

KiCad

The KiCad Team

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

1	&#x7b80;&#x4eb;	1
1.1	系统要求	1
1.2	KiCad 文件和文件夹	1
2	&#x5b89;&#x88c5;&#x548c;&#x5347;&#x7ea7; KiCad	4
2.1	导入设置	4
2.2	从早期版本迁移文件	5
3	&#x4f7f;&#x7528; KiCad &#x5de5;&#x7a0b;&#x7ba1;&#x7406;&#x5668;	6
3.1	工程管理器窗口	6
3.2	工程树视图	7
3.3	侧面工具条	8
3.4	创建一个新工程	8
3.5	从其他 EDA 工具导入工程	9
4	KiCad &#x914d;&#x7f6e;	10
4.1	共同偏好设置	10
4.2	鼠标和触摸板偏好设置	12
4.3	快捷键偏好设置	14
4.4	路径配置	15
4.4.1	Advanced environment variables	16
4.5	库配置	16
5	&#x5de5;&#x7a0b;&#x6a21;&#x677f;	17
5.1	使用模板	17
5.2	模板位置：	18
5.3	创建模板	18
5.3.1	模板示例	20
5.3.2	所需文件：	20
5.3.3	可选文件：	21
6	&#x63d2;&#x4ef6;&#x548c;&#x5185;&#x5bb9;&#x7ba1;&#x7406;&#x5668;	22
7	Actions reference	23
7.1	KiCad Project Manager	23

参考手册

版权

本文件的版权 © 2010-2021 由下列贡献者
GNU 通用公共许可证（http://www.gnu.org/licenses/gpl.html）&#x
3 版或更高版本，或知识共享署&#x
3.0 版或更高版本的条款发布它和/

本指南中的所有商标均属于其&#x

贡献者

Jean-Pierre Charras, Fabrizio Tappero, Jon Evans.

翻译人员

taotieren <admin@taotieren.com>, 2019, 2020, 2021.

Telegram 简体中文交流群: https://t.me/KiCad_zh_CN

反馈

KiCad 项目欢迎与本软件或其文档相
关于如何提交反馈意见或报告	
<https://www.kicad.org/help/report-an-issue/> 的说明

Chapter 1

简介

KiCad 是一款用于创建电子电路原理
 KiCad 支持综合设计工作流程，其中
 PCB 一起设计，也支持特殊用途的
 KiCad 还包括一些帮助电路和 PCB 设计&#x
 PCB 计算器、用于检查制造文件的
 Gerber 浏览器和用于检查电路行为ݨ
 SPICE 模拟器。

KiCad 可在所有主要的操作系统和广
 它支持多达 32 个铜层的 PCB，适合R
 KiCad 是由世界各地的软件和电气工
 该文件的最新版本可在 <https://docs.kicad.org> 。

1.1 系统要求

KiCad 能够在多种硬件和操作系统上&#x
 1920x1080 或更高分辨率的专用显卡和&#x
 有关最新的系统要求，请查看
 KiCad 网站：<https://kicad.org/help/system-requirements/>

1.2 KiCad 文件和文件夹

KiCad 创建并使用具有以下特定文件
 工程管理器文件：

*.kicad_pro	工程文件，包含原理图T PCB 之间共享的设置
*.pro	旧版 (KiCad 5.x 和更早版本) 工程文件。 可以读取，并将由工程{ .kicad_pro 文件。

原理图编辑器文件：

*.kicad_sch	包含所有信息和元件本
-------------	--

*.kicad_sym	原理图符号库文件，包T
*.sch	旧版 (KiCad 5.x 和更早版本) 原理图文件。 可以读取，并将在写入e
*.lib	旧版 (KiCad 5.x 和更早版本) 原理图库文件。可以读S
*.dcm	旧版 (KiCad 5.x 和更早版本) 原理图库文档。可以读S
*_cache.lib	旧版 (KiCad 5.x 和更早版本) 原理图元件库缓存文件0 正确加载旧版原理图 (.sch) 文件所需。
sym-lib-table	符号库列表 (符号库表)： 原理图编辑器中可用的{

板编辑器文件和文件夹：

*.kicad_pcb	包含除板框以外的所有含
*.pretty	封装库文件夹。 文件夹本身就是库。
*.kicad_mod	封装文件，每个文件包含某个
*.kicad_dru	设计规则文件，包含某个 .kicad_pcb 文件的自定义设计规则本
*.brd	旧版 (KiCad 4.x 和更早版本) 电路板文件。 当前电路板编辑器可以设含
*.mod	旧版 (KiCad 4.x 和更早版本) 封装库文件。 可由封装或线路板编辑器中可用的封装库的加设含
fp-lib-table	封装库列表（ 封装库表 ）： 线路板编辑器中可用的封装库的加设含
fp-info-cache	缓存以加速封装库的加设含

＊常用文件：,*

*.kicad_wks	图框页面布局 (图形边框和标题栏) 描述文件
*.net	原理图创建的网表文件，此文件与 .cmp 文件相关联，适用于偏r
*.kicad_prl	当前工程的本地设置，^ KiCad 记住上次使用的设置， 如图层可见性或选择过 可能不需要与工程一起

其他文件：

*.cmp	原理图中使用的元件与元它可以由 Pcbnew 创建并由 Eeschema 导入。 其目的是为用户导入从 Pcbnew 到 Eeschema 的更改 谁更改 Pcbnew 内的封装（例如使用 改变封装 命令） 并希望在原理图中导入
-------	---

其他文件：

它们由 KiCad 生成用于制作或文档。

*.gbr	Gerber 文件，用于制作。
*.drl	钻孔文件（Excellon 格式），用于制作。
*.pos	位置文件（ASCII 格式），用于自动插入某些文件 (如工程本地设置文件 (.kicad_prl) 和 fp-info-cache 文件) 不需要随工程一赿 Git 之类的版本控制系统来跟踪 Ki-Cad 工程，则可能需要将这些文件。
*.rpt	报告文件（ASCII 格式），用于文档。
*.ps	绘图文件（Postscript），用于文档。
*.pdf	绘图文件（PDF 格式），用于文档。
*.svg	绘图文件（SVG 格式），用于文档。
*.dxf	绘图文件（DXF 格式），用于文档。
*.plt	绘图文件（HPGL 格式），用于文档。

