to ensure that all parts linked to the model currently in the database continue to work with the version being checked in.

Tip

You can view the rules against which the library elements are verified in Rules Manager in Database Editor. One of the administration tasks for a librarian is to create new rules and to specify the rules to be used for each verification step.

The check-in also fails when mandatory relations of property values are missing. Check in auto sync models with data from the hard disk.After a successful check-in, the development state of the library element is changed to Checked-In & Verified.

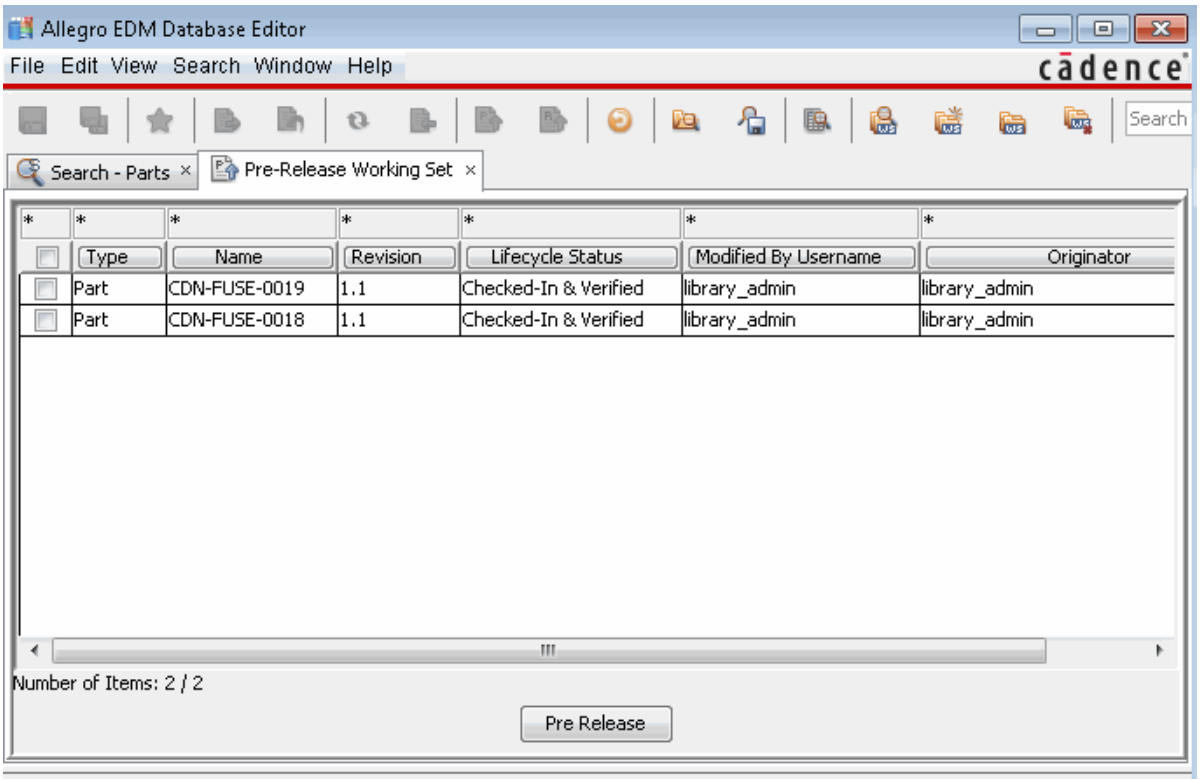
Prereleasing a Library Element

Prereleasing or releasing a library element makes it available to designers for use, even before all verification tasks have been completed for that library element.

Steps

1. Select *Pre-Release* from the Library Flow tree view.
2. In the *Flow Specific Tools* pane, click the *Pre-Release* button.
3. In the login screen, specify the login ID and password to log into Database Editor.

Allegro EDM Database Editor window appears with all the library elements that are in the `Checked-In & Verified` state listed in the *Pre-release Working Set* tab as shown in the following figure.



4. Select the library elements to be pre-released and click *Pre Release*.

Releasing a Library Element

Releasing a library element makes it available to the designers.

1. In Flow Manager, select *Release* from the Library Flow tree view.
2. In the *Flow Specific Tools* pane, click the *Release* button.

   Note: Alternatively, select the Release icon in the Allegro EDM Database Editor.

   The *Release Working Set* tab appears in Database Editor. All the elements that are in the pre-released or checked-in state are listed in this tab.
3. Select the check box against the elements to be released, and click *Release*.

   The Release Dependency Set dialog box displays along with the login screen.

Check Out Library Element

This step is required when you need to modify a library element that is already checked into the Allegro EDM database.

Communicating With Designers

The library elements created by librarians are used by the PCB designers for creating actual designs. The interaction between a PCB designer and a librarian is required when a designer requests a new part and when a librarian has created a new part or library.

The tasks of distributing the verified and released libraries to the designers is performed using the Library Distribution utility. For information on library distribution, refer to *Allegro EDM Library Distribution User Guide*.

What Next?

The rest of the user guide uses a design example to perform the steps required to create a new part from scratch.

2

Creating Schematic Models

This chapter demonstrates the steps to be followed for creating a schematic model using the Allegro EDM library flow. This, and the following chapters, focus on how to create a schematic model, footprint, and a part based on the information in a datasheet.

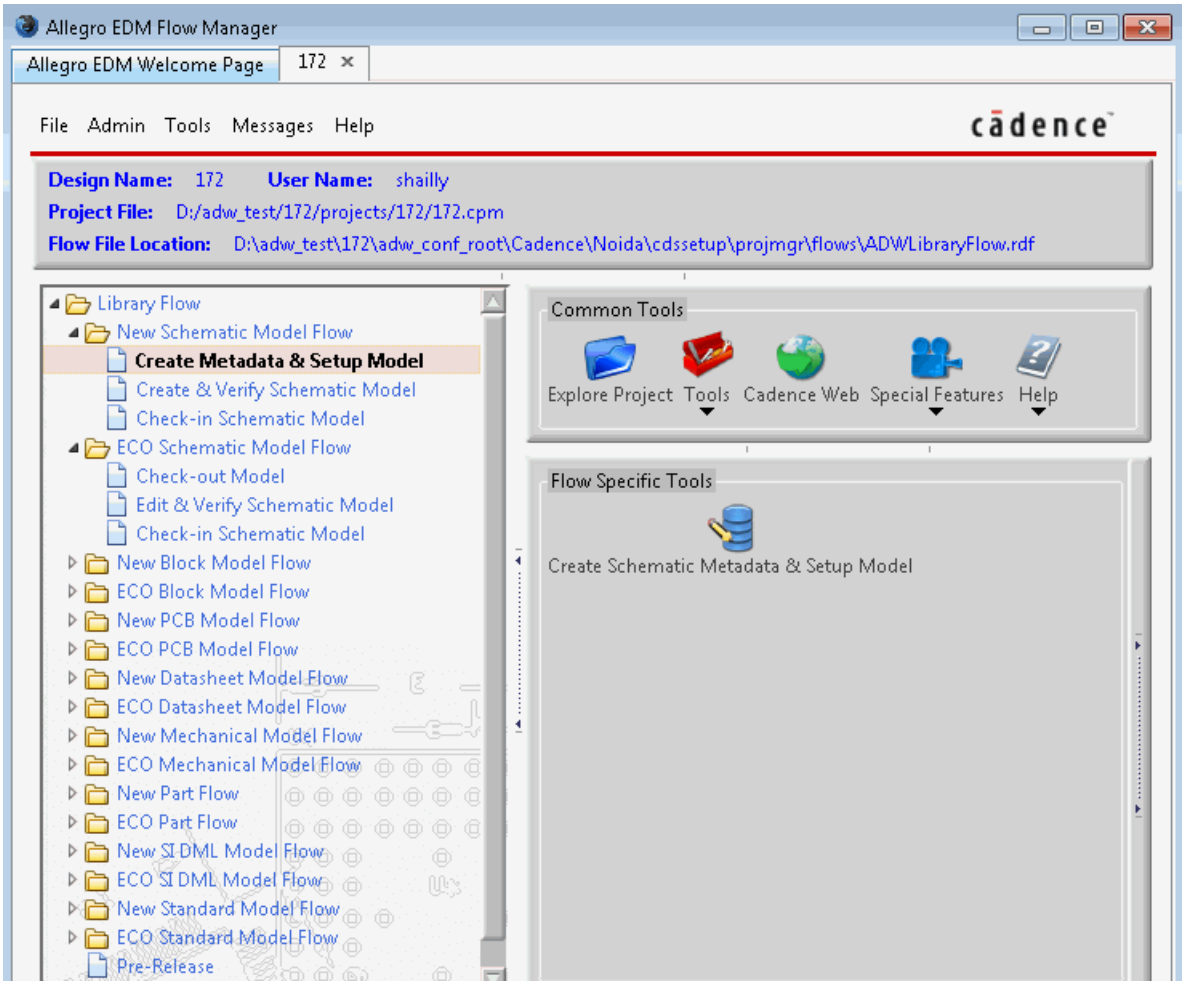
Prerequisites

You must be acquainted with the following design tools from Cadence.

