

第 5 章 原理图元件库的管理

在电子产品系统开发过程中，如何快速、准确地找到所需要的元件，对于设计者来说是至关重要的。

Altium Designer 系统提供了相当完整的内置集成库文件，所存放的库元件数量非常庞大，几乎涵盖了世界上所有芯片制造厂商的产品，此外还与 2 个元器件供应商（Newark 和 Farnell）的实时数据连接。因此，借助于其强大灵活的库管理功能，大多数情况下，用户能够轻松地找到所需要的元件，并进行放置使用。

但是，对于某些比较特殊的、非标准化的元件或者新开发出来的元件，有时可能无法直接找到，另外，某些现有元件的原理图符号外形及其他模型形式也有可能不符合实际电路的设计要求。在这些情况下，就要求用户能够自己对库元件进行创建或者编辑，为其绘制合适的原理图符号或者其他模型形式，以满足自己的设计需要。

Altium Designer 系统为用户提供了多功能的库文件编辑器，使用户能够随心所欲地编辑符合自己要求的库元件，并可建立相应的库文件，加入到工程中，使得工程自成一体，便于工程数据的统一管理，也增加了其安全性和可移植性。

5.1 原理图库文件编辑器

使用 Altium Designer 系统的库文件编辑器可以创建多种库文件，执行【文件】|【新的】|【库】命令后，弹出的菜单如图 5-1 所示。

该菜单显示了可以创建的库文件类型，有原理图库（扩展名是.SchLib）、PCB 元件库（扩展名是.PcbLib）、焊盘过孔库（扩展名是.VHDLIB）、数据库（扩展名是.DbLib）以及 SVN 数据库（扩展名是.SVNDbLib）等。在这里，我们主要来看一下原理图库文件的创建和编辑。我们知道，元件的原理图符号本身并没有任何实际上的意义，只不过是一种代表了引脚电气分布关系的符号而已。因此，同一个元件的原理图符号可以具有多种形式（即可以使用多种显示模式），只要保证其所包含的引脚信息是正确的就行。但是，为了便于交流和统一管理，用户在设计原理图符号时，也应该尽量符合标准的要求，以便与系统库文件中所提供的库元件原理图符号做到形式与结构上的一致。



图 5-1 库文件菜单

5.1.1 原理图库文件编辑器的启动

启动原理图库文件编辑器有多种方法，通过新建一个原理图库文件，或者打开一个已

有的原理图库文件，都可以进入原理图库文件的编辑环境中。

执行【文件】|【新的】|【库】|【原理图库】命令，则一个默认名为 SchLib1.SchLib 的原理图库文件被创建，同时原理图库文件编辑器被启动，如图 5-2 所示。

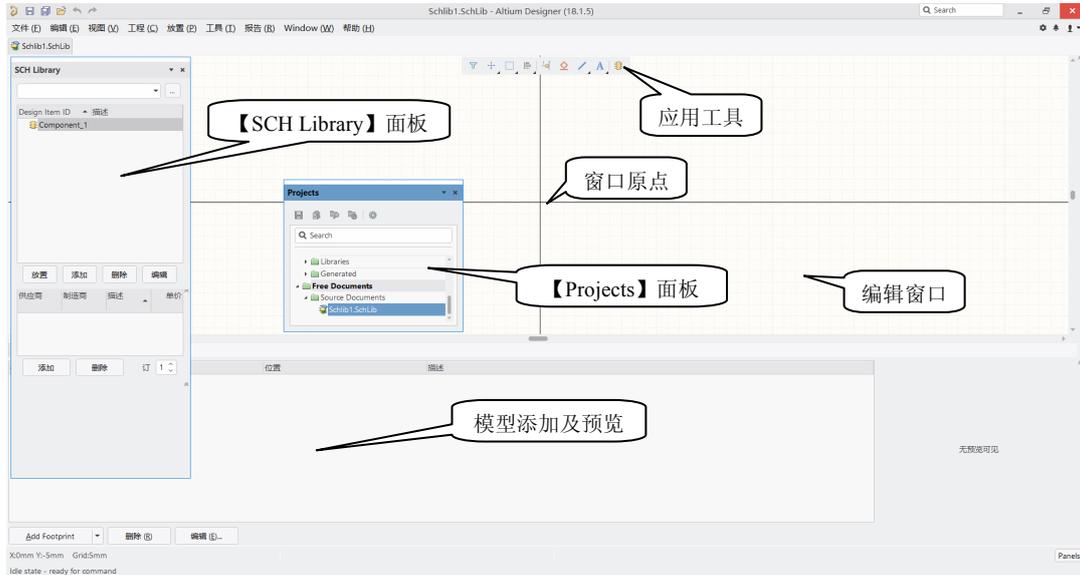


图 5-2 原理图库文件编辑器

5.1.2 原理图库文件编辑环境

原理图库文件编辑环境与前面的电路原理图编辑环境的界面非常相似，主要由主菜单栏、标准工具栏、应用工具、编辑窗口及面板等几部分组成，操作方法也几乎一样。但是也有不同的地方，具体表现在以下几个方面。

- ❑ 编辑窗口：窗口内不再有图纸框，而是被十字坐标轴划分为四个象限，坐标轴的交点即为该窗口的原点。一般在绘制元件时，其原点就放置在编辑窗口原点处，而具体元件的绘制、编辑则在第四象限内进行。
- ❑ 应用工具：提供了 3 个重要的工具，即原理图符号绘制工具栏 、IEEE 符号工具栏  和模式管理器 。这几个工具是原理图库文件编辑环境中所特有的，用于完成原理图符号的绘制以及通过模型管理器为元件添加相关的模型。
- ❑ 【Projects】面板：用于对项目进行编辑、管理。
- ❑ 【SCH Library】面板：原理图库文件编辑环境中特有的工作面板，用于对原理图库文件中的元件进行编辑、管理。
- ❑ 模型添加及预览：用于为元件添加相应模型，如 PCB 封装、仿真模型、信号完整性模型等，并可在右侧的窗口中进行预览。

 编辑窗口中的原点是为元件定位所设计的。在原理图中放置一个元件的时候，需要知道所放置的坐标或者位置，一般情况下默认第一个引脚为定位点。

5.1.3 原理图库应用工具栏

对于原理图库文件编辑环境中的主菜单栏及标准工具栏，由于功能和使用方法与原理图编辑环境中基本一致，在此不再赘述。我们主要对应用工具中的原理图符号绘制工具栏、IEEE 符号工具栏以及模式工具栏进行简要介绍，具体的使用方法在后面的实例中逐步介绍。

1. 原理图符号绘制工具栏

单击应用工具中的，会弹出相应的原理图符号绘制工具栏，如图 5-3 所示。其中各个图标的功能与【放置】子菜单中的各项命令具有对应的关系，如图 5-4 所示。



图 5-3 原理图符号绘制工具栏



图 5-4 【放置】子菜单

2. IEEE符号工具栏

单击应用工具中的，会弹出相应的 IEEE 符号工具栏，如图 5-5 所示，其上符合 IEEE 标准的一些图形符号。同样，由于该工具栏中各个符号的功能与执行【放置】【IEEE 符号】命令后弹出的菜单（如图 5-6 所示）中的各项操作具有对应的关系，所以不再逐项说明。

3. 模式工具栏

模式工具栏用来控制当前元件的显示模式，如图 5-7 所示。

- 模式** ▾：单击该图标可以为当前元件选择一种显示模式，系统默认为 Normal。
- +**：单击该图标可以为当前元件添加一种显示模式。
- ：单击该图标可以删除元件的当前显示模式。
- ←**：单击该图标可以切换到前一个显示模式。
- ：单击该图标可以切换到后一个显示模式。