存储和发送 KiCad 文件

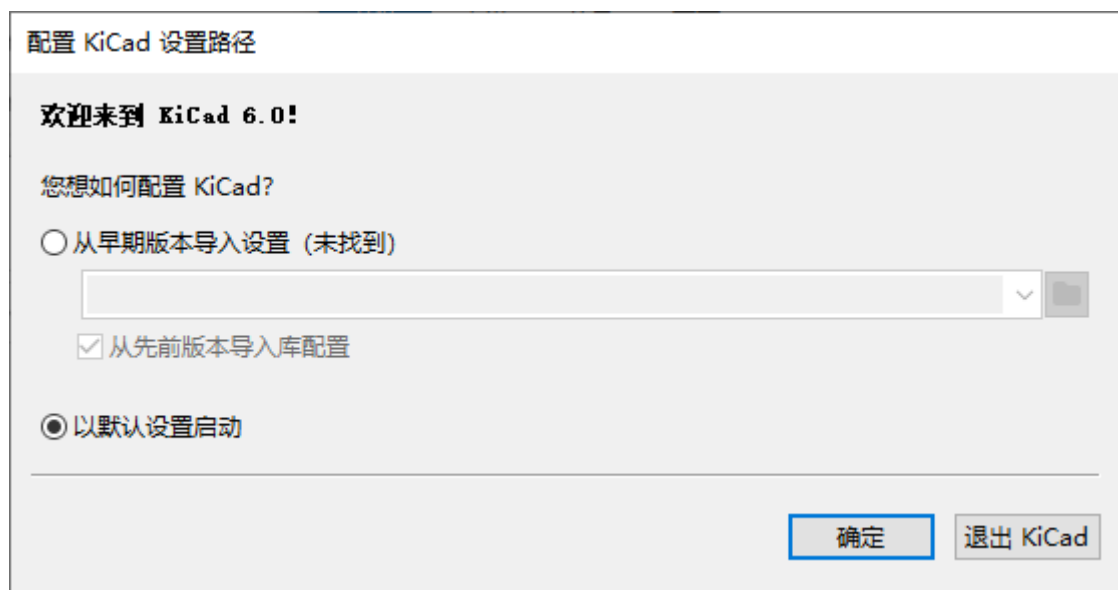
KiCad 原理图和电路板文件包含设计 (.kicad_pro) 中，因此，如果要发送完整某些文件 (如工程本地设置文件 (.kicad_prl) 和 fp-info-cache 文件) 不需要随工程一赿 Git 之类的版本控制系统来跟踪 Ki-Cad 工程，则可能需要将这些文件。

Chapter 2

Καδ

2.1 \mathbb{R}^n and \mathbb{C}^n

KiCad 的每个主要版本都有自己的配版本，而不会干扰配置。首次本有版本，系统将询问如何初始化设



如果检测到以前版本的 KiCad，您Ծ

Please note that, the schematic symbol and footprint library tables from the previous version of KiCad will **not** be imported.

如果您不想从以前的版本导入

KiCad 将设置文件存储在用户目录内
KiCad 版本都会将其设置存储在该文
(KiCad 5.1 及更早版本除外，它不使用内

Windows	%APPDATA%\kicad
Linux	~/.config/kicad
Mac OS	/Users/<username>/Library/Preferences/kicad

2.2 从早期版本迁移文件

现代版本的 KiCad 可以打开在早期版 KiCad 版本更改到下一个版本。打开一般来说，一个新版本的 KiCad 创建< b>不能< /b> 被旧版本的 KiCad 打开。因此， KiCad 版本时保留项目的备份副本非 KiCad 版本。

Note

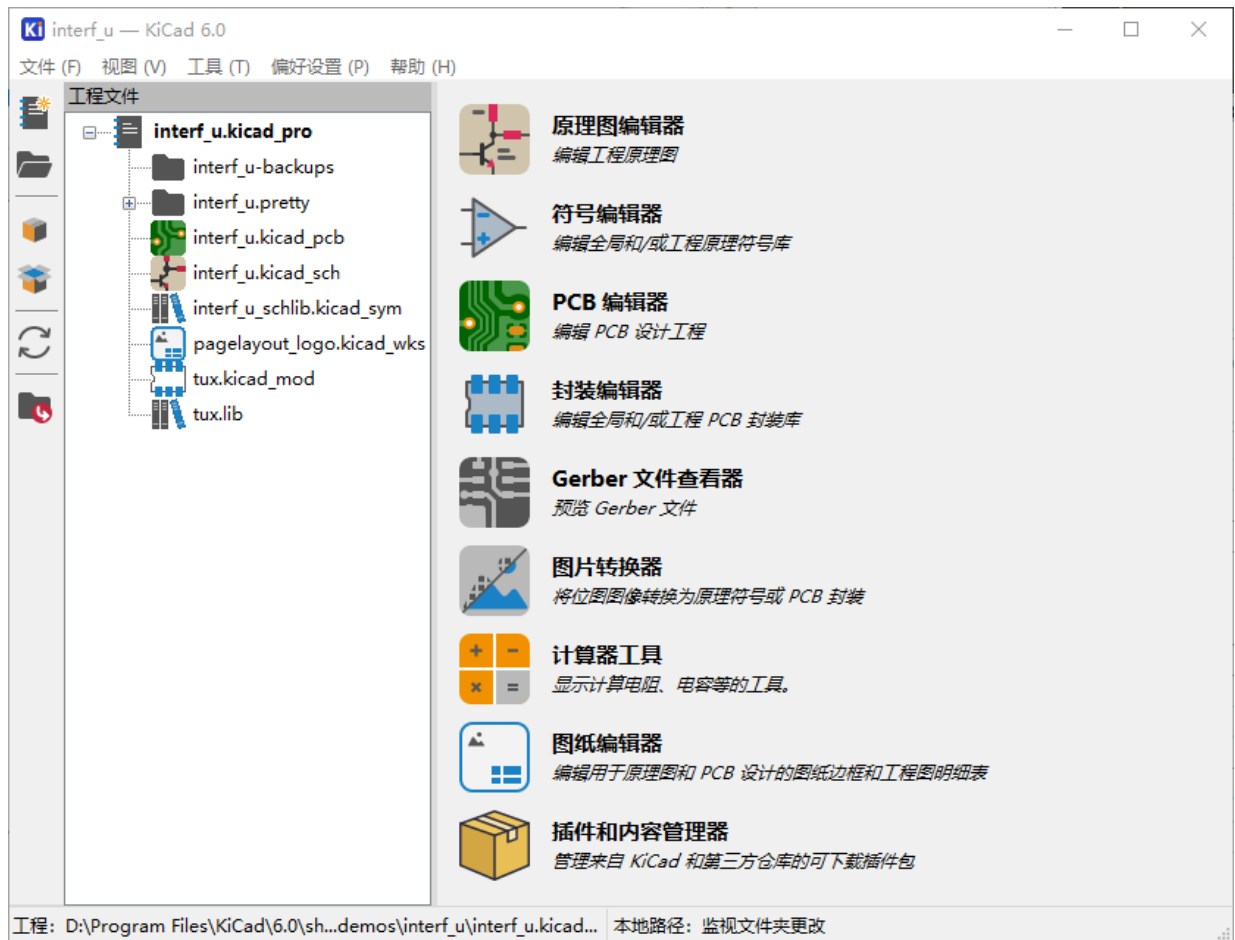
目前不会从以前的版本导入快捷&#x
*.hotkeys文件从旧版本配置目录复制刣
KiCad不会自动检测冲突，例如一个	

##x4f7f;##x7528; KiCad ##x5de5;##x7a0b;##x7ba

从 KiCad 管理器运行其他工具有一ө

- KiCad 目前一次只支持打开一个工程
KiCad 工程管理器运行原理图和电路
独立模式下运行时，您可以打开

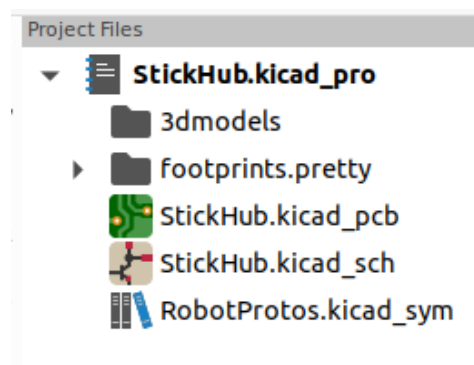
3.1 工程管理器窗口



KiCad 工程管理器窗口由左侧的树ܫ

3.2 $\times 5de5; \times 7a0b; \times 6811; \times 89c6; \times 56fe;$

树状视图显示工程文件夹内的
在树状视图中双击一个文件将
右键点击一个文件将打开一个



Note

只有 KiCad 知道如何打开的文件才会

3.3 3D Model Generation

3D model generation is a process that creates a 3D representation of a PCB design. This is useful for visualizing the physical layout and for manufacturing purposes.