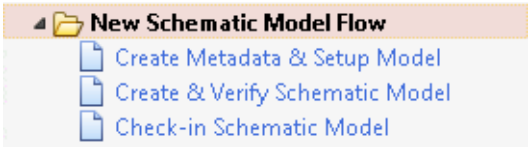
- Part Developer: Used for creating a symbol and a package
- Design Entry HDL: Used for creating a schematic design
- Allegro PCB Editor: Used for creating boards, shapes, padstacks, and footprints
- Design Entry HDL Rules Checker: Verification utility to check Allegro Design Entry HDL schematics for design rule violations. This tool can be run in the batch mode and in interactive mode.

Creating Sample Project

➤ Create a new library project in Allegro EDM Flow Manager.



The tasks to be performed for adding a new schematic model to the Allegro EDM library database are listed in the Library Flow tree view below the New Schematic Model Flow as shown in the following figure.

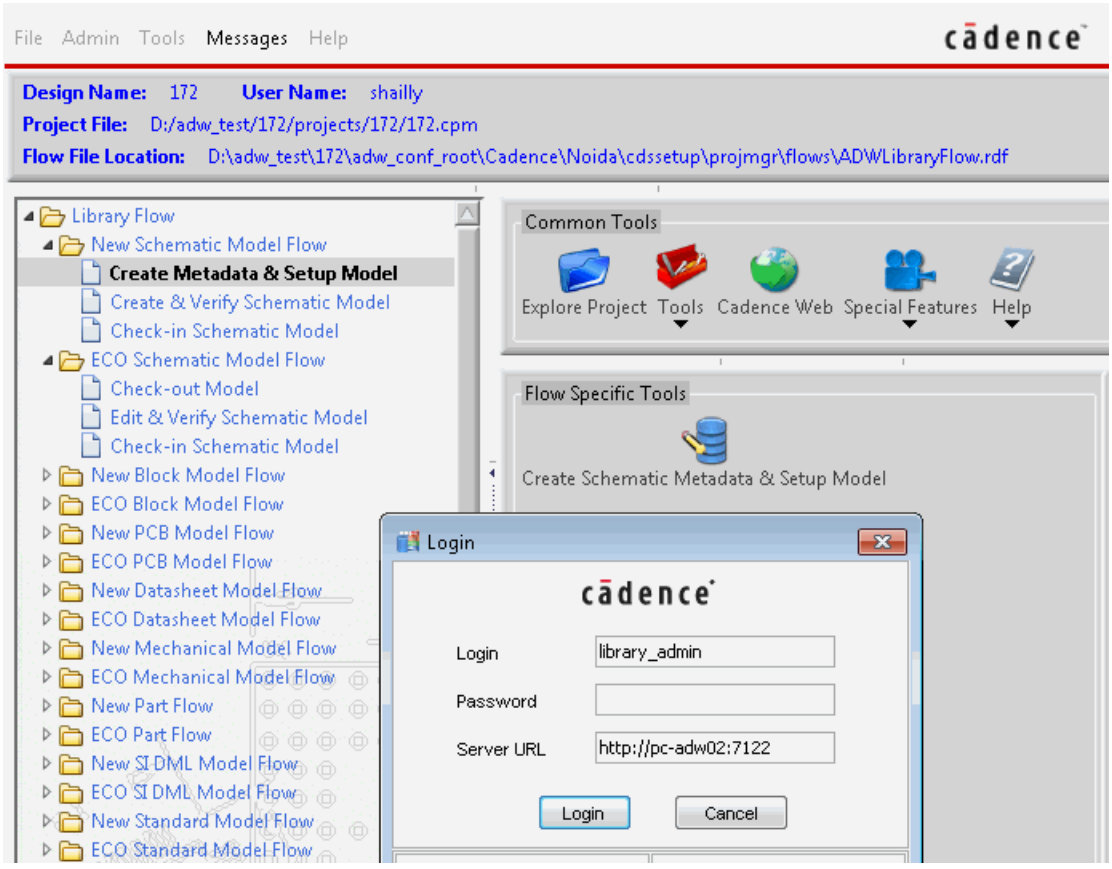


Creating Metadata

In this step, you add metadata for the schematic model to be created. This step is required to ensure that the new schematic model is searchable. For more information on this step, see Adding Metadata on page 22.

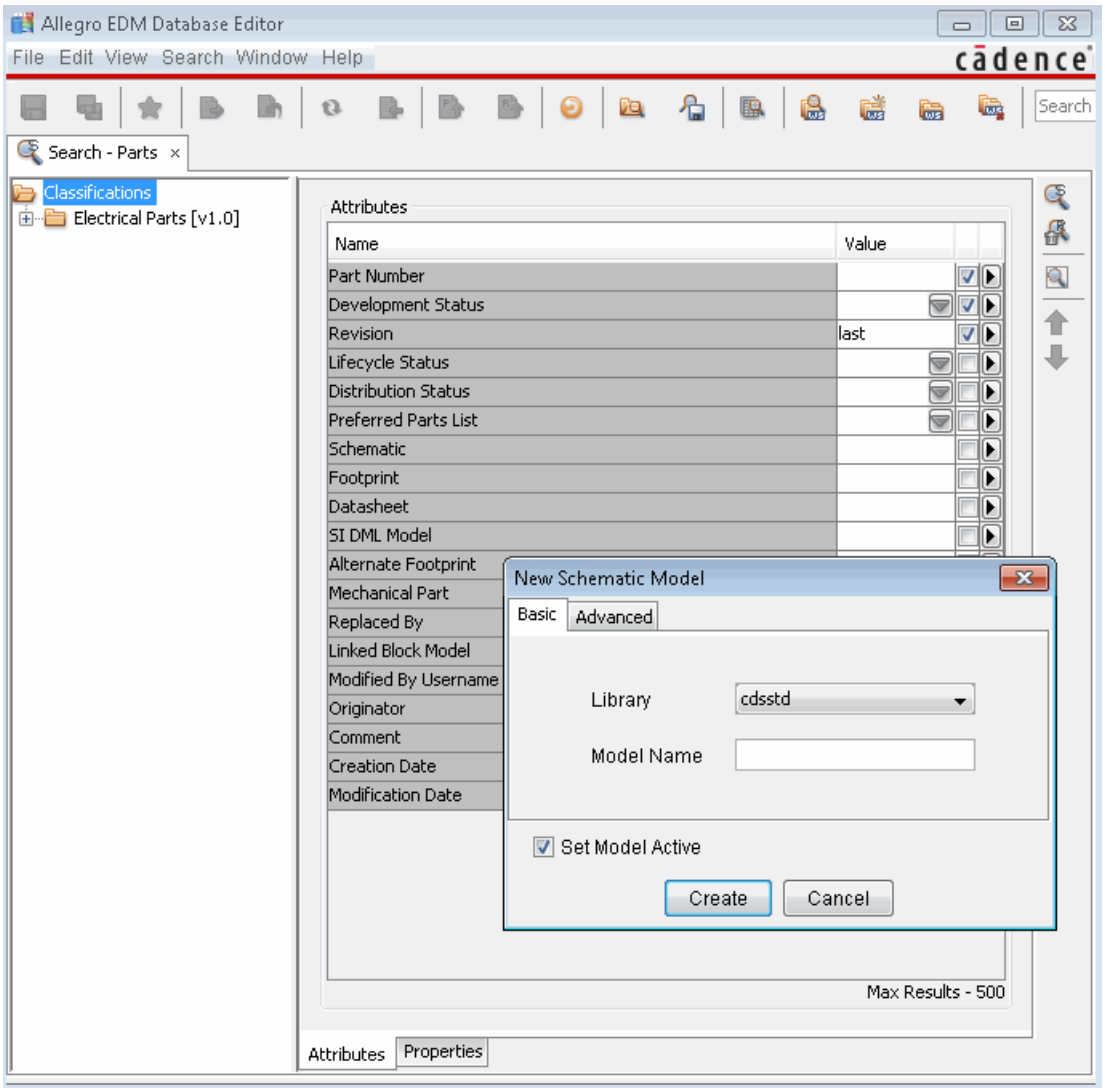
1. In the Library Flow tree view, under New Schematic Model Flow, select *Create Metadata & Setup Model*.
2. In the *Flow Specific Tools* pane, click the *Create Schematic Metadata & Setup Model* button.