图 5-5 IEEE 符号工具栏

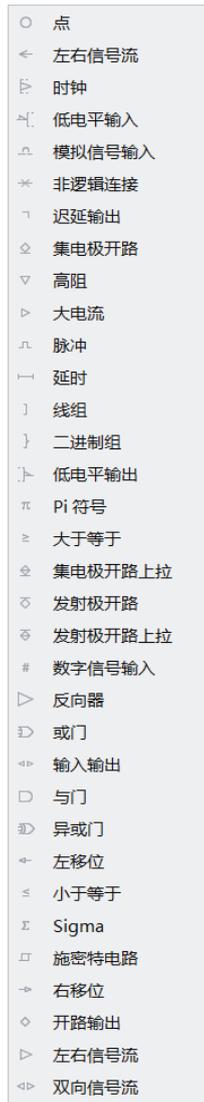


图 5-6 【IEEE 符号】菜单

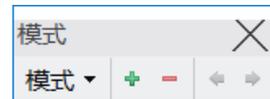


图 5-7 模式工具栏

5.1.4 【库】面板

【库】面板是原理图库文件编辑环境中的专用面板，用来对当前原理图库中的所有元件进行编辑和管理，如图 5-8 所示。

- **【设计条目】**：列出了当前原理图库中的所有库元件，包括元件名称及相应的描述等。选中某一库元件后，双击即可在当前打开的原理图文件中进行放置。
- **【模型名称】**：用于列出库元件的模型信息，如 PCB 封装、信号完整性分析模型、仿真模型等。单击相应的按钮，可为库元件添加模型或者编辑模型信息。
- **【供应商】**：用于显示可通过因特网提供元件供应商信息，包括供应商、制造商、描述等，这是系统默认显示的信息。



图 5-8 【库】面板

5.2 原理图库元件的创建

在对原理图库文件的编辑环境有所了解之后，这一节我们将通过一个具体元件的创建，使用户了解并熟练掌握建立原理图符号的方法和步骤，以便灵活地按照自己的需要，创建出美观大方、符合标准的原理图符号。

同样，与电路原理图的绘制类似，在创建库元件之前也应该对相关的工作区参数进行合理设置，以便提高效率 and 正确性，达到事半功倍的目的。

5.2.1 设置工作区参数

在原理图库文件的编辑环境中，执行【工具】|【文档选项】命令，弹出 Library Options 【Properties】属性面板，如图 5-9 所示，用户可以根据需要设置相应的参数。

Library Options 属性内容与原理图编辑环境中的 Document Options 属性内容相似，所以我们只介绍其中个别选项的含义，其他选项用户可以参考 Document Options 进行设置。

- **【Show Hidden Pins】**：用来设置是否显示库元件的隐藏引脚。选中该复选框后，元件的隐藏引脚将被显示出来。

隐藏引脚被显示出来，并没有改变引脚的隐藏属性。要改变其隐藏属性，只能通过 Pin 【Properties】 属性面板才能完成。

- 【Show Comment/Designator】：选中该复选框后，库元件的默认标识及注释将被显示出来。

5.2.2 库元件的创建

在对库元件进行具体创建之前，用户应参考相应元件的数据手册，充分了解其相关的参数，如引脚功能、封装形式等。

【例 5-1】 创建单片机芯片 STC11F02

下面我们将以第 4 章综合实例中所用到的单片机芯片 STC11F02 为例，详细讲述库元件原理图符号的绘制过程。

(1) 执行【文件】|【新的】|【库】|【原理图库】命令，启动原理图库文件编辑器，新建一个原理图库文件，命名为 R Radar.SchLib。在新建原理图库的同时，系统已自动为该库添加了一个默认名为 Component_1 的库元件，打开【SCH Library】面板即可以看到，如图 5-10 所示。

(2) 执行【工具】|【文档选项】命令，在 Library Options 【Properties】 属性面板中进行工作区参数设置。

集成元件的原理图符号外形，一般采用矩形或正方形表示，大小应根据引脚的多少来决定。由于我们使用的 STC11F02 是 20 引脚的 SOP/DIP 封装，所以应画成矩形。具体绘制时一般应画得大一些，以便于放置引脚，在引脚放置完毕后，可再调整为合适的尺寸。

(3) 单击原理图符号绘制工具栏中的放置矩形图标 ，则光标变为十字形，并附有一个矩形符号。以原点为基准，两次单击鼠标，在编辑窗口的第四象限内放置一个实心矩形。

(4) 单击放置引脚图标 ，则光标变为十字形，并粘附一个引脚符号，移动该引脚到矩形边框处，单击鼠标左键即可完成放置，如图 5-11 所示。

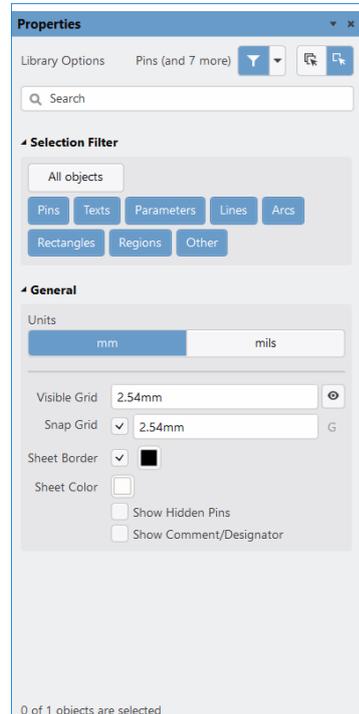


图 5-9 Library Options 属性面板

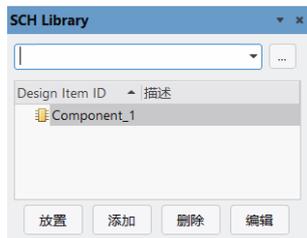


图 5-10 新建库元件

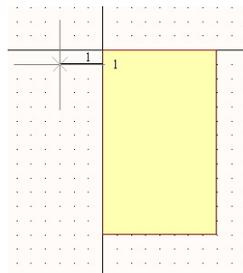
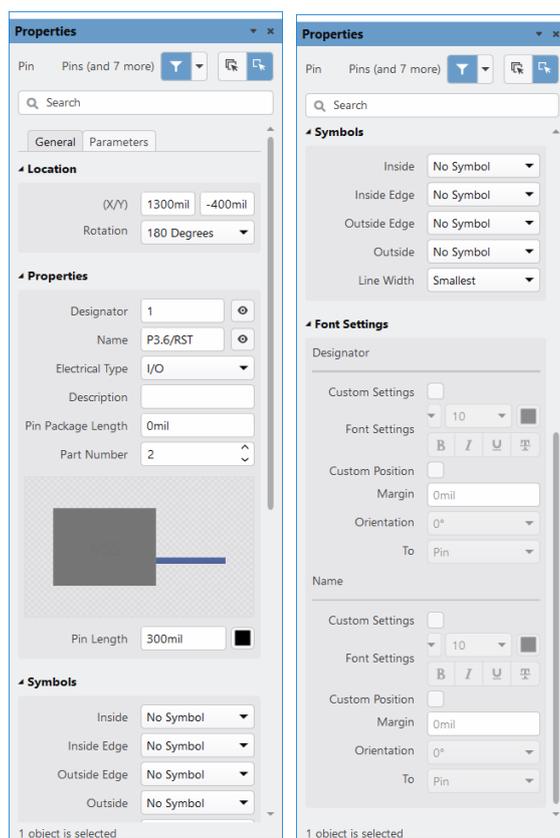


图 5-11 放置引脚

放置引脚时，一定要保证具有电气特性的一端，即带有 × 号的一端朝外，这可以通过在放置时按空格键旋转来实现。

(5) 在放置引脚时按下 Tab 键，或者双击已放置的引脚，则系统弹出 Pin 【Properties】属性面板，如图 5-12 (a) 所示，在面板中可以完成引脚的各项属性设置。



(a)

(b)

图 5-12 Pin 属性面板

面板各项属性含义如下。

- ❑ **【Name】**: 用于输入库元件引脚的功能名称。
- ❑ **【Designator】**: 用于设置引脚的编号，应该与实际的引脚编号相对应。选中这两项后面的 复选框后，显示形式如图 5-12 (b) 所示。
- ❑ **【Electrical Type】**: 用于设置库元件引脚的电气特性。单击右边的 ▾ 按钮可以选择设置，有 **【Input】**（输入引脚）、**【Output】**（输出引脚）、**【Power】**（电源引脚）、**【Open Emitter】**（发射极开路）、**【Open Collector】**（集电极开路）、**【HiZ】**（高阻）等 8 种。如果用户对各引脚的电气特性非常熟悉，也可以不必设置，以便简化原理图符号的形式。在这里，我们设置为 I/O，这是一个双向的输入/输出引脚。
- ❑ **【Description】**: 用于输入库元件引脚的特性描述。
- ❑ **【Symbols】**: 根据引脚的功能以及电气特性，用户可以为该引脚设置不同的 IEEE 符号，作为读图时的参考。可放置在原理图符号的里面、内边沿、外部边沿或外部等不同位置处，并没有任何电气意义。
- ❑ **【Font Settings】**: 用于设置该引脚的位置、长度、方位、颜色等基本属性以及是否锁定。

