

# KiCad 入门

The KiCad Team

# Table of Contents

KiCad 10.0 简介 .....	2
下载和安装 KiCad .....	2
支持 .....	2
Basic concepts and workflow .....	3
PCB design workflow .....	5
First time setup .....	6
教程第一部分：工程 .....	10
Tutorial part 2: schematic .....	12
Schematic editor basics .....	12
Schematic sheet setup .....	13
Adding symbols to the schematic .....	13
Selecting and moving objects .....	15
Wiring the schematic .....	16
Annotation, symbol properties, and footprints .....	17
Electrical rules check .....	19
Bill of materials .....	21
Tutorial part 3: circuit board .....	23
PCB editor basics .....	23
Board setup and stackup .....	23
Importing changes from schematic .....	26
Drawing a board outline .....	28
Placing footprints .....	30
Routing tracks .....	32
Placing copper zones .....	34
Design rule checking .....	36
3D viewer .....	38
Fabrication outputs .....	39
Tutorial part 4: custom symbols and footprints .....	42
Library and library table basics .....	42
Creating new global or project libraries .....	43
Creating new symbols .....	43
Creating new footprints .....	46
Linking symbols, footprints, and 3D models .....	54
Where to go from here .....	57
More learning resources .....	57
Help improve KiCad .....	57

## Copyright

This document is Copyright The KiCad Documentation Contributors. You may distribute it and/or modify it under the terms of either the GNU General Public License (<https://www.gnu.org/licenses/gpl.html>), version 3 or later, or the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/3.0/>), version 3.0 or later.

本指南中的所有商标均属于其合法所有者。

### Contributors

Graham Keeth, Jon Evans, Glenn Peterson.

### 翻译人员

taotieren <[admin@taotieren.com](mailto:admin@taotieren.com)>, 2019-2025.

### Former Contributors

David Jahshan, Phil Hutchinson, Fabrizio Tappero, Christina Jarron, Melroy van den Berg.

### Feedback

KiCad 项目欢迎与软件或其文档相关的反馈、错误报告和建议。有关如何提交反馈或报告问题的更多信息，请参阅 <https://www.kicad.org/help/report-an-issue/> 上的说明

### Software and Documentation Version

本用户手册基于 KiCad 10.0.3 版本编写。其他版本的 KiCad 在功能和外观上可能有所不同。

文档修订版本：`ff59d8ec`。

# KiCad 10.0 简介

KiCad 是一个开源的软件套件，用于创建电子电路原理图、印刷电路板（PCB）和相关的零件描述。KiCad 支持综合设计工作流程，其中原理图和相应的 PCB 一起设计，也支持特殊用途的独立工作流程。KiCad 还包括一些帮助电路和 PCB 设计的实用程序，包括用于确定电路结构电气特性的 PCB 计算器、用于检查制造文件的 Gerber 查看器、用于可视化成品 PCB 的 3D 查看器以及用于检查电路行为的集成 SPICE 仿真器。

KiCad 可在所有主要的操作系统和广泛的计算机硬件上运行。它支持多达 32 个铜层的 PCB，适合创建各种复杂的设计。KiCad 是由世界各地的软件和电气工程师组成的志愿者团队开发的，其使命是创建适合专业设计师的免费和开源的电子设计软件。

The latest documentation for KiCad is available at <https://docs.kicad.org>.

## 下载和安装 KiCad

KiCad 可在许多操作系统上运行，包括微软 Windows、苹果 macOS 和许多主要的 Linux 发行版。

You can find the most up to date download links and installation instructions at <https://www.kicad.org/download/>. These instructions are not included in this manual as they may change over time with the release of operating system updates.

If you run into problems running KiCad on your system, check the list of known system-related issues and workarounds at <https://www.kicad.org/help/known-system-related-issues/>.

### IMPORTANT

根据 [KiCad 稳定发布政策](#)，KiCad 的稳定发布会定期进行。新的功能正在不断地被添加到开发分支中。如果你想利用这些新功能，并通过测试它们来提供帮助，请下载你的平台的最新夜间构建包。夜间构建可能会引入一些错误，如文件损坏、生成不良 Gerbers 等，但 KiCad 开发团队的目标是在新功能开发期间尽可能保持开发分支的可用性。

## 支持

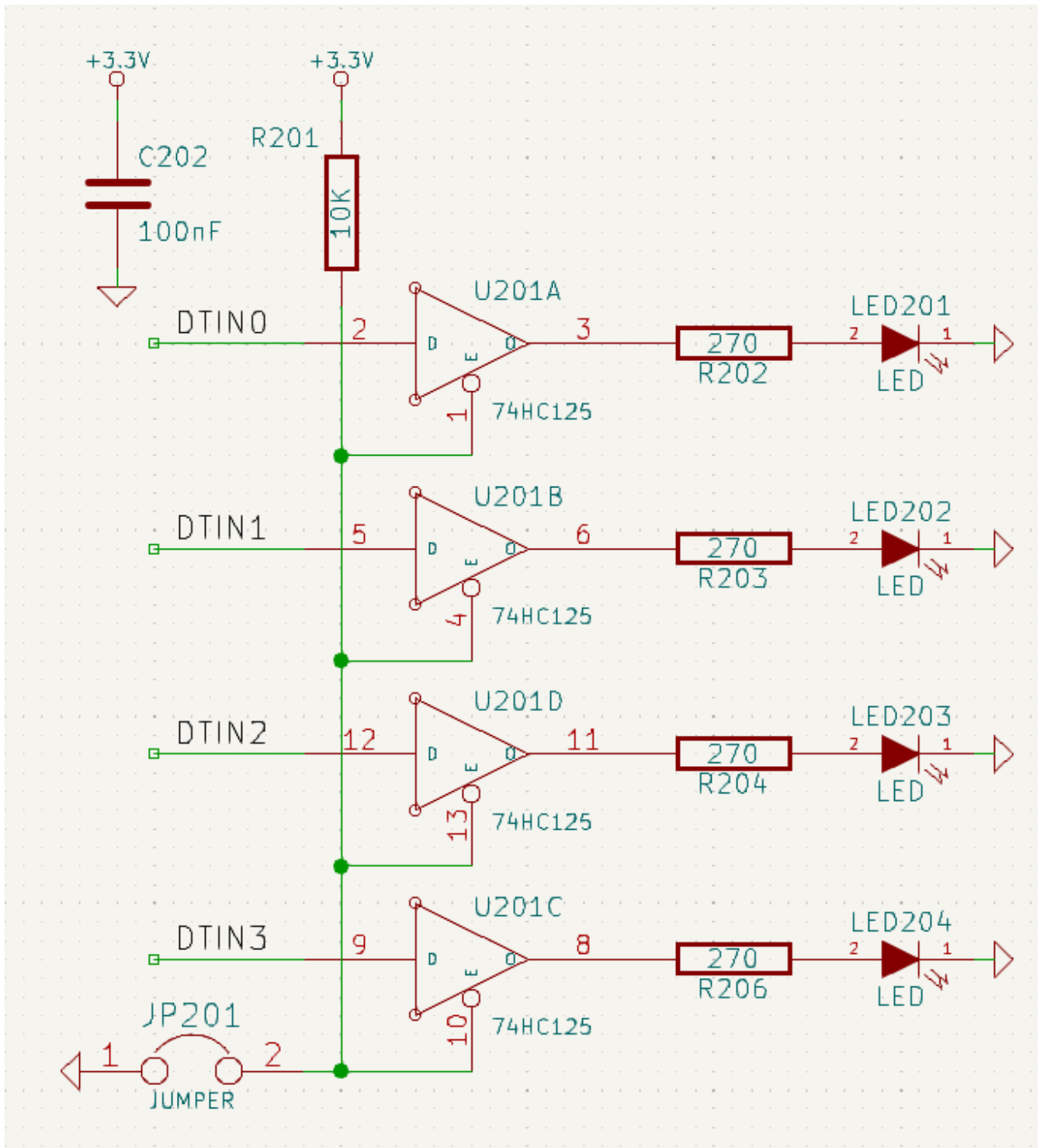
如果您有想法，评论或问题，或者您只是需要帮助：

- 这是 [KiCad 官方用户论坛](#) 是一个与其他 KiCad 用户联系并获得帮助的好地方。
- 加入我们在 [Discord](#) 或 [IRC](#) 的社区，与用户和开发者进行实时讨论。
- 查看 [KiCad 网站](#)，了解 KiCad 社区制作的 [学习资源](#)。

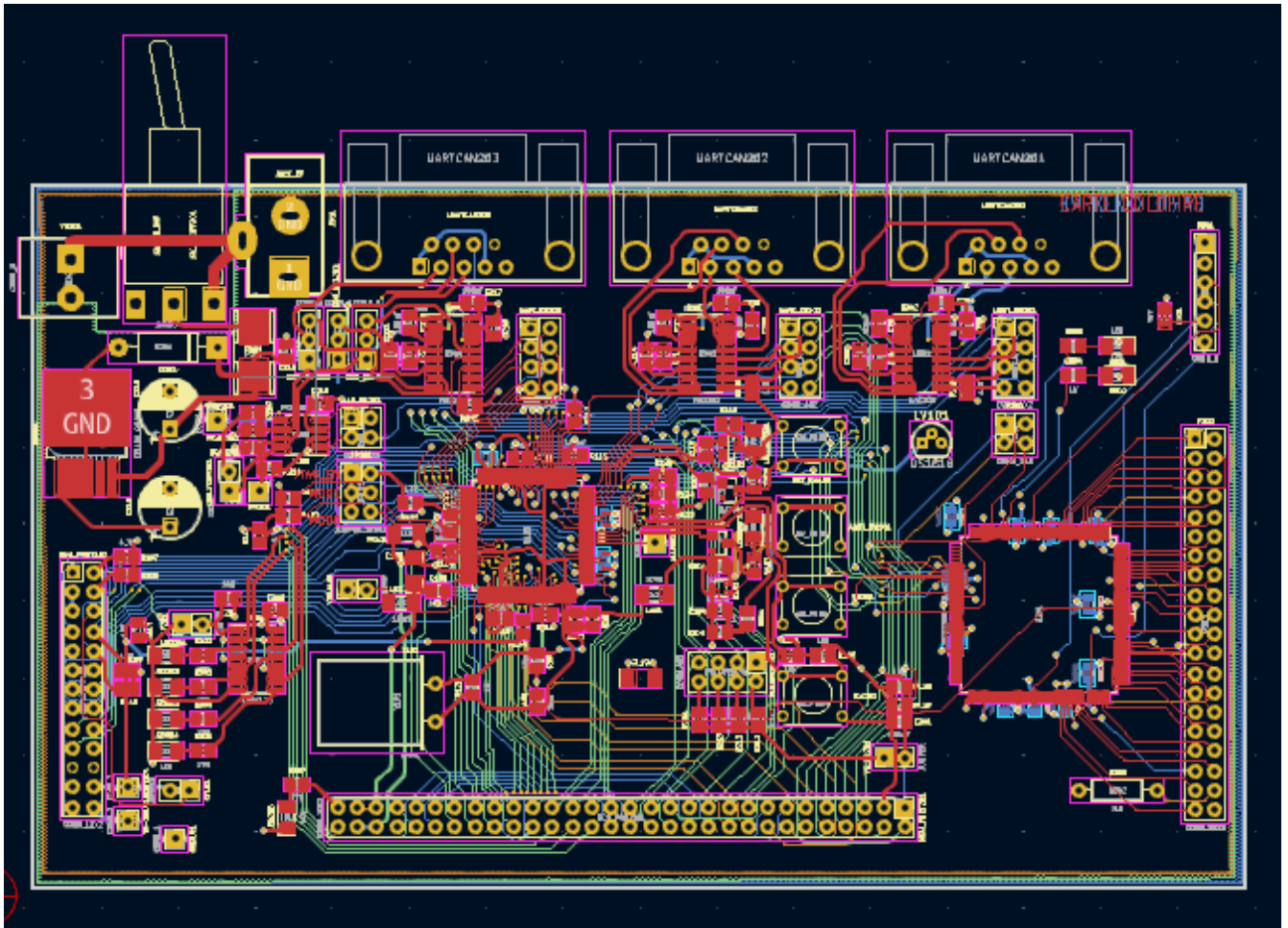
# Basic concepts and workflow

KiCad 的典型工作流程包括两个主要任务：绘制原理图和布局电路板。

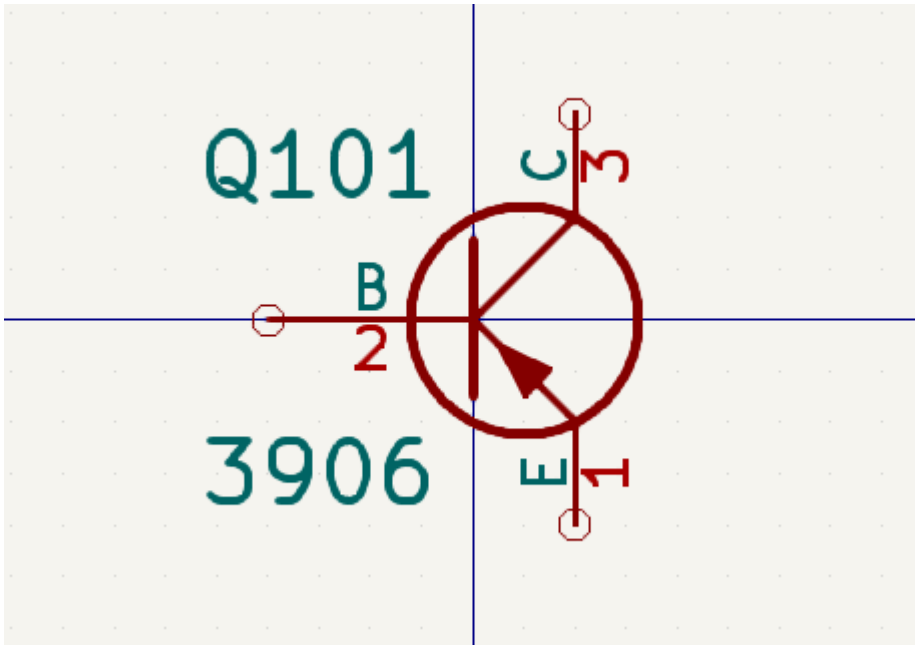
原理图是电路的符号表示：使用了哪些元件，它们之间有哪些连接。原理图符号是电子元件在原理图中的象形表示，如一个电阻的人字形或矩形，或一个运算放大器的三角形。原理图包含了设计中每个元件的符号，符号中的线连接着引脚。原理图通常是先画出来，然后再布线电路板。

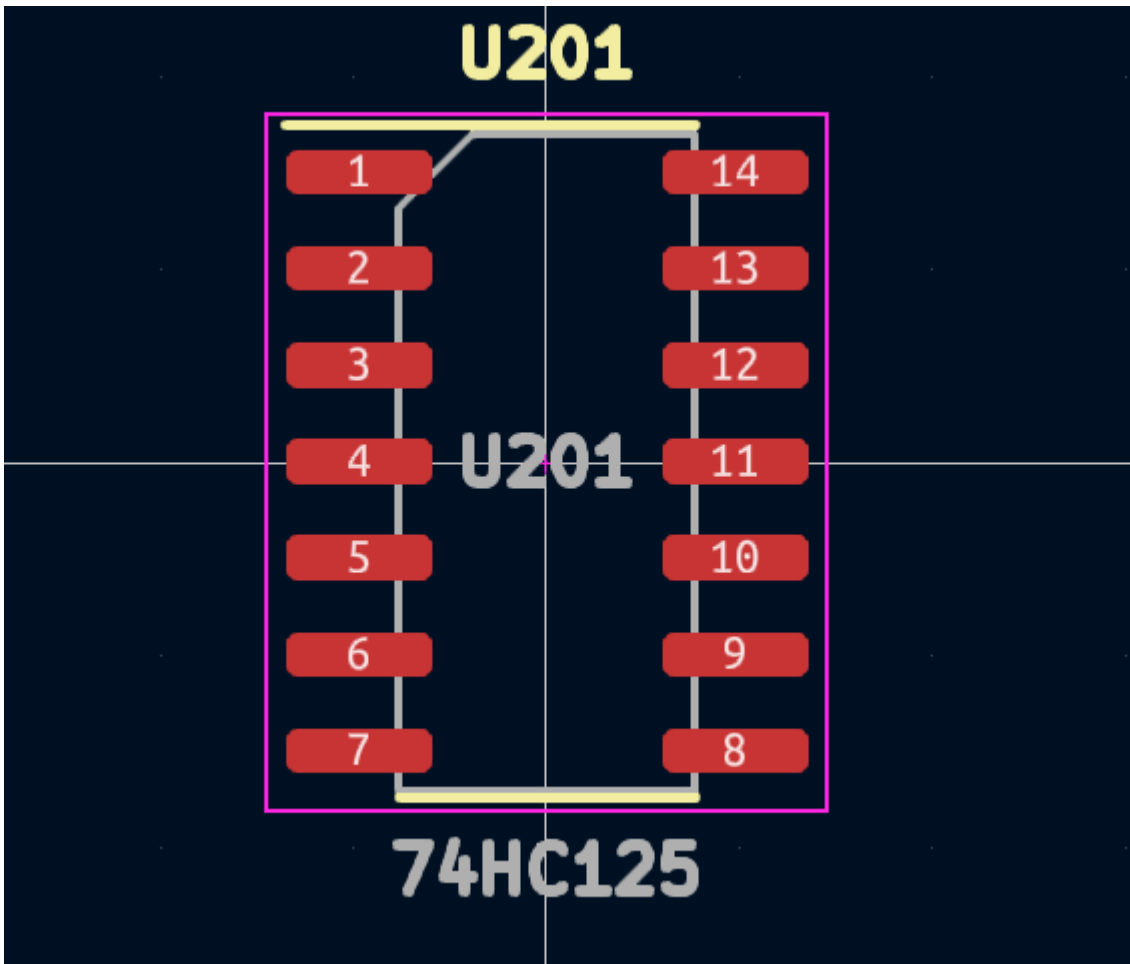


板子是原理图的物理实现，元件的封装被放置在板子上，铜线实现原理图中描述的连接。封装是一组铜焊盘，与物理元件上的引脚相匹配。当电路板被制造和组装时，该元件将被焊接到电路板上相应的封装上。



KiCad 有单独的窗口用于绘制原理图（“原理图编辑器”）、布局电路板（“PCB 编辑器”）以及编辑符号和封装（“符号编辑器”和“封装编辑器”）。KiCad 自带了大量高质量的、由用户提供的符号和封装库，但创建新的符号和封装或修改现有符号和封装也很简单。





最后，重要的是要了解 KiCad 有一个基于工程的工作流程。KiCad 工程是一个包含工程文件、原理图、电路板布局的文件夹，也可以选择其他相关文件，如符号和封装库、仿真数据、采购信息等。许多与项目相关的设置，包括网络类和设计规则，都存储在工程层面。在其相关工程之外打开一个电路板可能会导致设计信息的丢失，所以要确保将所有与工程相关的文件放在一起。

## PCB design workflow

通常情况下，首先绘制原理图。这意味着在原理图上添加符号，并绘制它们之间的连接。如果没有合适的符号，可能需要创建自定义符号。在这个阶段，还要为每个部件选择封装，必要时还要创建自定义封装。当原理图完成后，设计通过了电气规则检查（ERC），原理图中的设计信息被转移到电路板编辑器中，并开始布局。

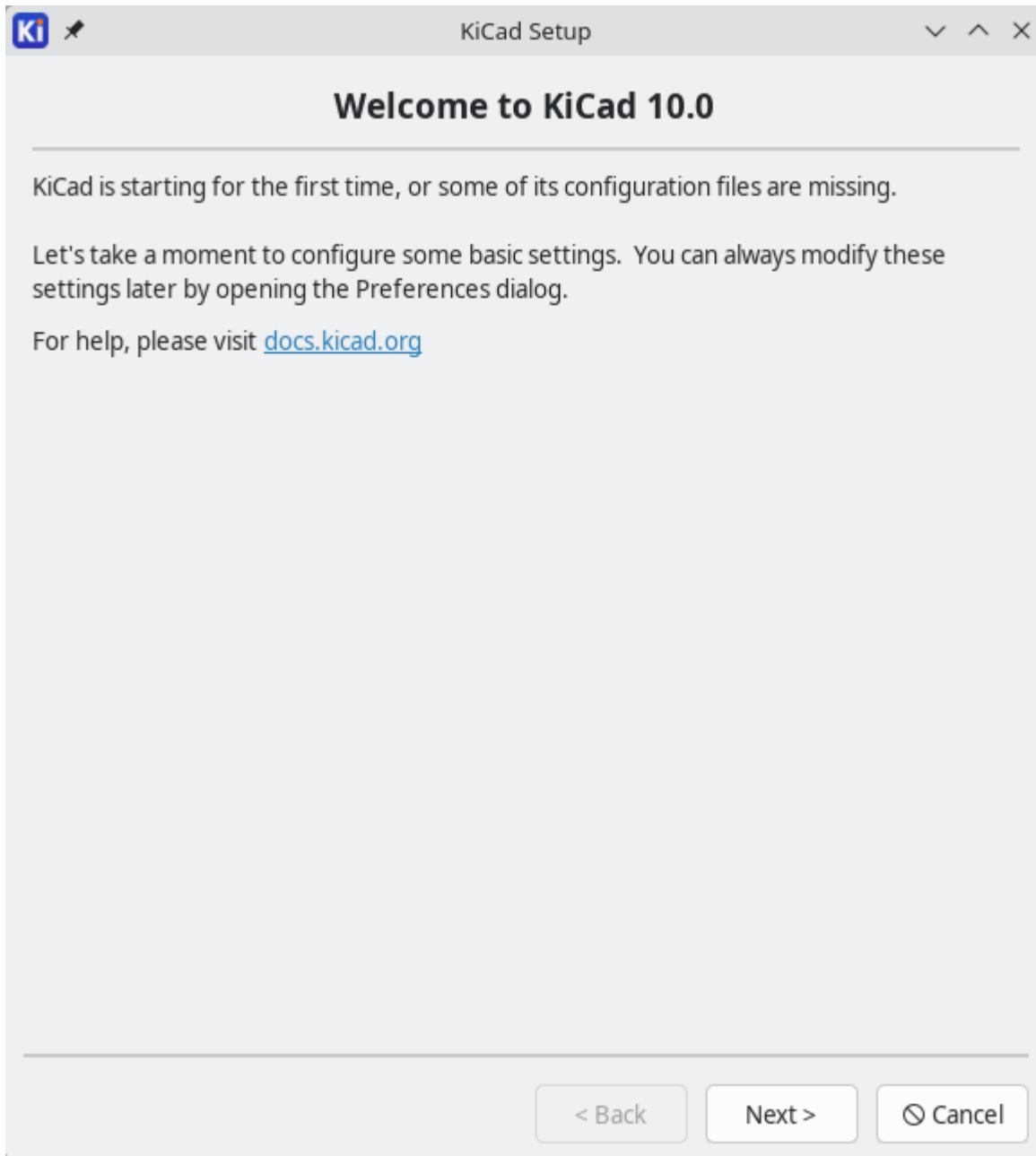
该原理图描述了设计中包含哪些元件以及它们之间的连接方式；电路板编辑器利用这些信息来简化布局过程，并防止原理图与 PCB 之间出现不匹配。布局过程需要仔细地将每个元件封装放置在电路板上。在元件放置完成后，根据原理图中的连接关系以及其他电气考虑因素（如走线电阻、受控阻抗要求、串扰等），在元件之间绘制铜走线。

通常情况下，原理图在布局开始后需要更新；原理图的变化可以很容易地拉到电路板设计中。相反的情况也经常发生：在电路板布局中的任何设计变化都可以被推回到原理图中，以保持两者的一致性。

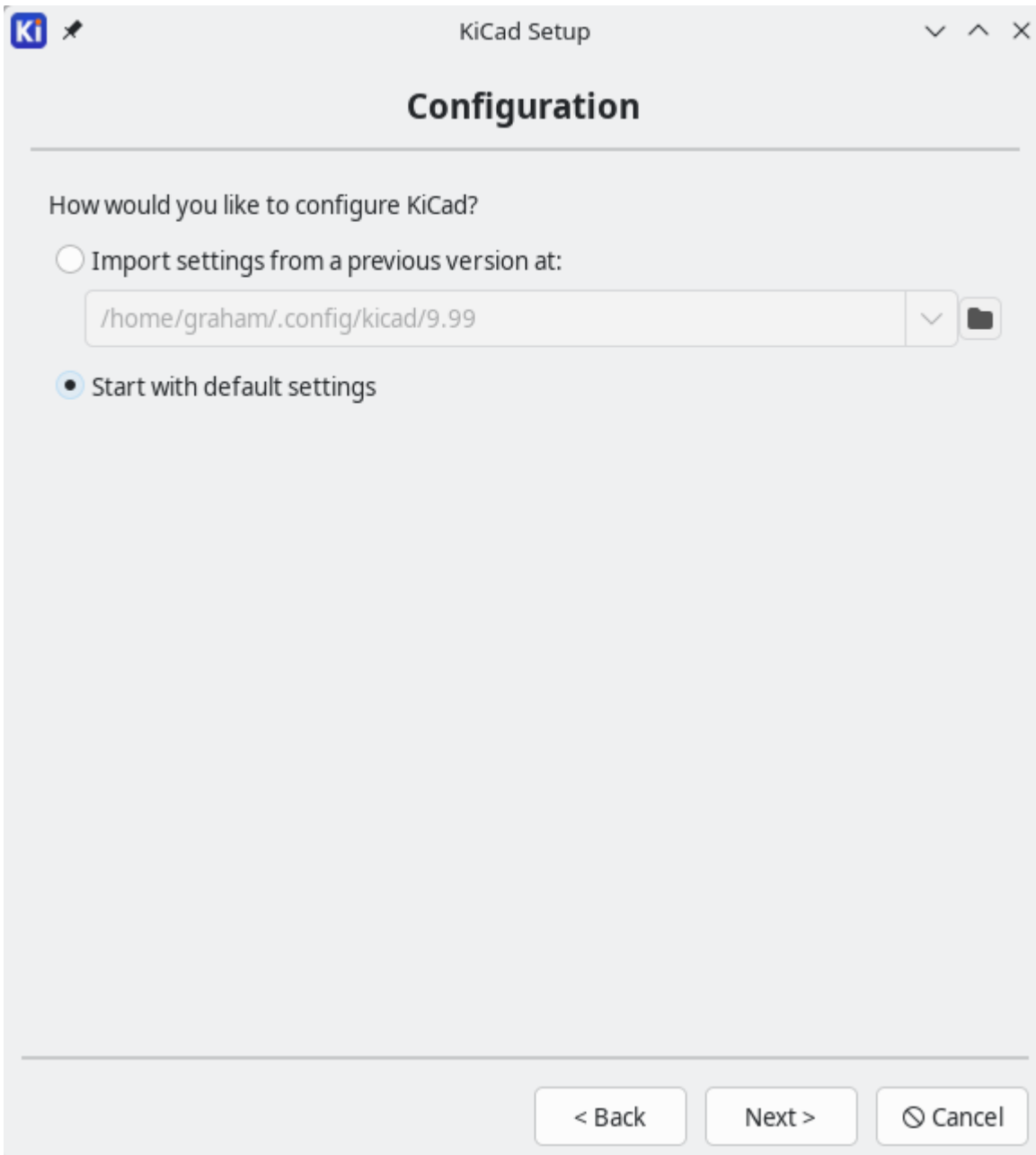
当电路板布局完成，并且电路板通过了设计规则检查（DRC），就会产生制造输出，这样电路板就可以由 PCB 制造商制造。

# First time setup

The first time you open KiCad, a Welcome window appears to guide you through the initial setup.



The next page asks you if you want to start with fresh preferences or if you want to import your settings from an older version of KiCad.



The next page asks you how you want to configure your symbol and footprint libraries. If you are setting up KiCad for the very first time, you should select the first option, **Start with the built-in KiCad libraries**. This initializes your library tables to include all of the libraries that come with KiCad.

If you have used an older version of KiCad and are updating it to the newest version, and therefore already have a library setup that worked in the old version, you can instead select the second option, **Import tables from the previous version**. In this case, it is recommended to also select the option **Migrate built-in libraries to the latest version**, which replaces references to the old-version built-in libraries with references to the updated libraries.

You also have the option to start with no libraries and do all of the configuration yourself. Refer to the [Schematic Editor](#) and [PCB Editor](#) documentation for details on how to manually configure libraries.