   The Login screen appears.



3. Specify the user name and password to log on to Allegro EDM Database Editor.

Database Editor is launched and the New Schematic Model dialog box appears.



4. From the library drop-down list, select the `ic` library.

   The cell for the schematic model will be created in the `ic` library.

5. In the *Model Name* field, enter the model name as `8XC196NT`.

   By default, the cell name created in the library for the schematic model is the same as the name of the model. However, if required, the cell name can be specified in the Cell text box in the *Advanced* tab of the New Schematic Model dialog box.

6. In the *Cell* text box of the *Advanced* tab, specify the cell name as `8XC196`.

Ensure that the *Set Model Active* check box is selected. This is required to ensure that all tools launched from the Flow Manager use the model being created as the active model.

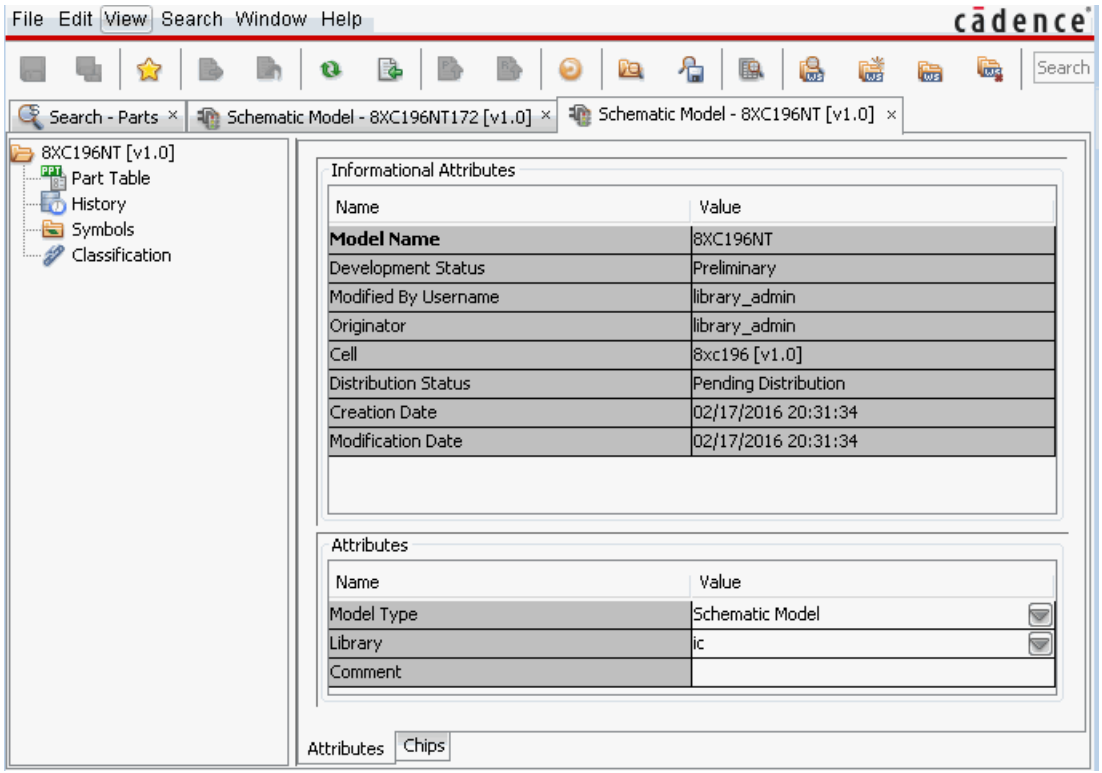
7. Click *Create*.

   If you do not have a working set created, a message appears prompting you to create a working set. If you create a working set, you need to specify a name for the set.

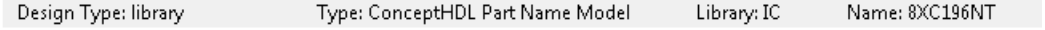
8. Click *Create*.

9. Specify the working set name and click *OK*.

   The metadata information is successfully added and the model attributes are displayed in Database Editor.



Note that the development status of the new schematic model is set to `Preliminary`, the cell name is `8xc196[v1.0]`, and the *Distribution Status* is set to `Pending Distribution`. After the metadata is created, the status bar of Flow Manager is also updated to display the current library and the Schematic Model name. This happens because the *Set Active* check box was selected in the New Schematic Model dialog box.



Creating and Verifying Schematic Model

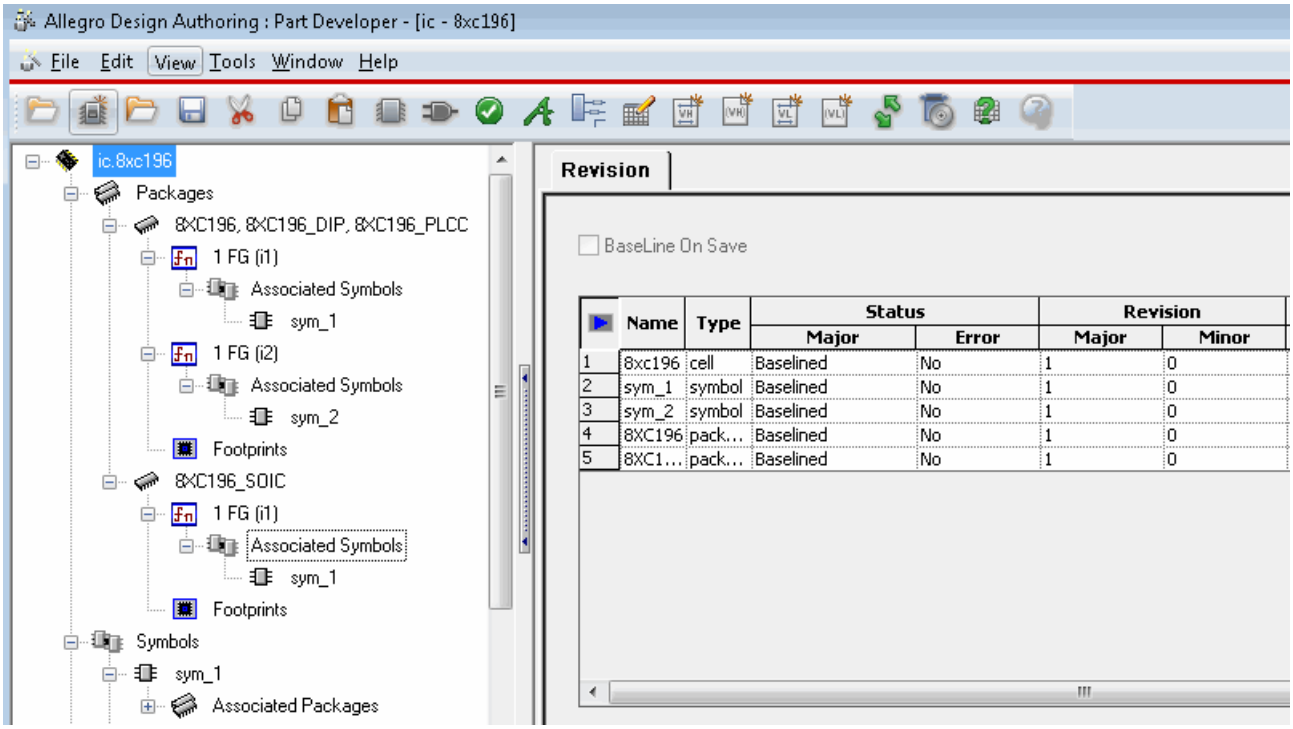
After adding the metadata information for the new schematic model to be created, the next step is to create the actual schematic model. To create the schematic model, the library design tool, Part Developer (PCB Librarian), is used.