一般来说，【Name】、【Designator】以及【Electrical Type】属性是必须设置的。其余的各项，用户可以自行选择设置，也可以不必设置。

(6) 设置好属性的引脚如图 5-13 所示。

(7) 按照同样的操作，或者使用阵列粘贴功能，完成其余的 19 个引脚放置，并设置好相应的属性，如图 5-14 所示。

为了更好地满足原理图设计的实际需要，可以对所绘制原理图符号的尺寸大小以及各引脚位置进行适当的调整。

(8) 调整后的原理图符号如图 5-15 所示。

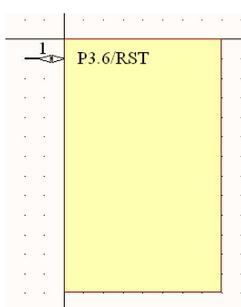


图 5-13 设置属性后的引脚

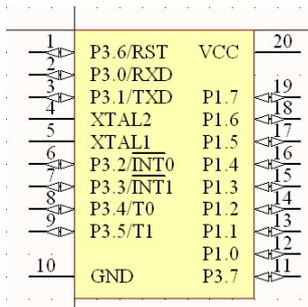


图 5-14 放置所有引脚

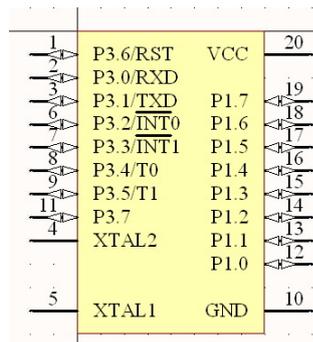


图 5-15 调整后的原理图符号

(9) 单击【SCH Library】面板上的 **编辑** 按钮，则系统弹出 Component 【Properties】属性面板，如图 5-16 所示。

在面板中的【General】标签页可以对所绘制的库元件进行特性描述以及其他属性参数的设置，主要设置有如下 6 项：

- ❑ **【Designator】**：默认的库元件标识符。即把该元件放置到原理图上时，系统最初默认显示的标识符。这里设置为 IC?，并选中右边 复选框，则放置该元件时，IC? 会显示在原理图上。
- ❑ **【Comment】**：库元件型号说明。这里设置为 STC11F02，并选中右边 复选框，则放置该元件时，STC11F02 会显示在原理图上。
- ❑ **【Description】**：库元件的性能描述，将显示在 SCH Library 面板上。这里输入 8-Bit Microcontroller, 2kB Flash。
- ❑ **【Type】**：库元件的符号类型设置。这里采用系统默认值 Standard 即可。
- ❑ **【Symbol Reference】**：在该文本编辑框中，用户可以为所绘制的库元件重新命名，这里输入 STC11F02。
- ❑ **【Local Colors】**：选中该复选框后，会显示 3 组颜色选

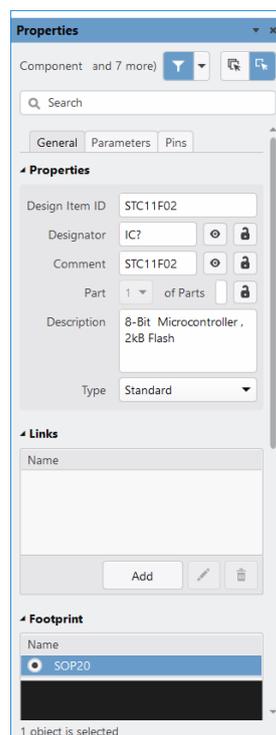


图 5-16 Component 属性面板

项，可分别设置原理图符号的填充、边线以及引脚的颜色。

在面板的【Parameters】标签页中，可以为库元件添加其他的参数，如版本、制造商、发布日期等。在【Models】选项区域中，则可以添加各种模型，如 PCB 封装、信号完整性模型、仿真模型、PCB 3D 模型等。

此外，双击【Pins】标签页下的标题框，则会打开【元件管脚编辑器】，可以对该元件的所有引脚进行一次性的编辑设置。

(10) 设置完毕后，关闭该面板。此时在 SCH Library 面板上显示了新建库元件 STC11F02 的有关信息。

至此，我们完成了元件 STC11F02 的创建。在设计电路原理图时，只需要将该元件所在的库文件加载，就可以随时取用该元件了。

为了方便用户之间的阅读和交流，有时还需要在绘制好的原理图符号上添加一些文本标注，如生产厂商、元件型号等。执行【放置】|【文本字符串】命令或者单击原理图符号绘制工具栏中的放置文本字符串图标，即可完成该项操作，在此不再过多说明。

5.3 原理图库元件的编辑

在建立了原理图库并创建了所需要的库元件以后，可以给用户的电路设计带来极大的方便。然而，随着电子技术的发展，各种新元件不断涌现，旧元件不断被淘汰，因此，用户对自己的原理图库也需要不断地随时更新，如添加新的库元件、删除不再使用的库元件或者编辑修改已有的库元件等，以满足电路设计的更高要求。

5.3.1 原理图库元件菜单命令

在原理图库文件的编辑环境中，系统提供了一系列对库元件进行管理编辑的命令，如图 5-17 所示的【工具】菜单。常用的命令主要有如下 13 项。

- 【新器件】：用于创建一个新的库元件。
- 【移除器件】：用于删除当前正在编辑的库元件。
- 【复制器件】：将当前库元件复制到目标库文件中。
- 【移动器件】：将当前库元件移送到目标库文件中。
- 【新部件】：用于为当前库元件添加一个子部件，与原理图符号绘制工具栏中的添加器件部件图标功能相同。
- 【移除部件】：用于删除当前库元件的一个子部件。
- 【模式】：用于对库元件的显示模式进行管理，包括添加、删除、切换等，功能与模式工具栏相同。
- 【查找器件】：用于打开【搜索库】对话框，进行库元件查找，与【库】面板上的按钮功能相同。

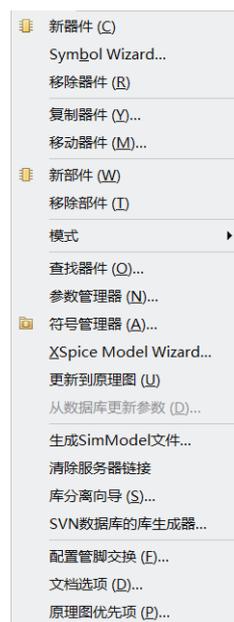


图 5-17 【工具】菜单

- **【参数管理器】**：用于打开**【参数编辑选项】**对话框，对当前的原理图库及其中库元件的相关参数进行查看、管理。
- **【符号管理器】**：用于打开**【模型管理器】**对话框，以便为当前库元件添加各种模型。
- **【XSpice Model Wizard】**：用于引导用户为当前库元件添加一个 SPICE 模型。
- **【更新到原理图】**：用于将编辑修改后的库元件更新到打开的电路原理图中。
- **【从数据库更新参数】**：用于将数据库更新到打开的电路原理图中。

5.3.2 原理图库文件添加模型

为满足不同的设计所需，一个元件中应包含多种模型。对于自行创建的库元件，其模型的来源主要有三种途径：一是由用户自己建立的模型，二是使用 Altium Designer 系统库中现有的模型，三是去相应的芯片供应商网站下载模型文件。

