

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	7
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	11
4.2	鼠标和触摸板偏好设置	13
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

Copyright

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

Contributors

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

Feedback

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

Project manager file:

*.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。 可以请取，并将由工程{ .kicad_pro 文件。

Schematic editor files:

*.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 .kicad_sch 文件。
*.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{

Board editor files and folders:

*.kicad_pcb	包含除板框以外的所有
*.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T
*.kicad_dru	设计规则文件，包含某 .kicad_pcb 文件的自定义设计规则
*.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‹
*.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V
fp-lib-table	封装库列表（ 封装库表 ）： 线路板编辑器中可用的\
fp-info-cache	缓存以加速封装库的加

Common files:

*.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ÿ 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

Other files:

*.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
-------	---

Other files:

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g
*.rpt	报告文件（ASCII 格式），用于文档。
*.ps	绘图文件（Postscript），用于文
*.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。

Storing and sending KiCad files

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&#x

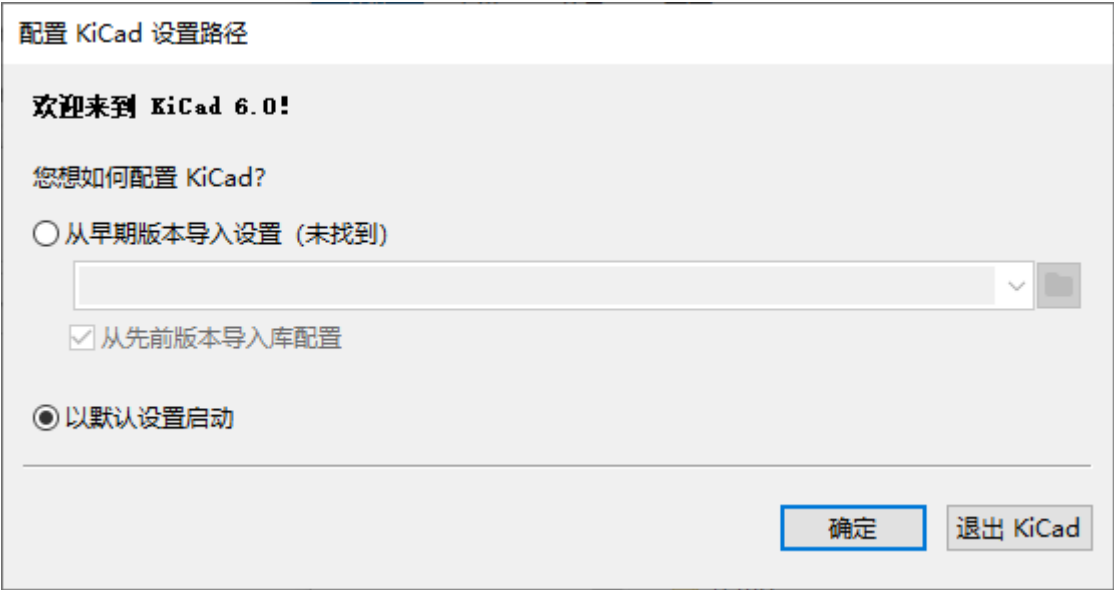
某些文件 (如工程本地设置文件 (.kicad_prl) 和 fp-info-cache 文件) 不需要随工程一 Git 之类的版本控制系统来跟踪 Ki-Cad 工程，则可能需要将这些文件&#x

Chapter 2

Getting Started with KiCad

2.1 Installing KiCad

KiCad is a free and open-source electronic design automation (EDA) software package. It is used to create printed circuit boards (PCBs) and is available for Windows, Linux, and macOS. KiCad is a powerful tool for designing and manufacturing PCBs, and it is used by many engineers and hobbyists around the world.



KiCad will automatically create a configuration directory in the user's home directory. The default location for the configuration directory is %APPDATA%\kicad on Windows and ~/.config/kicad on Linux.

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

KiCad will automatically create a configuration directory in the user's home directory. The default location for the configuration directory is %APPDATA%\kicad on Windows and ~/.config/kicad on Linux. (KiCad 5.1 and earlier versions used the location %HOMEDRIVE%\kicad on Windows and ~/.kicad on Linux.)