1. In Flow Manager, under New Schematic Model Flow, select *Create & Verify Schematic Model*.
2. In the *Flow Specific Tools* pane, select *Create Symbols & Chips*.

   Part Developer is launched. In Part Developer, create the following:

   - Multiple primitives for the `8xc196` cell
   - Multiple symbols for each primitive

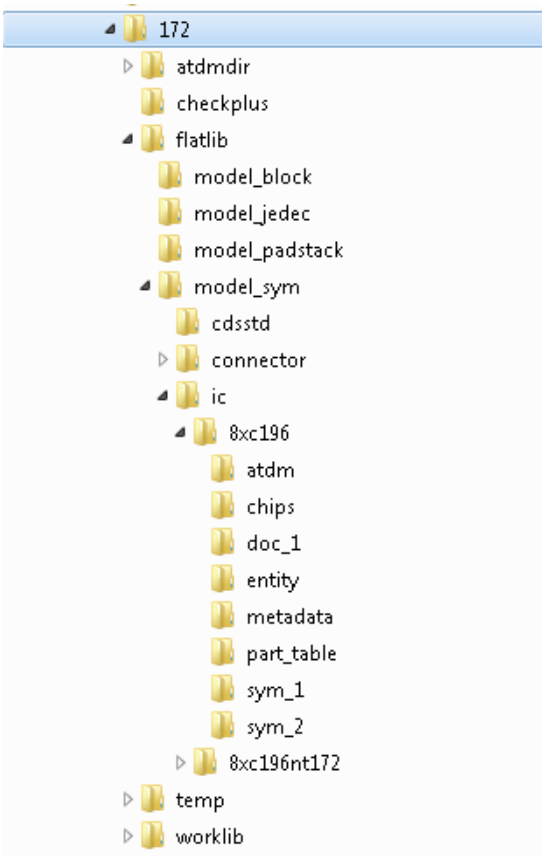
   As shown in the following figure, two parts— `8XC196` and `8XC196_SOIC` — have been created. There are multiple primitives for the `8XC196` part.



   For information on how to use Part Developer, see *Part Developer User Guide*.

3. Save and close Part Developer.

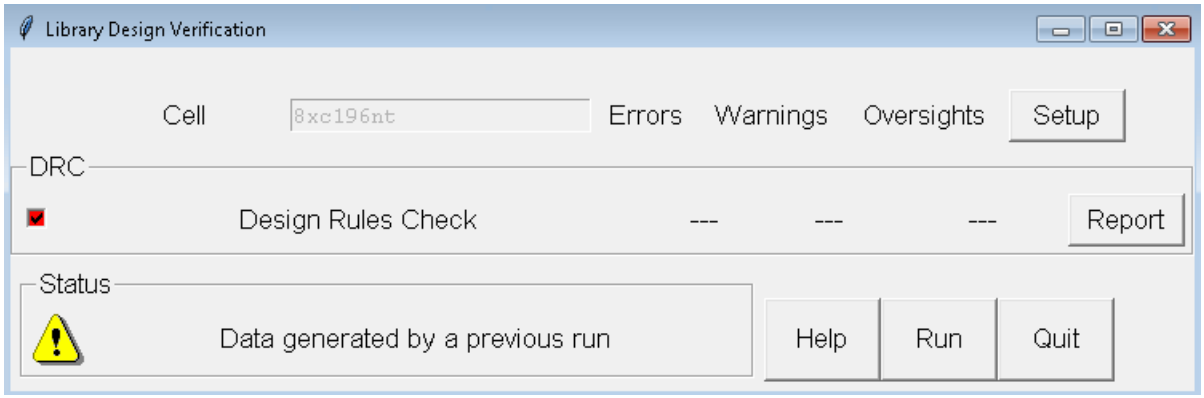
After you have created the symbol, the following directory structure is created in the work area.



4. In the *Flow Specific Tools* pane, select *Verify Symbol*.

   The Library Design Verification dialog box appears.

   a. Click *Run*.



b. To see any errors being reported, click the *Reports* button to open the log file in a text editor. Fix the errors in Part Developer, and verify the symbol again.

c. If no errors are reported, click the *Quit* button to close the Library Design Verification dialog box.

5. In the *Flow Specific Tools* pane, select *Create and Verify Test Schematic*.

Design Entry HDL is launched and a test schematic is created using the symbols in the `8xc196` cell.

The generated test schematic has two pages. This is because the number of pages in the test schematic is the same as the number of symbols in the cell with the schematic model.

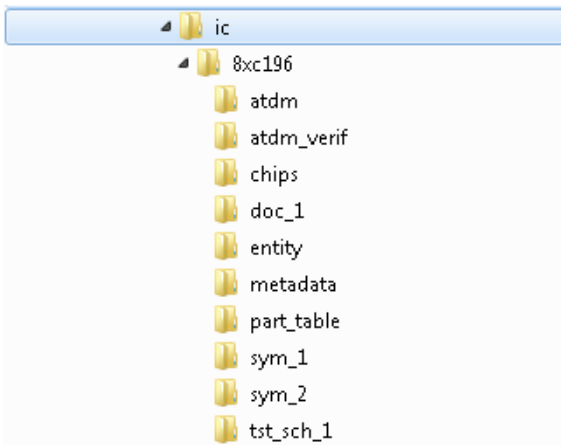
You have the option to manually package the schematic in DE-HDL and open it in PCB Editor. The packaged design can be checked on a board and provides options to verify pin swaps, PCB gate swapping, annotate classification names for Schematic Models, and so on.

This provides another level of verification that the symbol and footprint are correctly built and linked together at the part level, and allows for robust part testing across the library flow.

Note: When multiple classifications are associated to a single cell, only one of the classifications will be annotated on the test schematic that is generated.

6. Visually verify the symbols and exit Design Entry HDL.

This completes the task of creating and verifying the schematic model. The directory structure at this time is as shown in the following figure.



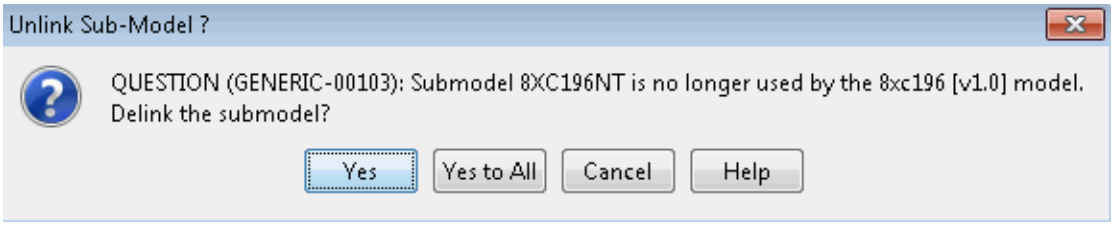
Synchronize Models

After you have created the schematic model, you can synchronize the model information in your hard disk (work area) with the information in the database (working set). This is an optional step because model synchronization is performed as a part of the check-in process.