模型的添加可在**【Properties】**面板中进行，或者通过**【模型管理器】**来完成。

【例 5-2】 使用**【模型管理器】**为库元件添加封装

本例中，将为上面创建的库元件 STC11F02 添加 1 个 PCB 封装。

(1) 执行**【工具】|【符号管理器】**命令，打开**【模型管理器】**对话框，在左侧的列表选中元件 STC11F02，如图 5-18 所示。

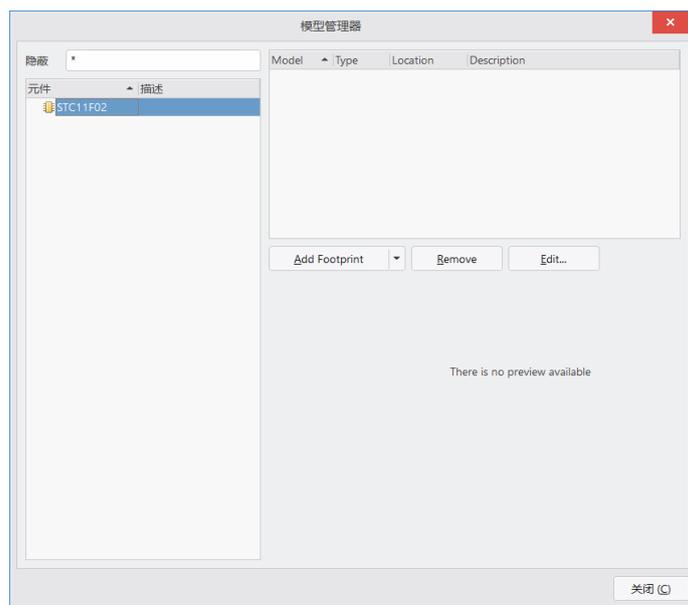


图 5-18 **【模型管理器】**对话框

(2) 单击 **Add Footprint** 按钮，打开**【PCB 模型】**对话框，如图 5-19 所示。

(3) 单击 **浏览 (B)...** 按钮，打开**【浏览库】**对话框，如图 5-20 所示。

【浏览库】对话框中显示的是当前可用的封装库（即已加载的封装库）信息，若用户从未加载过封装库，则该对话框中所有信息区域均为空白，如图 5-20 所示。



图 5-19 【PCB 模型】对话框

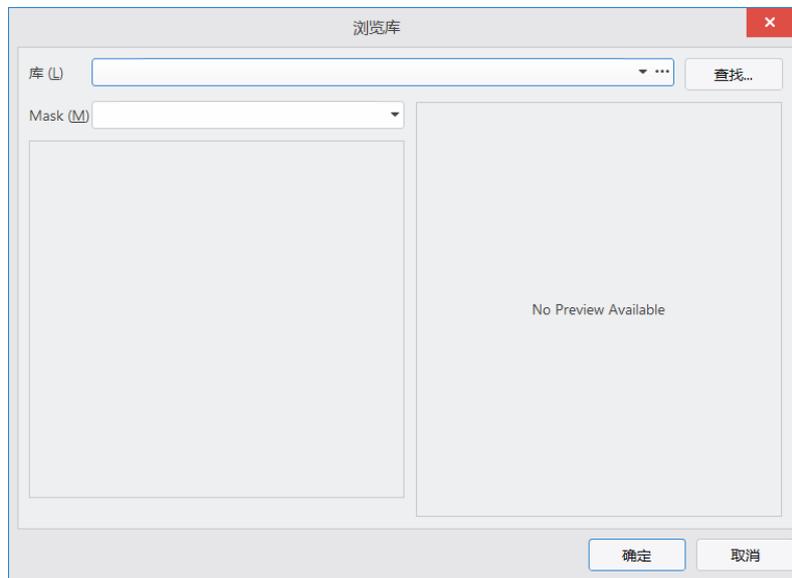


图 5-20 【浏览库】对话框

(4) 单击【浏览库】对话框中的 按钮，进入【搜索库】对话框。在【字段】

列表框的第一行选择 Name，在【运算符】列表框中选择 contains，在【值】列表框中输入要查找的封装名称 MHDR2X10，并选中【搜索路径中的库文件】单选按钮，如图 5-21 所示。

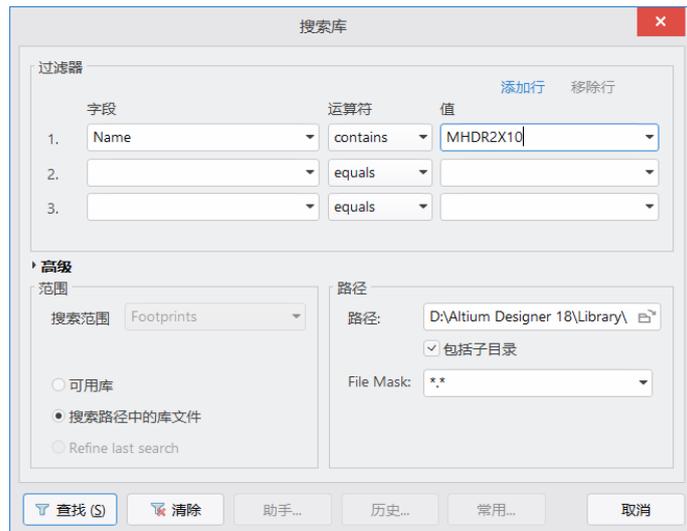


图 5-21 封装搜索设置

(5) 单击 **查找** 按钮，系统开始搜索。与此同时，搜索结果逐步显示在【浏览库】对话框中，如图 5-22 所示。可以看到，共有 1 个符合条件的封装，本例中选择 D:\Altium Designer 18\Library\Miscellaneous Connectors.IntLib 中的 MHDR2X10 封装。

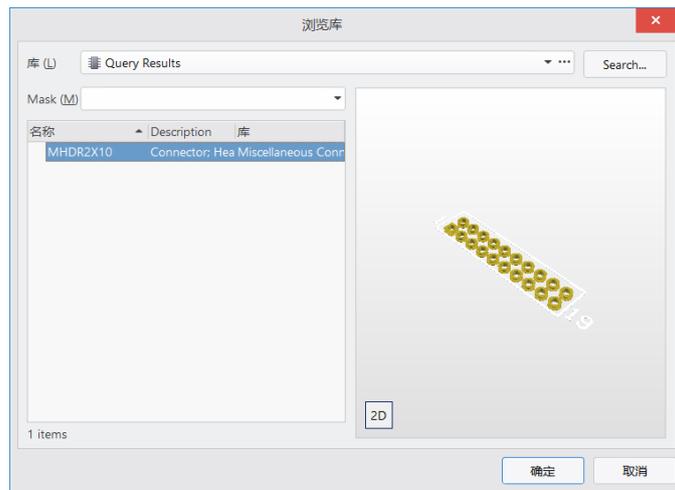


图 5-22 搜索结果显示

(6) 单击 **确定** 按钮，关闭【浏览库】对话框，加载相应的封装库。此时，【PCB 模型】对话框中已加载了选中的封装，如图 5-23 所示。

(7) 单击 **确定** 按钮，关闭【PCB 模型】对话框。可以看到，封装模型显示在了【模型管理器】中。同时，原理图库文件编辑器的模型区域也显示出了相应信息，如图 5-24 所示。