	3D model generation process
	3D model generation process
	3D model generation process
	3D model generation process
	3D model generation process
	3D model generation process

3.4 3D Model Generation

3D model generation is a process that creates a 3D representation of a PCB design. This is useful for visualizing the physical layout and for manufacturing purposes.

3D model generation is a process that creates a 3D representation of a PCB design. This is useful for visualizing the physical layout and for manufacturing purposes.

3D model generation is a process that creates a 3D representation of a PCB design. This is useful for visualizing the physical layout and for manufacturing purposes.

3D model generation is a process that creates a 3D representation of a PCB design. This is useful for visualizing the physical layout and for manufacturing purposes.

Note

3D model generation is a process that creates a 3D representation of a PCB design. This is useful for visualizing the physical layout and for manufacturing purposes.

3D model generation is a process that creates a 3D representation of a PCB design. This is useful for visualizing the physical layout and for manufacturing purposes.

example.kicad_pro	KiCad 3D model generation process
example.kicad_sch	3D model generation process
example.kicad_pcb	3D model generation process

KiCad 可以导入由其他一些软件包创

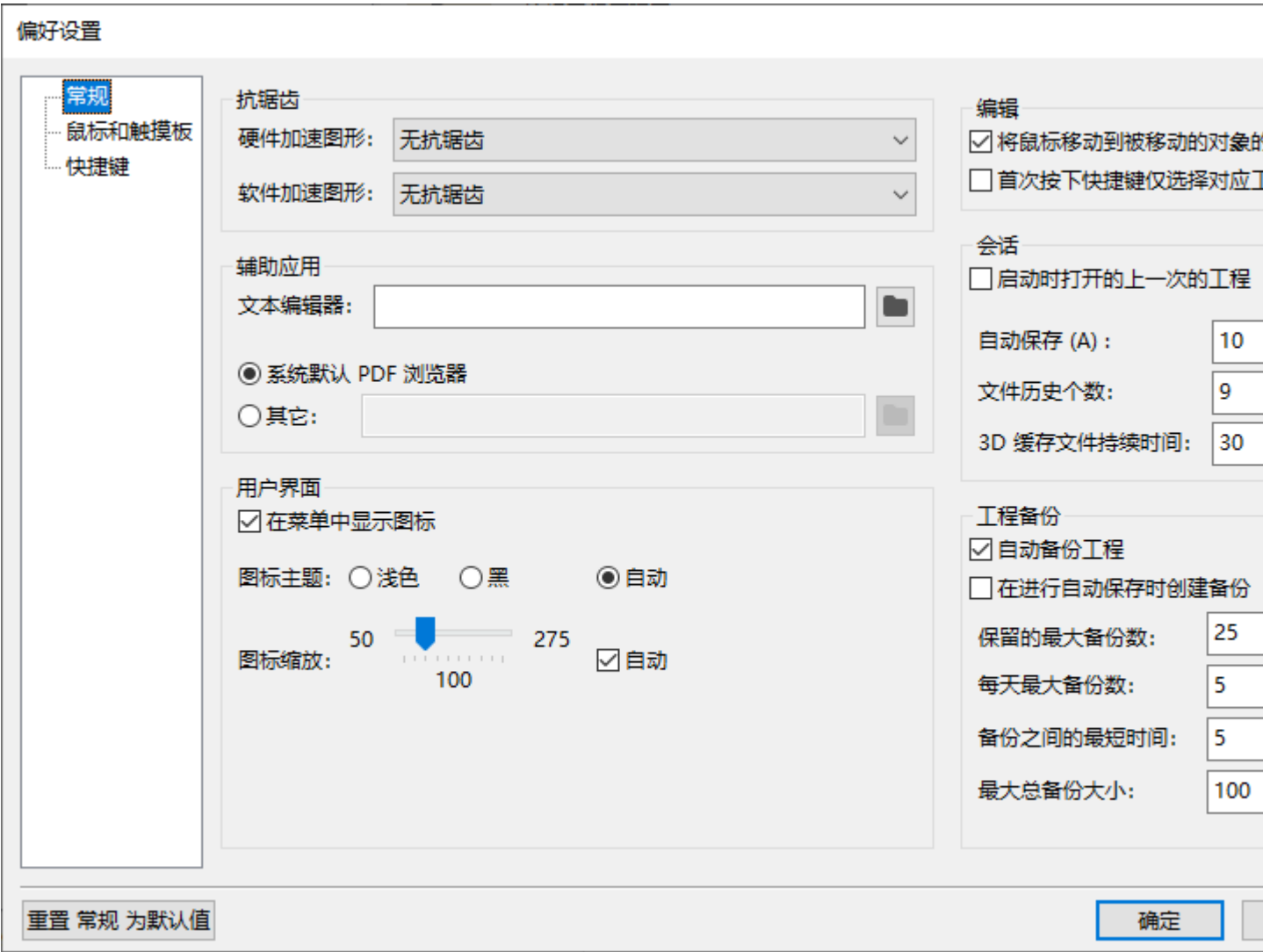
要从这些工具中导入工程，请W
文件 菜单的 **导入非 KiCad 工程** 子菜&#x
 系统将提示您在导入文件浏览V
 (例如，project.sch 和 project.brd)。一旦选择了请&
 KiCad 工程。

Chapter 4

KiCad 配置

KiCad 偏好设置始终可以从 偏好设置菜单访问，也可以使用快捷键 (৬trl+,)访问。 偏好设置对话框在运KiCad 工具之间共享。有些偏好设置 (如原理图或电路板编辑器)。

4.1 共同偏好设置



常规, KiCad 抗锯齿: 无抗锯齿, 软件加速图形: 无抗锯齿, 辅助应用: 文本编辑器, 系统默认 PDF 浏览器, 用户界面: 在菜单中显示图标, 图标主题: 自动, 图标缩放: 100, 自动, 编辑: 将鼠标移动到被移动的对象, 首次按下快捷键仅选择对应工, 会话: 启动时打开的上一次的工程, 自动保存 (A): 10, 文件历史个数: 9, 3D 缓存文件持续时间: 30, 工程备份: 自动备份工程, 在进行自动保存时创建备份, 保留的最大备份数: 25, 每天最大备份数: 5, 备份之间的最短时间: 5, 最大总备份大小: 100, 重置 常规 为默认值, 确定