1. To start synchronization, do one of the following tasks:

   - In Database Editor, choose *Edit – Synchronize Models*.
   - From the toolbar in Database Editor, click the *Synchronize Models (*  *)* button.

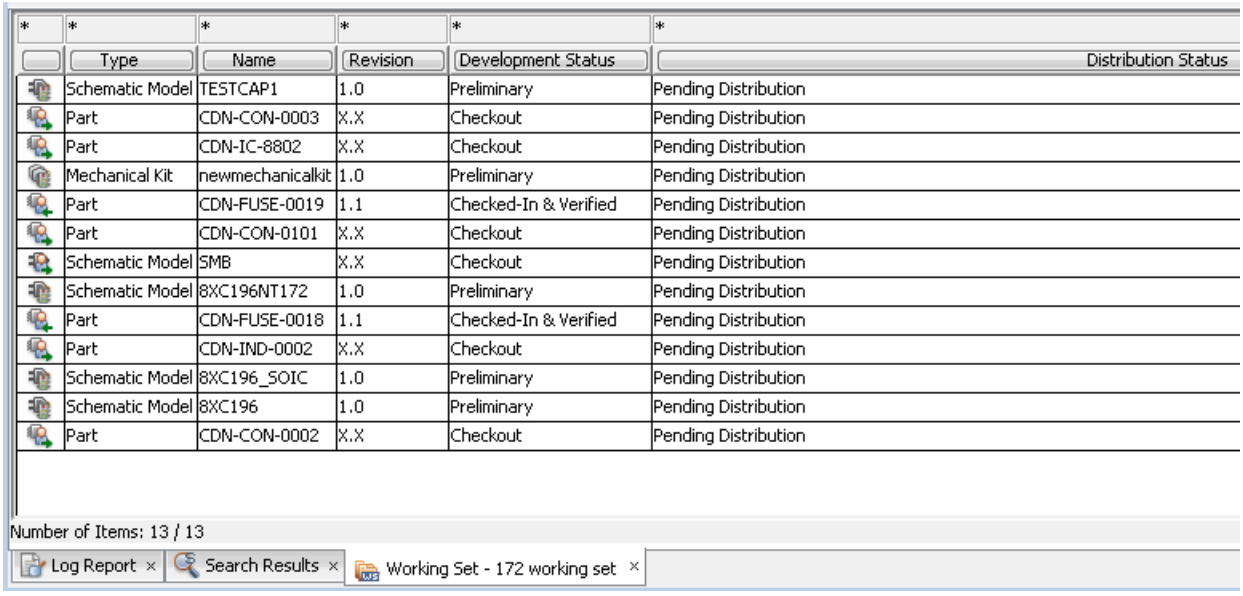
   On selecting *Synchronize Models*, the following message box appears.



   The message appears because none of the packages created in Part Developer for the `8xc196` cell have the name `8XC196NT`.

2. Click *Yes*.

   The model is delinked and the information about the schematic models created in Part Developer is loaded in the working set, as shown in the following figure.



Type	Name	Revision	Development Status	Distribution Status
Schematic Model	TESTCAP1	1.0	Preliminary	Pending Distribution
Part	CDN-CON-0003	X.X	Checkout	Pending Distribution
Part	CDN-IC-8802	X.X	Checkout	Pending Distribution
Mechanical Kit	newmechanicalkit	1.0	Preliminary	Pending Distribution
Part	CDN-FUSE-0019	1.1	Checked-In & Verified	Pending Distribution
Part	CDN-CON-0101	X.X	Checkout	Pending Distribution
Schematic Model	SMB	X.X	Checkout	Pending Distribution
Schematic Model	8XC196NT172	1.0	Preliminary	Pending Distribution
Part	CDN-FUSE-0018	1.1	Checked-In & Verified	Pending Distribution
Part	CDN-IND-0002	X.X	Checkout	Pending Distribution
Schematic Model	8XC196_SOIC	1.0	Preliminary	Pending Distribution
Schematic Model	8XC196	1.0	Preliminary	Pending Distribution
Part	CDN-CON-0002	X.X	Checkout	Pending Distribution

After synchronization, there are two schematic models displayed in the search pane. This is because we created two separate schematic models in Part Developer. Both the models, `8XC196` and `8XC196_SOIC`, are linked with the same cell, `8XC196`.

The schematic model information is read from the `chips` view for the `8XC196` cell. The following figure shows a section of the `chips.prt` file created for the cell.

chips.prt

```
FILE_TYPE=LIBRARY_PARTS;
primitive '8XC196','8XC196_DIP','8XC196_PLCC';
  pin
    'A1':
      PIN_NUMBER='(1,0)';
      PIN_TYPE='ANALOG';
      NO_LOAD_CHECK='Both';
      NO_IO_CHECK='Both';
      NO_ASSERT_CHECK='TRUE';
      NO_DIR_CHECK='TRUE';
      ALLOW_CONNECT='TRUE';
```

```
primitive '8XC196_SOIC';
  pin
    'A1':
      PIN_NUMBER='(1)';
      PIN_TYPE='ANALOG';
      NO_LOAD_CHECK='Both';
      NO_IO_CHECK='Both';
      NO_ASSERT_CHECK='TRUE';
      NO_DIR_CHECK='TRUE';
      ALLOW_CONNECT='TRUE';
    'A2':
      PIN_NUMBER='(2)';
      PIN_TYPE='ANALOG';
      NO_LOAD_CHECK='Both';
      NO_IO_CHECK='Both';
      NO_ASSERT_CHECK='TRUE';
      NO_DIR_CHECK='TRUE';
      ALLOW_CONNECT='TRUE';
  end_pin;
  body
    PART_NAME='8XC196_SOIC';
    BODY_NAME='8XC196';
    PHYS_DES_PREFIX='U';
    CLASS='IC';
  end_body;
end_primitive;
```

Tip

The value of the PART_NAME property in `chips.prt` is reported as the schematic model name in Database Editor.

Note: All the schematic models link to the same physical cell: `8XC196`. The `8XC196NT` schematic model was delinked from the `8XC196` cell as it does not appear in the chips view of the cell.

3. To view the data about `8XC196`, double-click the row with the schematic model name.

   The details open in a new tab.

   Note that the symbols attached to the schematic model and the schematic models linked to this model are listed in the tree view. The tree view also displays `8XC196_SOIC` as the schematic model linked to the `8XC196` schematic model.

   Note: By looking at the tree view and the `chips.prt` file, you can verify that the information is synchronized with the cell data in the work area.

4. To set this as the active model, do one of the following tasks.

   - In Database Editor, choose *Edit – Set Active*.
   - From the toolbar in Database Editor, click the *Set Active* () button.

   The status bar of the Library Flow UI is updated to indicate that `8XC196` is set as the active model.