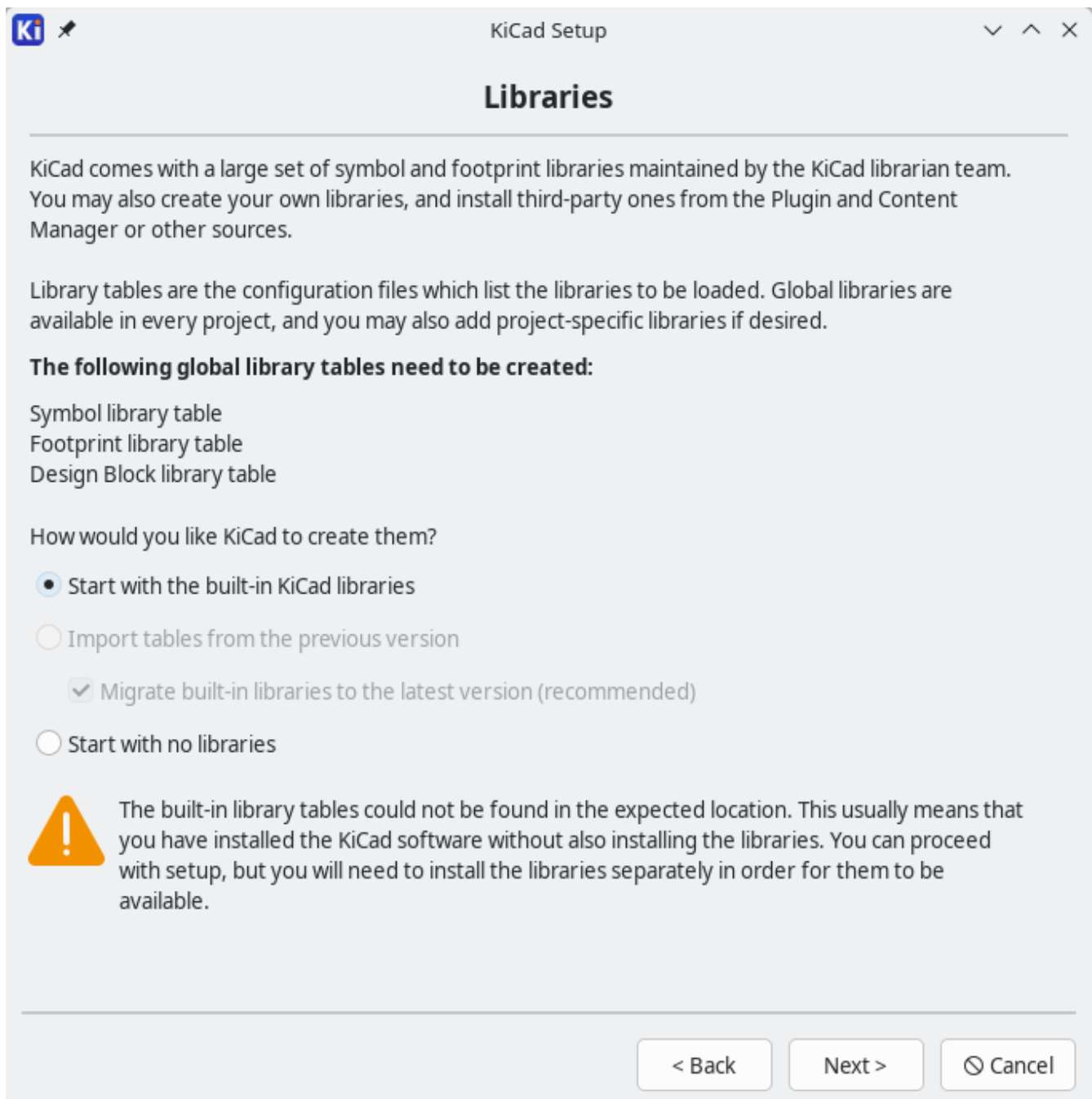
Depending on your system and how KiCad was installed, some options may not be available. KiCad may also not be able to locate any libraries installed on your system. In this case, you should ensure you have installed the libraries, or you can manually configure the libraries after completing setup.

**NOTE**

On some systems the KiCad libraries are installed as a separate package.

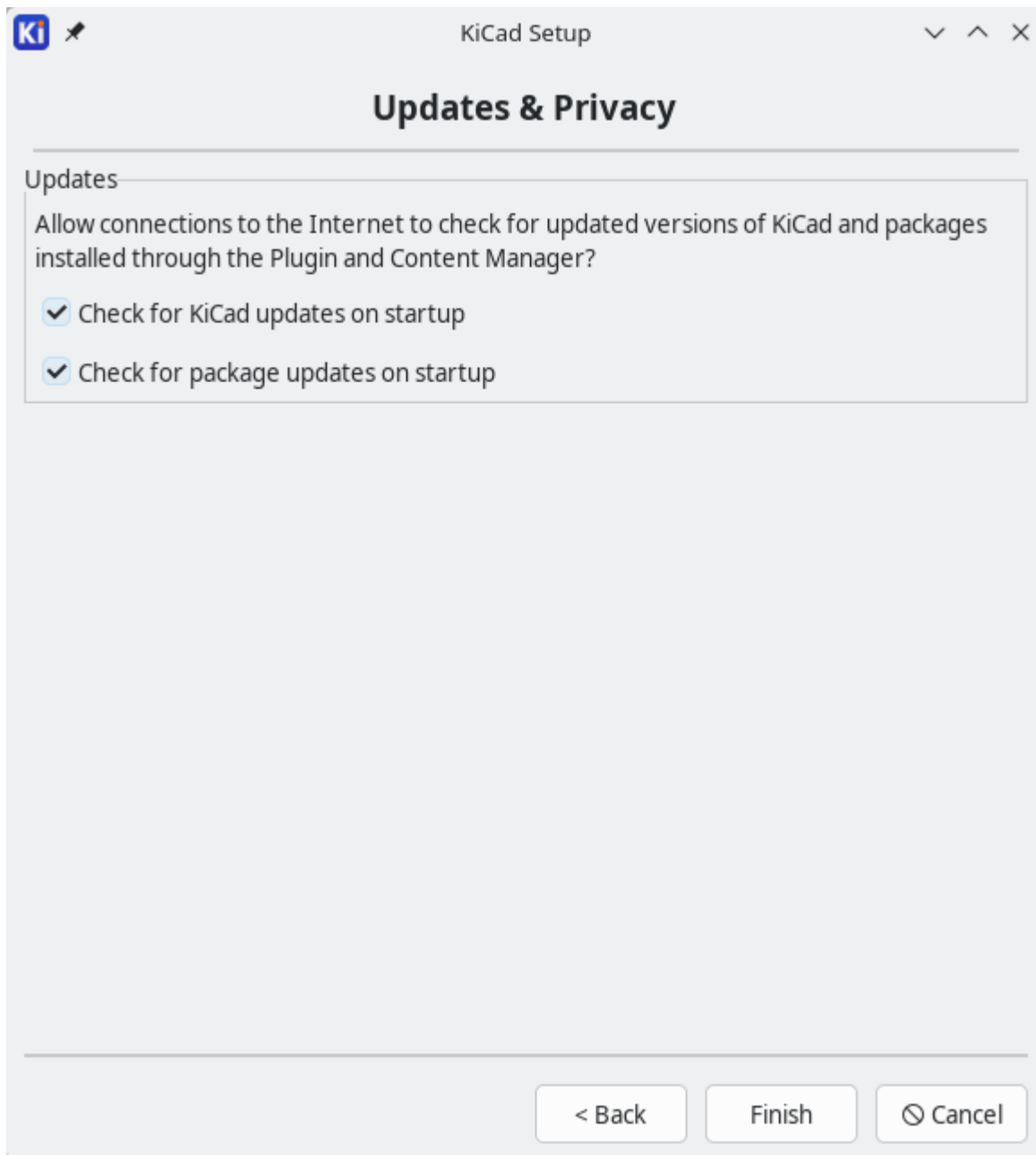
默认库表文件的位置取决于操作系统，并可能根据安装位置而有所不同。下面是每个操作系统的默认值：

Windows	C:\Program Files\KiCad\10.0\share\kicad\template\
Linux	/usr/share/kicad/template/
macOS	/Applications/KiCad/KiCad.app/Contents/SharedSupport/template/



The last page asks you whether you want to automatically check for updates to the KiCad application and to third-party packages you may have installed through KiCad's Plugin and Content Manager (plugins, themes, libraries, etc.).

On Windows, you also have the option to enable anonymous crash reporting, which helps the KiCad team diagnose and fix critical issues in the application.

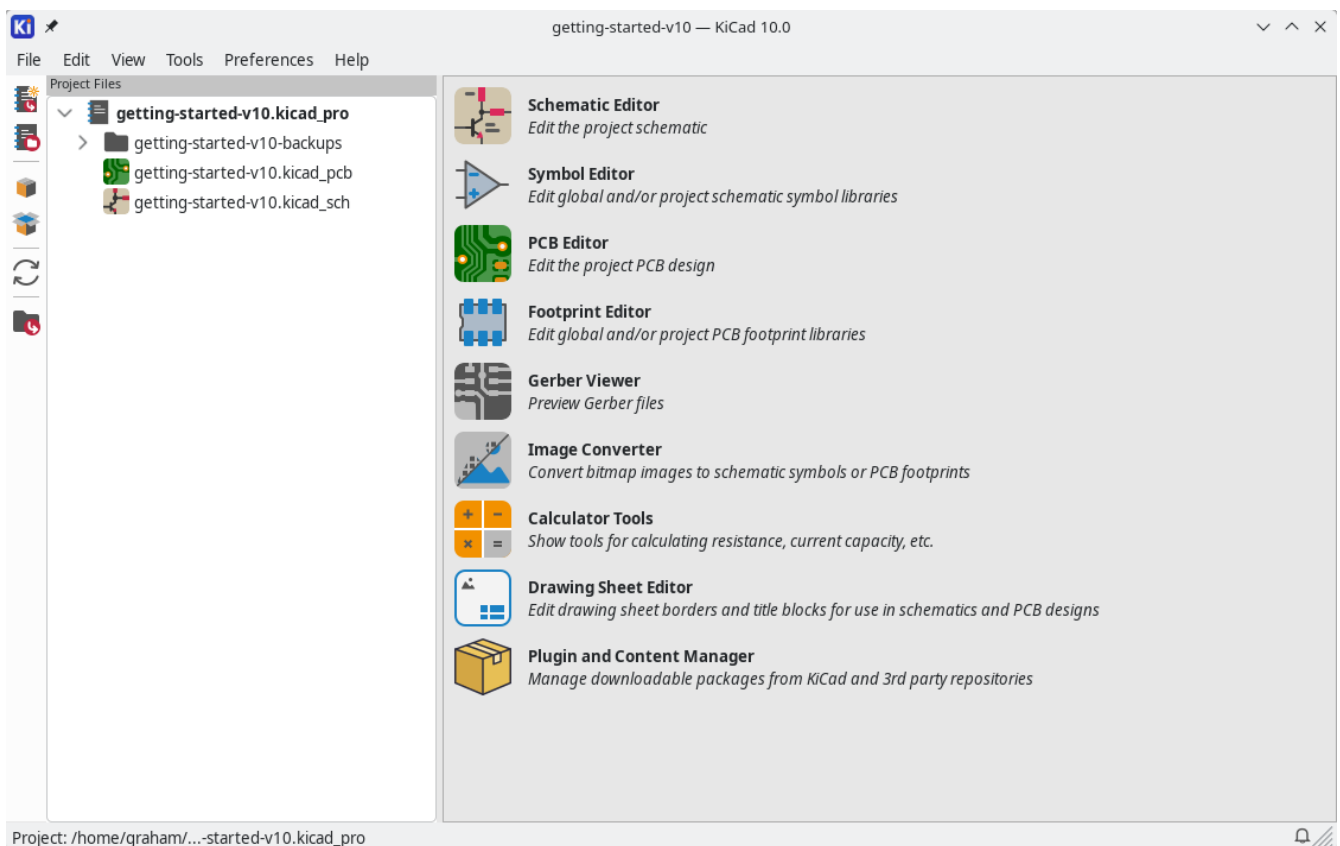


# 教程第一部分：工程

Opening KiCad will bring up the Project Manager. To create a new project:

1. Click **File** → **New Project**
2. Select the **Default** project template and click **OK**.
3. Browse to the location where you want to save your project, and give your project a name, such as `getting-started`.
4. Make sure the **Create a new folder for the project** checkbox is ticked.
5. Click **Save**.

This will create your project files in a new subfolder with the same name as your project.



在左边，工程文件窗格列出了新工程中的文件。有一个扩展名为 `.kicad_pro` 的工程文件，一个扩展名为 `.kicad_sch` 的原理图文件，以及一个扩展名为 `.kicad_pcb` 的电路板文件。这些文件都与你的工程共享一个名称。

There may also be a `-backups` directory: KiCad can automatically create backups of your project when you save and when there are unsaved changes after several seconds of inactivity. The backup settings are configurable by going to **Preferences** → **Preferences** → **Common** → **Project Backup**.

**NOTE** 打开偏好设置对话框可能会触发库表配置对话框。该对话框的说明 [下文章节](#)。

**NOTE** 在 macOS 上，偏好设置对话框位于 **KiCad** 菜单中，而非 **偏好设置** 菜单。

在工程窗口的右侧，有一些按钮可以启动 KiCad 提供的各种工具。启动这些工具将自动打开当前项目中的相关设计文件（原理图或 PCB）。首先打开原理图编辑器。

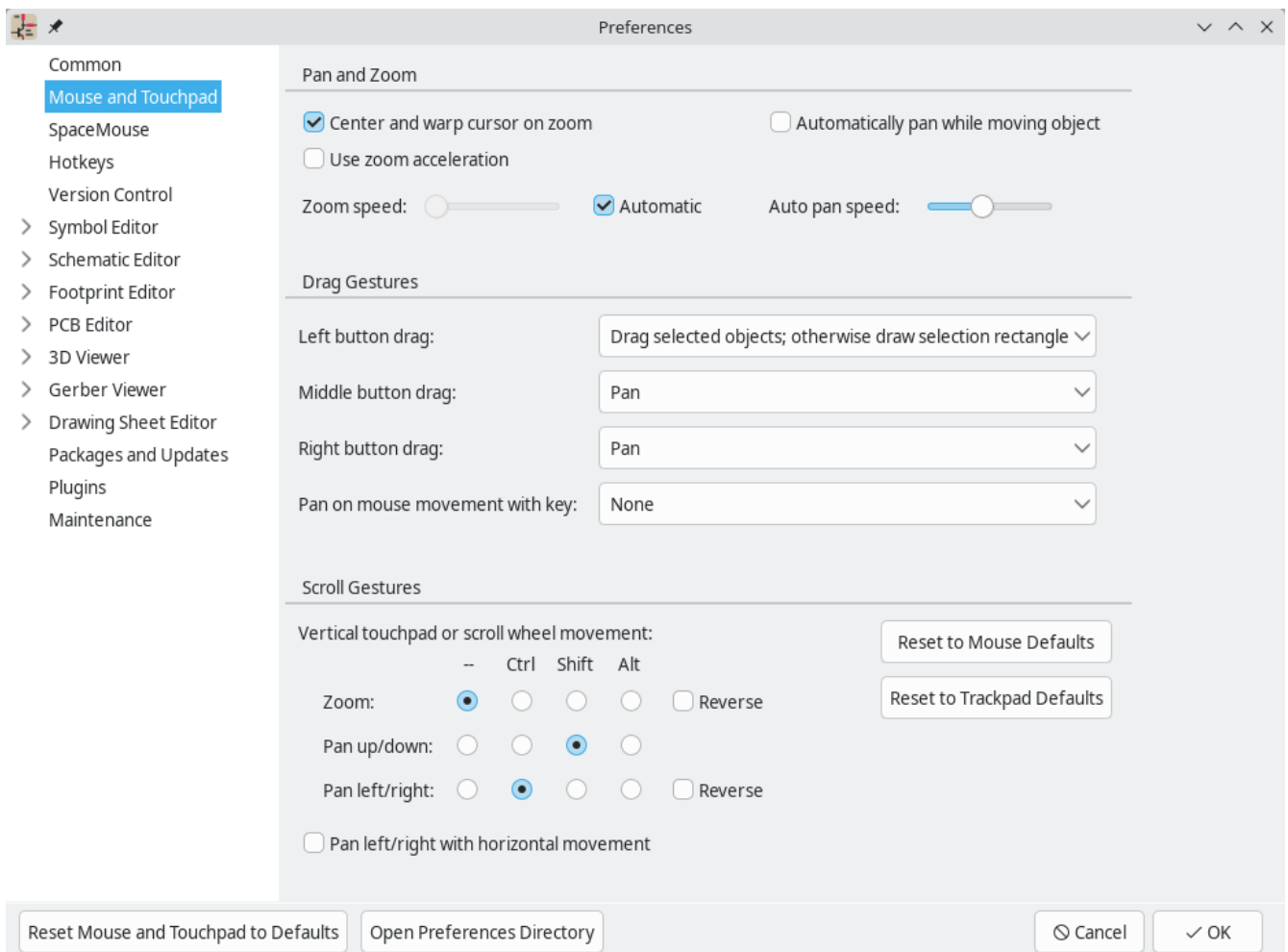
# Tutorial part 2: schematic

## Schematic editor basics

要在原理图上移动，用鼠标中键或鼠标右键点击并拖动。用鼠标滚轮或 **F1** 和 **F2** 进行放大和缩小。笔记本电脑用户可能会发现，改变鼠标控制，使其更适合于触摸板；鼠标控制可在 **偏好设置** → **偏好设置...** → **鼠标和触摸板** 中配置。


**NOTE** 在 macOS 上，偏好设置对话框位于 **KiCad** 菜单中，而非 **偏好设置** 菜单。

默认情况下，KiCad 启用了名为 **缩放时的中心和经纬光标的鼠标设置**。启用该功能后，当用户放大或缩小时，鼠标光标会自动移动到屏幕中央。这使缩放区域始终保持在中心位置。这个功能是不寻常的，但许多用户一旦习惯了它就会发现它很有用。试着用鼠标光标在画布的不同区域进行放大和缩小。如果默认的缩放行为让人不舒服，请在鼠标和触摸板偏好设置中禁用该功能。



原理图编辑器屏幕左侧的工具栏包含基本的显示设置。屏幕右侧的工具栏包含用于编辑原理图的工具。

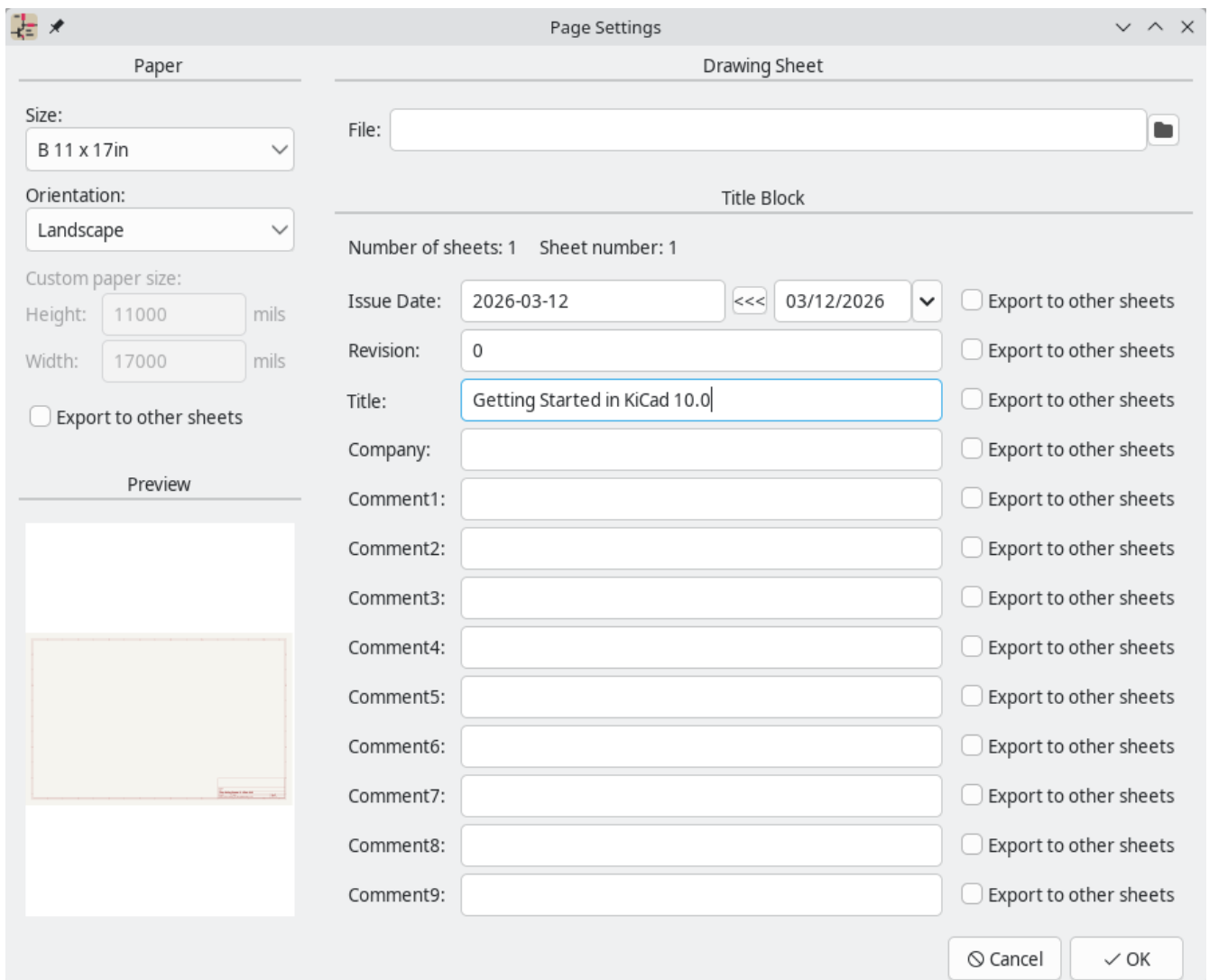
## NOTE

Some toolbar buttons have a small triangle in the bottom right corner . The triangle indicates that the button has an expanding palette containing several related tools, for example different kinds of labels. To select an alternate tool, click and hold on the button until the palette appears, then click on the alternate tool. Another way to use the palettes is to click on the button and drag to the left until the palette appears, then release the mouse button when the desired tool is highlighted.


KiCad 中的大多数工具要么已分配默认快捷键，要么可以自定义快捷键。要查看所有快捷键，请前往 **帮助** → **列出快捷键...**。快捷键可以在偏好设置对话框的 **快捷键** 面板中进行更改。

## Schematic sheet setup

在绘制原理图中的任何内容之前，先对原理图页面本身进行设置。点击 **文件** → **页面设置**。给原理图一个标题和日期，如果需要的话，改变原理图大小。

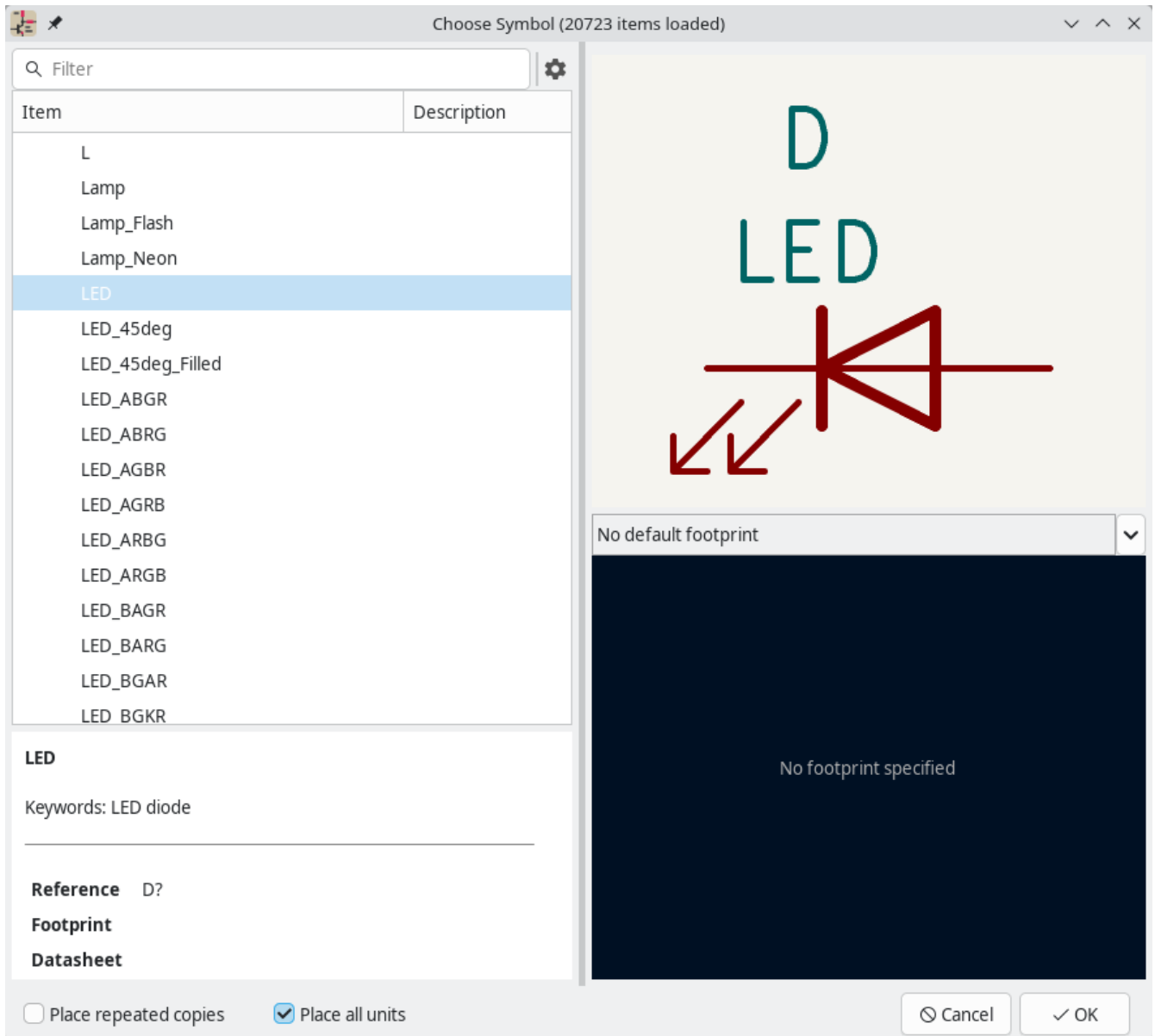


## Adding symbols to the schematic

在原理图中添加一些符号，开始制作电路。点击窗口右侧的 **添加符号** 按钮  或按 **A** 键，打开 **选择符号** 对话框。

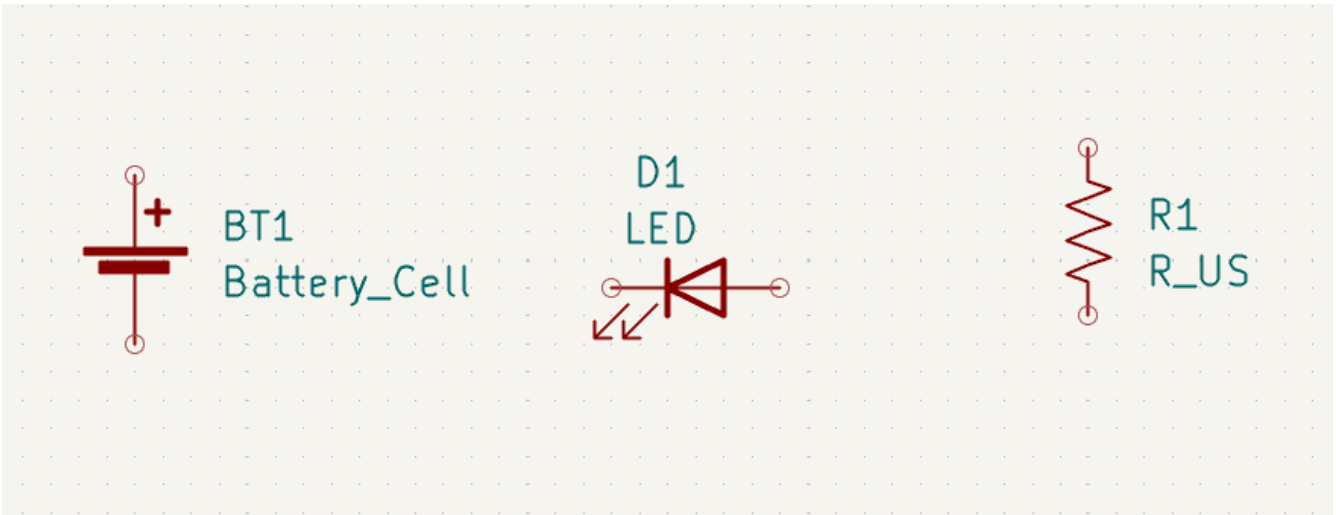
选择符号对话框列出了可用的符号库和其中包含的元件符号。基本设备如无源元件、二极管和其他通用符号在 **Device** 库中找到。特定的设备，如一个特定的 LED，可能在其他库中找到。