除了封装模型以外，还可以为元件添加仿真模型、3D 模型、信号完整性模型等，操作

过程与上面基本相同，在此不再重复。

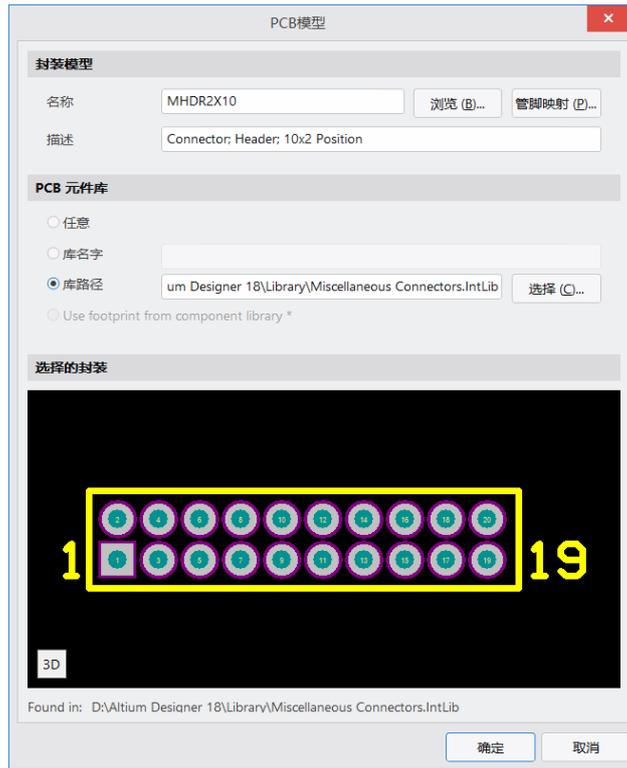


图 5-23 加载封装

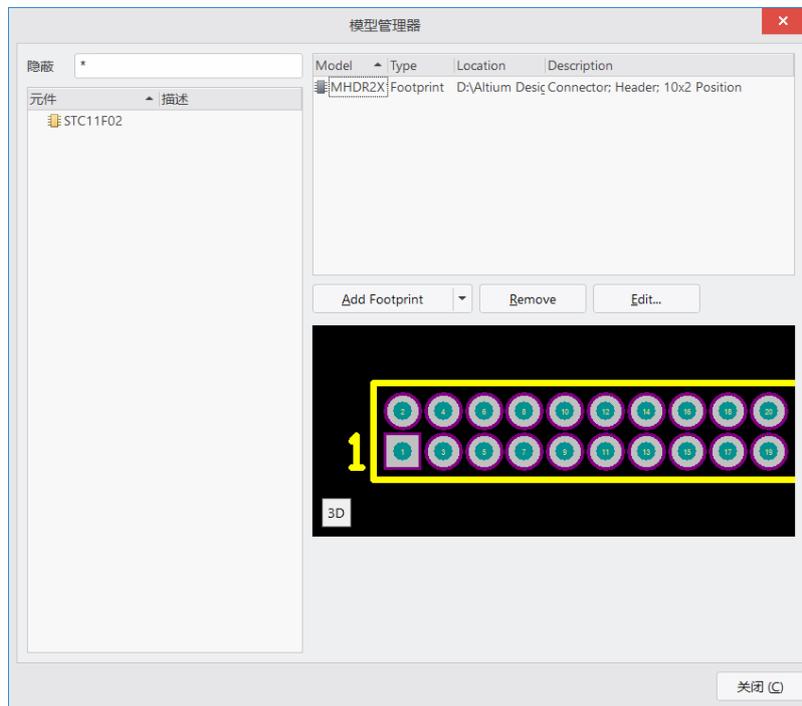


图 5-24 封装已添加

5.3.3 创建含有子部件的库元件

下面我们利用相应的库元件编辑命令，来创建一个含有子部件的库元件。

【例 5-3】 创建低噪声双运算放大器 NE5532

NE5532 是美国 TI 公司所生产的低噪声双运算放大器，在高速积分、采样保持等电路设计中常常用到，采用了 8 引脚的 DIP 封装形式。

(1) 打开前面建立的原理图库文件 R Radar.SchLib，使用所设置的默认工作区参数。

(2) 执行【工具】|【新器件】命令，则系统会弹出【New Component】对话框，输入新元件名称 NE5532，如图 5-25 所示。

(3) 单击原理图符号绘制工具栏  中的放置多边形图标 ，以编辑窗口的原点为基准，绘制一个三角形的运算放大器符号。

(4) 单击原理图符号绘制工具栏  中的放置引脚图标 ，放置引脚 1、2、3、4、8 在三角形符号上，并设置好每一引脚的相应属性，如图 5-26 所示，这样就完成了一个运算放大器原理图符号的绘制。

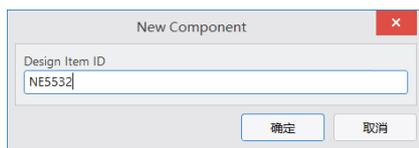


图 5-25 命名新元件

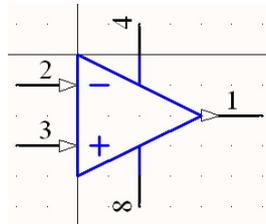


图 5-26 绘制一个子部件

 1 引脚为输出 OUT，2、3 引脚为输入 IN-、IN+，8、4 引脚为公共的电源引脚 V+、V-，可将其设置为隐藏引脚。多部件元件中，隐藏引脚不属于某一特定部件而是为所有子部件所共用的引脚。

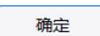
(5) 单击原理图标准工具栏中的图标 ，将图 5-26 所示的子部件原理图符号选中。

(6) 单击复制图标 ，将选中的子部件原理图符号进行复制。

(7) 执行【工具】|【新部件】命令。此时，在 SCH Library 面板上库元件 NE5532 的名称前面出现了一个  符号。单击  符号打开，可以看到该元件中有两个子部件，系统将刚才所绘制的子部件原理图符号命名为 Part A，还有一个子部件 Part B 是新创建的。

(8) 单击粘贴图标 ，将复制的子部件原理图符号粘贴在 Part B 中，并改变引脚序号：6、5 引脚为输入 IN-、IN+，7 引脚为输出 OUT，8、4 仍为公共的电源引脚 V+、V-。

(9) 在【SCH Library】面板上，双击库元件名称 NE5532，打开 Component【Properties】属性面板。在【Comment】文本编辑框中输入 NE5532，在【Description】文本编辑框中输入 Dual Low-Noise Operational Amplifier。

(10) 设置完毕后，单击  按钮，关闭对话框。

这样，一个含有 2 个子部件的库元件 NE5532 就建立好了，如图 5-27 所示。使用同样的方法，还可以创建含有多个子部件的库元件。

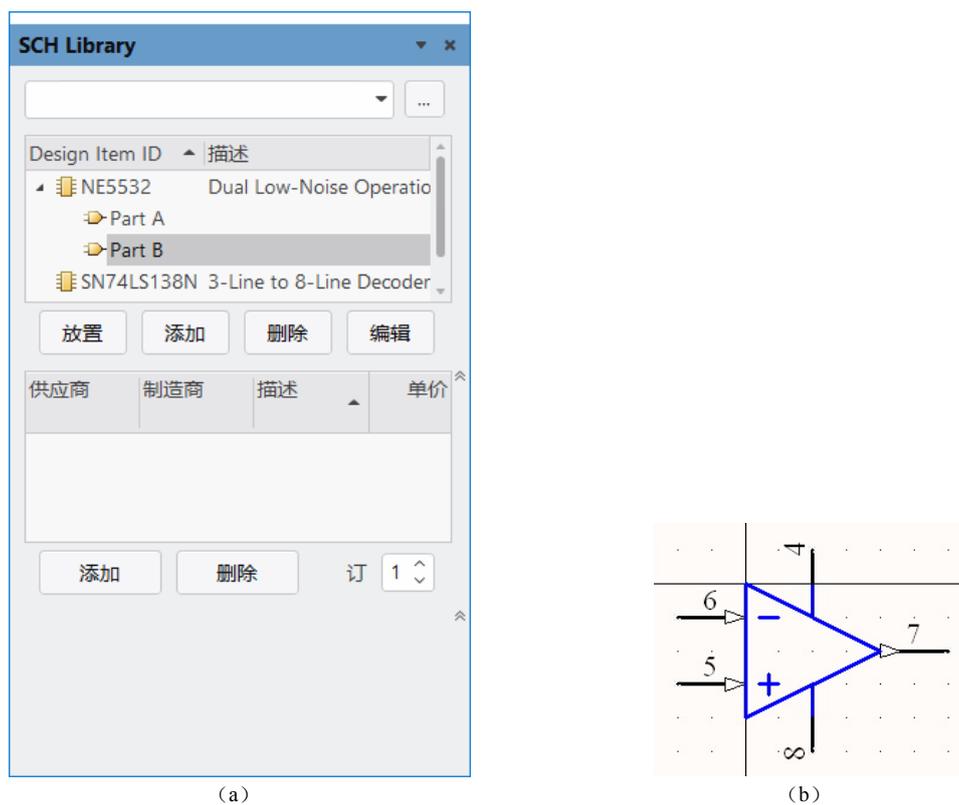


图 5-27 创建含有子部件的库元件

Altium Designer 系统中，执行【工具】|【模式】|【添加】命令，可为子部件建立多种显示模式，只是每种显示模式的引脚设置必须与普通模式一致。

5.3.4 复制库元件

用户要建立自己的原理图库，一种方式是自己创建各种库元件，绘制其原理图符号并编辑相应属性，就像前面我们所做的一样；还有一种方式是把现有库文件中的类似元件复制到自己的库文件中，直接使用或者在此基础上再进行编辑修改，创建出符合自己需要的原理图符号，这样可以大大提高设计效率，节省时间和精力。

下面我们以复制系统提供的集成库文件 TI Logic Decoder Demux.IntLib 中的元件 SN74LS138N 为例，介绍库元件的复制过程。

【例 5-4】复制库元件

把集成库 TI Logic Decoder Demux.IntLib 中的元件 SN74LS138N 复制到前面所创建的原理图库 R Radar.SchLib 中。

(1) 打开原理图库 R Radar.SchLib。

(2) 执行【文件】|【打开】命令，找到 D:\Altium Designer 18\Library\Texas Instruments 目录下的库文件 TI Logic Decoder Demux.IntLib，如图 5-28 所示。