Associating Schematic Classification

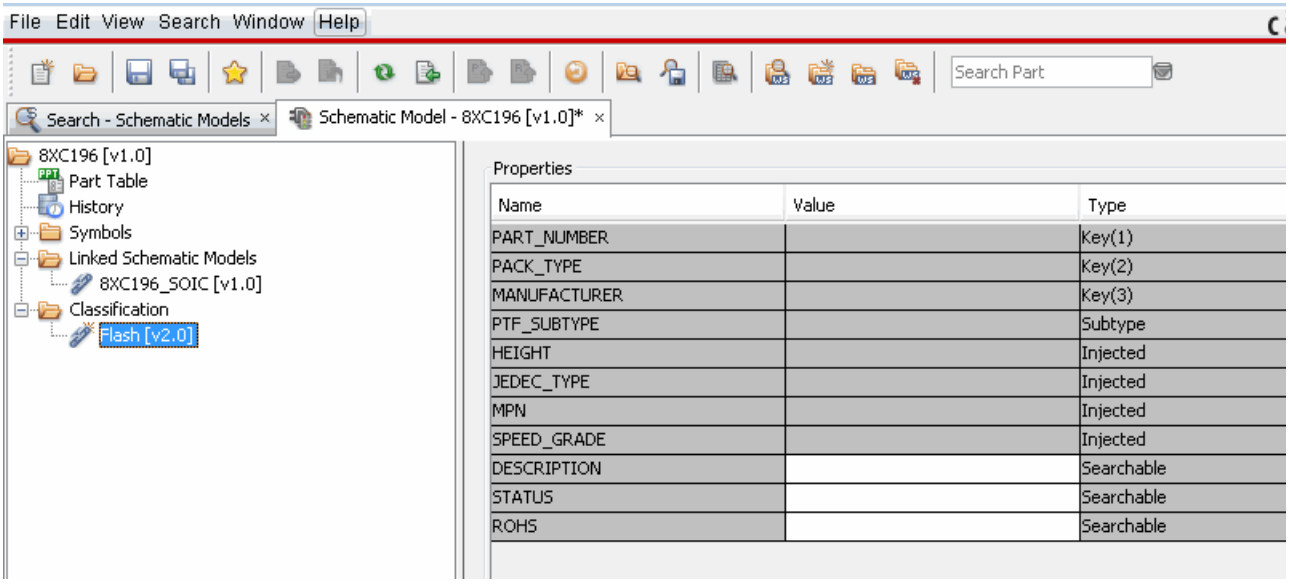
Once a schematic model is created, you need to associate a classification with the new schematic model. This step is required to obtain the PTF headers. The PTF header generated for the schematic model directly depends on the properties defined in the classification attached to the schematic model. The task of associating a schematic model classification is done in Database Editor.

To associate a schematic model classification to `8XC196`, complete the following steps.

1. In the left pane, select *Classification*.
2. Right-click and choose *Associate Classification From Tree*.

   The Associate Classification dialog box appears. In this dialog box, as you select a classification in the left pane, the properties in the classification are displayed in the right pane.
3. For this example, select classification Flash `[v2.0]` and click *Associate*.

   The classification properties are attached to the schematic model and are displayed in Database Editor.



Note that the *Value* column is editable. Wherever required, you can use this column to define the default property values.

4. Similarly, associate the `FLASH[v2.0]` classification to the `8XC196_SOIC` schematic model.

Tip

If you do not want to attach any property to the schematic model, associate an empty classification, `_EMPTYCLASSIFICATION[v1.0]`.

Important

Before you start creating a new schematic model, ensure that the required model classifications are available in the database. Creating new classifications or modifying an existing classification is a library administration task that can only be performed by librarians with administrator rights. For more information on managing classifications, refer to *Allegro EDM Database Editor User Guide*.

5. Choose *File – Save*.

The schematic model is now ready to be checked into the Allegro EDM database.

Check in Schematic Model

1. Initiate the check-in process.

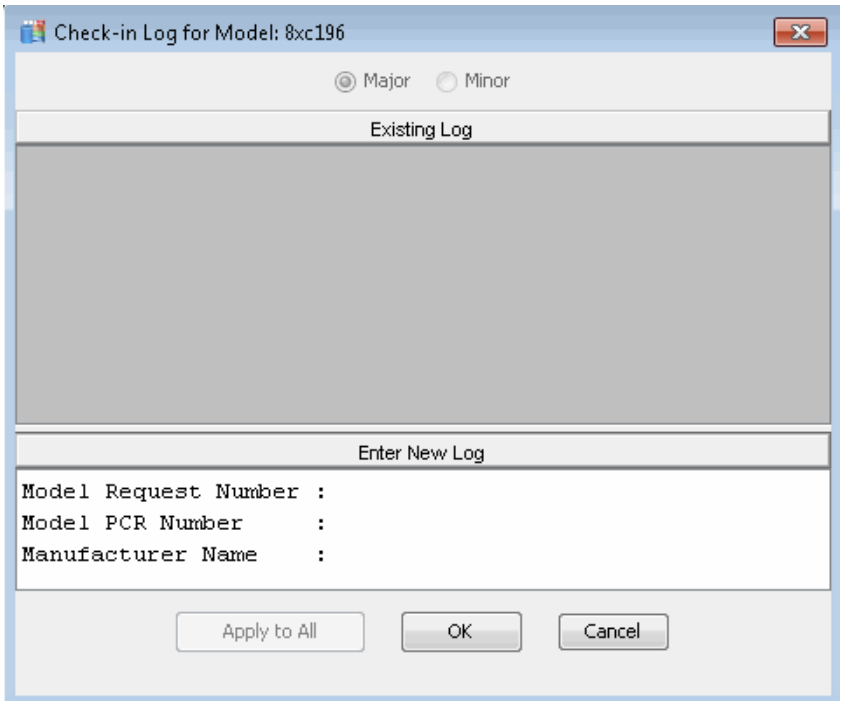
   A schematic model can be checked in using one of the following methods.

   - Using Library Flow
     - In Flow Manager, under New Schematic Model Flow, select *Check-in Schematic Model*.
     - In the *Flow Specific Tools* pane, click the *Check-in Schematic Model* button.
   - From Database Editor

     If Database Editor is open, check in the schematic model directly.

     - Choose *Edit – Check-in*.
     - From the toolbar, click the *Check-in (*  *)* button.

     The check-in window appears.



2. In the *Enter New Log* pane, enter the *Model Request Number*, *Model PCR Name*, and *Manufacturer Name*, and click *OK*.

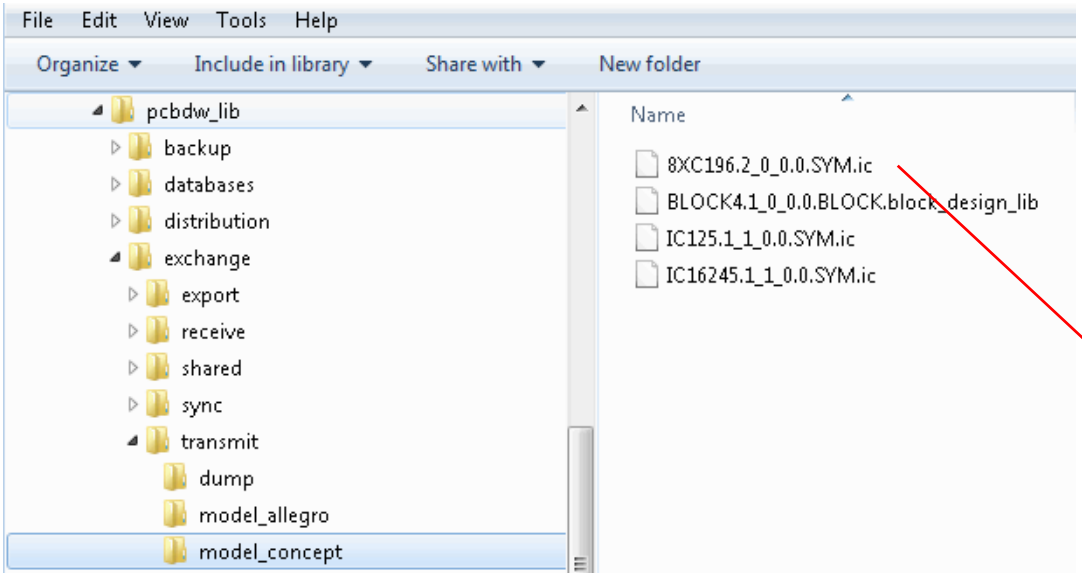
Tip

The information added in the log is displayed in Library Revision Manager (LRM), which is a tool used by PCB designer to display library modifications.

On successful check-in, the Development Status of the schematic model is changed to `Checked-In & Verified`.

Note that only the schematic models that were linked to the `8XC196` cell are checked in. The schematic model `8XC196NT` is not checked in as it was delinked from the `8XC196` cell.

Once the schematic model is checked in, the model information is copied to the integration area (in the folder specified by the `PCBDW_LIB` environment variable), as shown in the following figure.