Note

常规, KiCad 抗锯齿: 无抗锯齿, 软件加速图形: 无抗锯齿, 辅助应用: 文本编辑器, 系统默认 PDF 浏览器, 用户界面: 在菜单中显示图标, 图标主题: 自动, 图标缩放: 100, 自动, 编辑: 将鼠标移动到被移动的对象, 首次按下快捷键仅选择对应工, 会话: 启动时打开的上一次的工程, 自动保存 (A): 10, 文件历史个数: 9, 3D 缓存文件持续时间: 30, 工程备份: 自动备份工程, 在进行自动保存时创建备份, 保留的最大备份数: 25, 每天最大备份数: 5, 备份之间的最短时间: 5, 最大总备份大小: 100, 重置 常规 为默认值, 确定

常规, KiCad 抗锯齿: 无抗锯齿, 软件加速图形: 无抗锯齿, 辅助应用: 文本编辑器, 系统默认 PDF 浏览器, 用户界面: 在菜单中显示图标, 图标主题: 自动, 图标缩放: 100, 自动, 编辑: 将鼠标移动到被移动的对象, 首次按下快捷键仅选择对应工, 会话: 启动时打开的上一次的工程, 自动保存 (A): 10, 文件历史个数: 9, 3D 缓存文件持续时间: 30, 工程备份: 自动备份工程, 在进行自动保存时创建备份, 保留的最大备份数: 25, 每天最大备份数: 5, 备份之间的最短时间: 5, 最大总备份大小: 100, 重置 常规 为默认值, 确定

画布比例： 设置 KiCad 编辑器中使画选择 自动 根据您的操作系统设画对字体应用图标缩放： 该设置画中使用的字体进行缩放。 大多画显示器时，可能会改善某些 Linux 平台上 KiCad 的外观。

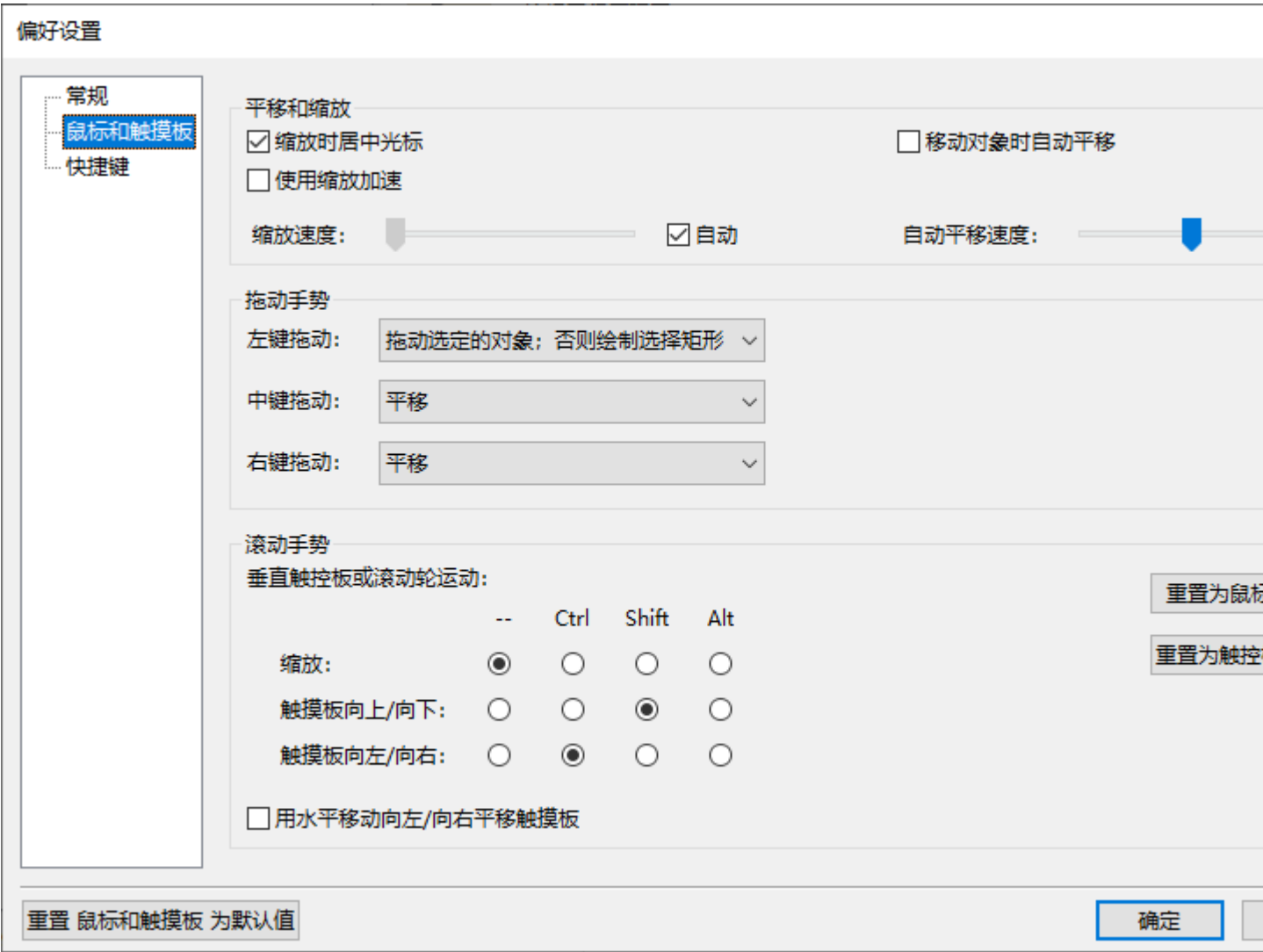
将鼠标扭曲到移动对象的原点启用后，当您在对象上启动移动扭曲) 到该对象的原点。

第一个快捷键选择工具： 禁用添加线 等命令的快捷键将立即启用后，第一次按快捷键将仅选添加线 工具，但不会立即开始上

记住下次启动项目时打开的文启用后，当重新打开工程时，KiCad 会自动重新打开之前打开的所文件历史大小： 配置最近打开画缓存文件时长： KiCad 创建 3D 模型缓 3D 观看速度。 您可以配置在删除自动备份工程：开启后，KiCad 工程文件。 存档将保存在工程文件在工程中保存文件时会创建备上自动保存时创建备份： 启用后如果下面的设置允许备份。 如禁用)，则此设置无效。

要保留的最大备份数： 创建新歯日最大备份次数： 新建备份最小备份间隔时间： 如果触发例如保存电路板文件)，如果已最大总备份大小： 创建新的备记住下次启动工程时打开的文勾选后，如果上次关闭工程管画将重新打开它们。