向下滚动到 Device 库，展开它，并选择 LED 符号。点击 确定，然后再次点击，将该符号放入原理图中。



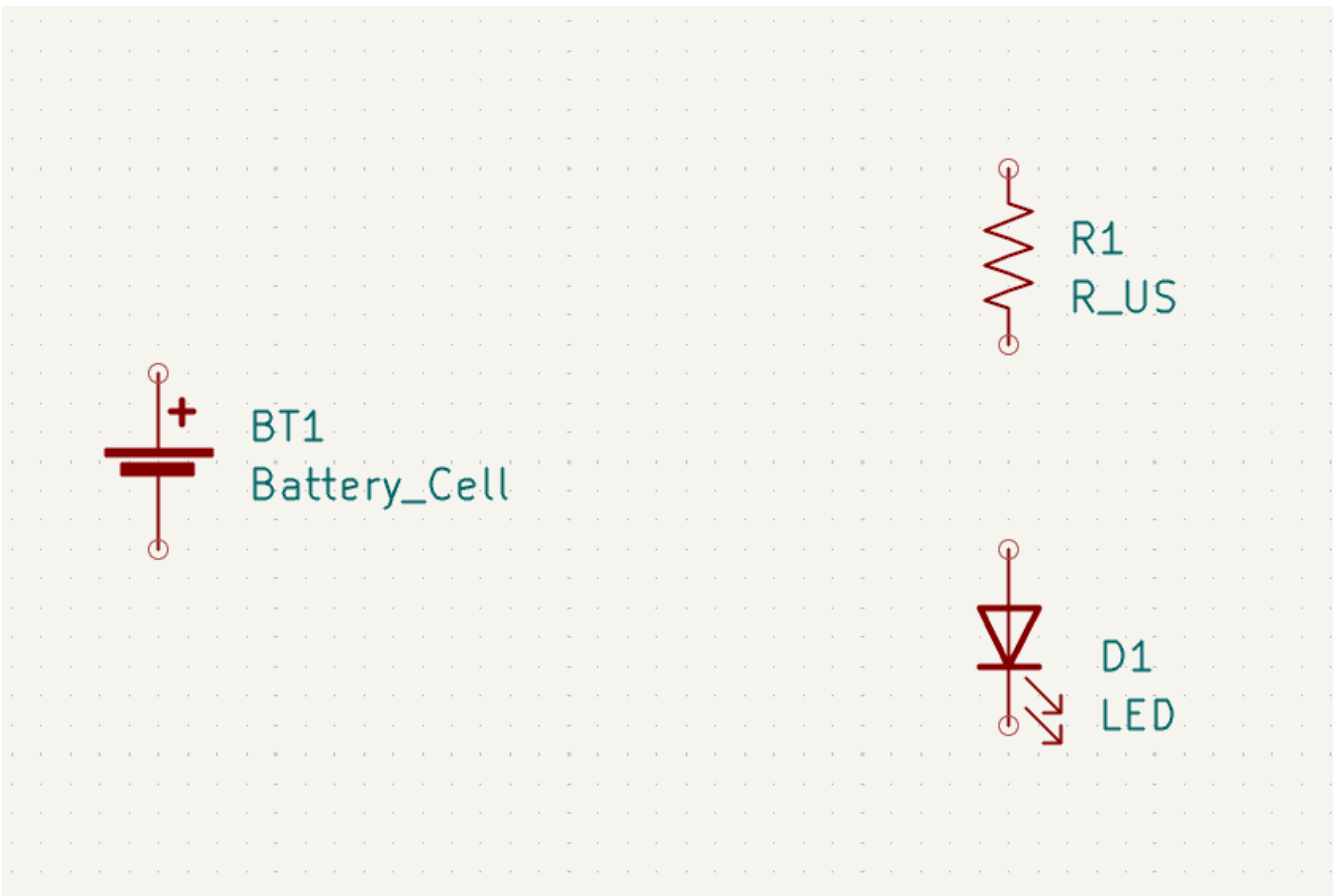
接下来，添加一个限流电阻。回到符号选择器，但这次尝试通过在顶部的过滤框中输入 R 来搜索一个电阻。同样，它可以在 Device 库中找到。R 设备是一个 IEC 风格的矩形电阻符号。对于喜欢 ANSI 风格人字形符号的用户，也有一个 R\_US 符号。选择一个电阻符号并将其添加到原理图中。


最后，添加一个电池为LED供电。Device 库有一个合适的 Battery\_Cell 符号。



## Selecting and moving objects

接下来，按照截图所示，将符号彼此间的位置正确摆放。您可以通过选择每个符号，然后移动和旋转它来完成这一步骤。



在 KiCad 中，当选择工具处于激活状态时，通过点击对象即可进行选择。选择工具在无其他工具激活时是默认工具，因此您可以通过退出任何当前激活的工具（`Esc`）或点击右侧工具栏中的  按钮来进入选择工具模式。


可以通过按下 `Shift` + 单击 来向选择中添加更多对象，或通过按下 `Ctrl` + `Shift` + 单击（macOS：`Cmd` + `Shift` + 单击）来移除对象。您可以通过按下 `Ctrl` + 单击（macOS：`Cmd` + 单击）来切换项目的选择状态。

拖动选择也是可以的；从左到右拖动可以选择完全被选择框包围的对象，而从右到左拖动也可以选择部分被选择框包围的对象。`Shift`，`Ctrl` + `Shift`（`Cmd` + `Shift`），和 `Ctrl`（`Cmd`）也可以和拖动选择一起使用，分别从选择中增加、减少或切换。

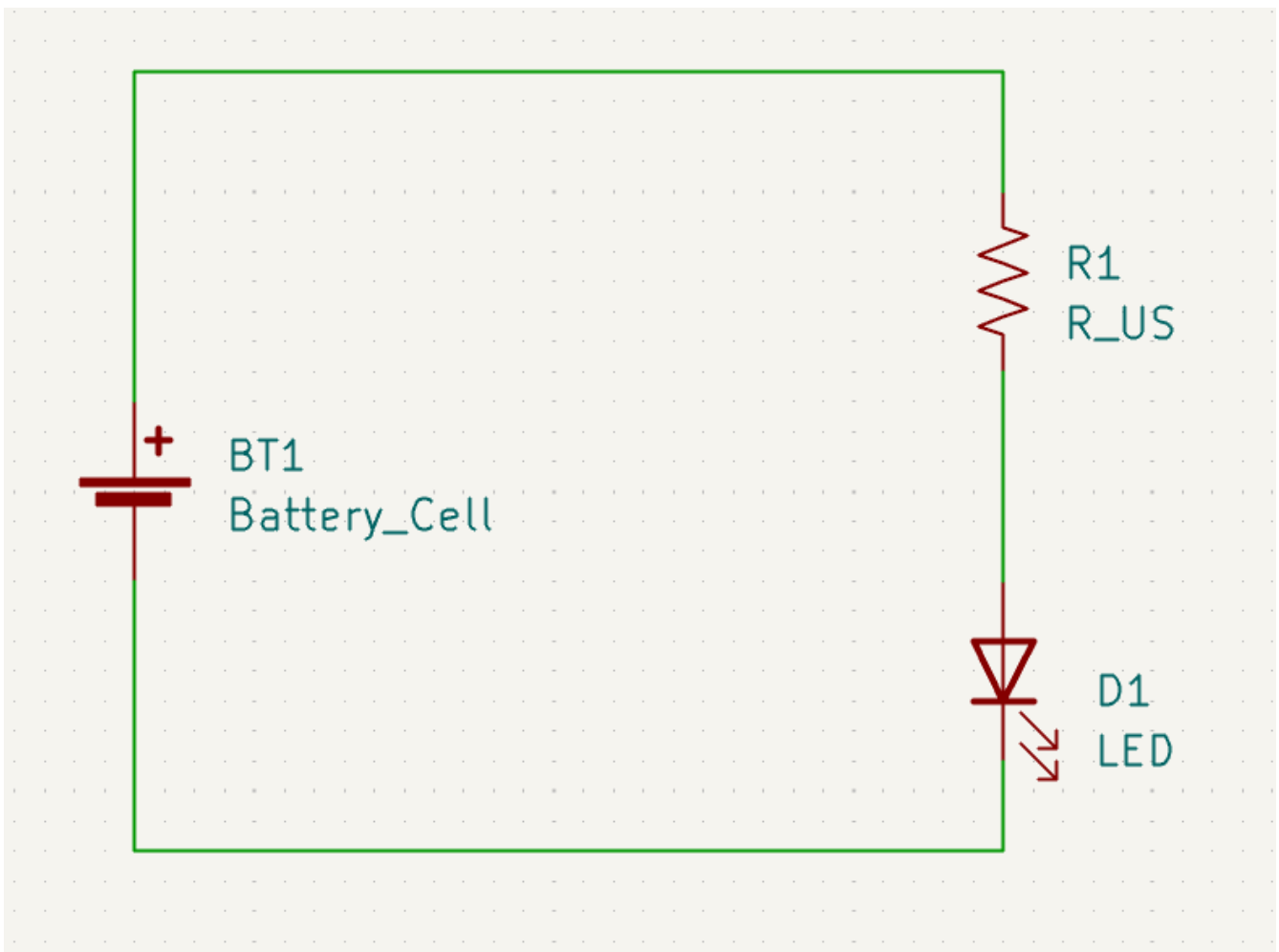
注意，可以选择整个符号（通过点击符号形状本身）或选择符号中的一个文本字段而不选择符号的其他部分（通过点击文本）。当只选择一个文本字段时，所执行的任何操作将只作用于所选择的文本，而不作用于符号的其他部分。

按 **M** 可以移动选中的对象，按 **R** 可以旋转。**G** 热快捷键（拖动）也可用于移动对象。对于移动未连接的符号，**G** 和 **M** 的行为是相同的，但对于连接有导线的符号，**G** 移动符号并保持导线连接，而 **M** 移动符号并留下导线。选定的对象可以用 **Del** 键删除。


## Wiring the schematic

符号引脚上均带有小圆圈，表示它们尚未连接。请按照截图所示，在符号引脚之间绘制导线以解决此问题。点击右侧工具栏上的 **添加导线** 按钮 ，或使用快捷键 **W**。点击并释放鼠标以开始绘制导线，通过点击符号引脚或双击任意位置完成绘制。按下 **Esc** 键将取消绘制导线。



另一种方便的画导线方法是将鼠标悬停在一个未连接的引脚上。鼠标指针会改变，表示可以从该位置开始画导线。点击该引脚就会自动开始画导线。

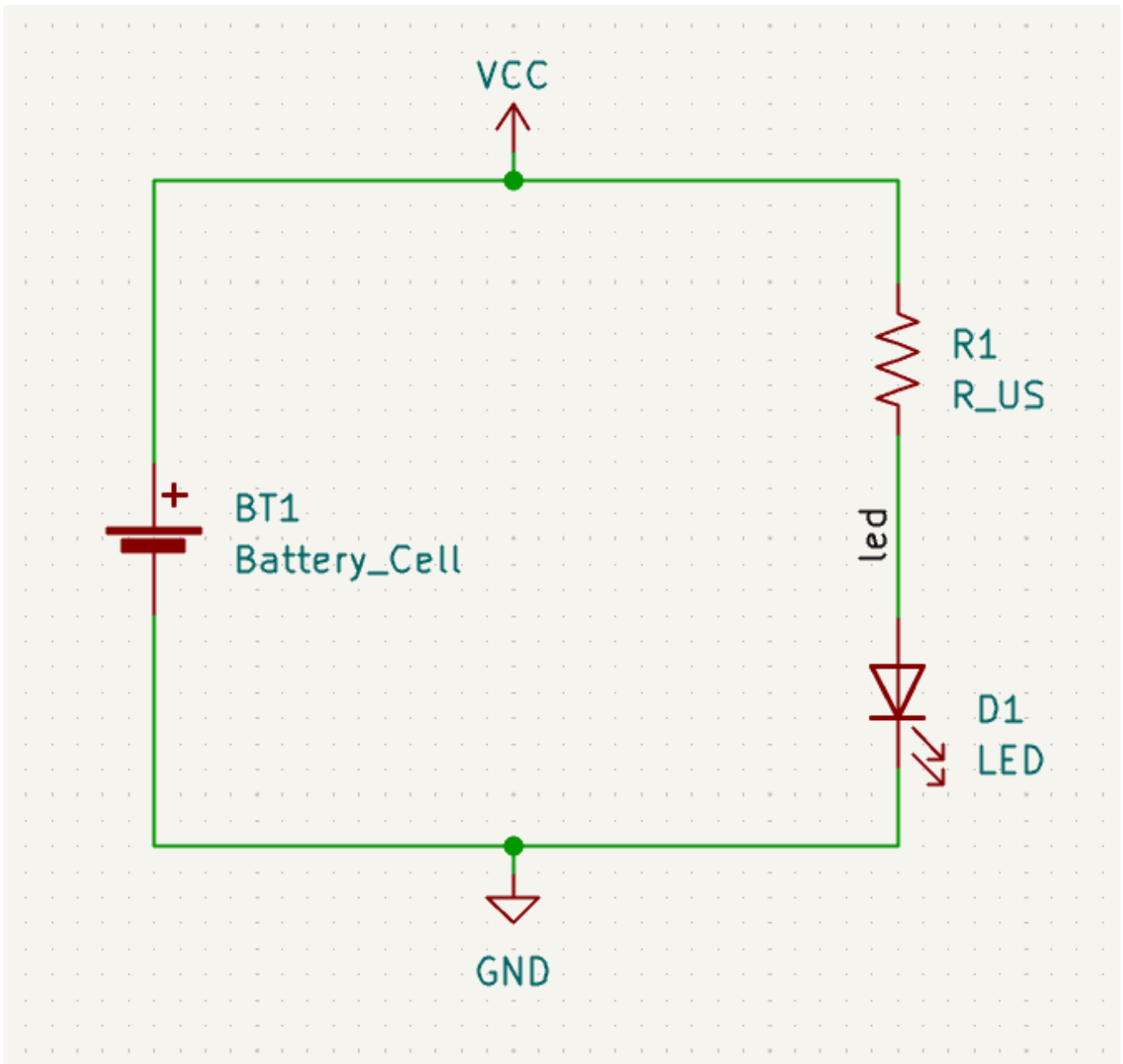


接下来，在原理图上添加电源和接地符号。虽然在这样一个简单的原理图中没有严格意义上的必要，但这些符号使大型原理图更容易理解。

在 **Power** 符号库中提供了多种电源和地符号。不过，添加这些符号有一个快捷方式：点击 **添加电源符号** 按钮  或使用 **P** 快捷键。这将弹出 **选择符号** 对话框，但仅显示包含电源符号的符号库。

添加一个 **VCC** 符号和一个 **GND** 符号，然后用导线将它们连接到电路中。

Finally, add a label to the wire between the LED and resistor. Again, this may not be necessary in a simple circuit, but it is good practice to label important nets. Click the **Place Net Labels** button  in the right toolbar () , type a label name ( led ), and place the label into the schematic so that the square attachment point overlaps with the wire. Rotate and align the label as necessary.





请注意，同名的标签和电源符号是连接在一起的。在这个原理图页面上，另一个 GND 符号或标记为 led 的导线将与现有的导线短路，即使没有导线将它们直观地连接起来。

## Annotation, symbol properties, and footprints


### 批注

每个符号都需要为其分配一个独特的位号。这个过程也被称为注解。

默认情况下，符号在添加到原理图时会自动进行批注。可以通过左侧工具栏中的  按钮来启用或禁用自动批注功能。

虽然在本指南中没有必要，但可以使用顶部工具栏中的 **填写原理图符号位号** 按钮 () 对符号进行手动批注或重新批注。


## Symbol properties

接下来，填入每个元件的数值。选择 LED，右键单击，并选择属性... ()。这个工程将使用一个红色的 LED，所以将 **值** 字段改为 **红色**。在一个真正的工程中，在这里写上 LED 制造商的零件编号可能会更好。注意，可以在每个符号的属性中单独编辑位号。

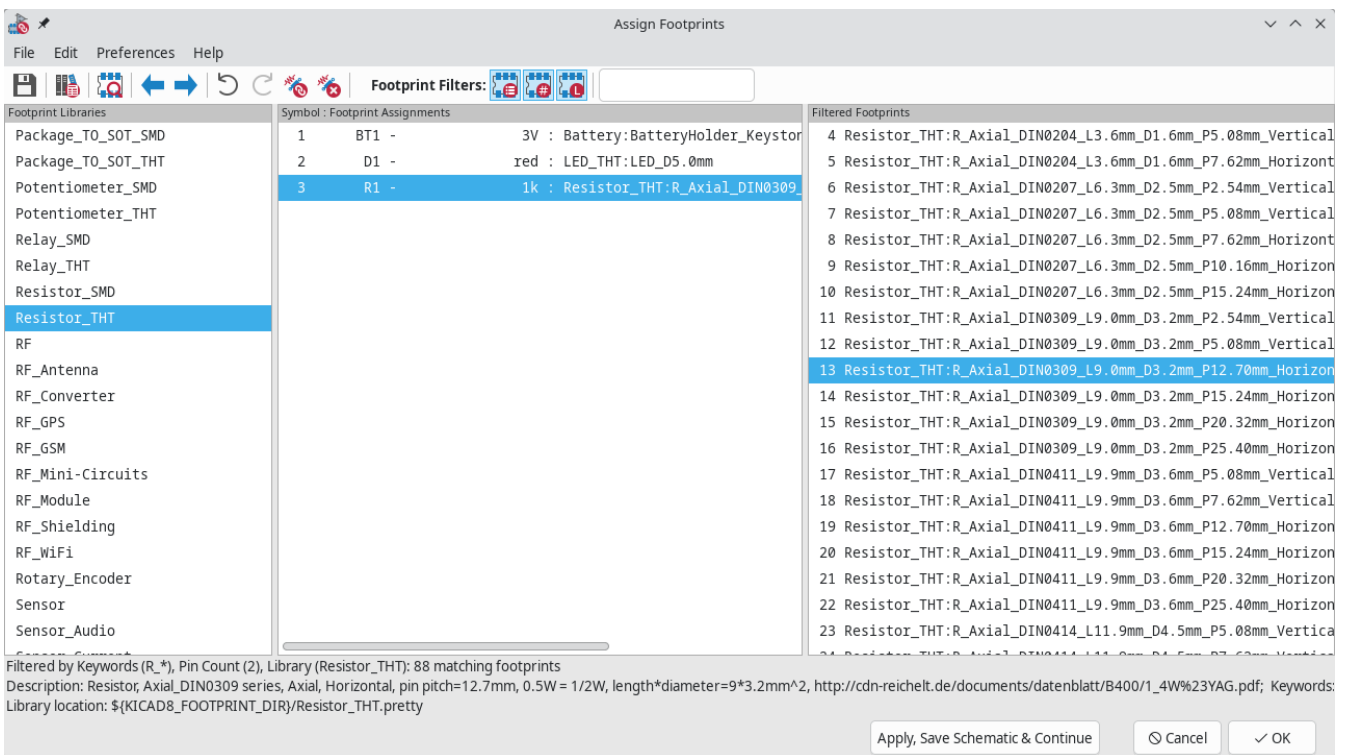
这个工程将使用一个 3V 的纽扣电池，所以把 BT1 的 **值** 字段改为 3V。将电阻的值改为 1k。

## Footprint assignment

最后，给每个元件指定一个封装。这定义了每个元件将如何连接到 PCB 上。有些符号预留了封装，但对许多元件来说，有多种可能的封装，所以用户需要选择适当的一个。


有几种分配封装的方法，但有一种方便的方法是使用封装分配工具，点击顶部工具栏的  按钮。


该窗口左侧窗格列出了可用的封装库。中间窗格显示原理图中的符号。所有这些符号都将分配封装。右侧窗格显示可为中间窗格所选符号选择的封装。要为某个符号分配封装，请在中间窗格中选择该符号，然后在右侧窗格中双击所需的封装。右击封装名称并选择 **查看所选封装**，即可预览封装。




KiCad 中包含了许多封装，因此封装分配工具提供了几种方法来筛选掉与有关符号不相关的封装。



- 最左侧的按钮  用于激活可以在每个符号中定义的筛选器。例如，一个运算放大器符号可能会定义仅显示 SOIC 和 DIP 封装的过滤器。有时这些预定义的筛选器可能缺失或过于严格，因此在某些情况下关闭此筛选器可能会很有用。

中间的按钮  按引脚数筛选，所以只有 8 个引脚的符号才会显示 8 个引脚的封装。这个筛选器几乎总是有用的。

- 右边的按钮  按所选库筛选。库在左侧窗格中被选中；不在所选库中的封装将被筛选掉。只要知道哪个库包含正确的封装，这个筛选器就很有用。通常情况下，最好使用这个筛选器或符号筛选器，但不能同时使用。
- 文本框筛选掉与框中文本不匹配的封装。当方框为空时，这个筛选器被禁用。


使用筛选器，在上方截图的中列中找到每个显示的封装。每个符号对应的封装也列在下表中。通过在中列中选择符号，然后在右列中双击相应的封装，将每个封装分配给符号。完成所有封装的分配后，点击 **确定**。

符号位号	分配封装
BT1	Battery:BatteryHolder_Keystone_1058_1x2032
D1	LED_THT:LED_D5.0mm
R1	Resistor_THT:R_Axial_DIN0309_L9.0mm_D3.2mm_P12.70mm_Horizontal

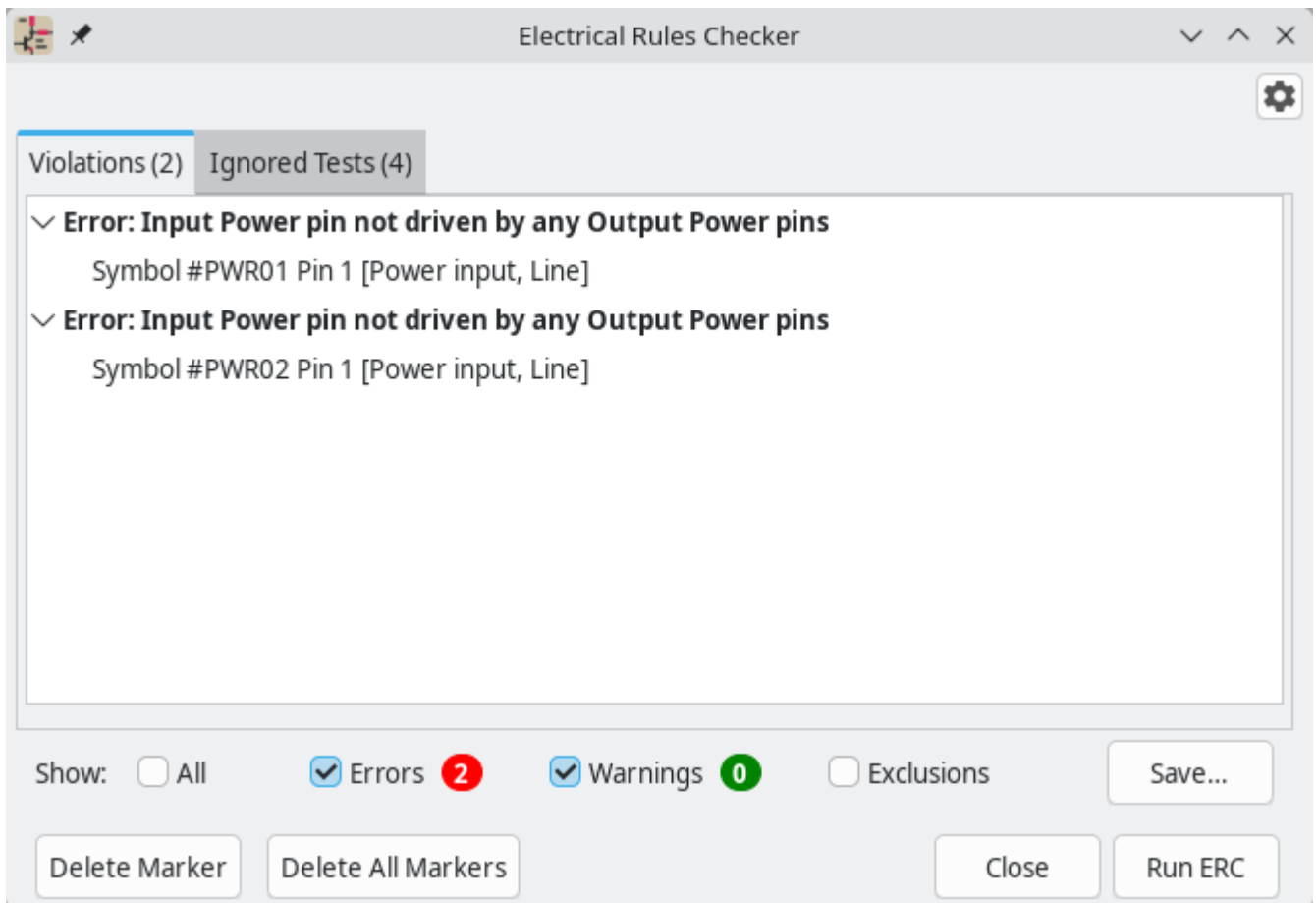
还有其他方法可以分配封装；其中一种方法是通过符号属性窗口。有关封装分配的更多信息，请参阅 [手册](#)。

## Electrical rules check

原理图中剩下的最后一件事就是检查电气错误。KiCad 的电气规则检查器（ERC）无法确保原理图中的设计能正常工作，但它可以检查一些常见的连接问题，如未连接的引脚、两个电源输出短路在一起或电源输入没有任何供电。它还能检查一些其他错误，如符号未加注释和网络标签中的错别字。要查看电气规则的完整列表并调整其严重程度，请访问 **文件** → **原理图设置...** → **电气规则** → **违反严重程度**。在开始布局之前，最好先运行 ERC。

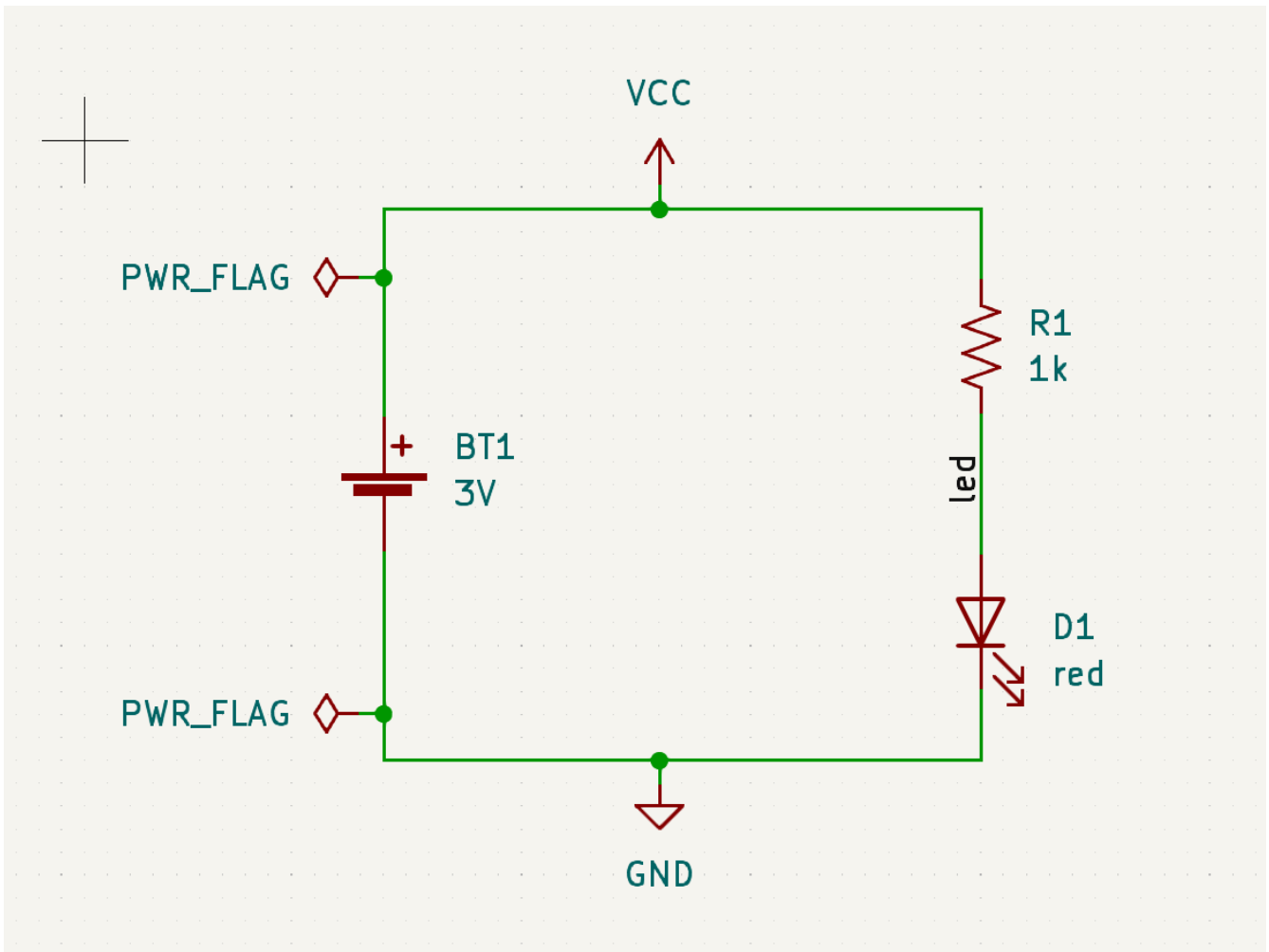
通过点击顶部工具栏的 **ERC** 按钮 ()，然后点击 **运行 ERC**，运行电气规则检查。