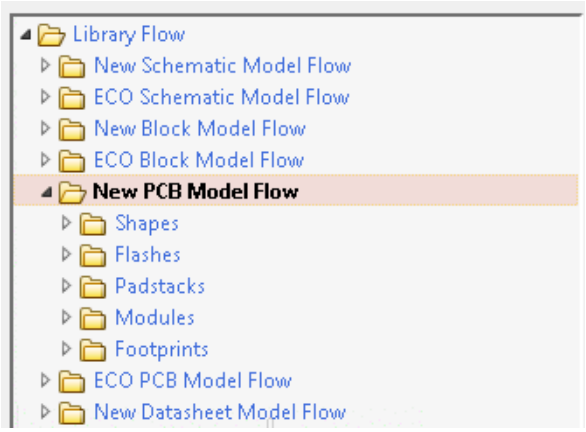
Summary

After you have created a schematic model using Part Developer, the `chips`, symbol (`sym_1`, `sym_2`, and so on), and entity views are created for the cell name specified in the New Schematic Model dialog box. The name of the schematic model appears as PART_NAME in the `chips.prt` file. The properties included in the classification attached to the schematic model provide the header information for the part table file generated for the part to which the schematic model is attached.

3

PCB Models

The Library Flow in Flow Manager allows you to add new padstack and footprints in the Allegro EDM database. Using the entries in the New PCB Model Flow, you can add new shapes, flashes, padstacks, and footprints.



Tip

While creating padstacks, you can use only the shapes and flashes that are available in the reference library. The reference library is at `<PCBDW_LIB>\reflib`. To add a new shape to the reference library, you need to distribute the library by running the Library Distribution utility.

Table 3-1 Tasks for Creating New PCB Models

Task	Tool(s) Used...	Steps	Output
Creating Shapes	Allegro EDM Database Editor	**1.** In the Library Flow tree view, select *New PCB Model Flow – Shapes – Create Padshape Metadata & Setup Model* **2.** In the *Flow Specific Tools* pane, click the *Create Padshape Metadata & Setup Model* button.	The New Shape Model window appears. Enter the model name and click *Create*.
	Allegro PCB Editor	**3.** In the Library Flow tree view, select *New PCB Model Flow – Shapes – Create Padshape Model*. **4.** In the *Flow Specific Tools* pane, click the *Create Pad Shape* button. Allegro PCB Editor is launched. In the Library Flow tree view, select *New PCB Model Flow – Shapes – Check-in Padshape Model*.	The `<model_name>.dra` file is created in the `<project_folder>\<project>\flatlib\model_padshape\<release>\shapes` folder.

Table 3-1 Tasks for Creating New PCB Models, *continued*

Task	Tool(s) Used...	Steps	Output
Creating a Flash Symbol	Allegro PCB Editor	**1.** In the Library Flow tree view, select *New PCB Model Flow – Flashes – Create Flash Metadata & Setup Model*. **2.** In the *Flow Specific Tools* pane, click the *Create Flash Metadata & Setup Model* button. **3.** In the Library Flow tree view, select *New PCB Model Flow – Flashes – Create Flash Model.* **4.** In the *Flow Specific Tools* pane, click the *Create Flash Symbol* button. Allegro PCB Editor is launched. **5.** In the Library Flow tree view, select *New PCB Model Flow – Flashes – Check-in Flash Model*.	The `<model_name>.dra` file is created in the `<project_folder>\<project>\flatlib\model_flash\<release>\flashes` folder. After check in, the model is available in the folder: `<pcbdw_lib>\exchange\transmit\model_allegro`

Table 3-1 Tasks for Creating New PCB Models, *continued*

Task	Tool(s) Used...	Steps	Output
Creating a Padstack	Pad Designer	**1.** In the Library Flow tree view, select *New PCB Model Flow – Padstacks – Create Padstack Metadata & Setup Model*. **2.** In the *Flow Specific Tools* pane, click the *Create Padstack Metadata & Setup Model* button. **3.** In the Library Flow tree view, select *New PCB Model Flow – Padstacks – Create Padstack Model*. **4.** In the Flow Specific Tools pane, click the *Create Padstack* button. Padstack Editor is launched. **5.** In the Library Flow tree view, select *New PCB Model Flow – Padstacks – Check-in Padstack Model*. **Note:** To use a new Shape or Flash for creating a Padstack, the padshape or the flash symbol must be in the Distributed state. This ensures that the padshape or the symbol is in the reference library, so that these models can be used while creating the padstacks.	The `<model_name>.pad` file is created in the `<project_folder>\project\flatlib\model_padstack\<release>\padstacks` folder. After check in, the model is available in the folder: `<pcbdw_lib>\exchange\transmit\model_allegro`

Table 3-1 Tasks for Creating New PCB Models, *continued*

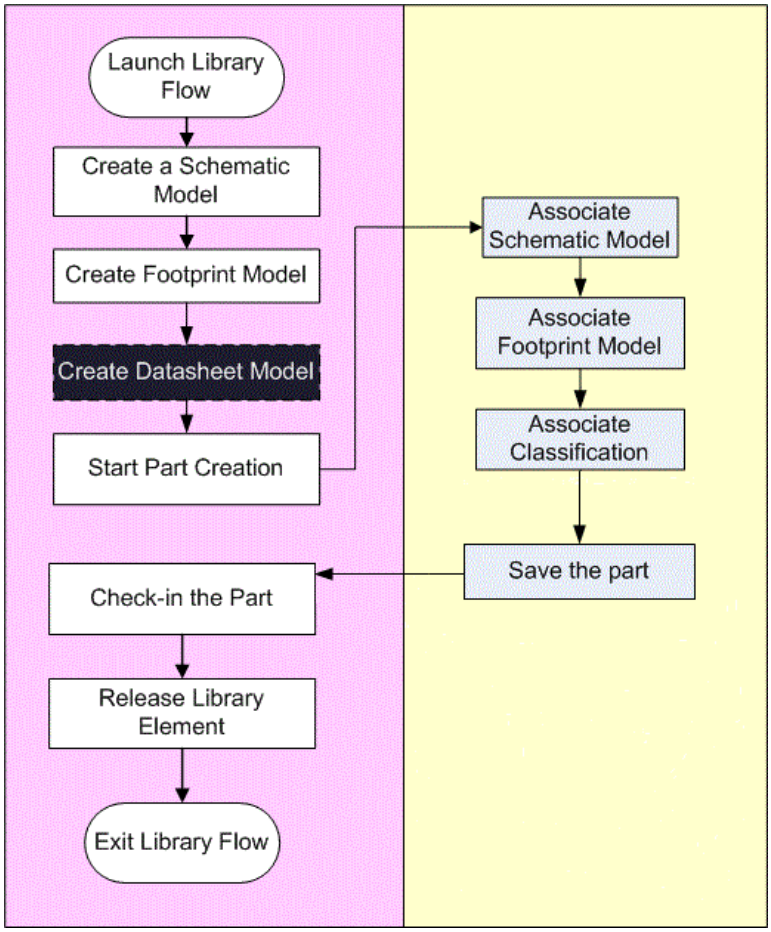
Task	Tool(s) Used...	Steps	Output
Creating Footprint	Allegro PCB Designer	**1.** In the Library Flow tree view, select *New PCB Model Flow – Footprints – Create Footprint Metadata & Setup Model*. **2.** In the *Flow Specific Tools* pane, click the *Create Footprint Metadata & Setup Model* button. **3.** In the Library Flow tree view, select *New PCB Model Flow – Footprints – Create &Verify Footprint Model*. **4.** In the *Flow Specific Tools* pane, click the *Create Footprint Model* button. Allegro PCB Designer is launched. **5.** In the *Flow Specific Tools* pane, click the *Verify PCB Symbol* button. **6.** In the Library Flow tree view, select *New PCB Model Flow – Footprints – Check-in Footprint Model*.	The `<model_name>.dra` and `<model_name>.psm` files are created in the *`<project_folder>`*`\project\flatlib\model_jedec\<`*`release`*`>\footprints` folder. After check in, the model is available in the folder: `<pcbdw_lib>\exchange\transmit\model_allegro`

4

Working With Parts

One of the main tasks performed by a librarian is to create new parts or components. Depending on the requirement, a librarian can either create a new part from scratch or can create a part using existing schematic models and footprints. This chapter covers the sequence of tasks to be followed for creating and updating a part. Figure 4-1 on page 49 depicts the sequence of tasks to be performed for creating a part using Library Flow.