4.2 鼠标和触摸板偏好设画



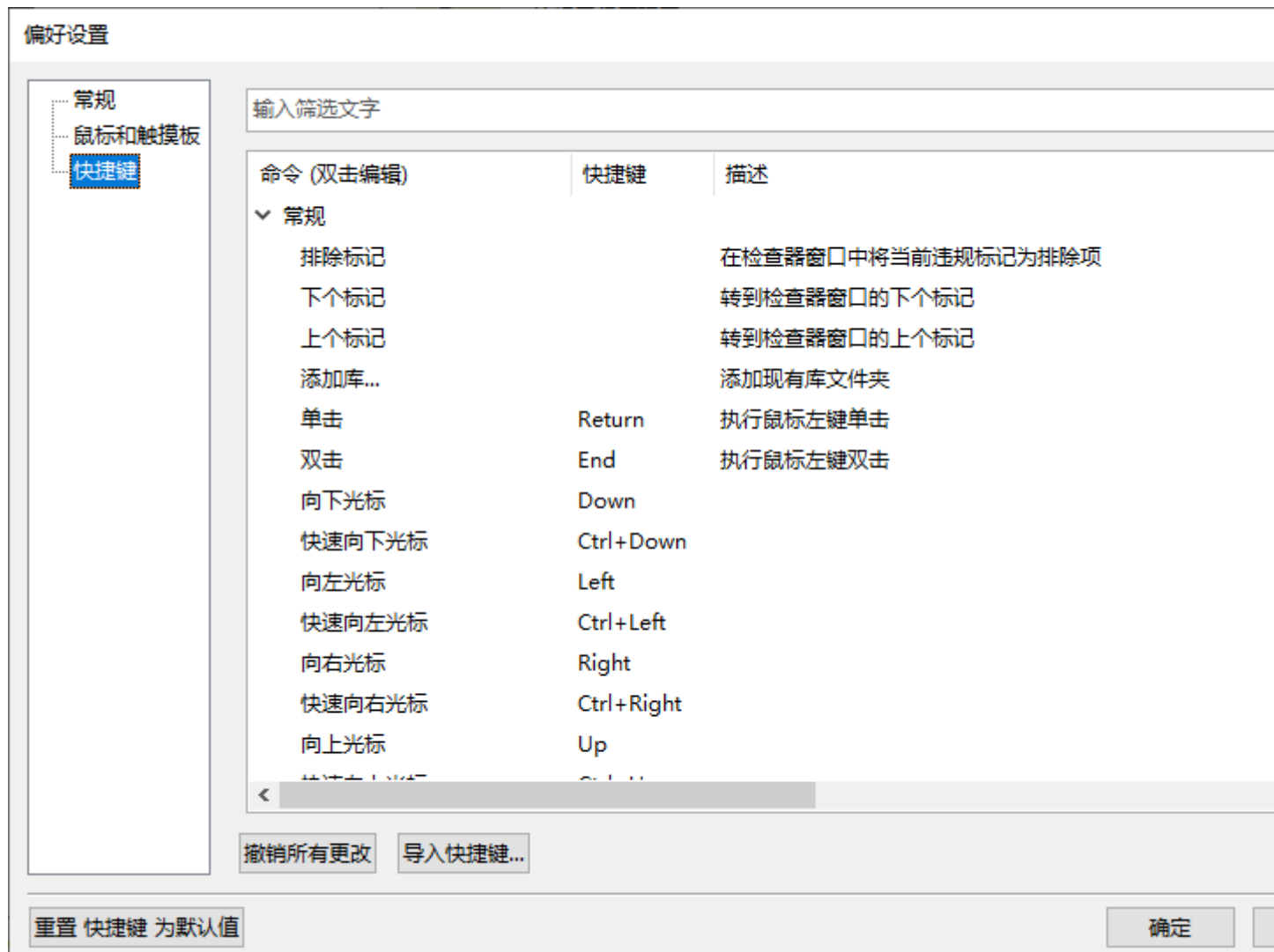
缩放时居中和扭曲光标： 启用使用缩放加速： 启用后，滚动速度： 控制鼠标滚轮或触使用 自动 根据您的操作系统设移动对象时自动平移： 开启后自动平移速度： 控制画布移动鼠标键： 您可以设置拖动鼠标䂨还可以根据编辑画布中是否\

Note

鼠标左键始终用于选择和操作对&

鼠标滚轮和触摸板滚动： 您可触摸板水平移动左右平移： 启如果鼠标上有) 平移视图。

4.3



可以使用此对话框自定义用于捷快捷键。公用部分中的快捷程序之间共享。当程序运行旻程序的快捷键。您可以将相同程序 (例如原理图编辑器和电跩中的不同操作，但不能将一个彤有许多可用命令，因此并非所您对快捷键分配所做的更改会字符。您可以通过右键单击特撤消更改来撤消对该命令的更导入快捷键

快捷键偏好设置存储在 KiCad 设置目
 .hotkeys 文件中 (有关设置目录在操作&#x
 《设置，设置》一节)。如果您
 KiCad 快捷键，则可以通过导入适当
 .hotkeys 文件将该配置传输到另一台&#x

4.4

KiCad environment variables: KICAD6_3DMODEL_DIR, KICAD6_3RD_PARTY, KICAD6_FOOTPRINT_DIR, KICAD6_SYMBOL_DIR, KICAD6_TEMPLATE_DIR, KICAD_USER_TEMPLATE_DIR

For instance, the path to the connect.pretty footprint library, when using the KICAD6_FOOTPRINT_DIR environment variable, would be defined as \${KICAD6_FOOTPRINT_DIR}/connect.pretty.

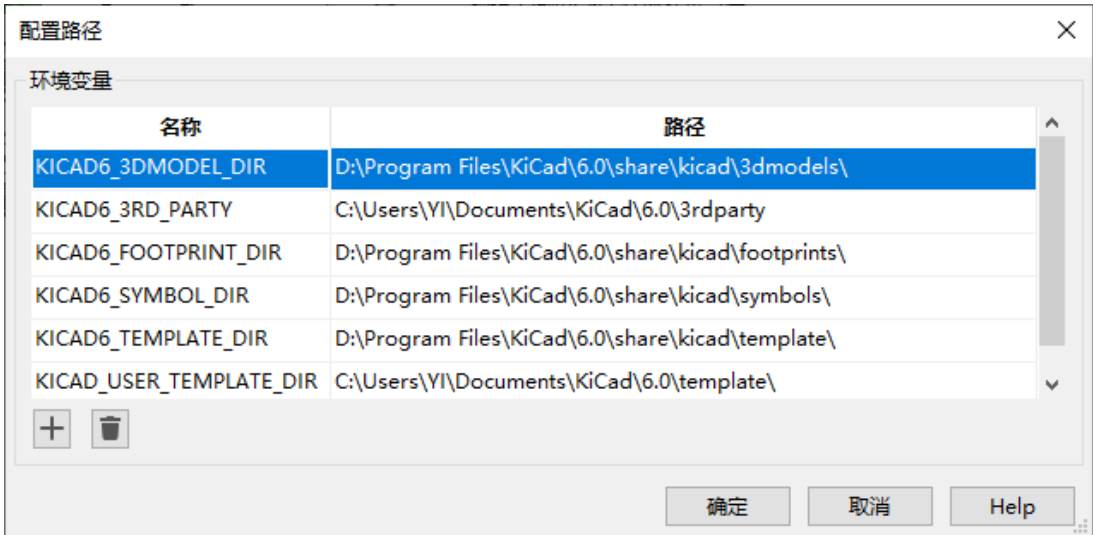
- KICAD6_3DMODEL_DIR: Base path of 3D models used in footprints.
- KICAD6_3RD_PARTY: Location for plugins, libraries, and color themes installed by the Plugin and Content Manager.
- KICAD6_FOOTPRINT_DIR: Base path of footprint library files.
- KICAD6_SYMBOL_DIR: Base path of symbol library files.
- KICAD6_TEMPLATE_DIR: Location of project templates installed with KiCad.
- KICAD_USER_TEMPLATE_DIR: Location of personal project templates.