即使在这个简单的原理图中，KiCad 也发现了两个潜在的错误。这些错误列在 ERC 窗口中，箭头指向原理图中的违规位置。在 ERC 窗口中选择一个错误，就会显示出相应的箭头。



你可以通过右键点击每条错误信息来排除个别违规行为或忽略整个违规类别。然而，为了得到一份干净的 ERC 报告，避免遗漏真正的问题，通常值得处理这些违规行为，即使它们不是实际的设计错误。

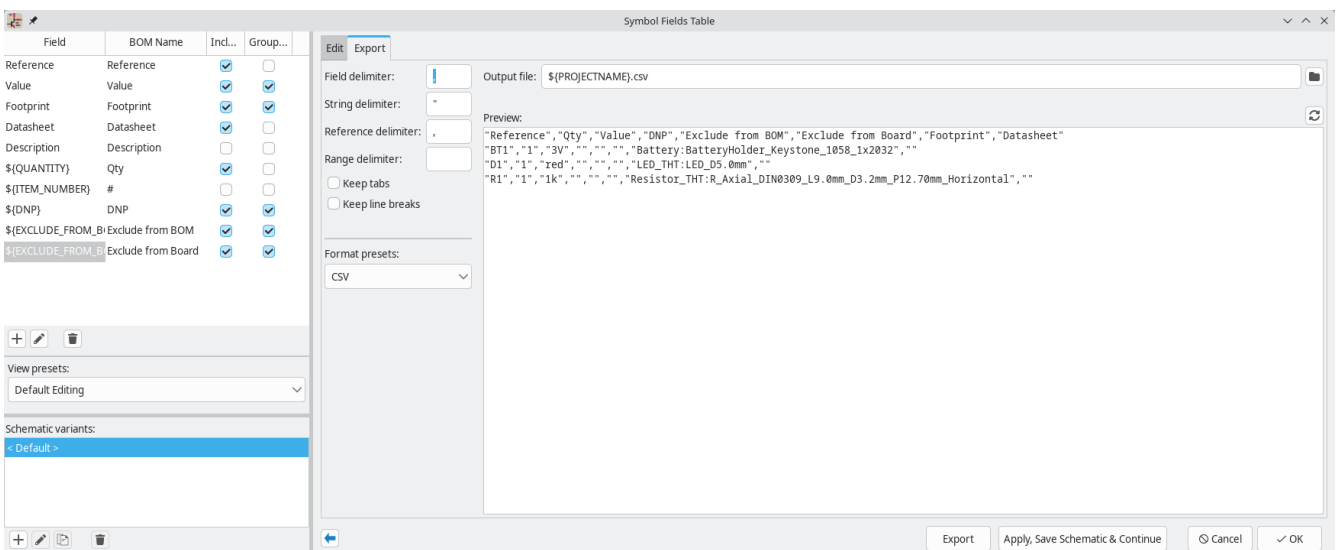
在这种情况下，KiCad 对 VCC 和 GND 网络报告“输入电源引脚没有被任何输出电源引脚驱动”。这是一个常见的 KiCad ERC 错误。电源符号被设置为需要一个电源输出引脚，如电压调节器的输出，在同一网络上；否则 KiCad 认为该网络未被驱动。对人来说，VCC 和 GND 显然是由电池驱动的，但有必要在原理图中明确显示。



在 Power 符号库中有一个特殊的 PWR\_FLAG 符号，用于解决这个问题，告诉 KiCad 这些网络实际上是被驱动的。将这个符号添加到 VCC 和 GND 网络上，然后重新运行 ERC。当 ERC 通过而没有任何违反时，原理图就完成了。

## Bill of materials

最后一步是可选的，即生成一份列出工程中所有元件的 BOM。点击 **工具** → **生成 BOM...**。



KiCad 提供了一个用于 BOM 导出的图形用户界面。您可以在 **编辑** 选项卡中配置要导出的符号元数据以及符号的分组方式，并在 **导出** 选项卡中配置输出格式。

目前，默认设置应该没问题，但你可以在预览窗格中看到将要导出的内容的原始视图。在 **导出** 选项卡中指定输出文件，然后按下 **导出** 按钮。

在电子表格程序中打开时，BOM 应如下图所示。

	A	B	C	D	E	F
1	Reference	Value	Datasheet	Footprint	Qty	DNP
2	BT1	3V	~	Battery:BatteryHolder_Keystone_1058_1x2032	1	
3	D1	red	~	LED_THT:LED_D5.0mm	1	
4	R1	1k	~	Resistor_THT:R_Axial_DIN0309_L9.0mm_D3.2mm_P12.70mm_Horizontal	1	
5						

# Tutorial part 3: circuit board


原理图完成后，回到工程窗口并打开 PCB 编辑器，可以通过点击 PCB 编辑器按钮或打开电路板文件。

## PCB editor basics

PCB 编辑器中的导航与原理图编辑器相同：用鼠标中键或右键拖动平移，用滚轮或 **F1** / **F2** 缩放。

PCB 编辑器的主要部分是一块画布，将在这里设计电路板。左边的工具栏有各种板子的显示选项，包括单位和布线、过孔、焊盘和区域的边框/填充显示模式的切换。画布右边的工具条包含了设计 PCB 的工具。

### NOTE

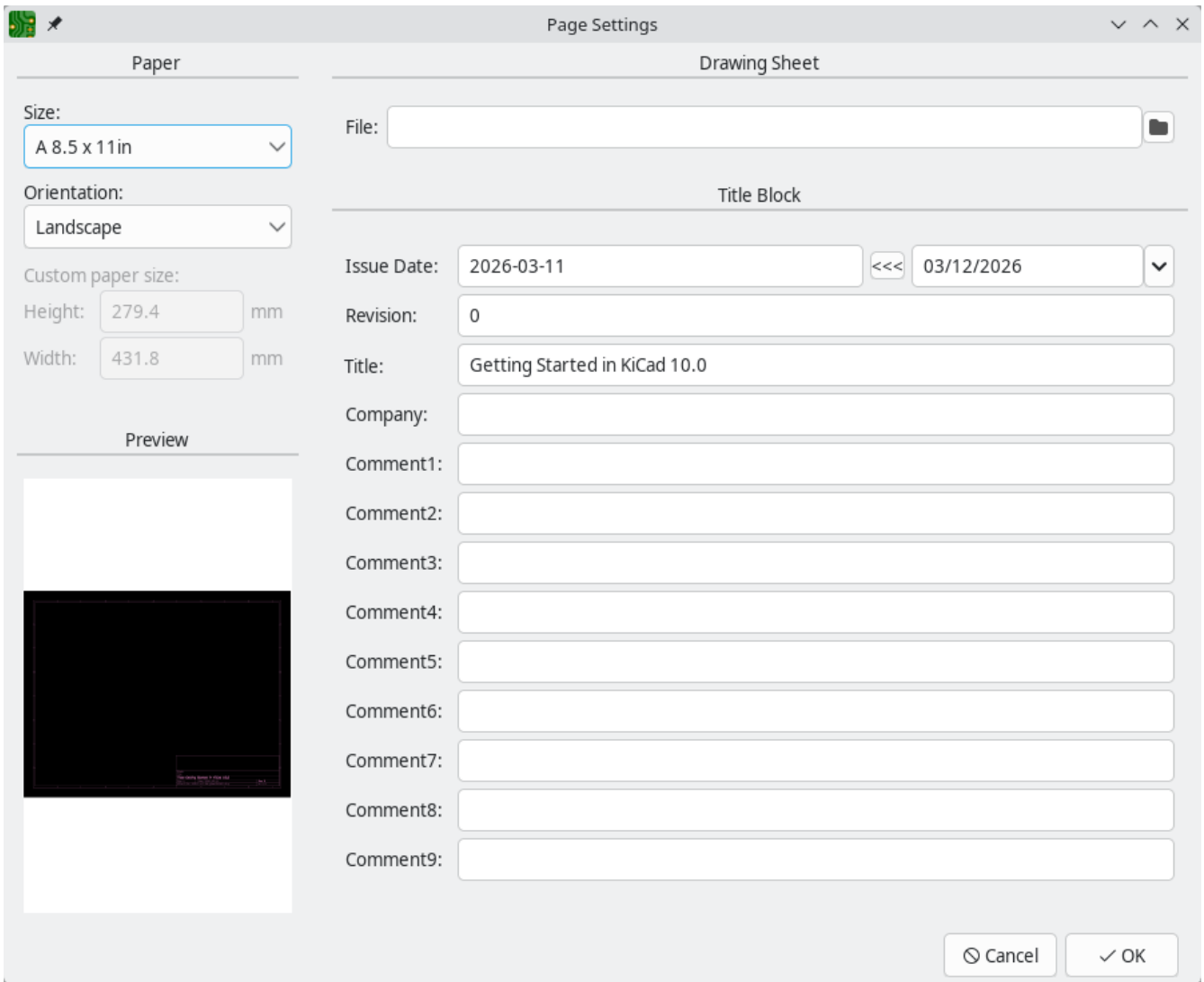
Some toolbar buttons have a small triangle in the bottom right corner . The triangle indicates that the button has an expanding palette containing several related tools, for example different kinds of dimensions. To select an alternate tool, click and hold on the button until the palette appears, then click on the alternate tool. Another way to use the palettes is to click on the button and drag to the left until the palette appears, then release the mouse button when the desired tool is highlighted.

最右边的是外观面板和选择筛选器。外观面板用于改变 PCB 层、对象和网络的可见性、颜色和透明度。通过点击层的名称可以改变活动层。

在外观面板下面是选择筛选器，它可以启用和禁用各种类型的 PCB 对象的选择。这对于在拥挤的布局中选择特定的项目很有用。

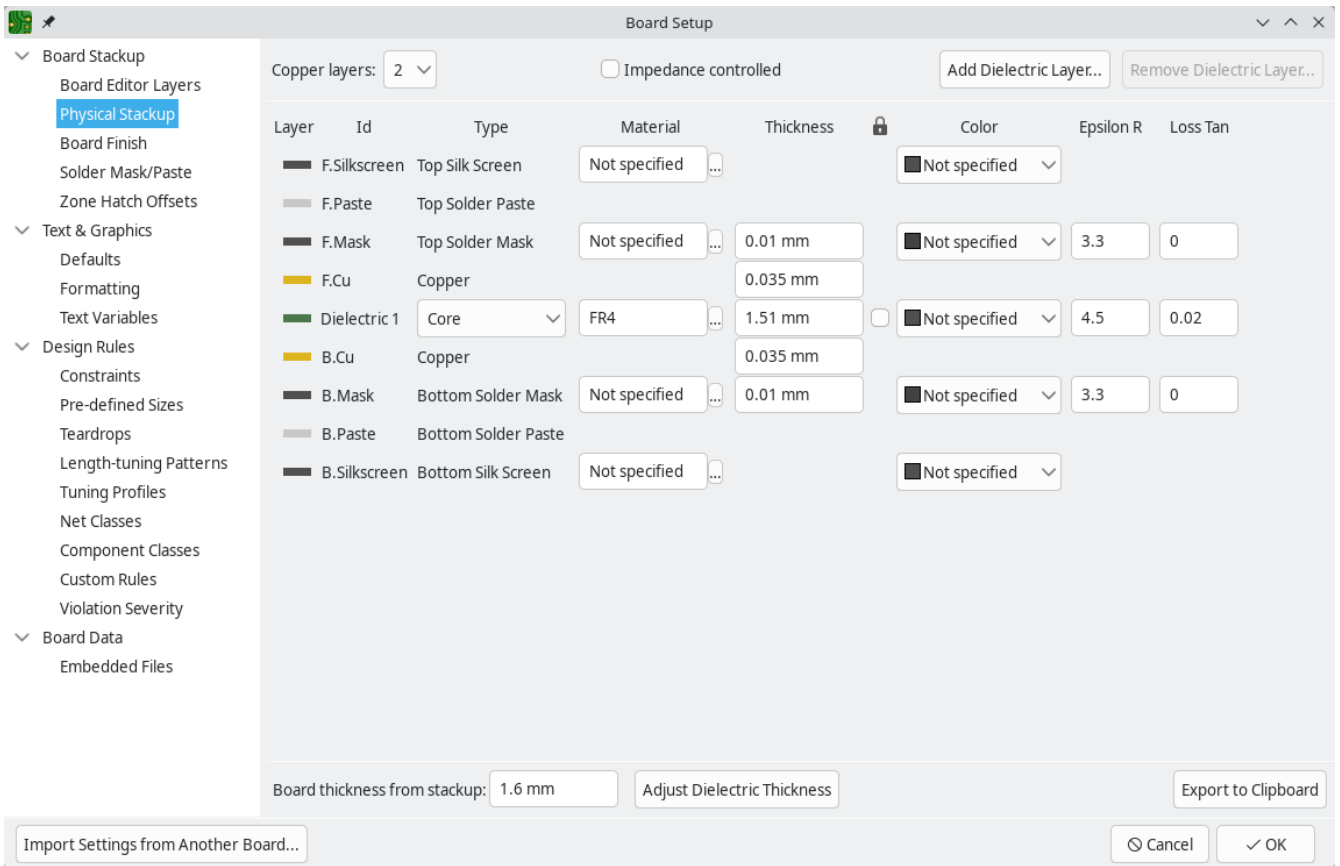
## Board setup and stackup

在设计电路板之前，先设置页面大小，并在标题栏中添加信息。点击 **文件** → **页面设置...**，然后选择一个合适的纸张尺寸，并输入日期、修订和标题。

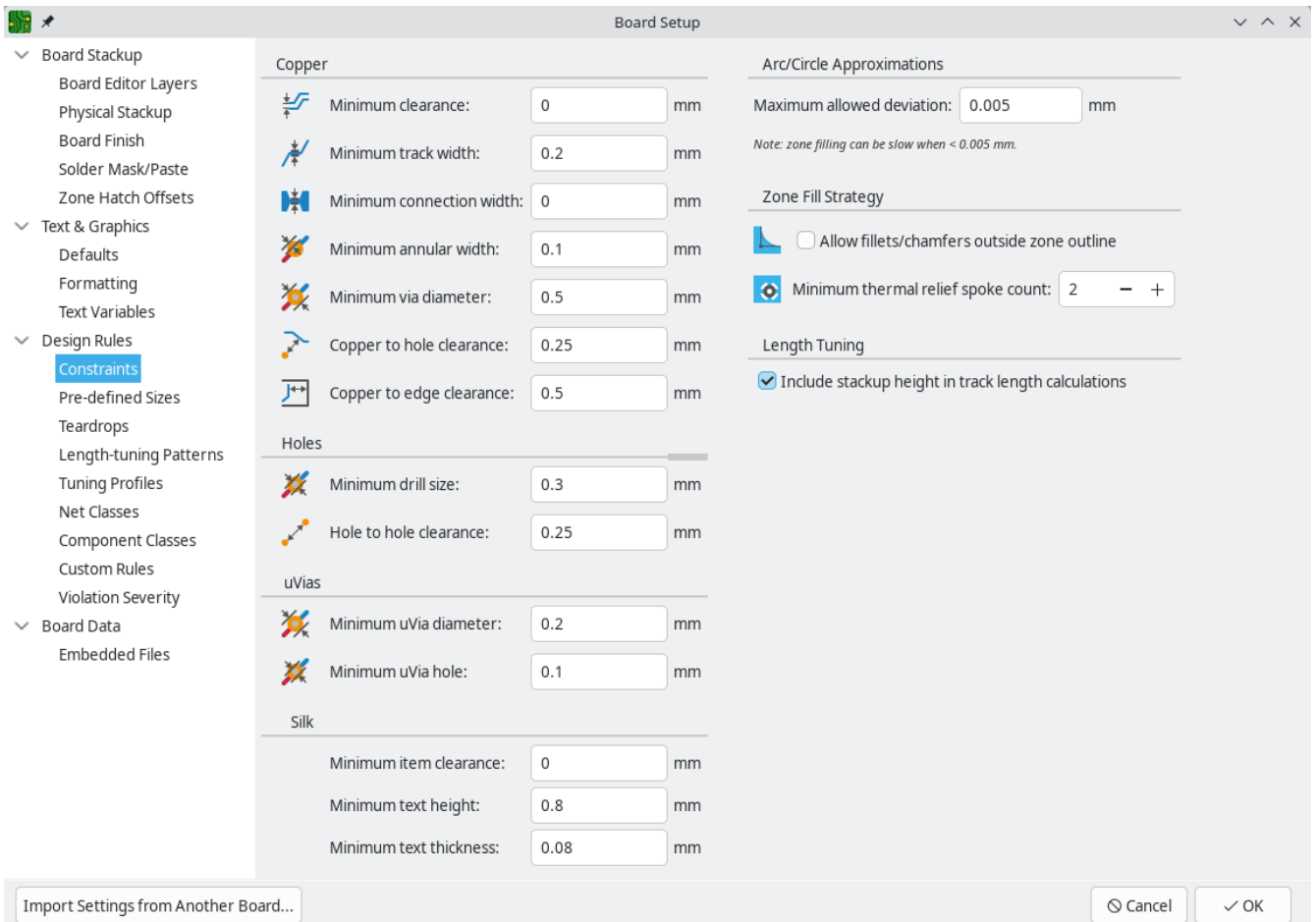


接下来，进入 **文件** → **电路板设置...**，定义如何制造 PCB。最重要的设置是压层，即 PCB 将有哪些铜层和介质层（以及它们的厚度），以及设计规则，例如布线和过孔的尺寸和间距。

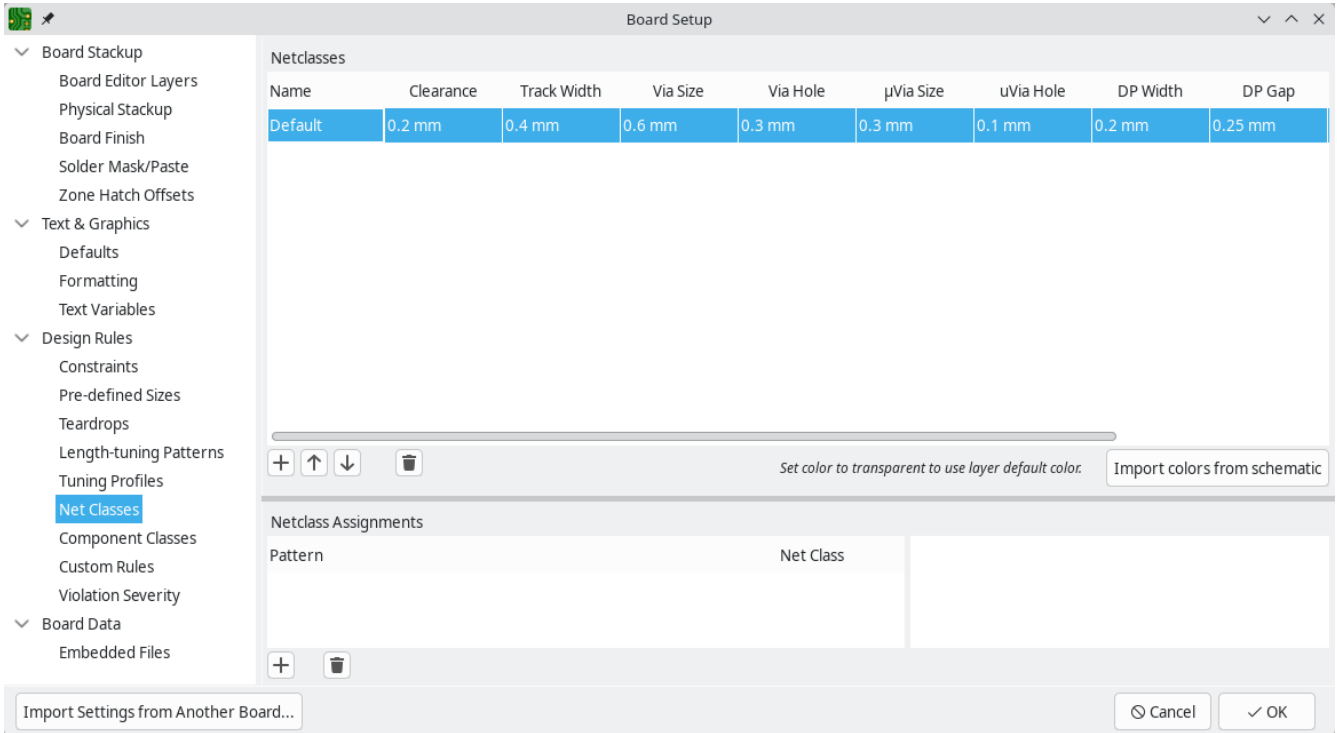
要设置压层，打开电路板设置窗口的 **电路板压层** → **物理压层** 页面。在本指南中，将铜层的数量定为 2，但更复杂的工程可能需要更多的铜层。



接下来，进入 **设计规则** → **约束** 页面。这个页面上的设置为电路板设计中的所有内容指定了首要的设计规则。就本指南的目的而言，默认值是好的。然而，对于一个真正的工程来说，应该根据 PCB 工厂的能力来设置这些规则，这样才能使 PCB 设计具有可制造性。



最后，打开 **设计规则** → **网络类** 页面。网络类是与特定网络组相关联的一组设计规则。此页面列出了设计中每个网络类的设计规则，并允许将网络分配到各个网络类（网络也可以在原理图编辑器中分配到网络类）。



布线的宽度和间距可以由设计者在布局过程中手动管理，但推荐使用网络类，因为它们提供了一种自动管理和检查设计规则的方法。

In this design, no net classes are specified, so all nets will belong to the **Default** net class. A single net class is acceptable for this project, but other designs may have multiple net classes, each with different design rules. For example a board might have a **High Current** netclass with wide tracks, or a **50 ohm** netclass with specific width and clearance rules for 50 ohm controlled-impedance tracks.

This project has relatively few components and is not space constrained, so we can be generous with track sizing. Larger tracks can carry more current and are more mechanically robust. Increase the **Default** net class's **Track Width** from 0.2 mm to 0.4 mm. When we route tracks, they will automatically have a width of 0.4 mm.

## Importing changes from schematic

原理图已经完成，但布局中还没有任何元件。要从原理图中导入设计数据到布局中，请点击 **工具** → **从原理图更新 PCB...**，或按 **F8**。在顶部的工具栏上还有一个按钮 。

仔细阅读 **要应用的改变** 窗口中的信息，其中会说原理图中的三个元件将被添加到电路板上。点击 **更新 PCB**，**关闭**，然后点击画布来放置这三个封装。每个封装相对于其他封装的位置将在以后被改变。

Update PCB from Schematic

Options

- Re-link footprints to schematic symbols based on their reference designators
- Group footprints based on symbol group

Update Footprints

- Replace footprints with those specified by symbols
- Delete footprints with no symbols
- Override locks

Update Fields

- Update footprint fields from symbols
- Remove footprint fields not found in symbols

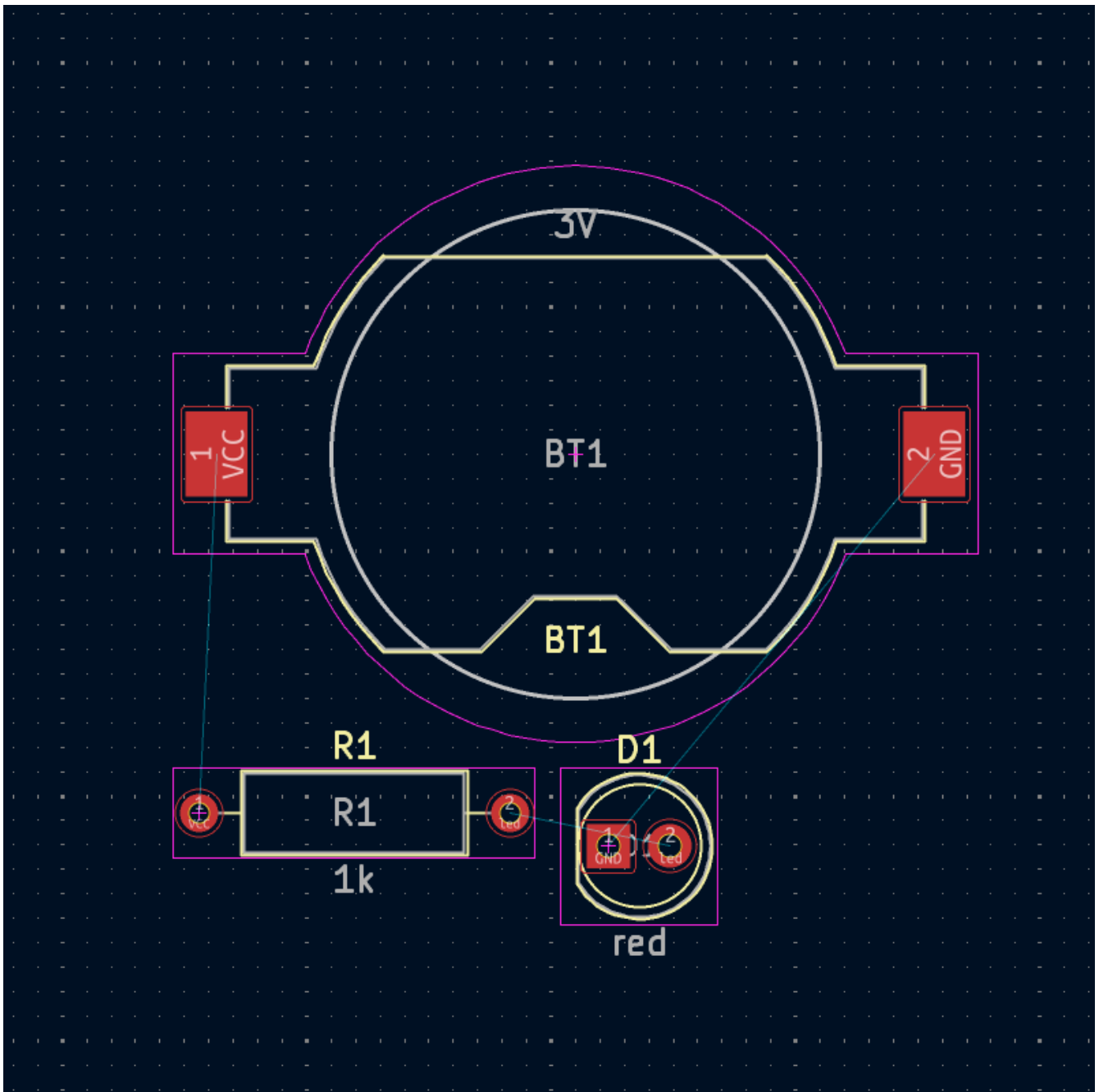
Changes to Be Applied

Processing symbol 'BT1:Battery:BatteryHolder\_Keystone\_1058\_1x2032'.  
Processing symbol 'D1:LED\_THT:LED\_D5.0mm'.  
Processing symbol 'R1:Resistor\_THT:R\_Axial\_DIN0309\_L9.0mm\_D3.2mm\_P12.70mm\_Horizontal'.  
Add BT1 (footprint 'Battery:BatteryHolder\_Keystone\_1058\_1x2032').  
Add D1 (footprint 'LED\_THT:LED\_D5.0mm').  
Add R1 (footprint 'Resistor\_THT:R\_Axial\_DIN0309\_L9.0mm\_D3.2mm\_P12.70mm\_Horizontal').

Total warnings: 0, errors: 0.

Show:  All     Errors **0**     Warnings **0**     Actions     Infos    Save...