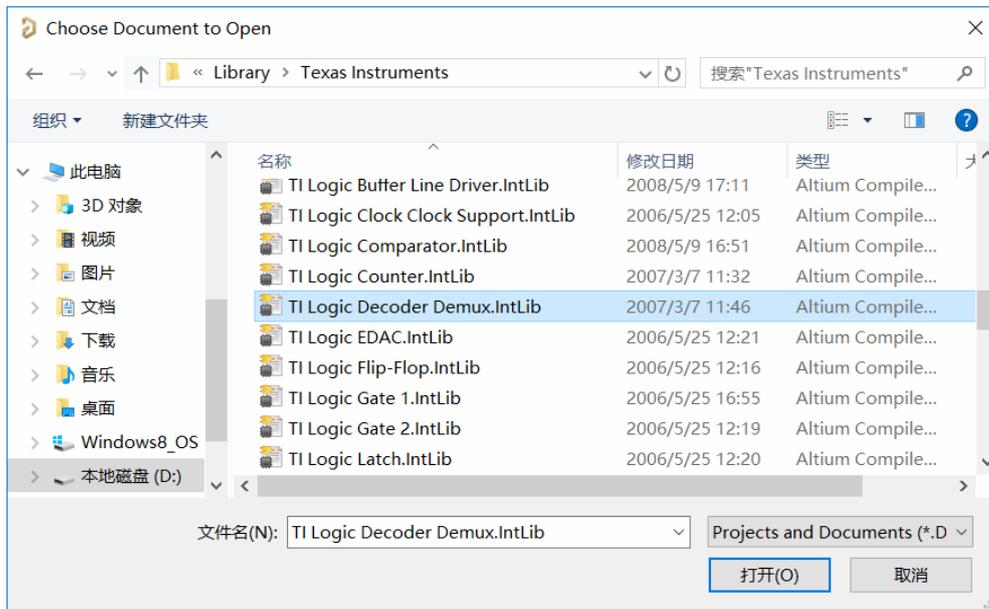


图 5-28 TI Logic Decoder Demux. IntLib 库文件

(3) 单击 **打开(O)** 按钮，则系统弹出如图 5-29 所示的【解压源文件或安装】提示框。

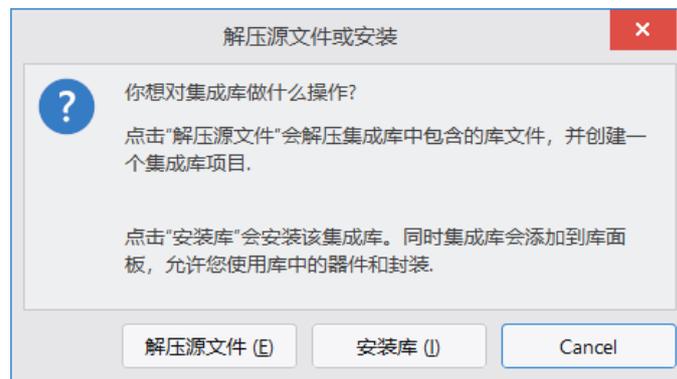


图 5-29 【解压源文件或安装】提示框

单击 **解压源文件 (E)** 按钮，系统会建立一个集成库工程，将该集成库分解为源库文件（原理图库和 PCB 库），供用户选择使用；单击 **安装库 (I)** 按钮，则系统只将该集成库加载到【库】面板上，而不会打开其源库文件。

(4) 单击 **解压源文件 (E)** 按钮，在【Projects】面板上显示出系统所建立的集成库工程 TI Logic Decoder Demux.LibPkg 以及分解成的两个源库文件：TI Logic Decoder Demux.PcbLib 和 TI Logic Decoder Demux.SchLib，如图 5-30 所示。

(5) 双击原理图库 TI Logic Decoder Demux.SchLib, 则该库文件被打开, 在 SCH Library 面板的元件栏中显示出库中的所有库元件, 如图 5-31 所示。

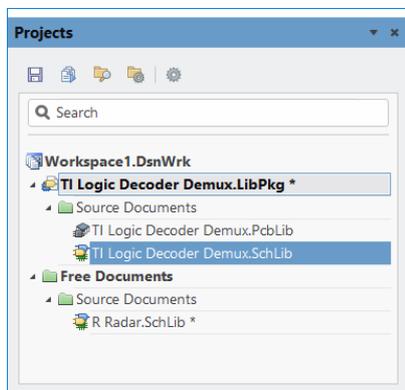


图 5-30 摘取源文件

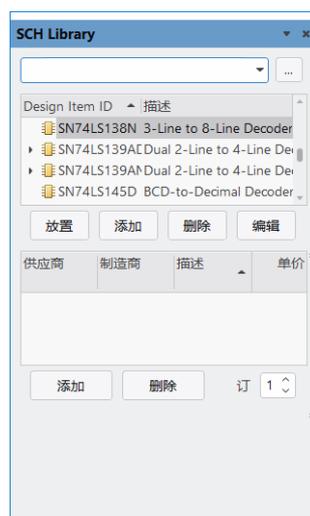


图 5-31 打开原理图库

(6) 选中库元件 SN74LS138N, 执行【工具】|【复制器件】命令, 则系统弹出【Destination Library】对话框, 如图 5-32 所示。

该对话框列出了当前处于打开状态的所有原理图库, 供用户选择将选中的库元件复制到哪个目标库中。

(7) 选择原理图库 R Radar.SchLib, 单击 按钮, 关闭对话框。

(8) 打开原理图库 R Radar.SchLib。通过【SCH Library】面板可以看到, 库元件 SN74LS138N 已复制到该原理图库中, 如图 5-33 所示。

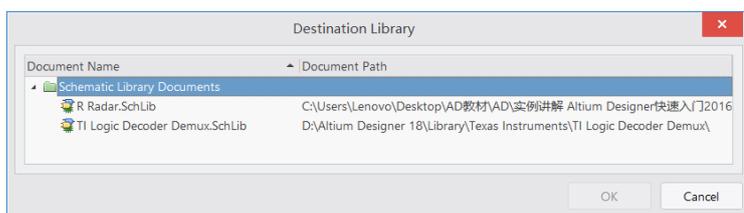


图 5-32 【Destination Library】对话框

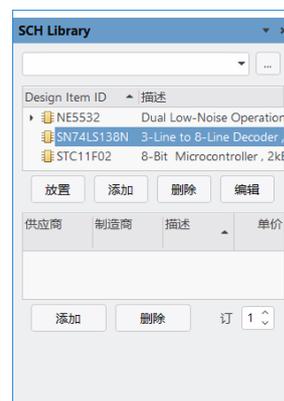


图 5-33 完成库元件复制

按照同样的操作, 可完成多个库元件的复制。对于复制过来的库元件, 用户可以进一步编辑、修改, 如重新设置引脚属性等, 以满足自己的实际设计需要。

📖 库元件复制完毕，应及时关闭集成库的源库文件。注意不要保存，以免对系统的集成库文件造成不必要的破坏。

5.4 制作工程原理图库

在一个设计工程中，所用到的元件由于性能、类型等诸多方面的不同，大多数情况下都来自于很多个不同的库文件。这些库文件中，有系统提供的若干个集成库，也有用户自己建立的原理图库，非常不便于管理，更不便于用户彼此之间的交流。

基于这一点，在原理图编辑环境中，用户可为自己的工程生成一个特定的原理图库，把工程中所用到的元件原理图符号都汇总到该原理图库中，脱离其他的库文件而独立存在，极大地方便了工程的统一管理。

下面，我们以工程 Audio AMP.PrjPCB 为例，为该工程生成自己的原理图库。

【例 5-5】生成工程原理图库

(1) 打开工程 Audio AMP.PrjPCB，进入电路原理图的编辑环境。

(2) 执行【设计】|【生成原理图库】命令，开始生成。生成过程中，对于有相同参考库的不同元件，系统会弹出如图 5-34 所示的对话框。这里，我们选中【处理所有元器件，并给予唯一名称】单选按钮，并选中【记住，下次不再询问】。