For instance, the path to the connect.pretty footprint library, when using the KICAD6_FOOTPRINT_DIR environment variable, would be defined as \${KICAD6_FOOTPRINT_DIR}/connect.pretty.

KiCad environment variables: KICAD6_3DMODEL_DIR, KICAD6_3RD_PARTY, KICAD6_FOOTPRINT_DIR, KICAD6_SYMBOL_DIR, KICAD6_TEMPLATE_DIR, KICAD_USER_TEMPLATE_DIR

KiCad environment variables: KICAD6_3DMODEL_DIR, KICAD6_3RD_PARTY, KICAD6_FOOTPRINT_DIR, KICAD6_SYMBOL_DIR, KICAD6_TEMPLATE_DIR, KICAD_USER_TEMPLATE_DIR

KICAD6_3DMODEL_DIR	Base path of 3D models used in footprints.
KICAD6_3RD_PARTY	Location for plugins, libraries, and color themes installed by the Plugin and Content Manager .
KICAD6_FOOTPRINT_DIR	Base path of footprint library files.
KICAD6_SYMBOL_DIR	Base path of symbol library files.
KICAD6_TEMPLATE_DIR	Location of project templates installed with KiCad.
KICAD_USER_TEMPLATE_DIR	Location of personal project templates.



Paths set in the Configure Paths dialog are internal to KiCad and are not visible as environment variables outside of KiCad. They are stored in [KiCad's user configuration files](#).

Paths can also be set as environment variables outside of KiCad, which will override any settings in the user's configuration.

Note

You cannot override an environment variable that has been set outside of KiCad by using the Configure Paths dialog. Any variable that has been set externally will be shown as read-only in the dialog.

Note also that the environment variable `KIPRJMOD` is **always** internally defined by KiCad, and expands to the **current project absolute path**.

For instance, `${KIPRJMOD}/connect.pretty` is always the `connect.pretty` folder (the footprint library) inside **the current project folder**.

The `KIPRJMOD` variable cannot be changed in the Configure Paths dialog or overridden by an external environment variable.

4.4.1 Advanced environment variables

Some advanced environment variables can be set to customize KiCad’s behavior. These variables are not shown in the environment variable configuration and cannot be used in path substitutions. They cannot be modified in the Configure Paths dialog, but they can be overridden by system environment variables.

Changing these variables will not result in KiCad moving any files from the default location to the new location, so if you change these variables you will need to copy any desired settings or files manually.

Warning

KICAD_CONFIG_HOME	Base path of KiCad configuration files. Subdirectories will be created within this directory for each KiCad minor version.
KICAD_DOCUMENTS_HOME	Base path of KiCad user-modifiable documents, such as projects, templates, Python scripts, libraries, etc. Subdirectories will be created within this directory for each KiCad minor version. This directory is provided as a suggested user data location, but does not need to be used.



Warning
If you modify the configuration of paths, please quit and restart KiCad to avoid any issues in path handling.

4.5 Managing Libraries

The **Preferences** → **Manage Symbol Libraries...** menu lets you manage the list of symbol libraries (**symbol library table**). Likewise, use the **Preferences** → **Manage Footprint Libraries...** menu to manage the list of footprint libraries (**footprint library table**).

For each type of library (symbol and footprint), there are 2 library tables: global and project specific. The global library table is located in the [user configuration directory](#) and contains a list of libraries available to all projects. The project-specific library table is optional and contains a list of libraries specific to the project. It is located in the project directory.