Close    Update PCB

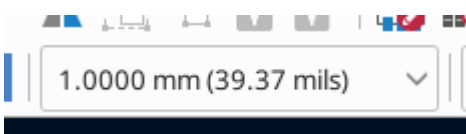








在 KiCad 中，根据原理图的变化更新 PCB 是一个手动过程：设计者决定何时适合根据原理图的修改来更新 PCB。每次编辑原理图时，设计者必须使用 **从原理图更新 PCB** 工具来保持原理图和布局的同步。

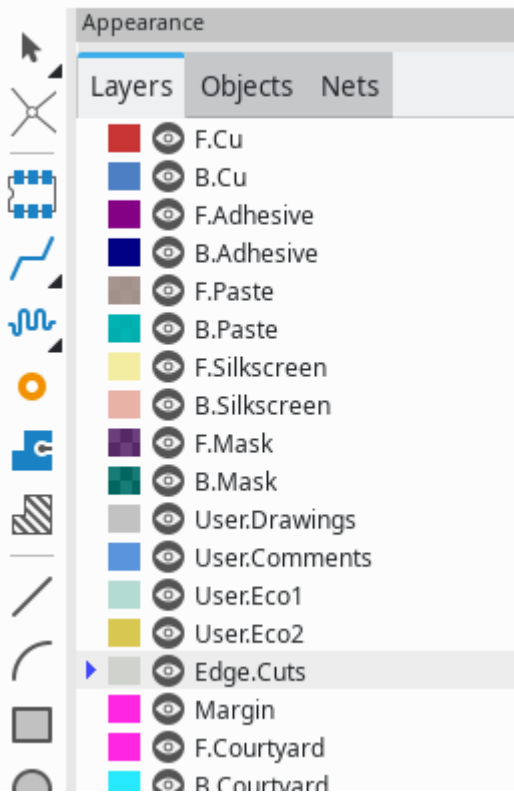
## Drawing a board outline

现在这三个元件已经被放置好了，但是电路板本身还没有被定义。通过在 `Edge.Cuts` 层上画一个板的边框来定义电路板。

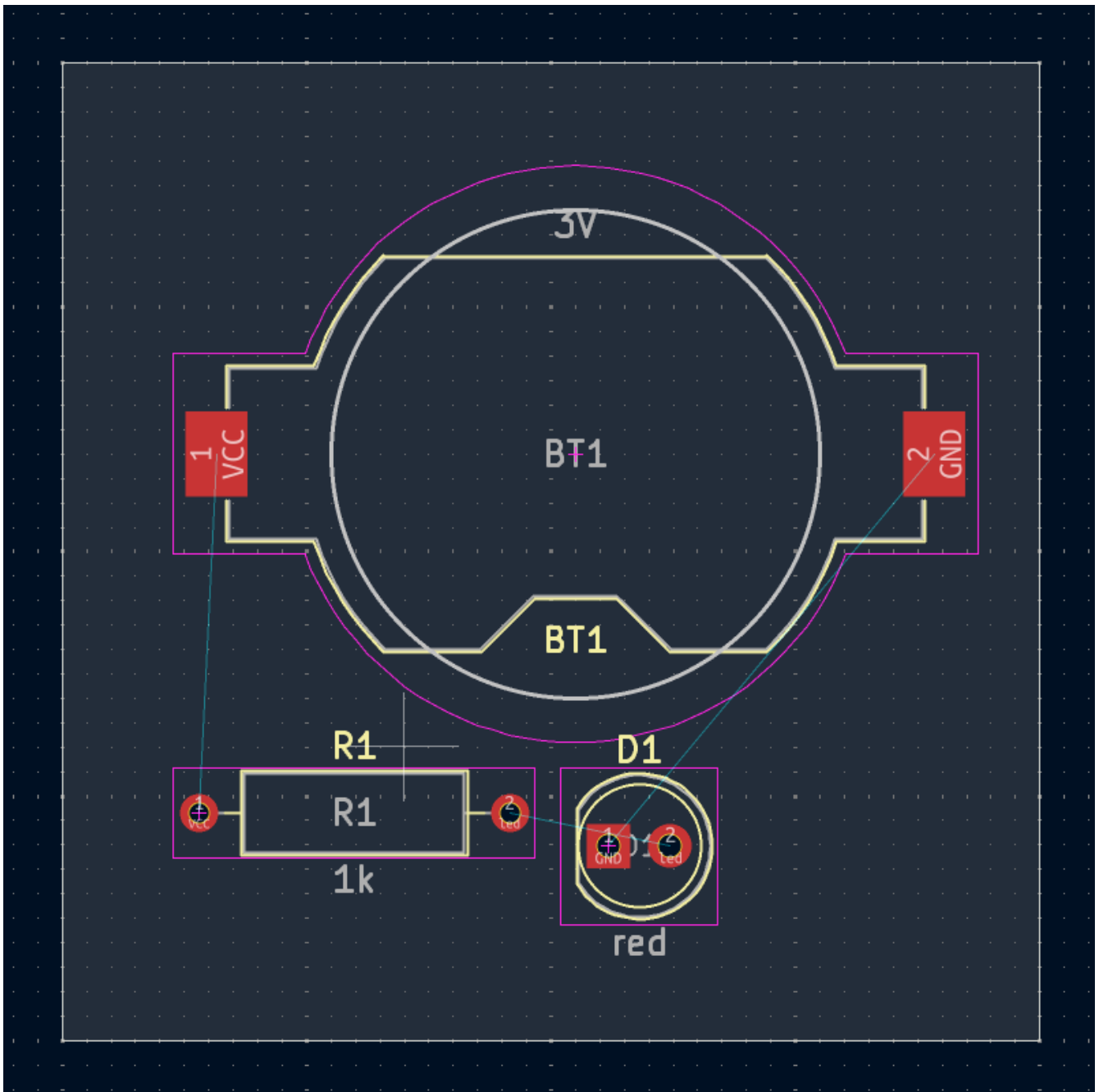
用粗大的网格来绘制电路的板框通常是很有用的，这样可以很容易得到画布尺寸的整数。在画布上方的网格下拉菜单中选择 1 毫米，就可以切换到粗略的网格。



要在 Edge.Cuts 层上绘制，请点击右侧外观面板的“图层”选项卡中的 **Edge.Cuts**。选择右侧工具栏中的矩形工具 ，在画布上点击以放置第一个角点，然后再次点击以放置对角点，使矩形大致包围三个封装。其他图形工具（如直线 、圆弧 、圆形 、多边形 、贝塞尔曲线 ，或它们的组合）也可用于定义板框；唯一的要求是板框必须是一个不自交的闭合形状。



The board outline should look like this:



The lighter grey color inside the outline indicates the interior of the board.

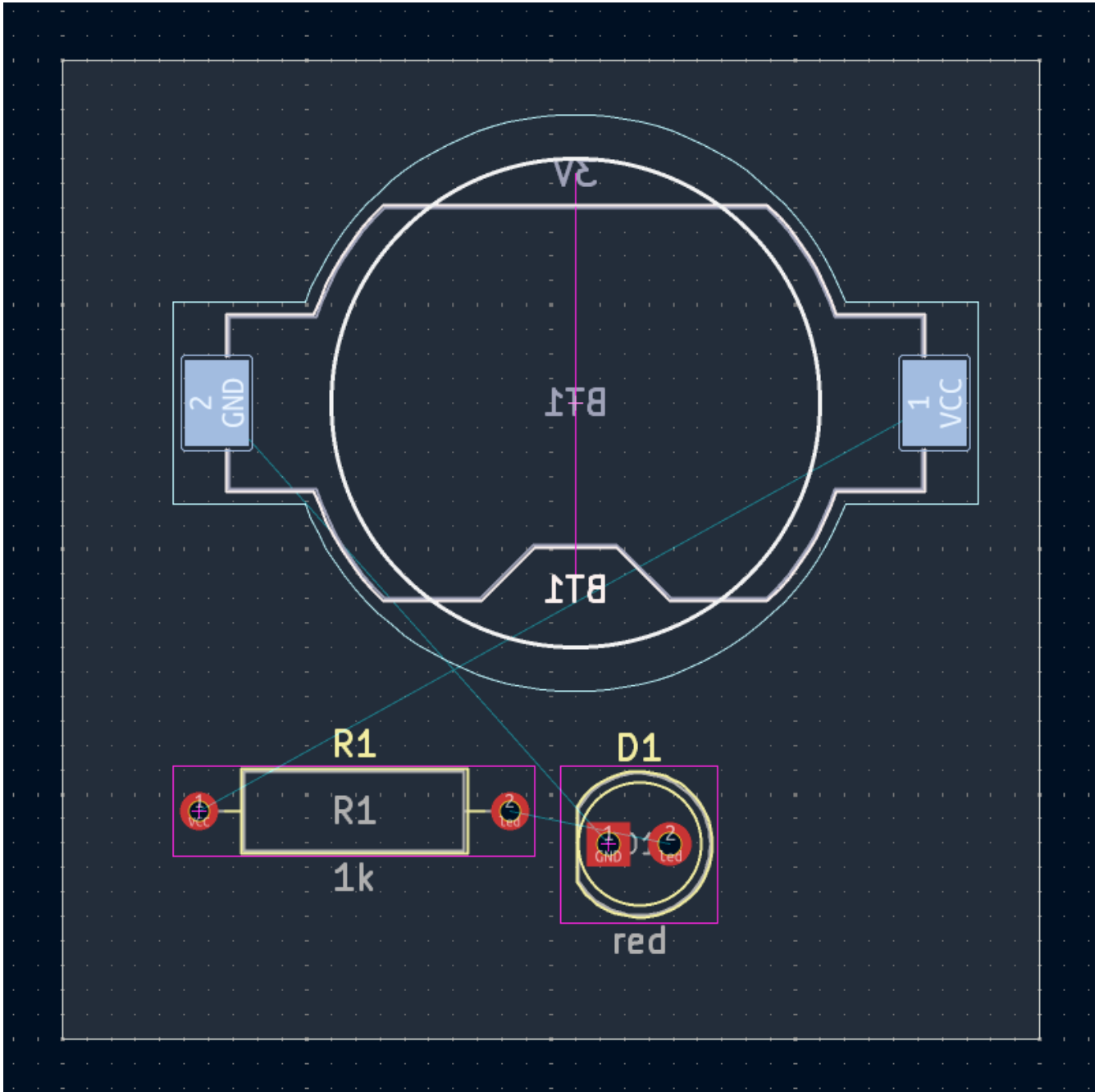
## Placing footprints

布局过程中的下一步是在电路板上安排封装。一般来说，在定位封装时有几个考虑因素：

- 有些封装可能对其位置有确切的要求，如连接器、指示灯或按钮和开关。
- 一些元件可能需要根据电气方面的考虑来放置。旁路电容应靠近相关 IC 的电源引脚，敏感的模拟元件应远离数字干扰。
- 几乎所有元件都有一个“边框”（如果定义了正面和背面，则可能有两个）。通常情况下，边框之间不应相互交叉。
- Otherwise components should be positioned for ease of routing. Connected components should generally be close together, and arranged to minimize routing complexity. The *ratsnest* (the thin lines indicating connections between pads) is useful for determining how best position footprints relative to other footprints.

在本指南中，唯一的放置目标是使布线过程尽可能简单。

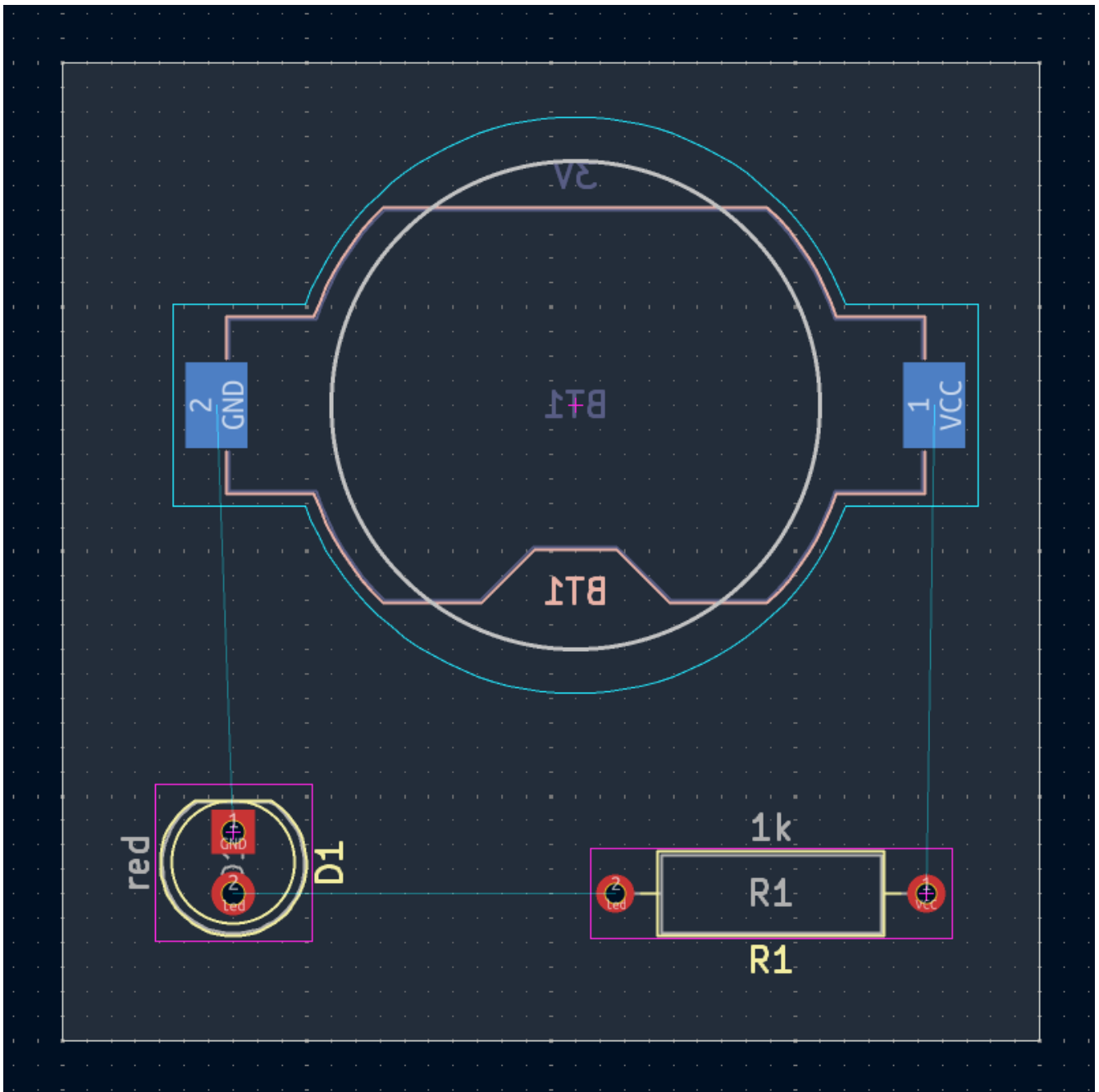
Start by moving the battery holder BT1 onto the back side of the board. Click it to select it, then press **M** to move it. Press **F** to flip it to the opposite side, then click to place it. It now appears mirrored and its pads have changed from red to blue.



所有的 PCB 层都是从板子的正面看的。因此，电路板底部的封装是颠倒的，看起来是镜像的。

每个 PCB 层都有一个独特的颜色，由外观面板的层标签中的色板显示。在默认的颜色方案中，F.Cu（正面铜）层上的项目是红色的，而 B.Cu（背面铜）上的项目是蓝色的。


Now place the other two components. One at a time, select each component, then move and rotate it with **M** and **R**. The thin lines between pads are called *ratsnest* lines; they indicate connections in the schematic that you have not yet created in the PCB. Watch the ratsnest lines between each pad to choose the simplest arrangement of components; a good arrangement will leave the lines untangled, which means it will be easy to route the tracks between pads. One possible arrangement is shown in the screenshot below.

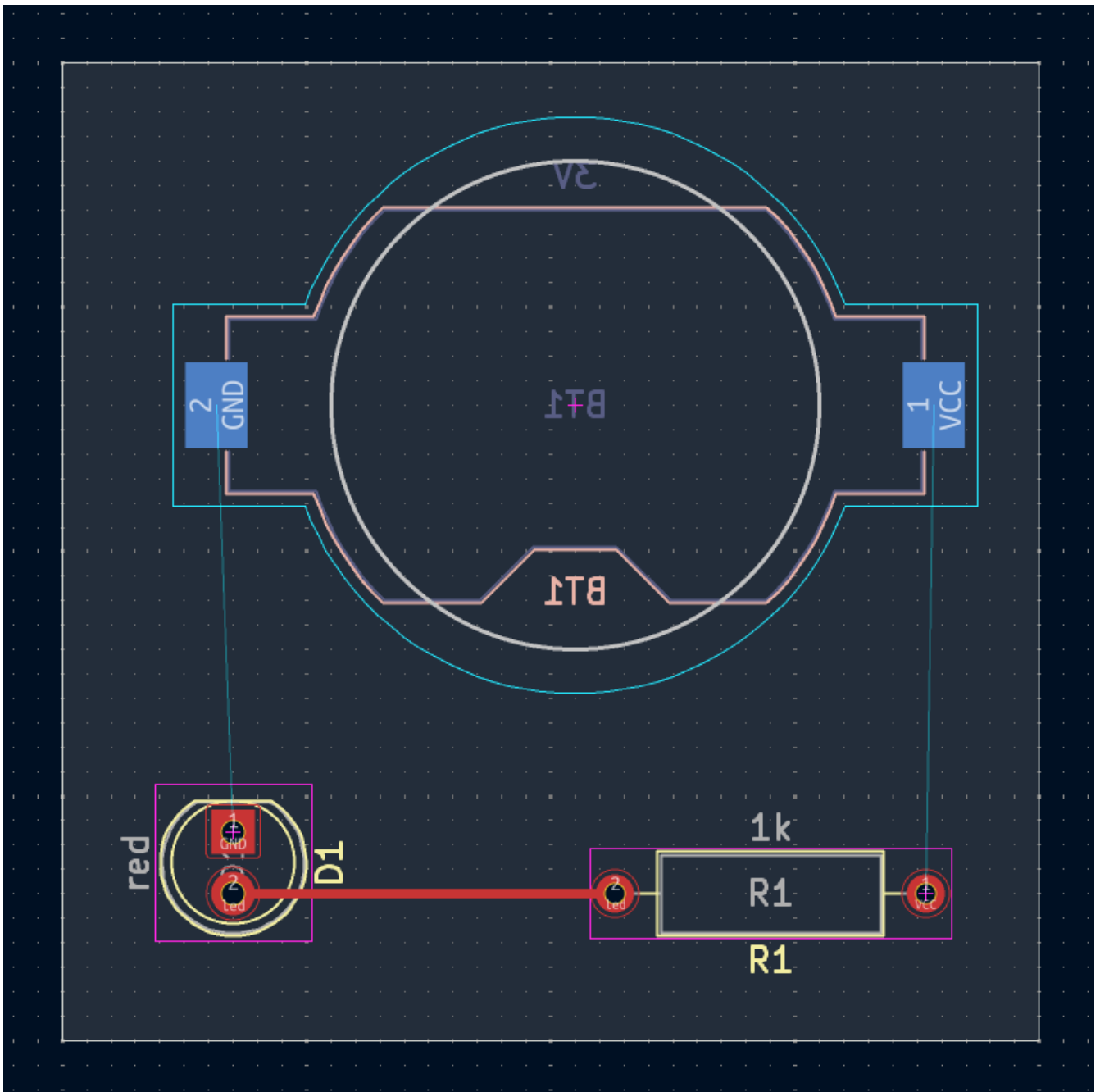


## Routing tracks

元件就位后，接下来便是用铜线将焊盘连接起来。

第一条走线将在电路板的正面绘制，因此请在“外观”面板的“图层”选项卡中将活动层更改为 F.Cu。

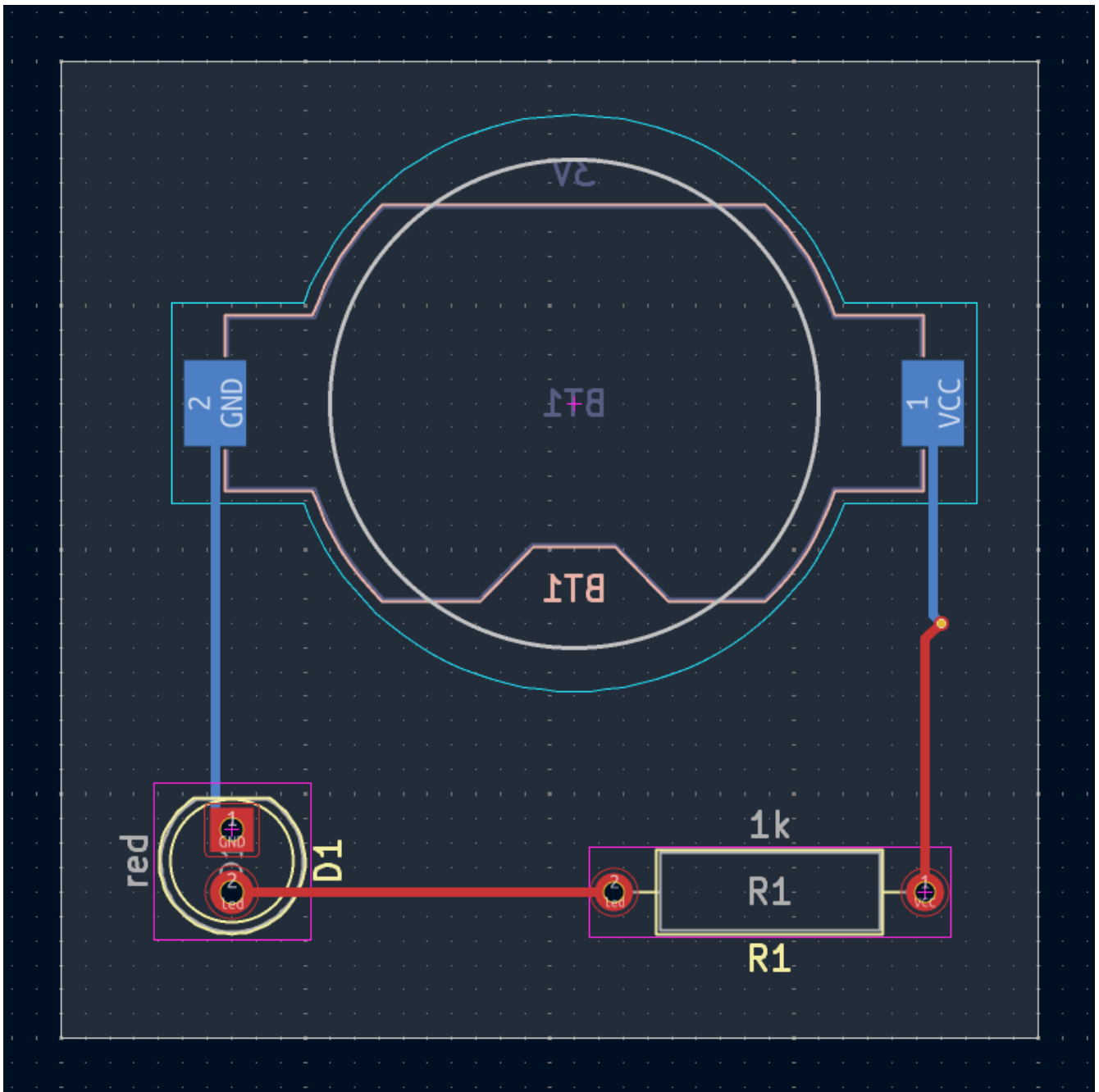
点击右侧工具栏中的 **布线**  或按下 **X**。点击 D1 的 led 焊盘并松开鼠标按钮。此时，飞线布线 (ratsnest line) 显示 R1 的 led 焊盘存在未布线的连接，因此点击该焊盘以绘制连接两个焊盘的走线。点击第二个焊盘后，走线即完成。由于铜层已建立连接，led 引脚之间的飞线将不再显示。



现在，在 BT1 和 D1 的 GND 焊盘之间绘制一条走线，从电路板背面的 BT1 焊盘开始。请注意，点击 BT1 焊盘后，活动层自动切换为 B.Cu。点击 D1 焊盘以完成走线。

虽然 BT1 的表面贴装焊盘仅位于电路板的底部，但 D1 的通孔焊盘可以连接到电路板正面和背面的走线。通孔焊盘是实现多层之间连接的一种方式。在本例中，D1 是电路板正面上的一个元件，但其通孔焊盘用于连接到电路板背面的走线。


另一种跨层连接的方法是用过孔。在电路板背面的 BT1 的 VCC 焊盘上开始布线。按 **V** 并在 BT1 和 R1 之间点击，插入一个过孔，同时将活动层切换到 F.Cu。通过点击 R1 的 VCC 焊盘完成电路板顶部的布线。



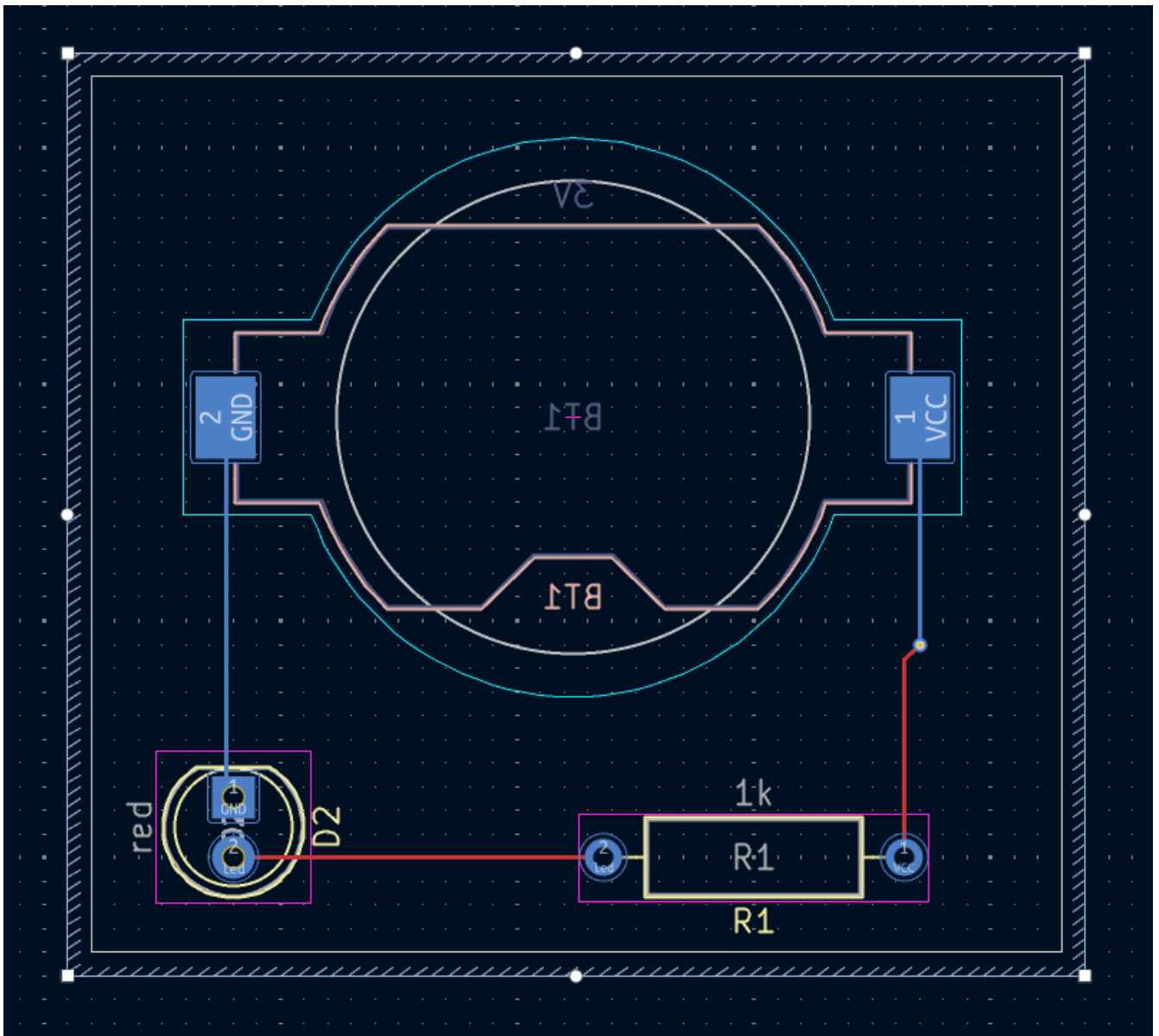
在这一点上，所有的连接都被布线了。这可以通过查看窗口左下方的状态屏幕来确认，其中未被布线的网络数量为 0。

## Placing copper zones

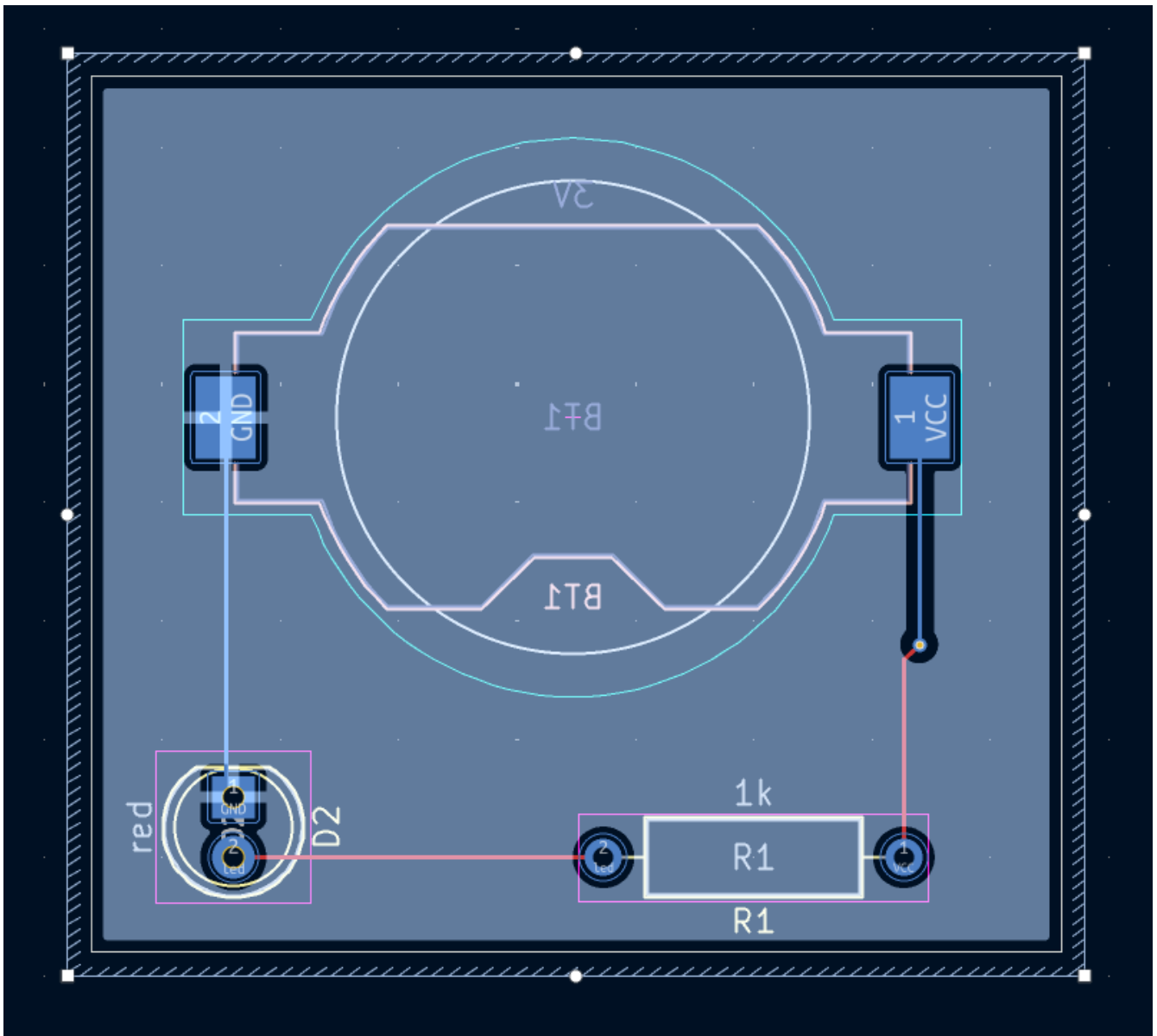
铜区域是指带有相关网络的铜质区域。这些区域会自动连接到属于同一网络的铜质对象，并避开属于其他网络的对象。铜区域常用于接地和电源连接，因为它们提供的连接阻抗比导线更低。

在电路板底部添加一个 GND 区域，方法是切换到底部铜层，然后点击右侧工具栏中的 **添加填充区域** 按钮 。点击 PCB 以放置区域的第一个角。


In the Copper Zone Properties dialog that appears, select the GND net from the **Net name** dropdown and make sure that the B.Cu layer is selected. Click **OK**, then click to place the other three corners of the zone. Double click when placing the last corner to complete the zone.