Figure 4-1 Flow For Creating a Part



Prerequisites for Creating a Part

As seen in Figure 4-1 on page 49, before you can create a part using Allegro Library Manager, the following information about the part should be available in the database.

- Schematic symbol to be associated with the part

  To know more about adding schematic models to the Allegro EDM database, see Chapter 2, “Creating Schematic Models.”

- Footprint information about the part
- Datasheet models

Important

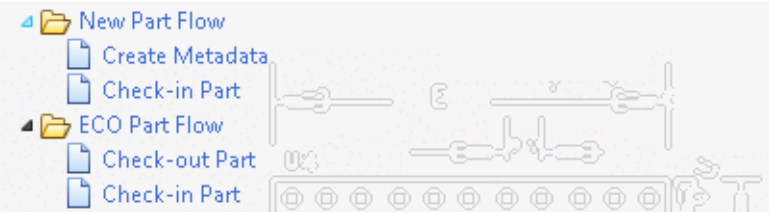
It is recommended that the prerequisite models should not be in the checked out state.

Creating a Part

To create a part, you need to perform two sets of tasks. As the first step, you need to add the metadata for the part in the database. The next step is to link together all the models — schematic model, PCB footprint, and datasheet — associated with the part. All these tasks are performed in Database Editor.

Adding Metadata Information

1. To create a part using Library Flow, select *New Part Flow* in the Library Flow tree view.



2. Select *Create Metadata*.
3. In the Flow Specific Tools pane, select *Create Part Metadata*.

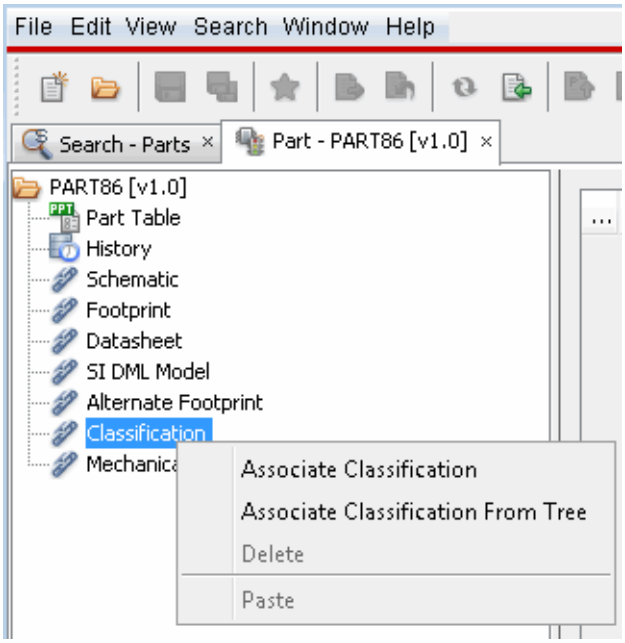
   Note that in the case of parts, the option to set Set Active is missing.

4. In the Login screen, enter the required details to log on to Database Editor.

5. In the New Part dialog box, enter the Part Number of the new part to be created and click *Create*.

   The data associated with the part is displayed in Database Editor. At this stage, the Development Status of the new part is set to Preliminary.

6. Next, you add the classification for the part. To do this, in Database Editor, right-click the Classification and choose *Associate Classification From Tree*.



7. In the Associate Classification dialog box, select the required classification and click *Associate*.

   Note: The properties in boldface in the Properties list box indicate the key properties in the part table file to be generated for the part.

8. In the left pane, select the classification attached to the part.

   The properties attached with the classification are listed in the *Properties* pane.

9. In the *Value* column, specify the value of each property.

Important

It is mandatory to specify the values for all property names in bold, as these are the key PTF properties. Specifying values for all other properties is optional.

10. Choose *File – Save* to save the part.

This completes the tasks of adding metadata for the part. Tasks covered in this section ensure that the new part is searchable in the database.

Linking Models

For any part to be usable, it should have an associated schematic model to enable designers to use the part while creating the logical part. Similarly, a part also needs to have a footprint and datasheet information to enable placement on board and to provide data required to manufacture the part. This section covers the steps to be performed to link different models to the part being created.

1. To associate a schematic model, right-click the *Schematic* node in the part tree view and choose *Associate Schematic From Tree*.
2. In the Associate Schematic dialog box, select the schematic model and click *Associate*.
3. Similarly, to link footprint data to the part, right-click the *Footprint* node, and choose *Associate Footprint From Tree*.
4. In the Associate Footprint dialog box, select the footprint and click *Associate*.
5. If required, you can also link datasheets to a part, though this is optional.
6. Choose *File – Save* to save the part.

Check in Parts

You can check in the part in the Allegro EDM component database using the Database Editor user interface as well as Flow Manager.

Using Database Editor

Use one of the following methods:

- Right-click the Part Number and choose *Check-in*.

  With the part information displayed, choose *Edit – Check-in*.

  The Check-in dialog box is displayed with a list of the rules against which the new part is to be verified.
- Click the Check-in button on the toolbar.
- Select *Edit – Library Flow – Check-in Parts.*

The *Check-in Parts* tab appears, listing all the parts that are in the Preliminary state and can be checked in.

Select the check box corresponding the part to be checked in and click *Check-in*.

The Check-in dialog box displays listing the rules against which new part is to be verified.

Using Flow Manager

1. In the Library Flow tree view, select *New Part Flow – Check-in Part*.
2. In the *Flow Specific Tools* pane, select *Check-in Part*.

   The *Check-in Parts* tab appears, listing all the parts that are in the `Preliminary` state and can be checked in.
3. Select the check box corresponding the part to be checked in and click *Check-in*.

   The Check-in dialog box displays listing the rules against which the new part is to be verified.

Modifying a Part

To modify a part using Library Flow, do the following:

1. Select the *ECO Part Flow – Check-out Part*

   The Search Parts tab opens in Database Editor.

   Note: A login screen appears if you are not already logged on to Database Editor.
2. Specify the search parameters and click the Search button.
3. In the *Search Results* tab, select the part row to be checked out.
4. Right-click and choose *Show Details*.
5. Check out the part using one of the following methods.
   - Right-click the part node and choose *Check-out*.
   - From the *Part - <Details>* tab, choose *Edit – Check-out*.
   - Click the Check-out button on the toolbar.

   If the part is checked out successfully, the *Development Status* of the part, displayed in the *Informational Attributes* pane, is changed to `Checkout`.
6. Modify the part as required.

Some of the modifications that can be done are:

- Delete one or all models linked to the part and link a new model.
- Add optional information, such as datasheet model, Preferred Part List, or Alternate Footprints.

7. Save the modified part and check in the part, using one of the options listed in the Check in Parts section.

5

Dialog Box Help

Library Design Verification

This dialog box lets you automatically check the new cell for DRC violations.

Option...	Description...
Setup	Select this button to review or configure the required rule set to be run.
Design Rule Check (DRC)	This check box is selected by default. This section displays the number of errors, warnings, and oversights found while running the design rule checks. The Rules Checker utility runs these checks.
Report	Displays the report generated for the schematic model after you click Run. The report is generated at: `<project_folder>\flatlib\model_sym\<library_name>\<cell_name>\atdm_verif\cp.log`
Run	Click this button to start the schematic verification process.
Status	This section displays a message indicating whether or not the information displayed is up-to-date or generated during any previous session.

New Schematic Model

Interface Element	Description
Library	Select the library in which the new schematic model is to be created.
Model Name	Specify the name of the new model to be created.
Set Model Active	Check box selected by default. Makes the new schematic model the active model to ensure that all tools launched from Flow Manager are launched with respect to this new schematic model.
Cell	Displays the cell name to be created based on the inputs provided in the *Basic* tab. You can modify the default value to specify a new value.

Options

This dialog box is displayed when you choose *Edit – Options* in Database Editor. For details see the *Configuring Data and Lifecycle Management Options* section in the *Allegro EDM Database Editor User Guide*.