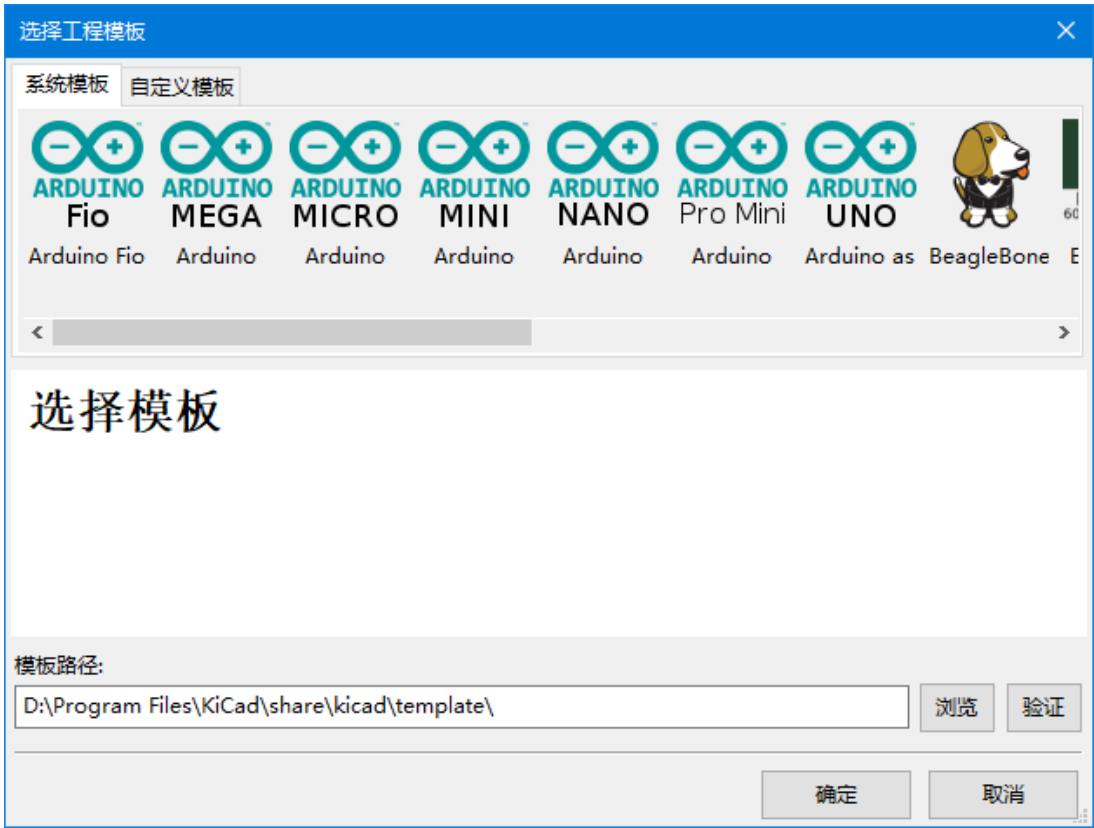
Chapter 5

5. 选择工程模板

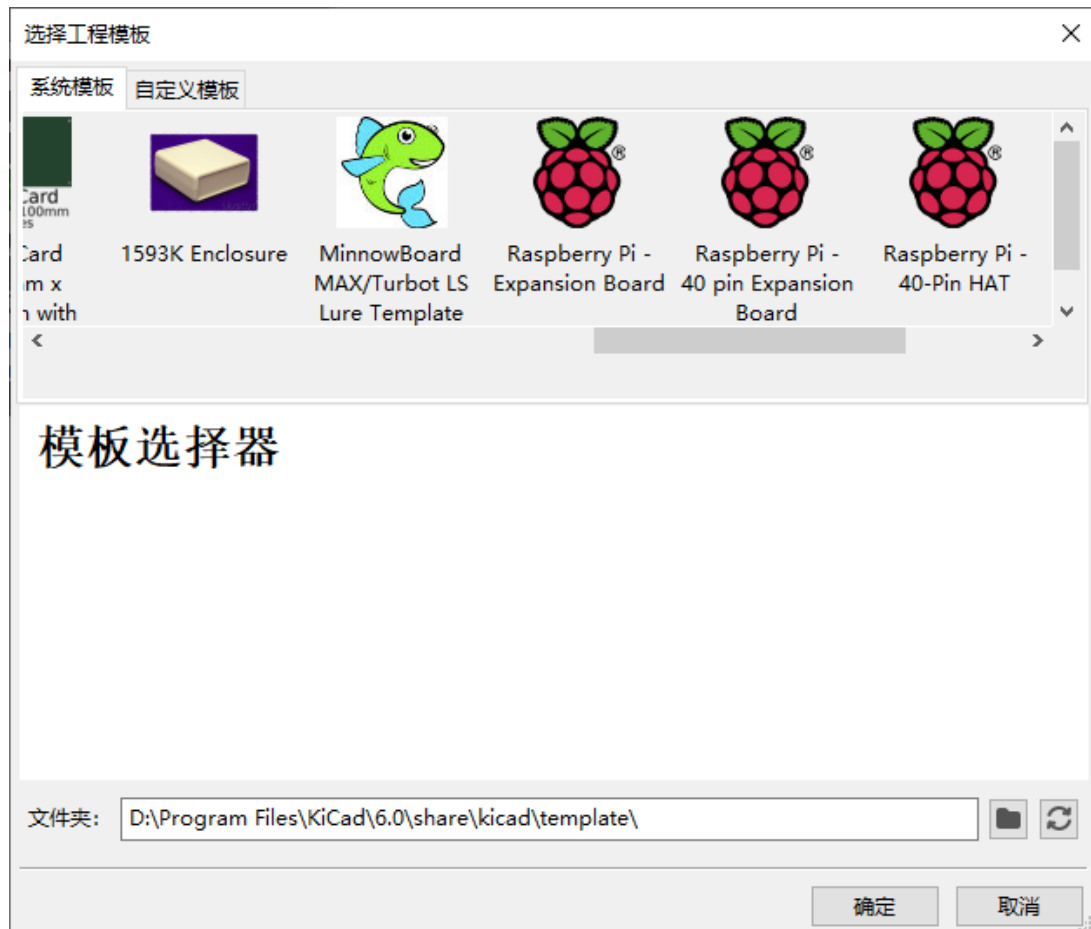
选择工程模板对话框显示了系统模板和自定义模板。系统模板包括 Arduino Fio, Arduino MEGA, Arduino MICRO, Arduino MINI, Arduino NANO, Arduino Pro Mini, 和 Arduino UNO。自定义模板路径为 D:\Program Files\KiCad\share\kicad\template\。

5.1 选择工程模板

选择工程模板对话框显示了系统模板和自定义模板。系统模板包括 Arduino Fio, Arduino MEGA, Arduino MICRO, Arduino MINI, Arduino NANO, Arduino Pro Mini, 和 Arduino UNO。自定义模板路径为 D:\Program Files\KiCad\share\kicad\template\。



选择工程模板对话框显示了系统模板和自定义模板。系统模板包括 Arduino Fio, Arduino MEGA, Arduino MICRO, Arduino MINI, Arduino NANO, Arduino Pro Mini, 和 Arduino UNO。自定义模板路径为 D:\Program Files\KiCad\share\kicad\template\。



5.2 \mathbb{R}^n and \mathbb{C}^n

KiCad 在以下路径中查找模板文件：

- 环境变量 KICAD_USER_TEMPLATE_DIR 中定义的路径
- 环境变量 KICAD_TEMPLATE_DIR 中定义的路径
- 系统模板：<kicad bin dir>/../share/kicad/template/
- 用户模板：
 - Unix: ~/kicad/template/
 - Windows: C:\Documents and Settings\username\My Documents\kicad\template or C:\Users\username\Documents\kicad\template
 - Mac: ~/Documents/kicad/template/

5.3 \mathbb{R}^n and \mathbb{C}^n

模板名称是存储模板文件的目元数据目录是名为 **meta** 的子目录，元数据由一个必需文件组成，嘭所有文件必须由用户使用文本的 KiCad 工程文件创建，并放入所需的

使用模板创建工程时，模板中的除外。包含模板名称的文件和例如，从名为 示例 的模板创建和新工程 的工程：

模板 示例 目录中的文件名称的文	在工程 新工程 目录中创建的文
示例.kicad_pro 示例.kicad_sch 示例.kicad_pcb 示例- 第一次.kicad_sch 第二次- 示例.kicad_sch 第三次.kicad_sch 第三次.kicad_pcb	新工程.kicad_pro 新工程.kicad_sch 新工程.kicad_pcb 新工程- 第一次.kicad_sch 第二次- 新工程.kicad_sch 第三次.kicad_sch 第三次.kicad_pcb

模板不需要包含完整的工程，将使用其默认的创建工程行为外。包含模板名称重命名规则的例和将基于该工程文件名称进行重

模板 示例 目录中的文件名称的文	在 新工程 新工程 目录中创建的文
示例.kicad_sch 第一次- 示例.kicad_sch 第一次- 示例.kicad_pcb 第二次- 示例.kicad_sch 第二次- 示例.kicad_pcb	新工程.kicad_sch 第一次- 新工程.kicad_sch 第一次- 新工程.kicad_pcb 第二次- 新工程.kicad_sch 第二次- 新工程.kicad_pcb 新工程.kicad_pro (默认) 新工程.kicad_pcb (默认)