敷铜边框已在画布上显示，但敷铜尚未填充——敷铜内没有铜箔，因此该敷铜尚未形成任何电气连接。通过 **编辑** → **填充所有区域** (B) 来填充敷铜。此时，铜箔已添加到敷铜中，但它并未连接到 VCC 或 led 焊盘和走线，且被板边裁剪。它与之前绘制的 GND 走线重叠，并通过细小的走线与 GND 焊盘相连。这些细小的走线是热焊盘 (Thermal Relief)，它们使焊盘更易于焊接。热焊盘及其他敷铜设置可以在敷铜属性对话框中进行修改。




在 KiCad 中，当第一次绘制或修改敷铜时，或移动敷铜内的封装时，敷铜不会被自动填充。敷铜是通过手动填充和运行 [DRC 章节](#) 时重新填充。在生成 [制造输出章节](#) 之前，请确保敷铜填充是最新的。

有时，在一个拥挤的画布设计中，填满的敷铜会让人很难看到其他的对象。可以使用左侧工具栏上的 **只显示敷铜边界** 按钮隐藏敷铜，。当只显示敷铜的边框时，敷铜会保留其填充状态—隐藏敷铜的填充与取消填充是不同的。

敷铜也可以使用外观面板使之透明，非活动层也可以使用外观面板中的 **层显示选项** 隐藏或调暗。

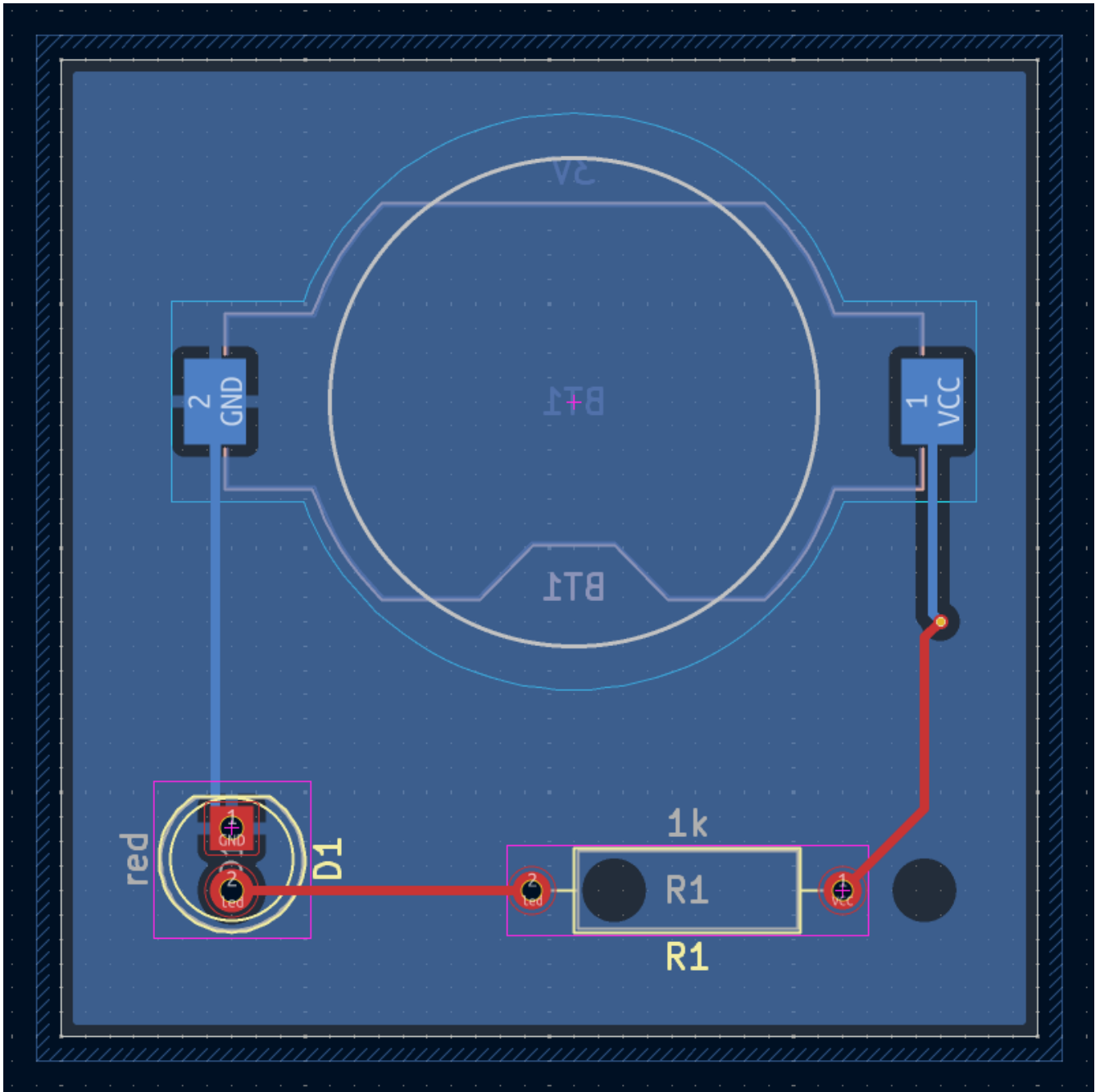
## Design rule checking

设计规则检查 (DRC) 相当于原理图中的电气规则检查 (ERC)。DRC 用于查找设计错误，例如原理图与布局之间的不匹配、铜区域间距不足或短路，以及未连接到任何地方的走线。您还可以编写自定义的 DRC 规则。要查看完整的设计规则列表并调整其严重性，请前往 **文件** → **板设置...** → **设计规则** → **违规严重性**。在生成 [制造输出章节](#) 之前，强烈建议运行 DRC 并修复所有错误。

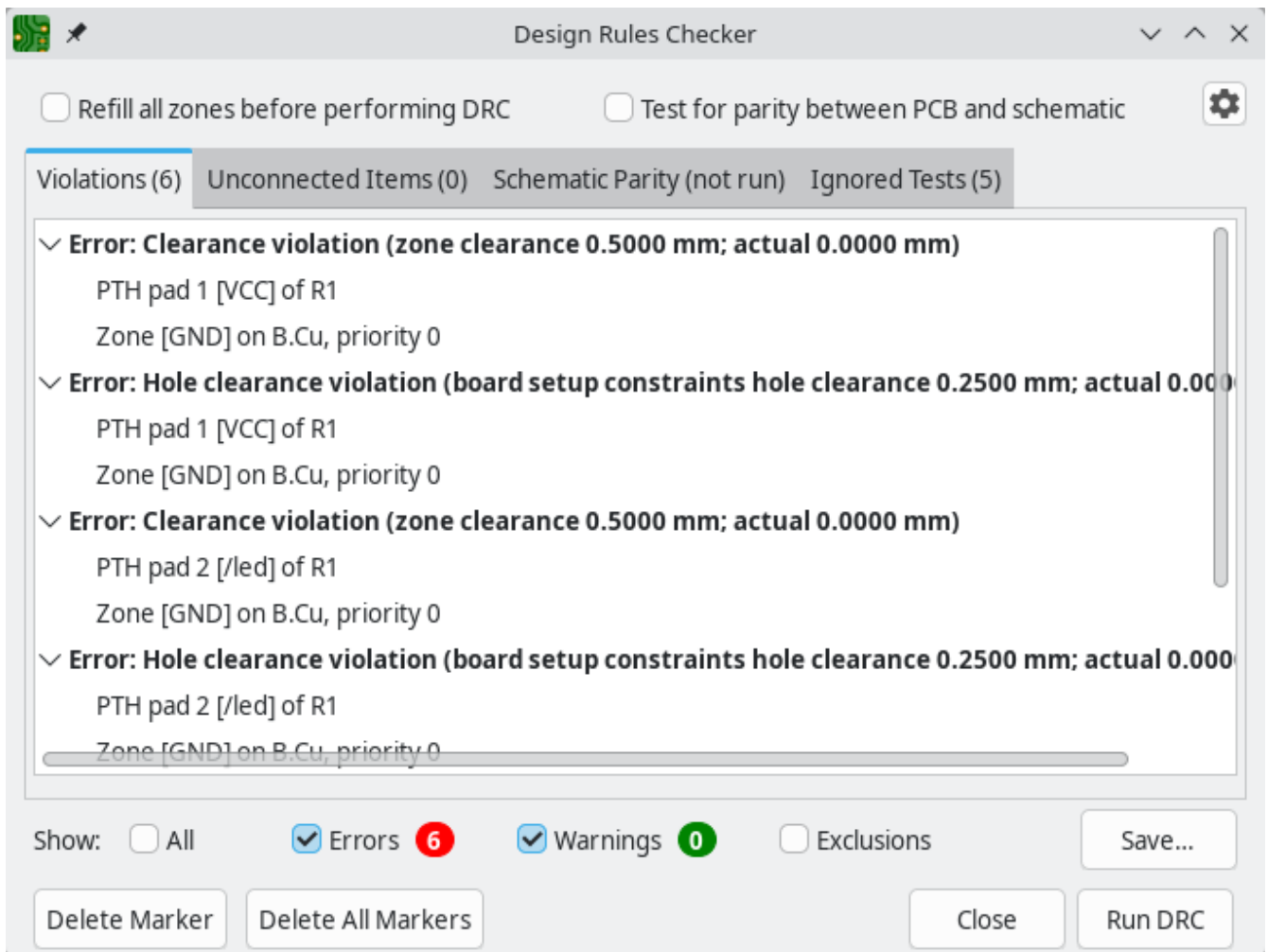
用 **检查** → **设计规则检查** 来运行 DRC 检查，或者使用顶部工具栏的  按钮。单击 **运行 DRC**。当检查完成后，不应该有错误或警告报告。关闭 DRC 窗口。

现在，故意通过移动电阻器封装使其与填充区域重叠来引发 DRC 错误。使用 **D** (拖动) 功能轻微移动电阻器封装，同时保持其焊盘上的走线连接。这将导致一个间距违规，因为电阻器的 VCC 和 led 焊盘与 GND 区域填充短路。通

常，这种情况可以通过重新填充区域来修复，但暂时不要重新填充区域。



再次运行 DRC，但请确保取消勾选 **在执行 DRC 前重新填充所有区域** 复选框。DRC 报告了 6 个违规：对于 R1 的每个焊盘，存在焊盘与区域之间的间距违规，焊盘通孔与区域之间的另一间距违规，以及第三个违规，即焊盘的阻焊层开口暴露了两个不同网络（GND 填充和连接到焊盘的走线）的铜层。画布上的箭头指向每个违规。点击每个违规信息会放大显示相应的违规位置。

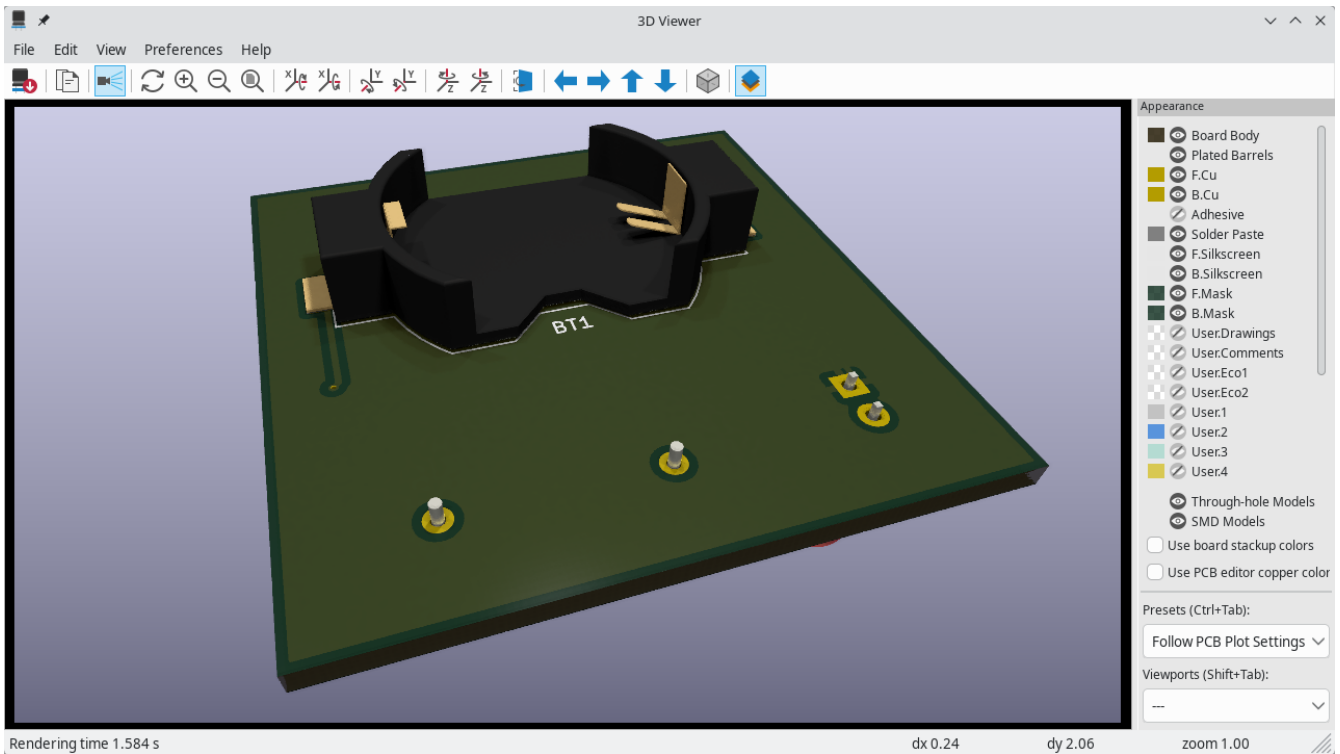


关闭 DRC 对话框，按 **B** 重新填充敷铜，并重新运行 DRC。或者，勾选 **在执行 DRC 之前重新填充所有敷铜** 复选框并重新运行 DRC。所有的违规行为都被修复。

## 3D viewer

KiCad 提供了一个 3D 查看器，对检查 PCB 很有用。用 **查看** → **3D 查看器** 打开 3D 查看器。用鼠标中键拖动进行平移，用鼠标左键拖动进行绕行。绕着 PCB 板运行，可以看到顶部的 LED 和电阻，以及底部的电池座。

提供了一种光线追踪模式，虽然速度较慢，但能提供更精确的渲染效果。您可以通过 **偏好设置** → **光线追踪** 切换到该模式。



KiCad 库中的许多封装都带有 3D 模型，包括本指南中使用的所有封装。有些封装没有附带 3D 模型，[但用户可以添加自己的模型章节](#)。

## Fabrication outputs

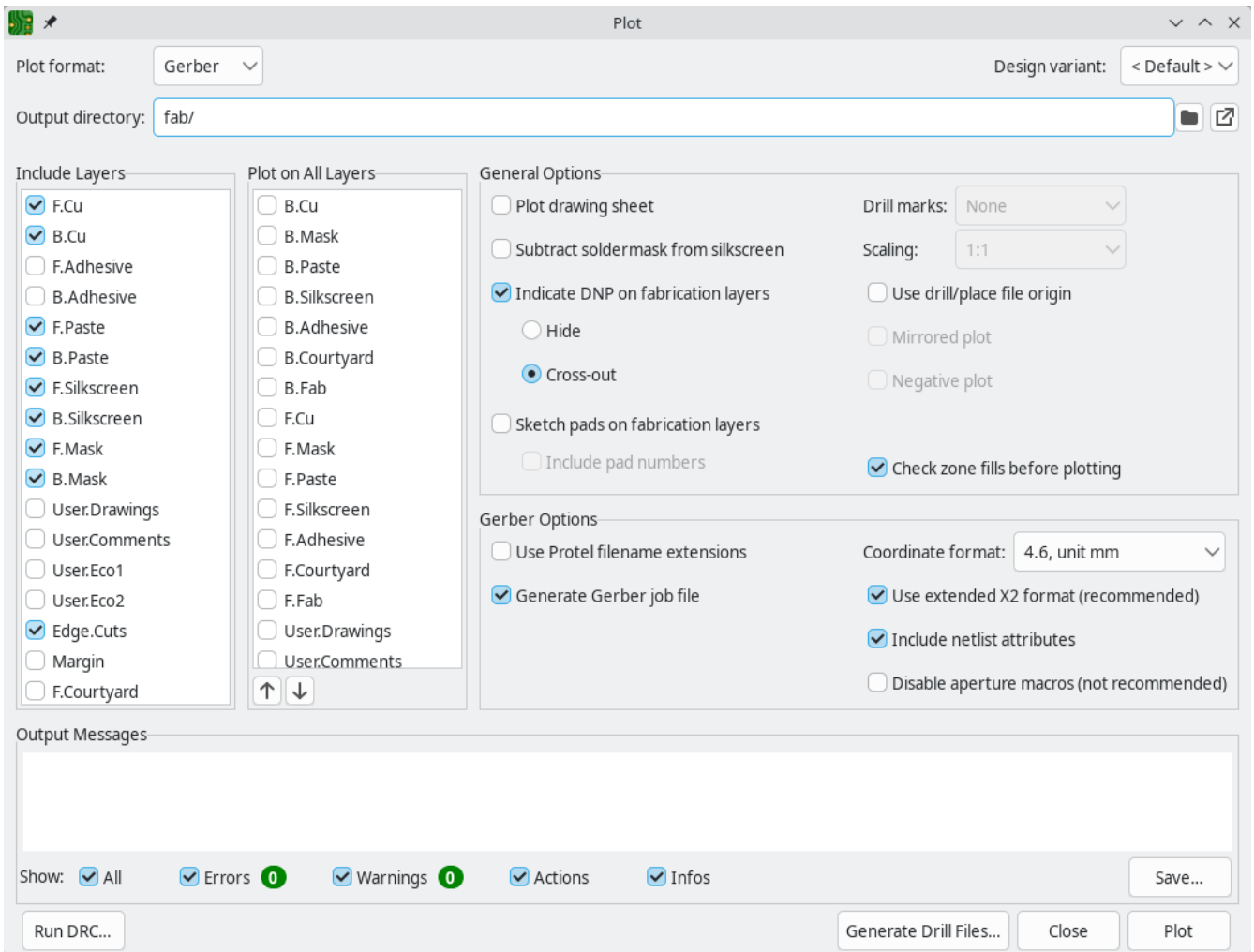
随着电路板设计的完成，最后一步是生成制造输出，这样电路板就可以被制造出来。

用 **文件** → **绘制...** 打开绘制对话框。这个对话框可以将设计绘制成几种格式，但 Gerber 通常是向 PCB 制造厂家制造的正确格式。

Specify an output directory so that the plotted files will be collected in a folder. Otherwise, the default settings are fine, but make sure all the necessary layers are checked:


- Copper layers ( \*.Cu ).
- Board outline ( Edge.Cuts ).
- Solder mask ( \*.Mask ).
- Silkscreen ( \*.Silkscreen ).
- Solder paste ( \*.Paste ). This layer is needed for manufacturing solder paste stencils, but is otherwise optional.





Other layers may be useful to plot, but are not typically necessary for PCB fabrication.



点击 **绘制** 来生成 Gerber 文件。同时点击 **生成钻孔文件...**，然后点击 **生成钻孔文件** 来创建文件，指定所有将在电路板上钻孔的位置。最后，关闭 "绘制" 对话框。设计就完成了。

Generate Drill Files

Output folder:  

Format	Options
<input checked="" type="radio"/> Excellon <ul style="list-style-type: none"><li><input type="checkbox"/> Mirror Y axis</li><li><input type="checkbox"/> Minimal header</li><li><input type="checkbox"/> PTH and NPTH in single file</li><li><input type="checkbox"/> Use alternate drill mode for oval holes</li></ul>	Origin: <input type="text" value="Absolute"/>  Units: <input type="text" value="Millimeters"/>  Zeros: <input type="text" value="Decimal format (recommended)"/>  Precision: 3:3
<input type="radio"/> Gerber X2 <ul style="list-style-type: none"><li><input type="checkbox"/> Generate tenting layers</li></ul>	
<input type="checkbox"/> Generate map: <input type="text" value="Gerber X2"/> 	

Messages

# Tutorial part 4: custom symbols and footprints

通过添加一个开关来打开和关闭 LED，该电路将得到改善。添加这个开关的过程需要创建一个新的符号和封装库，绘制一个开关符号，并为开关创建一个封装。

本指南将使用的具体开关是 [NKK M2011S3A1W03](#)，一种 SPST（单刀单置）拨动开关。许多其他开关也可以使用，但引脚编号和封装尺寸可能需要调整。

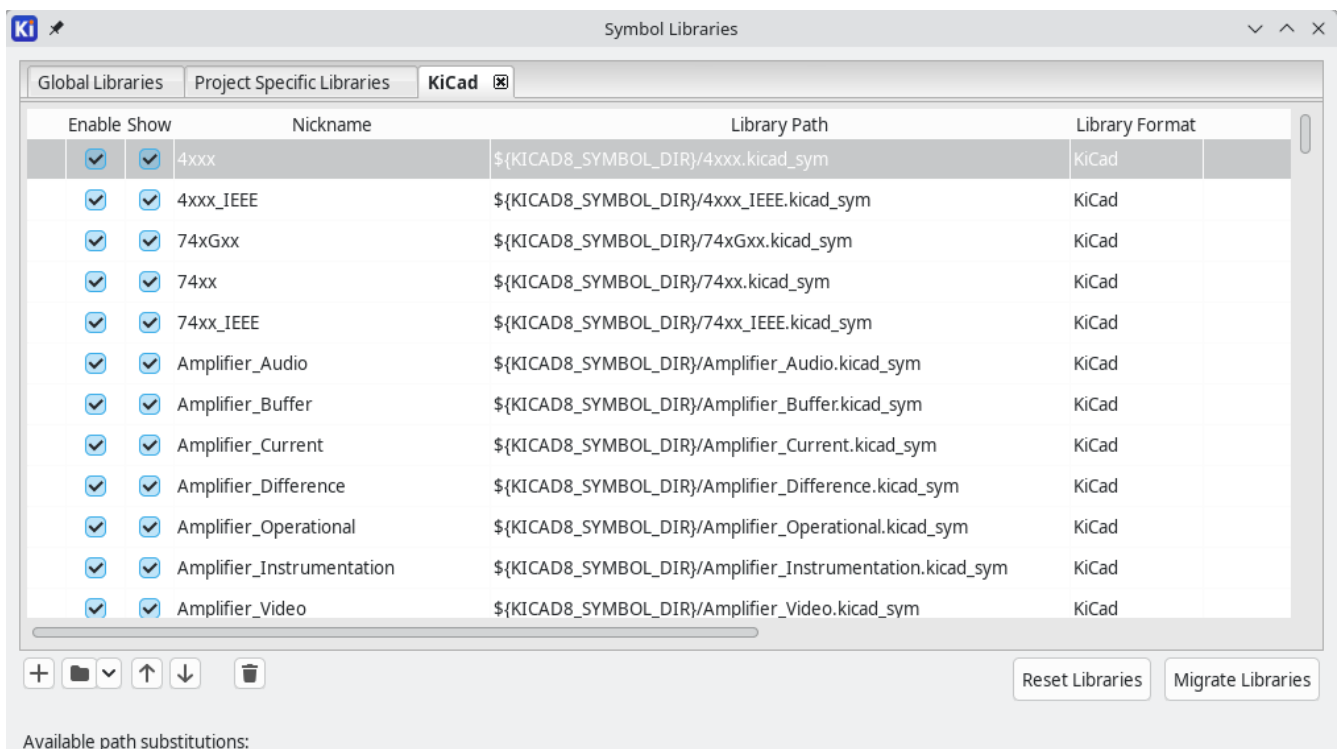
## Library and library table basics

Symbols and footprints are organized into *libraries*. A library can hold symbols or footprints, but not both.

KiCad keeps track of the user's symbol libraries and footprint libraries in the *symbol library table* and *footprint library table*, respectively. Each library table is a list of library names and the location of where each library exists on disk. Library tables can also include other library tables, which effectively adds all of the libraries in the child table to the parent table.

In addition to *global* symbol and footprint library tables, there are also *project* library tables for symbols and footprints. Symbols and footprints that are added to the global tables are available in all projects, while symbols and footprints in the project-specific tables are available only for that specific project. Users can add their own libraries to the global library tables or to project-specific tables.

符号库表可以在原理图编辑器或符号编辑器窗口中通过 **偏好设置** → **管理符号库...** 来查看或编辑。封装库表可以在电路板编辑器或封装编辑器中通过 **偏好设置** → **管理封装库...** 查看或编辑。这两个库表也可以从工程管理器中访问。



通常情况下，库的路径是通过路径替换变量定义的。这样，用户就可以在不修改库表的情况下，将所有库移动到新的位置。唯一需要更改的是重新定义变量，使其指向新位置。KiCad 的路径替换变量可通过项目管理器或任何编辑器窗口中的 **偏好设置** → **配置路径...** 进行编辑。

一个有用的路径替换变量是 `${KIPRJMOD}`。这个变量总是指向当前的工程目录，所以它可以用来包括存储在工程目录内的工程专用库。

On first run, KiCad [prompts the user to set up the symbol and footprint library tables](#). To reset your libraries to default, press the **Reset Libraries** button in the library table dialog.