(3) 单击 按钮，关闭对话框后，工程原理图库已自动生成，系统同时弹出如图 5-35 所示的提示信息。



图 5-34 【复制的元器件】对话框

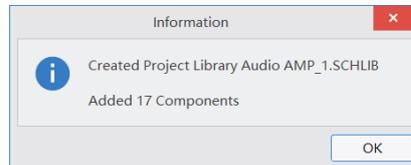


图 5-35 生成原理图库的提示信息

该提示信息告诉用户，当前工程的原理图库 Audio AMP.SCHLIB 已经生成，共添加了 17 个库元件。

(4) 单击 按钮确认，则系统自动切换到原理图库文件编辑环境中。在【SCH Library】面板上，列出了所生成的原理图库中的全部库元件及相应信息。

(5) 打开【Projects】面板，可以看到，工程 Audio AMP.PrjPcb 下的 Schematic Library Documents 文件夹中，已经存放了生成的原理图库 Audio AMP.SCHLIB，如图 5-36 所示。

📖 在生成的原理图库中，所存放的并不仅仅是元件的原理图符号，还有各种模型形式以及描述等。因此，准确地说，该库文件其实是工程的一个集成库，但是由于其扩展名为 .SCHLIB，所以在这里我们还是称之为原理图库。

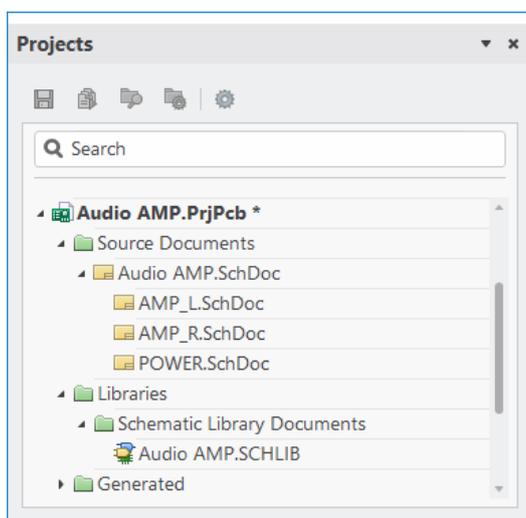


图 5-36 生成工程原理图库

建立了工程原理图库后，用户可以根据需要，很方便地对工程中所有用到的元件进行整体的编辑、修改，包括元件属性、引脚信息以及原理图符号形式等。更重要的是，如果我们在设计过程中，多次用到了同一个元件，则在该元件需要重新修改编辑时，不必到原理图中去逐一修改，而只需要在原理图库中修改相应的库元件，然后更新原理图即可。

5.5 器件报表输出及原理图库报告生成

在原理图库文件编辑器中，还可以生成各种报表及库报告，作为对库文件进行管理的辅助工具。用户在创建了自己的库元件并建立好自己的元件库以后，通过各种相应的报表，可查看库元件的详细信息、进行元件规则的有关检查等，以进一步完善所创建的库及库元件。

5.5.1 输出报表

下面以我们前面所建立的原理图库 R Radar.SchLib 为例，看一下各种报表的生成过程，并了解它们的不同作用。

【例 5-6】 生成器件报表

(1) 打开原理图库 R Radar.SchLib。

(2) 在【SCH Library】面板的元件栏中选择一个需要生成报表的库元件。如 STC11F02。

(3) 执行【报告】|【器件】命令，则系统自动生成了该库元件的报表，如图 5-37 所示。

器件报表是一个扩展名为.cmp 的文本文件，列出了库元件的属性及其引脚的配置情况，便于用户查看。

【例 5-7】 生成器件规则检查报表

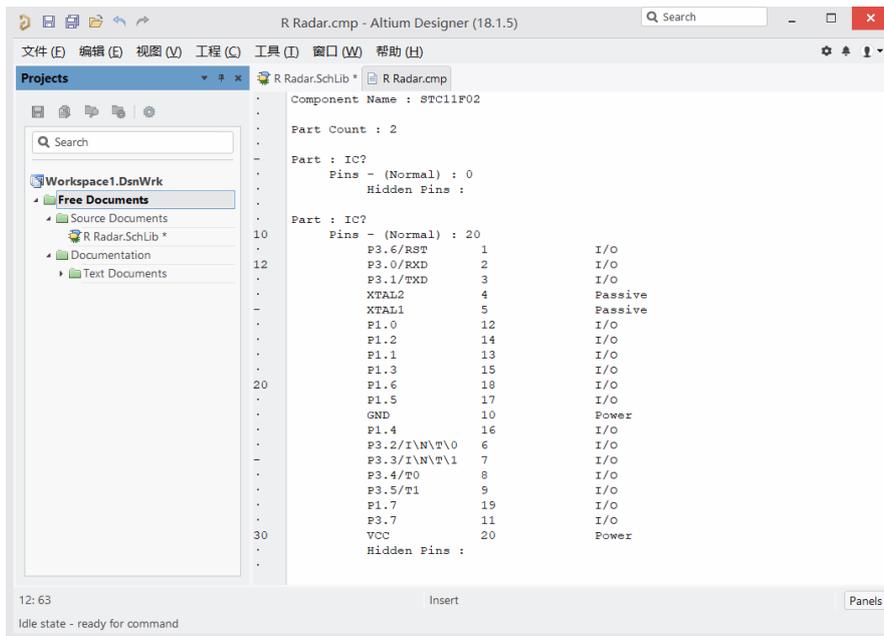


图 5-37 器件报表

- (1) 打开原理图库 R Radar.SchLib。
- (2) 执行【报告】|【器件规则检查】命令，系统弹出【库元件规则检测】设置对话框，如图 5-38 所示。



图 5-38 【库元件规则检测】对话框

对话框中有若干个复选框，供用户进行选择设置，含义如下：

- ❑ **【元件名称】**：用于设置是否检查重复的库元件名称。选中该复选框后，如果库中存在重复的库元件名称，则系统会把这种情况视为规则错误，显示在错误报表中；否则，不进行该项检查。
- ❑ **【管脚】**：用于设置是否检查重复的引脚名称。选中该复选框后，系统会检查每一库元件的引脚是否存在同名错误，并给出相应报告；否则，不进行该项检查。
- ❑ **【描述】**：选中该复选框后，系统将检查每一库元件属性中的**【描述】**框是否空缺，若空缺，则给出错误报告。

- **【封装】**：选中该复选框后，系统将检查每一库元件的封装模型是否空缺，若空缺，则给出错误报告。
- **【默认标识】**：选中该复选框后，系统将检查每一库元件的默认标识符是否空缺，若空缺，则给出错误报告。
- **【管脚名】**：选中该复选框后，系统将检查每一库元件是否存在引脚名称空缺的情况，若空缺，则给出错误报告。
- **【管脚号】**：选中该复选框后，系统将检查每一库元件是否存在引脚编号空缺的情况，若空缺，则给出错误报告。
- **【序列中丢失管脚】**：选中该复选框后，系统将检查每一库元件是否存在引脚编号不连续的情况，若存在，则给出错误报告。