作为模板名称重命名规则的例和将基于该工程文件名称进行重

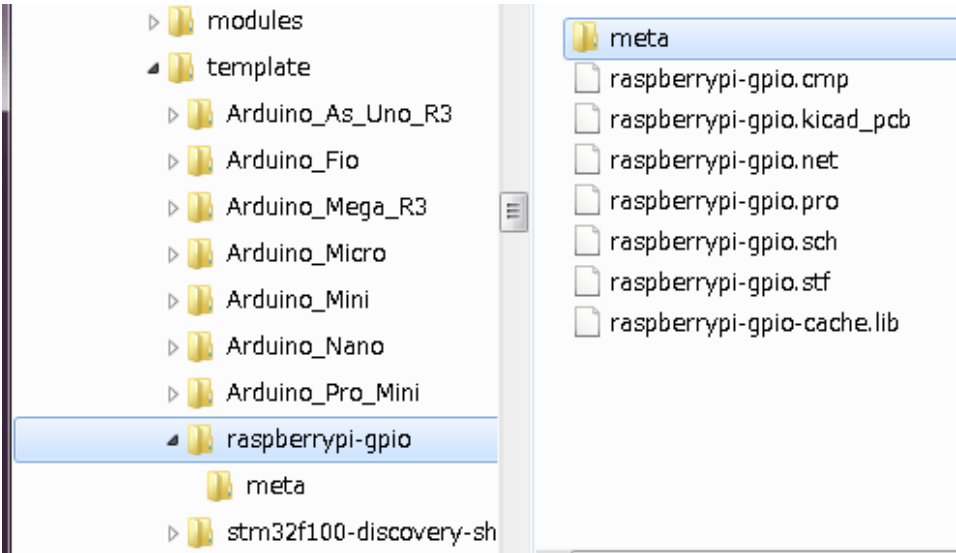
模板 示例 目录中的文件名称的文	在 新工程 新工程 目录中创建的文
示例.kicad_sch 示例.kicad_pcb 第一次- 示例.kicad_pro 第一次- 示例.kicad_sch 第一次- 示例.kicad_pcb 第二次- 示例.kicad_sch 第二次- 示例.kicad_pcb	示例.kicad_sch 示例.kicad_pcb 新工程.kicad_pro 新工程.kicad_sch 新工程.kicad_pcb 第二次- 示例.kicad_sch 第二次- 示例.kicad_pcb

Note

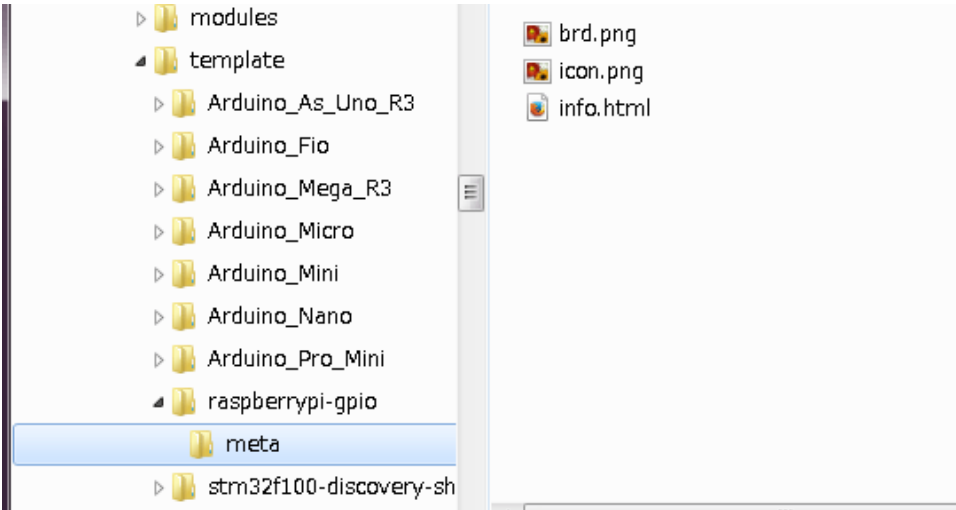
不建议创建包含多个工程文件的&

5.3.1 模板示例

这是一个显示 raspberry-gpio 模板的工程e



和元数据文件：



5.3.2 所需文件：

meta/info.html	描述模板的 HTML 格式信息。
----------------	---

<title> 标记确定向用户公开以供选خ
使用 HTML 意味着可以轻松地对图像

本文档中只能使用基本 HTML 标记　
这是一个示例 **info.html** 文件：

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD>
<META HTTP-EQUIV="CONTENT-TYPE" CONTENT="text/html;
charset=windows-1252">
<TITLE>Raspberry Pi - &#x6269;&#x5c55;&#x677f;</TITLE>
</HEAD>
<BODY LANG="fr-FR" DIR="LTR">
<P> ↳
    &#x8fd9;&#x4e2a;&#x5de5;&#x7a0b;&#x6a21;&#x677f;&#x662f;&#x6269;&#x5c55;&#x677f;&#x7684;&#x57fa;&#x677f;
<A HREF="http://www.raspberrypi.org/" TARGET="blank">Raspberry Pi $25
ARM &#x677f;&#x3002;</A> <BR><BR> ↳
    &#x8be5;&#x57fa;&#x7840;&#x5de5;&#x7a0b;&#x5305;&#x62ec;&#x4e00;&#x4e2a;PCB ↳
    &#x8fb9;&#x7f18;&#xff0c;
&#x5176;&#x5c3a;&#x5bf8;&#x4e0e;Raspberry-Pi PCB&#x76f8;&#x540c;&#xff0c;
&#x8fde;&#x63a5;&#x5668;&#x6b63;&#x786e;&#x653e;&#x7f6e;&#x4ee5;&#x5bf9;&#x9f50;&#x4e24;&#x5757;&#x6
-Pi &#x677f;&#x4e0a;&#x7684;&#x6240;&#x6709; IO &#x90fd;&#x901a;&#x8fc7;
0.1&quot;&#x8fde;&#x63a5;&#x5230;&#x5de5;&#x7a0b;&#x3002;&#x6269;&#x5c55;
&#x5934;&#x3002; <BR><BR>&#x677f;&#x8f6e;&#x5ed3;&#x5982;&#x4e0b;&#x6240;&#x793a;&#xff1a;
</P>
<P><IMG SRC="brd.png" NAME="brd" ALIGN=BOTTOM WIDTH=680 HEIGHT=378
BORDER=0><BR><BR><BR><BR>
</P>
<P>(c)2012 Brian Sidebotham<BR>(c)2012 KiCad Developers</P>
</BODY>
</HTML>
```

5.3.3 可选文件：

meta/icon.png	一个 64 x 64 像素的 PNG 图标文件，用作 模板选择对话框中的可
---------------	---

meta/info.html 使用的任何其他图像文件（&

Chapter 6

插件和内容

Note

TODO: 撰写本节

Chapter 7

Actions reference

Below is a list of every available **action** in the KiCad Project Manager: a command that can be assigned to a hotkey.

7.1 KiCad Project Manager

The actions below are available in the KiCad Project Manager. Hotkeys can be assigned to any of these actions in the **Hotkeys** section of the preferences.

Action	Default Hotkey	Description
New Project. . .	kbd:[Ctrl+N]	Create new blank project
New Project from Template. . .	kbd:[Ctrl+T]	Create new project from template
Open Project. . .	kbd:[Ctrl+O]	Open an existing project
Open Demo Project. . .		Open a demo project
Close Project		Close the current project
Schematic Editor	kbd:[Ctrl+E]	Edit schematic
Symbol Editor	kbd:[Ctrl+L]	Edit schematic symbols
PCB Editor	kbd:[Ctrl+P]	Edit PCB
Footprint Editor	kbd:[Ctrl+F]	Edit PCB footprints
Gerber Viewer	kbd:[Ctrl+G]	Preview Gerber output files
Image Converter	kbd:[Ctrl+B]	Convert bitmap images to schematic or PCB components
Calculator Tools		Run component calculations, track width calculations, etc.
Drawing Sheet Editor	kbd:[Ctrl+Y]	Edit drawing sheet borders and title block
Plugin and Content Manager	kbd:[Ctrl+M]	Manage downloadable packages from KiCad and 3rd party repositories
Open Text Editor		Launch preferred text editor