## Creating new global or project libraries

绘制一个新的符号或封装的第一步是选择一个库来存储它。在本指南中，开关符号和封装将进入新的工程专用库。

1. Open the Symbol Editor from the Project Manager.
2. Click **File** → **New Library**.
3. Select the **Add new library to project library table** option in the file browser.
4. Choose a name for the new library (e.g. `getting-started.kicad_sym`).
5. Save the new symbol library in the project directory.

The empty new library is now selected in the Libraries pane at left, and has been automatically added to the project library table (check the Project Specific Libraries tab in **Preferences** → **Manage Symbol Libraries...**).


## Creating new symbols

Now create the switch symbol in the new library.

1. Select the `getting-started` library in the Libraries pane at left.
2. Click **File** → **New Symbol...**
3. In the **Symbol name** field, enter the part number: `M2011S3A1W03`.
4. Switch symbols should have reference designators that start with `SW`, so change the **Default reference designator** field to `SW`.
5. Uncheck **Show pin name text**.
6. Leave all other fields as the defaults.
7. Press **OK**.

在库面板中，`M2011S3A1W03` 符号现在出现在启动库下。在画布中，一个十字表示封装的中心，并为位号添加了文本。现在，请将文本从符号中心移开，使其不碍事。

## Symbol pins

开始绘制符号时，先添加一个引脚。由于该符号没有引脚 1，因此我们将从编号为 2 的引脚开始。点击右侧工具栏中的 **添加引脚** 按钮 。随后将弹出“引脚属性”对话框。请按以下方式设置引脚属性：

- **Pin name:** `A`
- **Pin number:** `2`
- **Electrical type:** `Passive`
- **Orientation:** `Right`

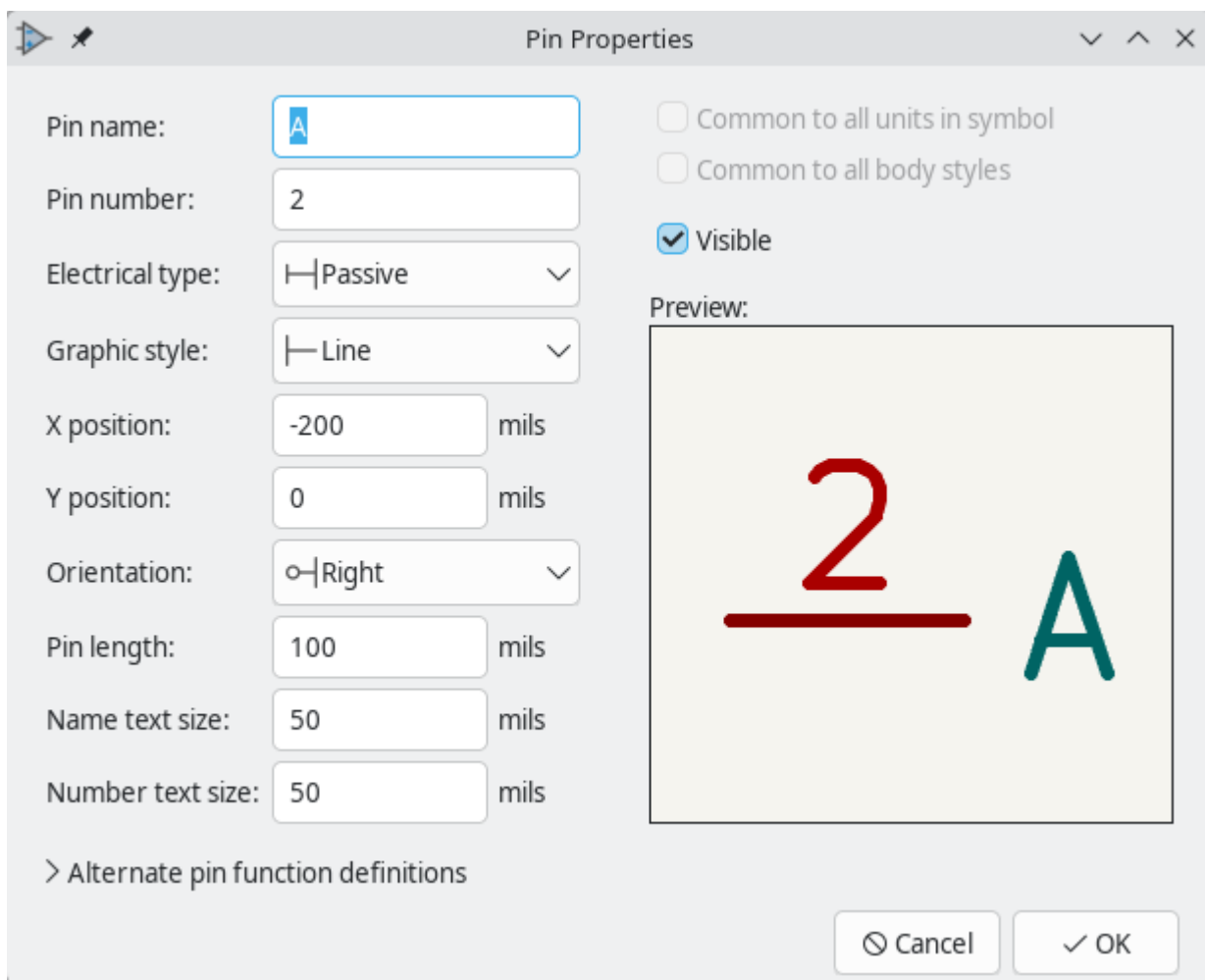
X Position: -200 mils

- Y Position: 0

点击 **确定**，然后在画布上点击以放置图钉。如果点击 **确定** 后引脚移动了，可能需要再次编辑引脚的属性（双击引脚，或右键点击引脚 → **属性...**）以正确设置位置。

#### NOTE


强烈建议在原理图编辑器和符号编辑器中始终使用 mils（或 in），尤其是符号引脚。混合使用英制和公制单位将导致符号引脚无法正确连接，因为它们无法与导线或其他引脚对齐。KiCad 库的引脚使用 50 mils 网格；使用不同的单位系统将导致符号和原理图无法与 KiCad 的内置符号连接。请注意，与 PCB 编辑器不同，原理图中的单位没有物理意义。



添加第二个引脚，但这次不要使用 **添加引脚** 工具。而是按下 **Insert**。一个新的编号为 3 的引脚被添加到符号中，正好位于引脚 2 的下方。

#### TIP

在 KiCad 的许多地方，按 **Insert** 会重复上次的操作。新项目的位置将被移位，并根据情况自动增加编号。在符号编辑器中，这可以用来快速放置大量的引脚。在原理图编辑器中，它可以用来重复放置一个元件，或者用编号标签标记一个大元件的引脚。**Insert** 在封装和电路板编辑器中也很实用。



我们希望引脚 3 位于开关符号的右侧，因此将编辑其属性以更改位置和方向。这次，我们不使用引脚属性对话框，而是利用属性面板。这是一个停靠在编辑画布左侧的面板，允许您查看和编辑所选对象的属性。要显示或隐藏属性面板，请使用左侧工具栏中的  按钮。

在选中引脚 3 的情况下，使用属性面板按如下方式设置其属性：

- **Pin Name:** B
- **Position X:** 200 mils
- **Position Y:** 0
- **Orientation:** Left

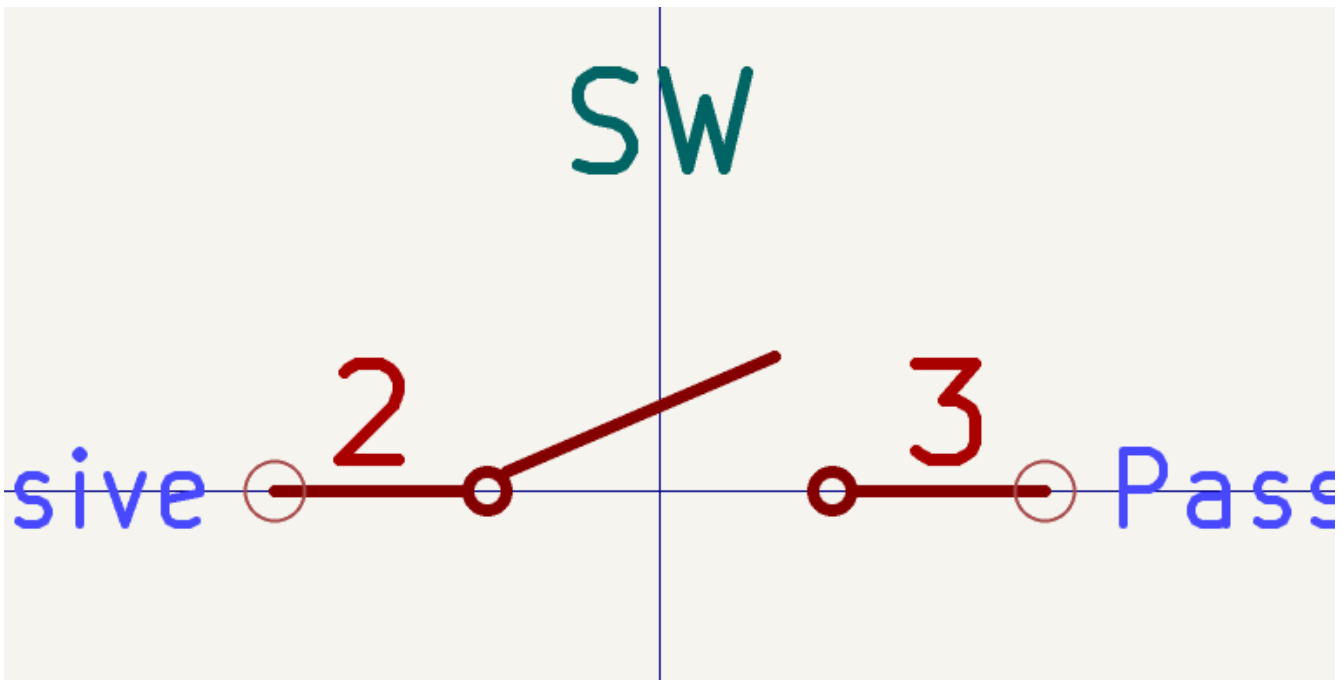
**NOTE** | 属性面板在 KiCad 的每个编辑器（原理图、PCB、符号和封装）中均可使用。

## Symbol graphics

放好引脚后，使用圆圈  和直线  工具，使符号看起来像一个 SPST（单刀单置）开关。对于这一步，切换到一个更小的网格将是有用的：右击画布，在 **网格** 子菜单中选择一个更小的网格。添加图形后，再切换回 50 mil 的网格。

### WARNING

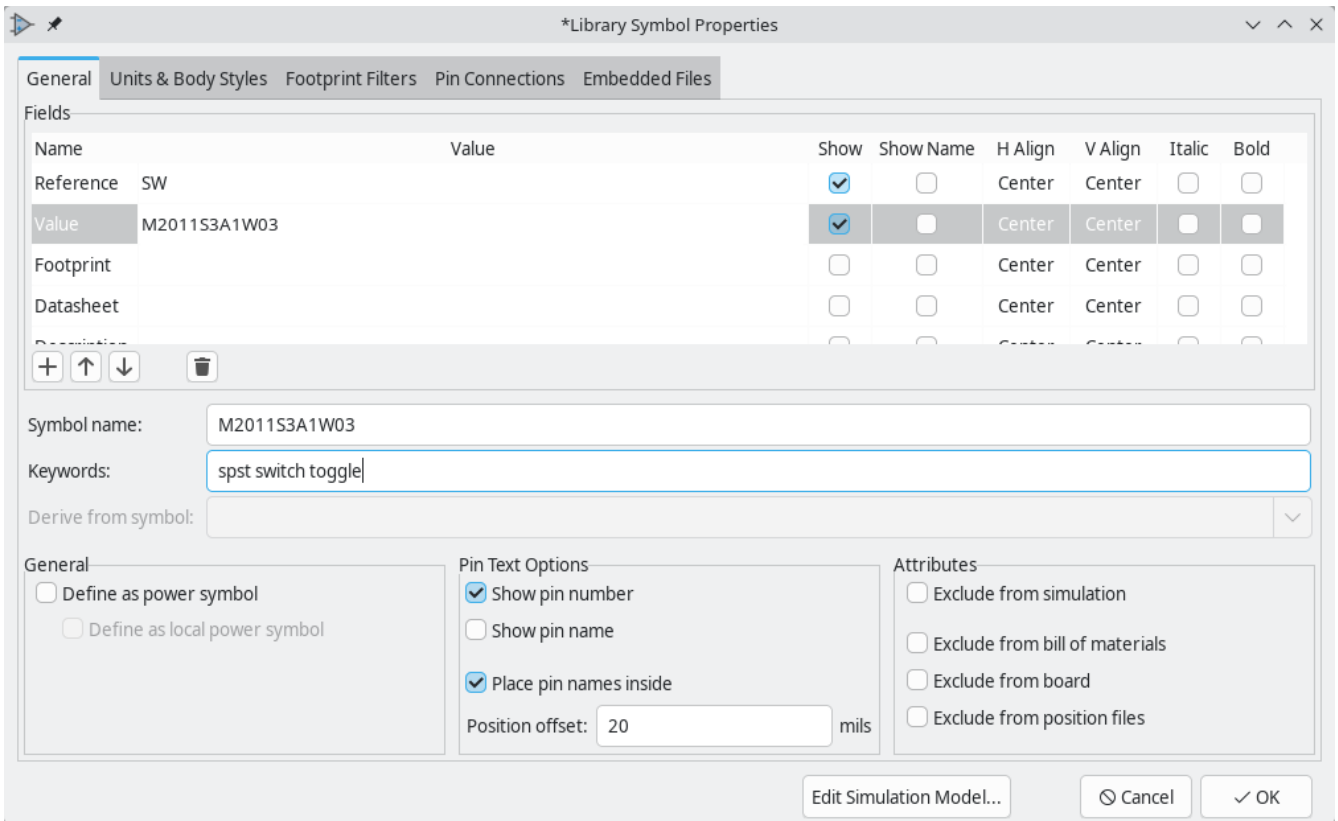
小网格对图形特征很有用，但 **符号引脚必须始终放在 50 mil（1.27 mm）的网格上**。没有对准 50 mil 网格的引脚将无法与原理图中的导线连接。



## Symbol properties

Now edit the properties for the whole symbol with **File** → **Symbol Properties**, or by double clicking on the canvas. Change the **Value** field to M2011S3A1W03, and add spst switch toggle to the **Keyword** field to make it easier to find the symbol by searching.

点击 **确定** 后，将数值字段的文本向下移动，以避免与符号图形重叠。



符号现在已经完成。保存它并继续创建一个封装。

## Creating new footprints


To create a new footprint, follow a similar process as for symbols. First you need to create a library, then create a new footprint in the library.

1. Open the Footprint Editor from the Project Manager.
2. Click **File** → **New Library...**
3. Select the **Add new library to project library table** option in the file browser.
4. Choose a name for the new library (e.g. `getting-started.pretty`)
5. Save the new footprint library in the project directory.

As with the symbol library, the new footprint library is added to the project library table. You can now add a footprint to the new library.

1. Select the `getting-started` library in the Libraries pane at left.
2. Click **File** → **New Footprint**. This creates an empty, untitled footprint.
3. Edit the footprint's properties (⚙️ button in the top toolbar) and set the following properties:
  - **Footprint name:** `Switch_Toggle_SPST_NKK_M2011S3A1x03`
  - **Value:** `Switch_Toggle_SPST_NKK_M2011S3A1x03`
  - **Component type:** Through hole


## Footprint pads

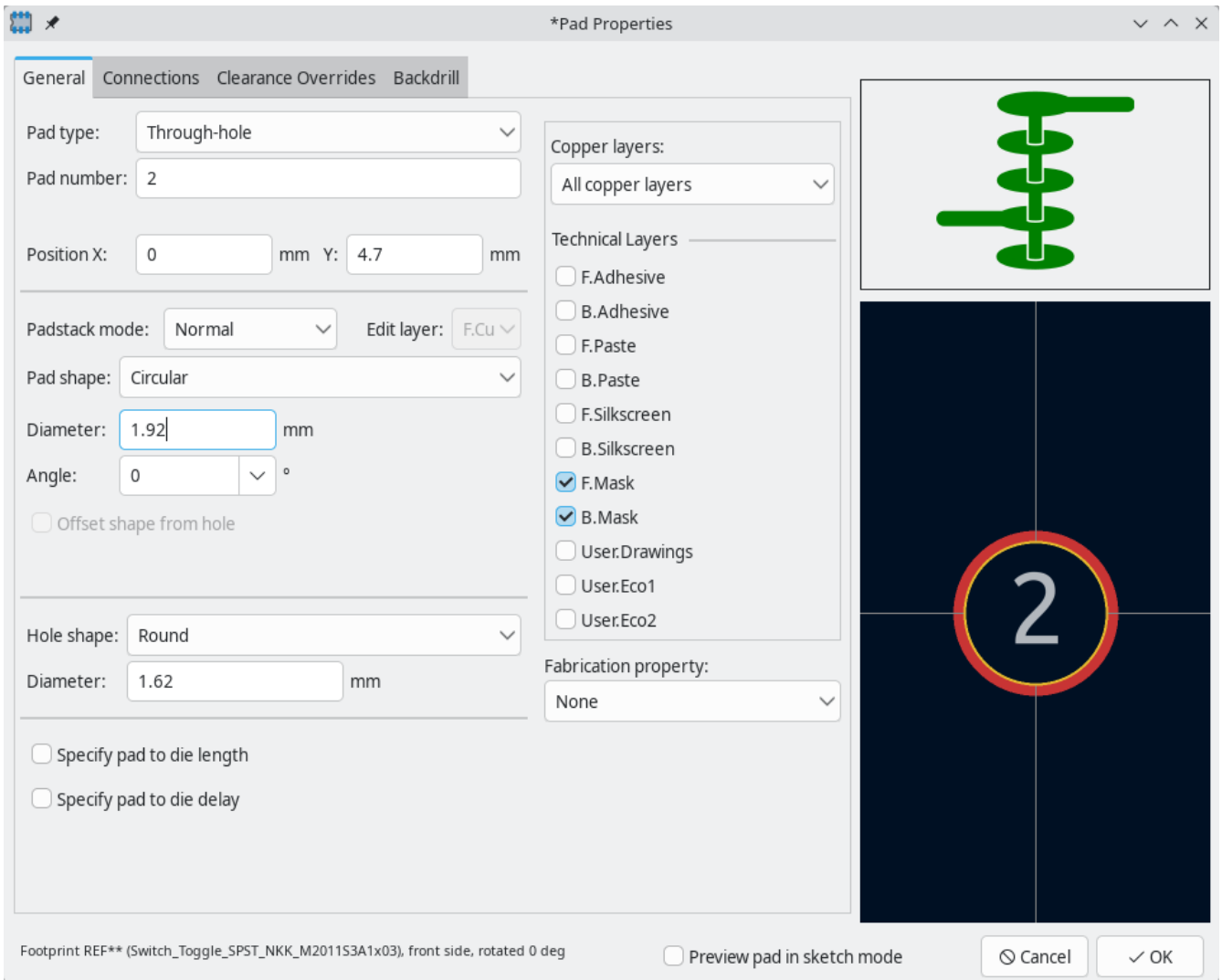
该开关有两个引脚，在数据手册中编号为 2 和 3，引脚间距为 4.7 mm。为了方便放置，请调整网格以匹配焊盘间距。通过右键单击左侧工具栏中的按钮 ，选择 **编辑网格...**，打开网格设置。点击底部的加号按钮添加一个新网格，将新网格的 X 尺寸设置为 4.7 mm。确认对话框后，返回编辑画布，如果尚未选中，请从顶部工具栏的网格下拉菜单中选择新网格。

By convention, through-hole footprints have pin 1 located at (0,0) and are oriented with pin 1 in the top left. We are using the SPST version of this switch which does not have pin 1, so the footprint will leave (0,0) empty and place pads 2 and 3 at (0, 4.7 mm) and (0, 9.4 mm). Note that in KiCad's default coordinate system, the positive Y-axis is oriented downwards.

### NOTE

光标的 X 和 Y 坐标显示在窗口底部的状态栏中。您可以在绘图或放置对象时使用此功能来检查坐标。

1. Use the **Add a pad** tool  in the right toolbar to place a pad one grid division below the origin, which is (0, 4.7 mm).
2. Press **Esc** to exit the pad tool.
3. Double click on the pad to edit its properties.
4. Change the pad number to 2.
5. Verify that the position is correct.
6. Set the pad shape to Circular.
7. Set the hole diameter. The switch pins are 1.17 mm x 0.8 mm, which gives a diagonal (maximum pin dimension) of 1.42 mm. Therefore set the hole diameter to  $1.42 \text{ mm} + 0.2 \text{ mm} = 1.62 \text{ mm}$ .
8. Set the pad diameter. For a hole diameter of 1.62 mm, set the pad diameter to  $1.62 \text{ mm} + 2 * 0.15 \text{ mm} = 1.92 \text{ mm}$  to provide a sufficient annular ring.
9. Press **OK** to apply the pad modifications.



现在再次使用 **添加焊盘** 工具，将另一个焊盘放置在 (0, 9.4 mm) 的位置。请注意，焊盘编号会自动递增，并且包括焊盘尺寸和孔径在内的属性会从上一个焊盘复制过来。

在放置了两个焊盘后，圆环形环看起来略小。如果将圆环形环做大，开关将更容易焊接，机械上也更坚固。通过编辑焊盘 2 并将焊盘尺寸改为  $1.62 \text{ mm} + 2 * 0.3 \text{ mm} = 2.22 \text{ mm}$ ，将圆环形环的厚度从 0.15 mm 增加到 0.3 mm。不要改变通孔的尺寸。请注意，焊盘尺寸字段接受数学表达式，所以  $1.62 + 2 * 0.3$  可以直接输入，并将评估为 2.22 mm。


#### TIP

KiCad 中的许多文本框支持数学表达，包括单位转换。

对另一个焊盘也做同样的圆环形圈修改。作为一个快捷方式，在焊盘 2 上点击右键，点击 **将焊盘属性推到其它焊盘上...**，然后点击 **改变当前封装上的焊盘**。




## Footprint graphics

一个好的封装会在制造层 ( F.Fab ) 上绘制精确的零件轮廓，在丝印层 ( F.Silkscreen ) 上绘制稍大的轮廓，并在整个封装周围设置一个禁布区 ( F.Courtyard )，以防止与其他封装发生重叠。





The fabrication outline should precisely match the physical dimensions of the part, which is 7.9 mm wide and 13 mm tall. One way to precisely place the outline is to create a new grid with the correct X and Y spacing, as we did for the pads. Make sure to uncheck the **Linked** checkbox in the Grid Settings dialog to allow unequal X and Y grid spacing. You can then adjust the grid origin by right clicking the  button and selecting **Grid Origin....**

按以下方式配置网格间距和原点：

- X 网格：7.9 mm
- Y 网格：13 mm
- X 原点：7.9 mm / 2 = 3.95 mm
- Y 原点：4.7 mm - (13 mm / 2) = -1.8 mm。

To draw the fab outline, switch to the front fabrication layer by clicking **F.Fab** in the Layers panel at right. Use the line , rectangle , or polygon  tools to draw the outline of the part as shown in the screenshot below, with corner coordinates annotated.

**NOTE**

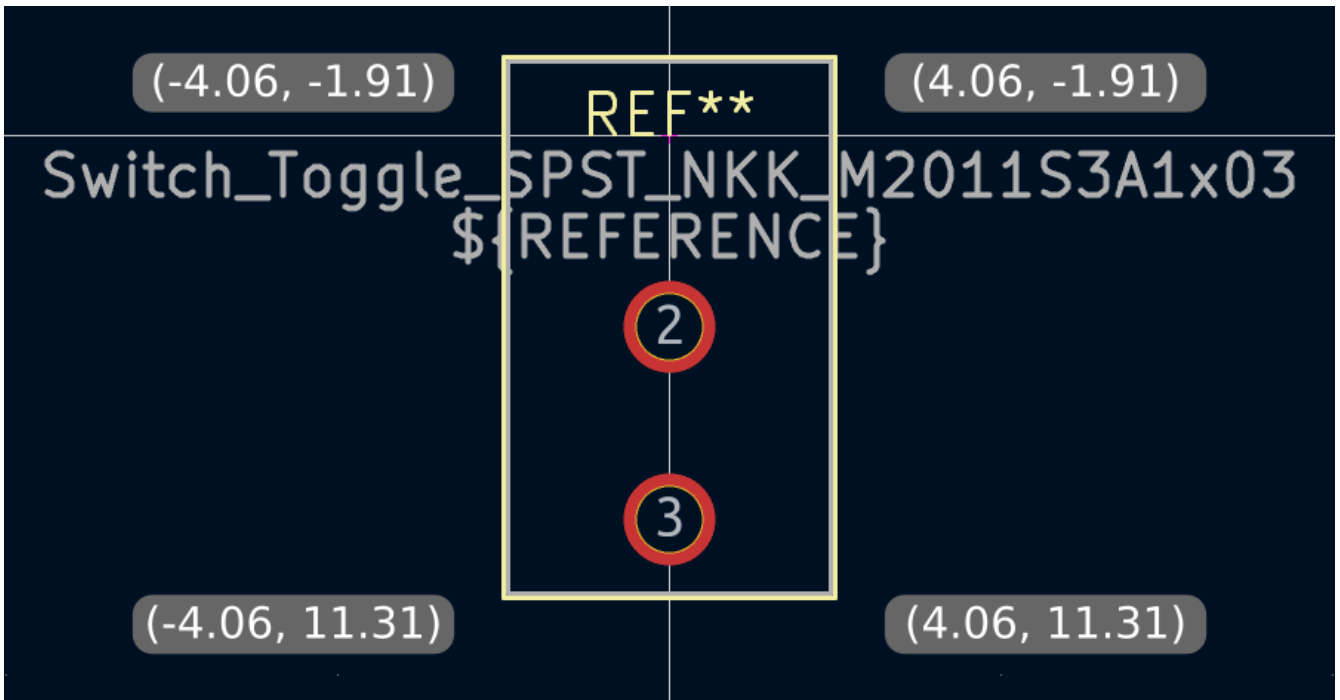
If you use the rectangle tool, you will need to make sure the *line mode* is set to the free-angle mode or you will only be able to draw squares. To change the line mode to free-angle, click the  /  /  button on the left toolbar until it shows . You can also use **Shift** + **Space** to cycle the line mode.



接下来，切换到 **F.Silkscreen** 层。我们希望丝印轮廓刚好位于元件轮廓的外侧，因此丝印线将绘制在制造层线条外侧 0.11 mm 处（0.11 mm 是 0.10 mm 制造线宽的一半，加上 0.12 mm 丝印线宽的一半）。具体坐标如下方截图所示，辅助网格设置为：

- X 网格：7.9 mm + 2 \* 0.11 mm = 8.12 mm
- Y 网格：13 mm + 2 \* 0.11 mm = 13.22 mm
- X 原点：8.12 mm / 2 = 4.06 mm
- Y 原点：-1.8 mm - 0.11 mm = -1.91 mm

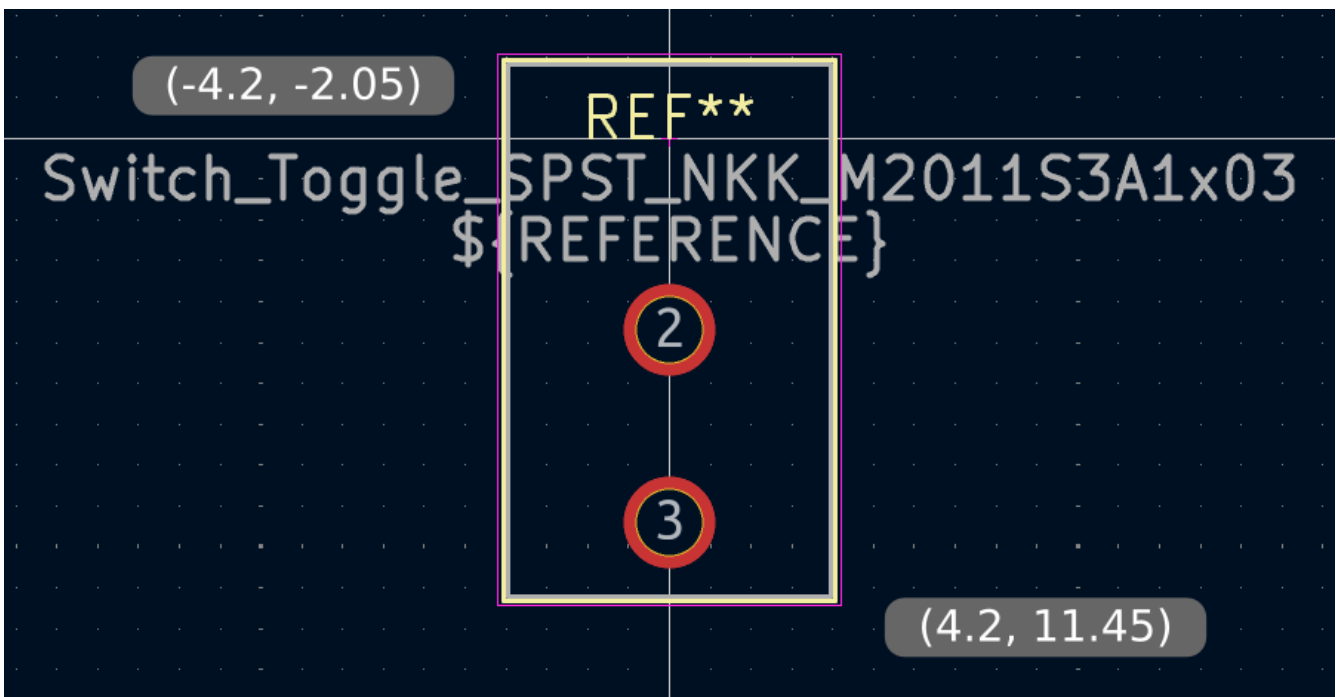
Also change the silk outline thickness to 0.12 mm by selecting each outline segment and editing its properties (right click → **Properties...** or press **E**).