用户可自行选择设置想要检测的选项，对于不需要检测的选项，忽略掉即可，以免产生不必要的错误报告。

(3) 设置完毕，单击 **确定** 按钮，关闭对话框，此时系统自动生成了该库文件的器件规则检查报表，是扩展名为.ERR 的文本文件，如图 5-39 所示。

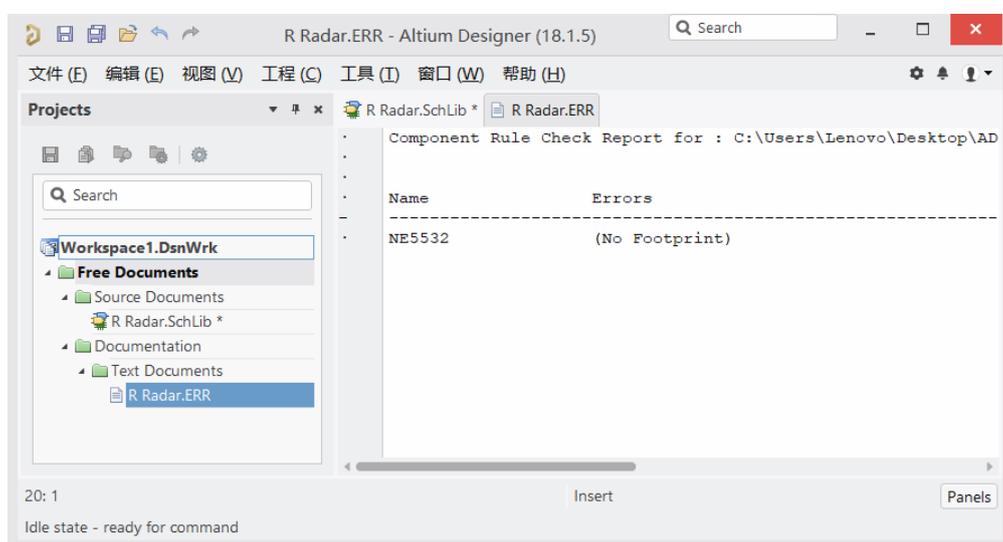


图 5-39 器件规则检查报表

根据所生成的器件规则检查报表，用户可以对相应的库元件进一步加以编辑、修改和完善。

【例 5-8】 生成库列表

(1) 打开原理图库 R Radar.SchLib。

(2) 执行 **【报告】|【库列表】** 命令，系统自动生成了 2 个描述库中所有元件信息的文件，扩展名分别为.csv 和.rep，如图 5-40 所示。

库列表是扩展名为.rep 的文本文件，列出了当前原理图库 R Radar.SchLib 中的元件数量、名称及相关的描述信息。

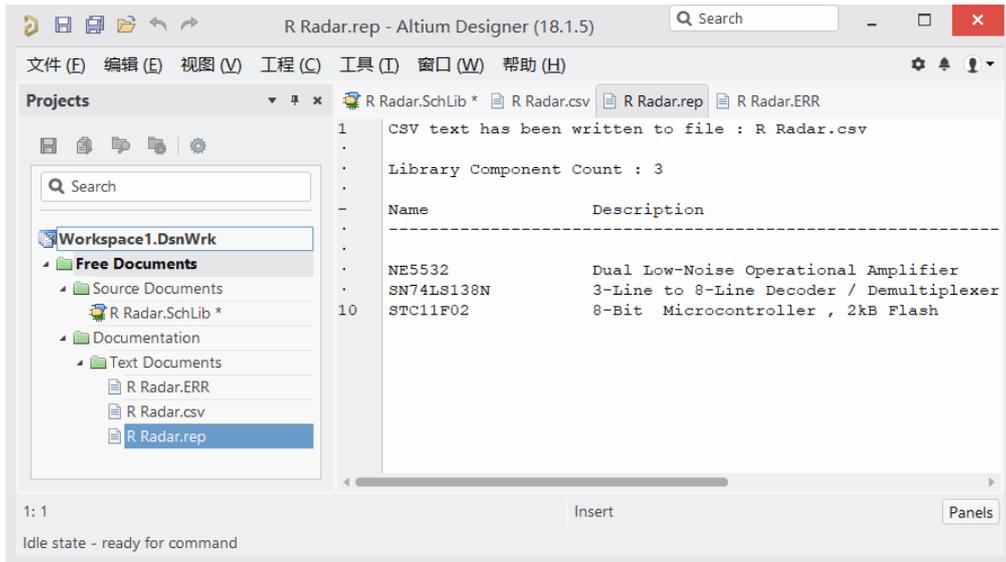


图 5-40 库列表

库列表列出库元件的数量时，是把一个库元件的别名作为另外一个库元件来进行统计的，用户应注意区分。

5.5.2 生成库报告

除了生成各种报表，Altium Designer 系统还可以快速便捷地生成综合的元件库报告，用来描述特定库中所有器件的详细信息。

报告可选择生成为 Word 格式或者 HTML 格式，包含了综合的器件参数、引脚和模型信息、原理图符号预览以及 PCB 封装和 3D 模型等，实现了对元件重要参数的完整多功能管理。HTML 格式的报告还可以提供库中所有元件的超级链接列表，便于通过网络进行发布。

【例 5-9】生成库报告

(1) 打开原理图库 R Radar.SchLib。

(2) 执行【报告】|【库报告】命令，系统弹出如图 5-41 所示的【库报告设置】对话框。

该对话框用于对库报告的格式进行设置，是【文档类型】(Word 格式)还是【浏览器类型】(HTML 格式)，并可选择设置报告里所包含的内容，有【器件参数】、【器件管脚】、【器件模型】等。

这里，我们选择了生成浏览器类型的库报告，并选中【打开生成的报告】、【添加生成的报告到当前工程】复选框，如图 5-41 所示。

(3) 单击 按钮，关闭对话框，即生成了 HTML 格式的库报告。

库报告提供了库文件 R Radar.SchLib 中所有元件的超级链接列表。单击列表中的任一项，即可链接到相应元件的详细信息处，供用户查看浏览。例如：单击 NE5532，显示的

信息如图 5-42 所示。



图 5-41 【库报告设置】对话框

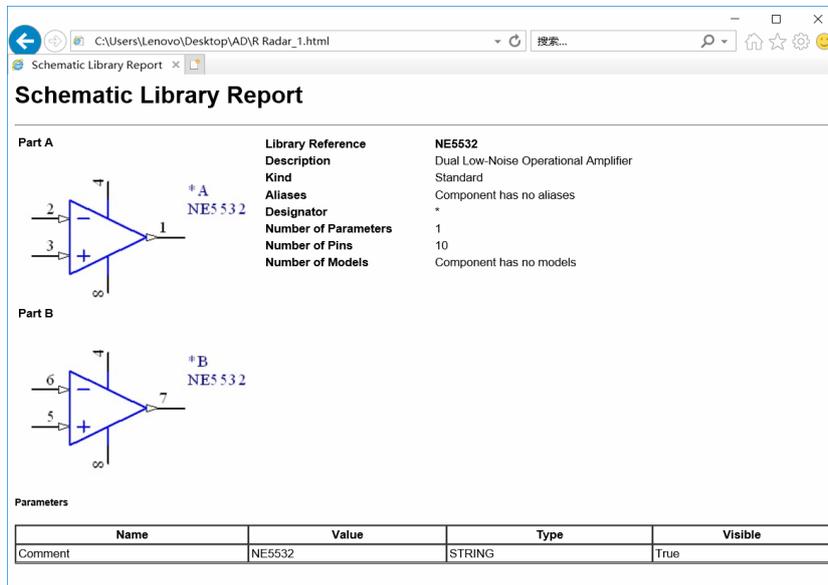


图 5-42 链接到元件的详细信息

5.6 思考与练习

1. 概念题

(1) 了解 Altium Designer 的原理图库文件编辑环境，并简述其主要组成。

- (2) 简述创建库元件的具体操作步骤。
- (3) 对于用户自行创建的库元件，其模型的来源主要有哪几种途径？

2. 操作题

- (1) 查阅相关资料，创建一个库元件——RC 滤波器芯片 LT1568。
- (2) 绘制如图 5-43 所示的电路原理图，并生成相应的原理图库及库报告。

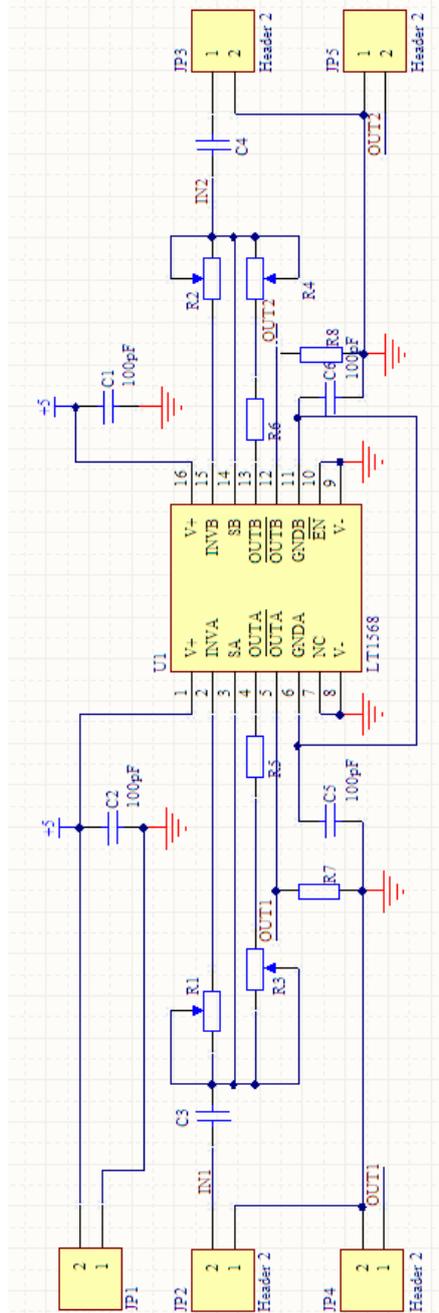


图 5-43 LT1568 电路