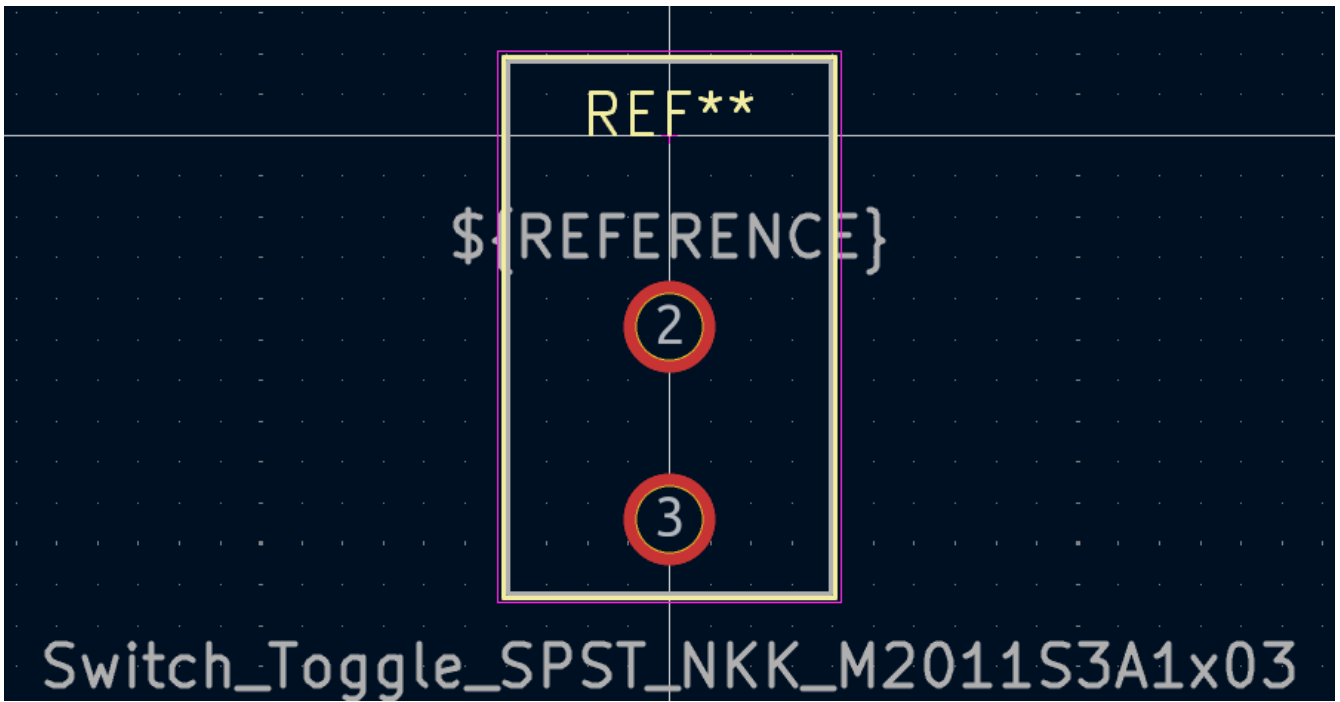
最后，选择 F.Courtyard 图层。我们希望边框的轮廓线围绕该部分，并留有 0.25 mm 的间隙。

为了多样化，我们可以采用不同的策略在这一层上绘制形状。将自定义网格切换为 1 mm 的网格，并大致围绕轮廓绘制一个矩形。选中矩形，按下键盘上的 **E** 键以编辑其属性，直接输入矩形的角坐标。角坐标如下：

- 左上角：(-4.2, -2.05)
- 右下角：(4.2, 11.45)



完成边框后，将文字定位如下图所示。封装就完成了。




## Kicad 库公约

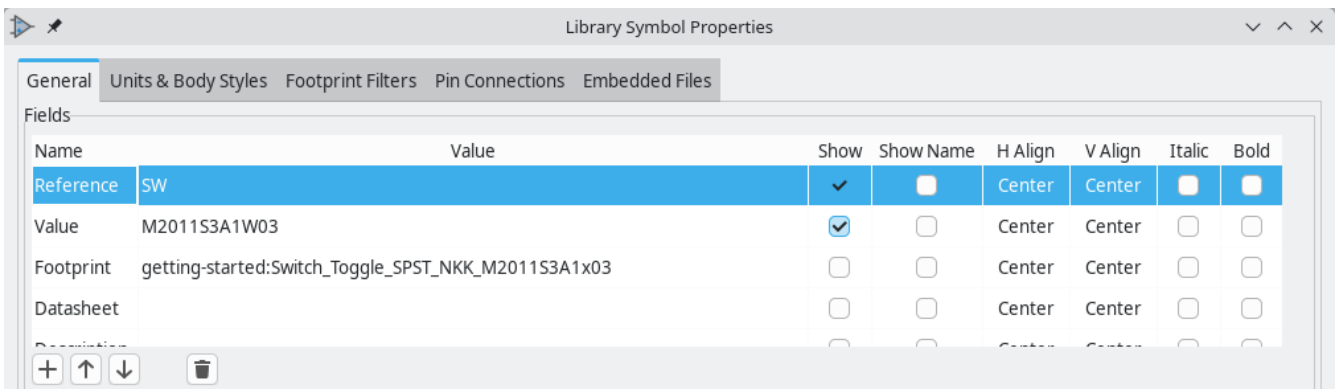
为了维护高质量的符号和封装库，KiCad 有一个 [KiCad 库公约](#) 文件，这是一套符号和封装的准则。个人库不一定要遵循这些约定，但它们是一个很好的出发点。官方库中的封装和符号都需要遵循 KLC。KLC 被用作本指南中的符号和封装的基础。

## Add switch to schematic

现在，封装已经完成，可以修改开关符号，以便默认为其使用匹配的封装。

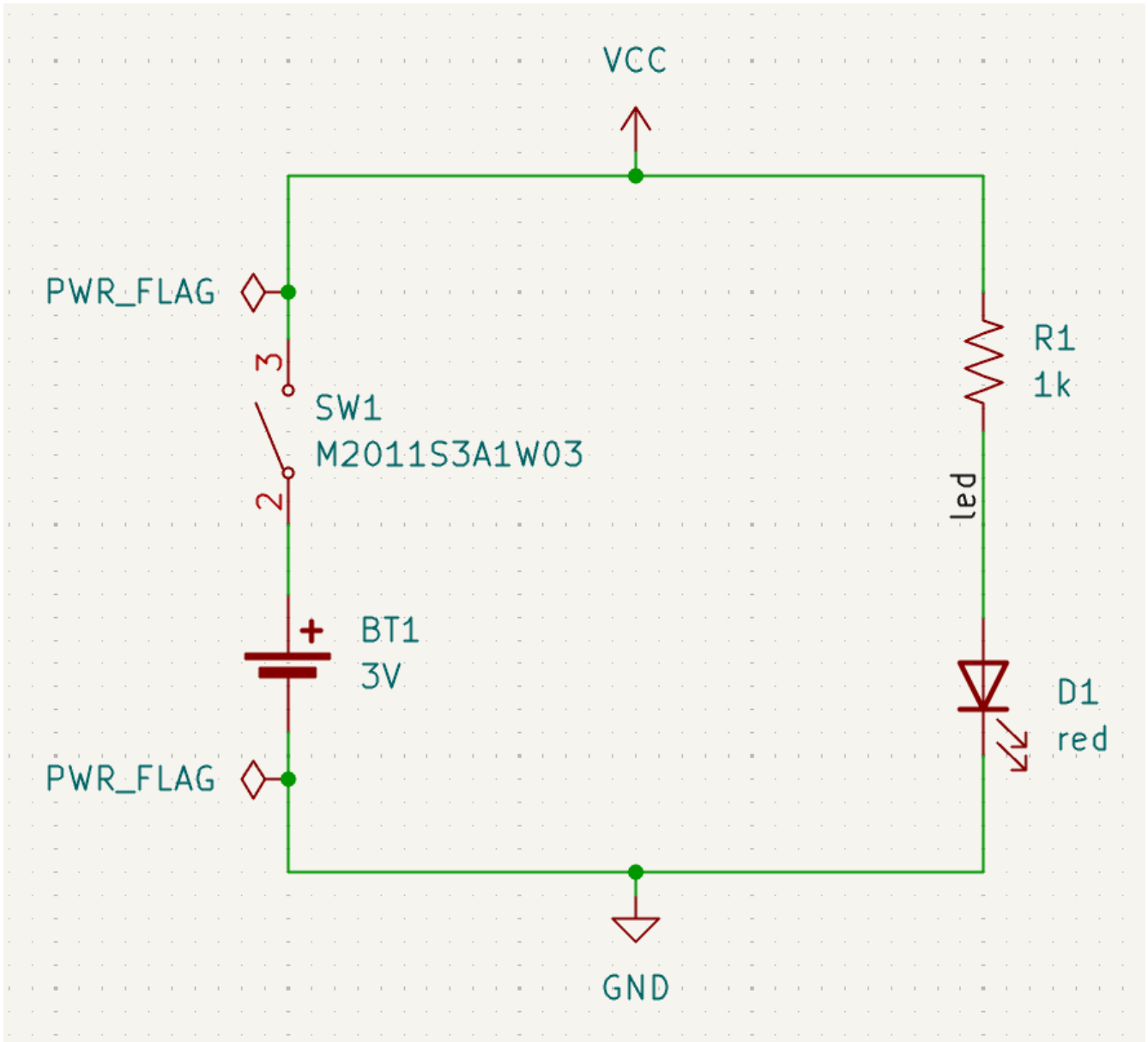
回到符号编辑器中，打开开关符号。编辑符号属性。点击 封装 字段，然后点击出现的库图标 。浏览到工程封装库，双击开关封装。保存该符号。

现在，开关封装被默认分配给该符号；每次将该符号添加到原理图中时，都不需要手动选择封装。



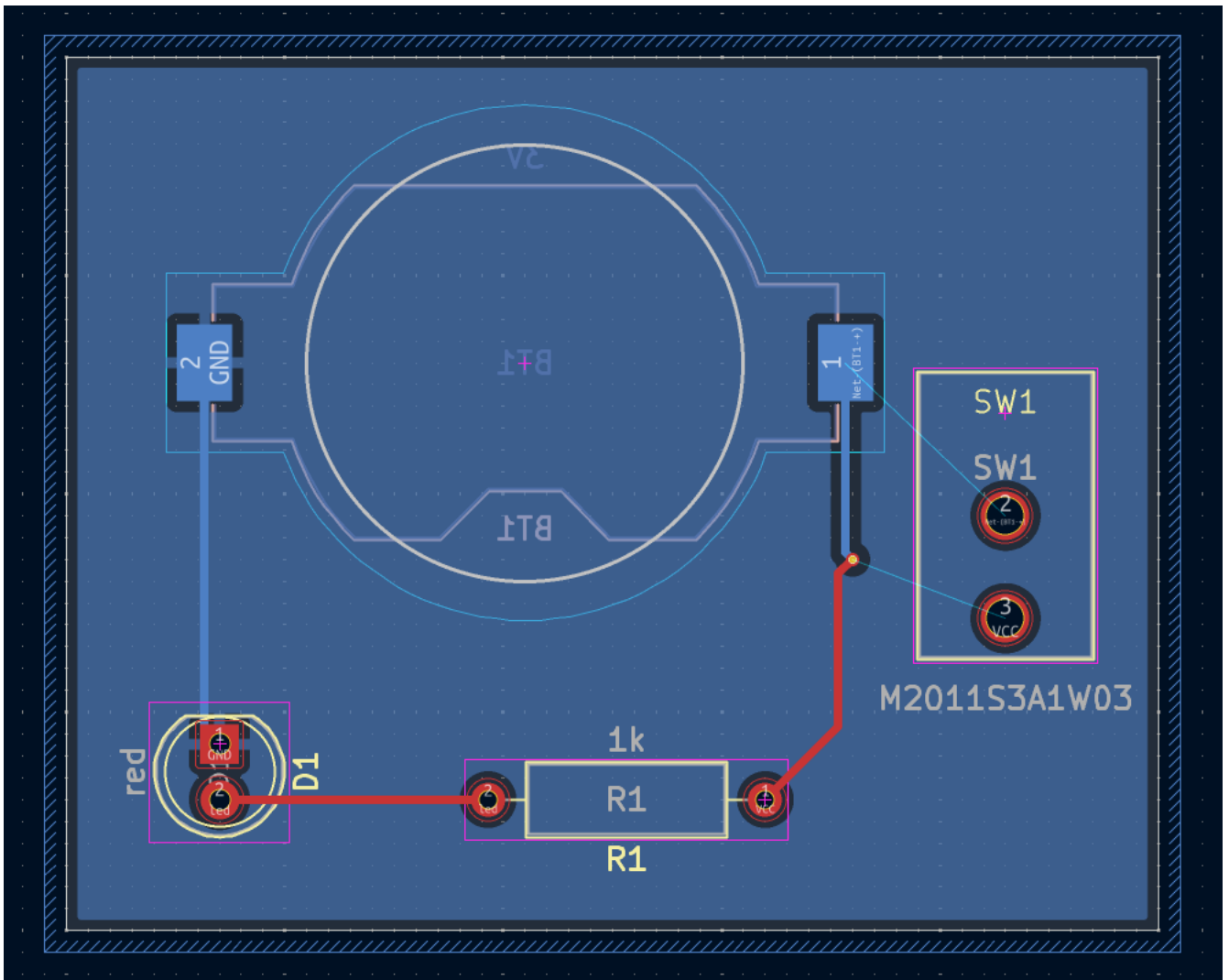
打开原理图，添加一个新符号，并选择新的开关符号。给它接线以连接或断开 LED 和电池。

无需手动为开关选择封装，因为开关符号已经指定了封装。运行 ERC，确保修改后的原理图不违反任何电气规则。



## Add switch to layout

确保原理图已保存，然后打开板编辑器，将符号添加到布局中。使用 **工具** → **从原理图更新PCB...** 将原理图的更改更新到 PCB 上，并按照图示将开关封装放置到板上。如有需要，调整板框和铜区的大小（调整铜区大小时，铜区会取消填充，但你可以根据需要按 **B** 重新填充）。

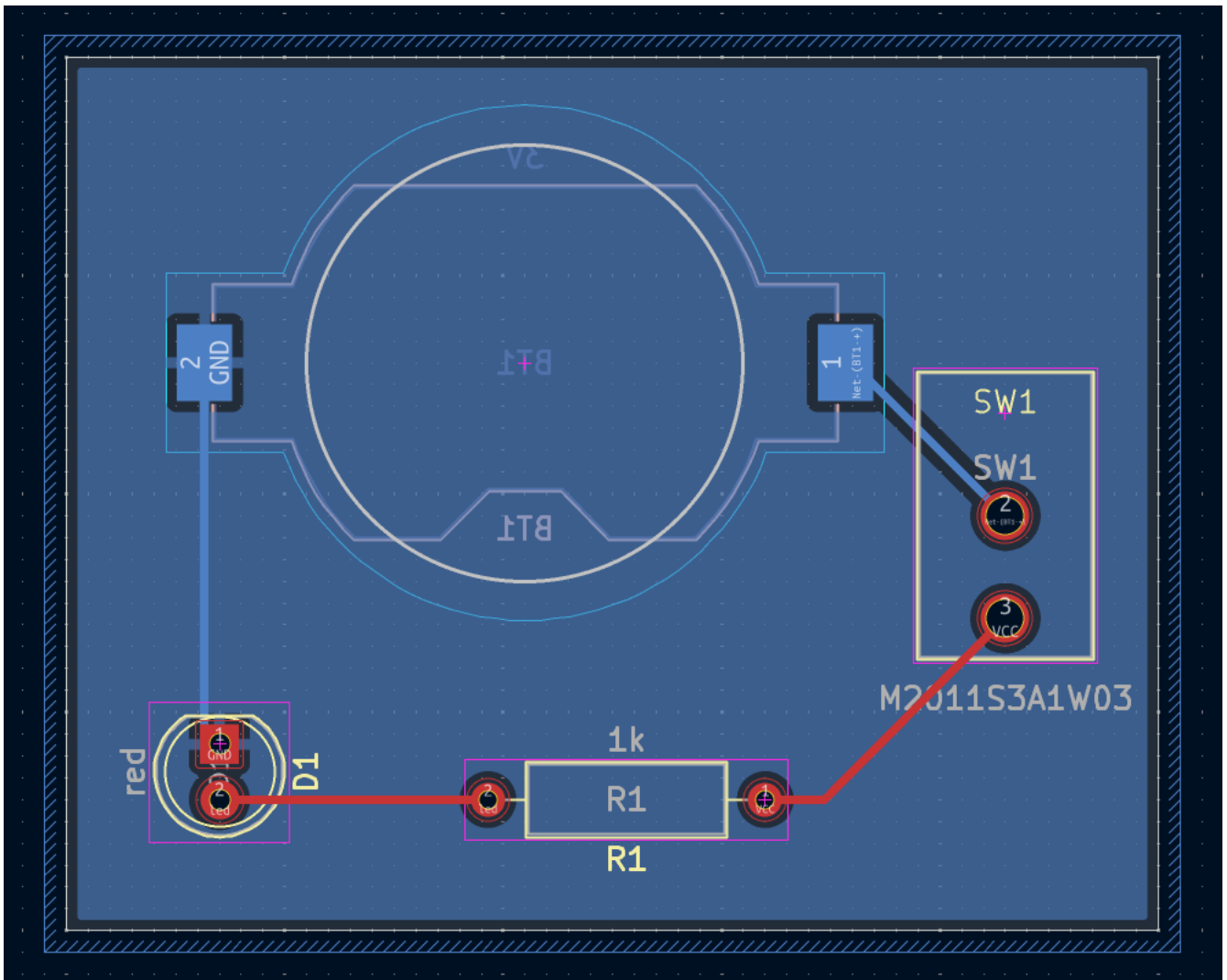


飞线显示了需要布线的新连接。此外，电池和电阻器之间的连接也需要删除。

首先，删除不需要的导线。在电池和电阻器之间选择其中一条导线——选择哪一条都无关紧要。多次按下 **U** 键，以扩展选择范围，使其包含电池和电阻器之间的所有线段。按下 **Delete** 键，移除该连接。

将新布线布置在电池与开关之间，以及开关与电阻器之间。按下 **B** 键以重新填充敷铜。

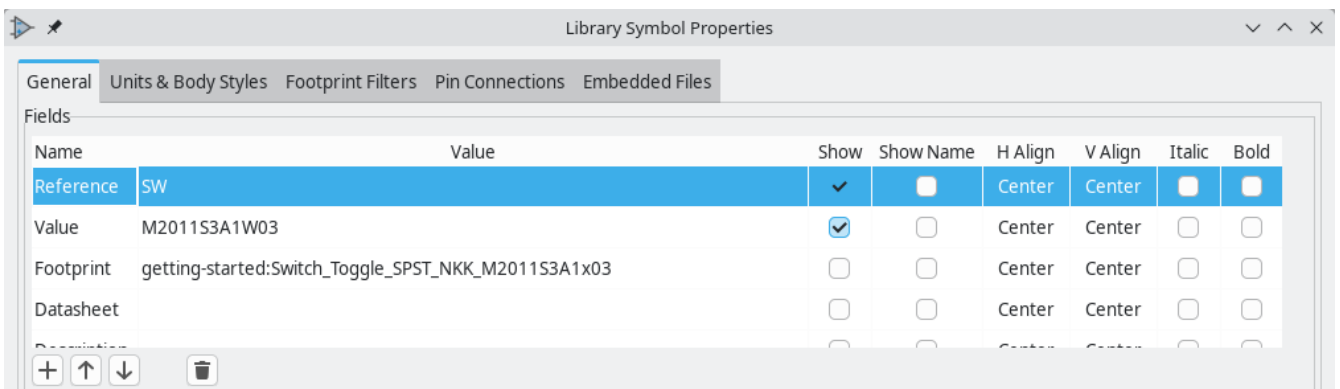
最后，重新运行 DRC 以确保修改后的电路板不违反任何设计规则。



## Linking symbols, footprints, and 3D models

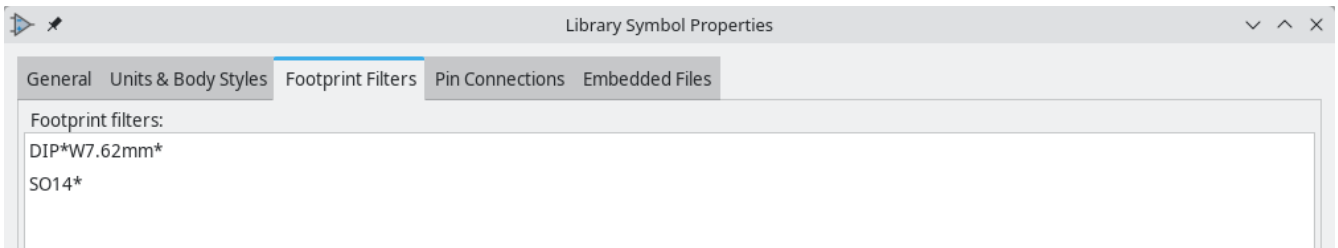
### Symbols and footprints

正如 [封装分配章节](#) 中所述，在原理图输入过程中，原理图中的每个符号都需要为其分配一个封装。分配给每个符号的封装名称被存储在符号属性的 `封装` 字段中。



符号可以指定一个预选的封装。封装不需要手动分配给这些符号，因为在创建符号的时候就已经选择了一个封装。用户可以在封装分配过程中正常覆盖预选的封装。定义一个默认的封装对于那些通常或总是被分配相同封装的符号来说是个好主意，例如一个只在一个包中可用的元件。开关封装 [被设置为其匹配符号的默认封装章节](#)

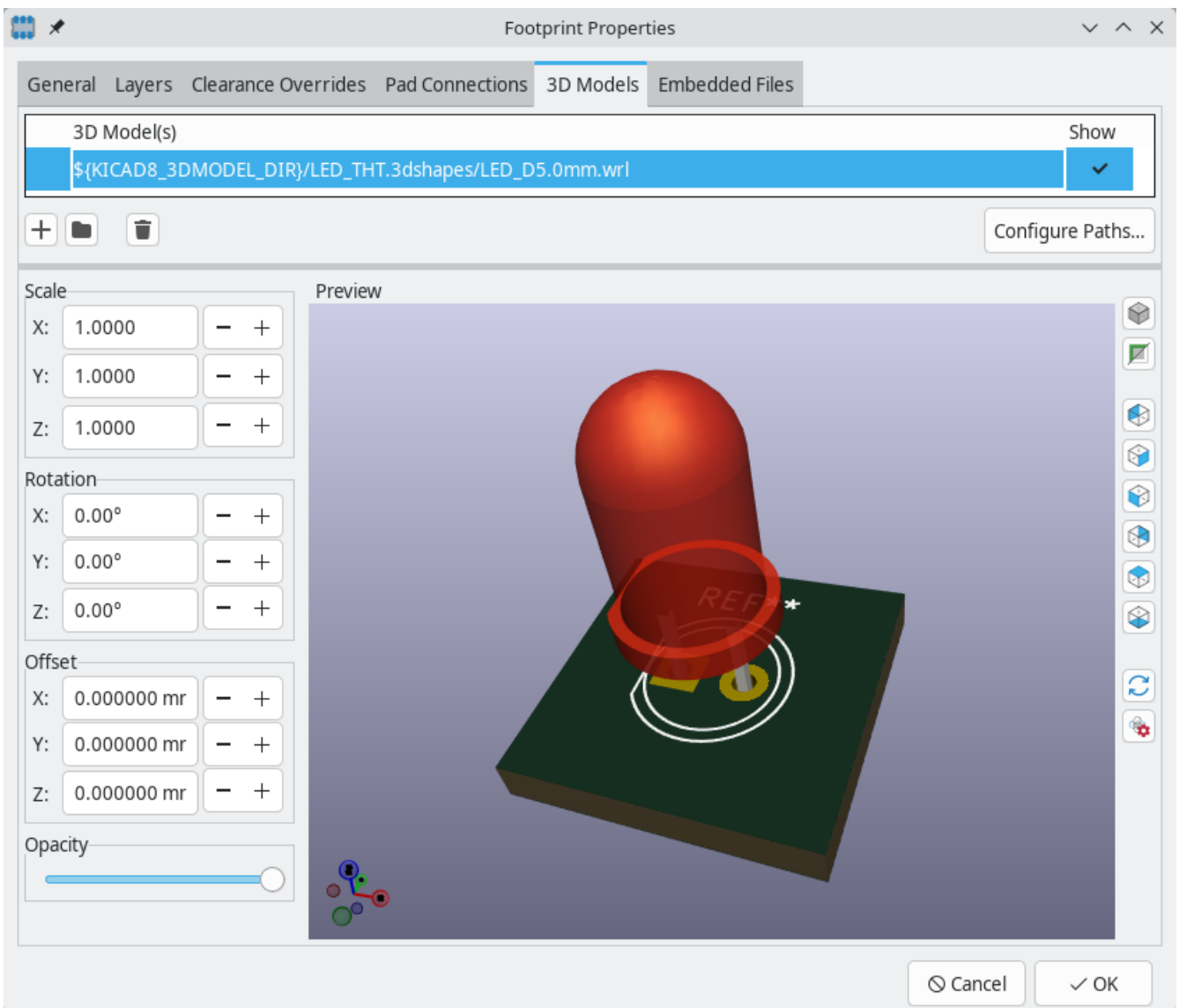
符号也可以指定封装筛选器，可以用来隐藏与符号不兼容的封装。例如，`74HC00` 符号有封装过滤器，导致只有适用的 DIP 和 SO14 封装显示在 [分配封装工具章节](#) 中。



这个 [KLC 对封装筛选器的要求](#) 包含一些有效的封装筛选器的提示。

## Footprints and 3D models

元件的 3D 模型存储在单独的文件中。元件的 3D 模型文件名保存在封装中。每个封装可以添加任意数量的 3D 模型。3D 模型文件名以及模型的缩放、旋转、偏移和透明度均在封装属性的 **3D 模型** 选项卡中设置。



Both STEP (.step) and VRML (.wrl) 3D model formats are supported. STEP files are useful where dimensional accuracy is needed, while VRML files can be used for more visually attractive renders. Many footprints in KiCad's library have associated 3D models; these models are provided in STEP format.

### NOTE

并非所有 KiCad 库中的封装都提供了 3D 模型，但所有封装都列出了 3D 模型的文件名，即使 3D 模型不存在。这样做是为了以后可以添加 3D 模型，而不需要编辑封装。

[FreeCAD](#) together with the [StepUp Workbench](#) are useful for creating component 3D models; they are used for many of the models in KiCad's library. StepUp is used to generate STEP and VRML files with correct placement, scaling, and rotation.

# Where to go from here

## More learning resources

关于如何使用 KiCad 的更多信息，请参阅 [手册](#)。

其他资源包括 [KiCad 官方用户论坛](#)，[https://www.kicad.org/community/chat/\[Discord 或 IRC\]](https://www.kicad.org/community/chat/[Discord或IRC])，以及来自 KiCad 社区的其他 [学习资源](#)。

要想了解 KiCad 的更多功能，请浏览网站的 [用 KiCad 设计](#) 部分，或打开 KiCad 附带的演示工程（[文件](#) → [打开演示工程...](#)）。

## Help improve KiCad

要报告一个错误或要求一个功能，请使用 [帮助](#) → [报告一个错误](#) 或在 [Gitlab](#) 上打开一个问题。

要为 KiCad 的发展做出贡献，请参见 [开发者贡献页面](#)。用户也可以通过为 [库](#) 或 [文档和翻译](#) 提供帮助。最后，请考虑 [捐赠支持](#) 继续开发 KiCad。