Windows	%APPDATA%\kicad
Linux	~/.config/kicad

Chapter 3

4f7f;7528; KiCad 5de5;7a0b;7ba

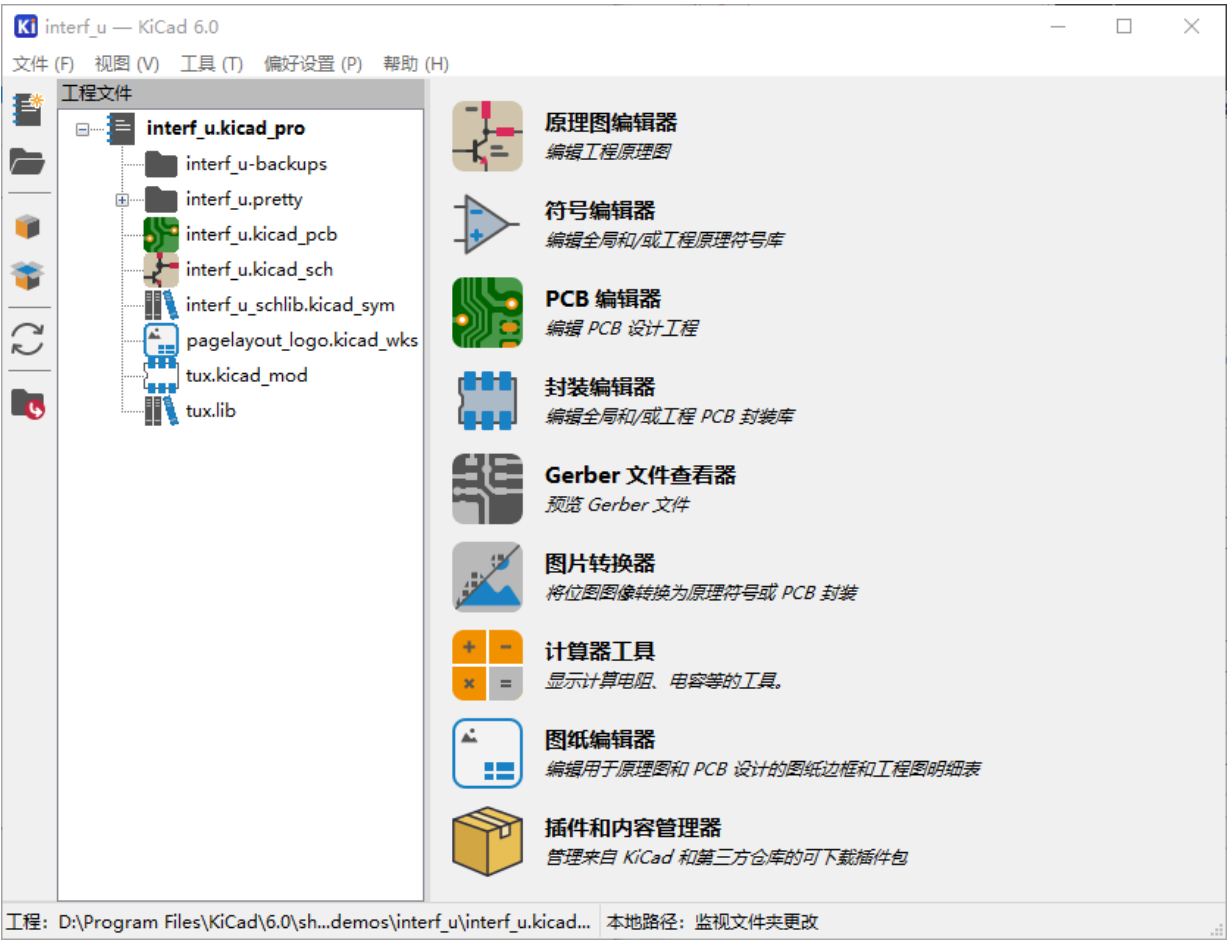
KiCad 5de5;7a0b;7ba1;7406;5668; (kicad 6216; kicad.exe) 662f;4e00;4e2a;5728;521b
 (539f;7406;56fe;548c;7535;8def;677f;7f16;8f91;5668;3001;Gerber 67e5;770
 7684;5de5;5177;3002;

4ece; KiCad 7ba1;7406;5668;8fd0;884c;5176;4ed6;5de5;5177;6709;4e00;4e9

- 5728;539f;7406;56fe;7f16;8f91;5668;548c;7535;8def;677f;7f16;8f91;5
- 539f;7406;56fe;7f16;8f91;5668;548c;7535;8def;677f;7f16;8f91;5668;5
 (65e0;9700;521b;5efa;7f51;8868;6587;4ef6;)

KiCad 76ee;524d;4e00;6b21;53ea;652f;6301;6253;5f00;4e00;4e2a;5de5;7a0b
 KiCad 5de5;7a0b;7ba1;7406;5668;8fd0;884c;539f;7406;56fe;548c;7535;8def
 72ec;7acb; 6a21;5f0f;4e0b;8fd0;884c;65f6;ffc;60a8;53ef;4ee5;6253;5f0

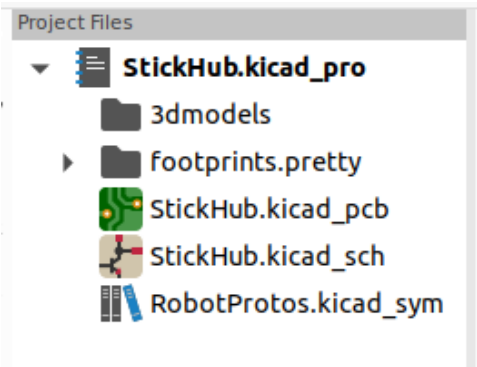
3.1 原理图编辑器



KiCad 原理图编辑器

3.2 原理图符号库

原理图符号库是 KiCad 的一个重要组成部分，它允许用户创建和管理自己的符号库。在 KiCad 中，符号库通常以 .kicad_sym 文件的形式存在。用户可以通过创建新的符号库或加载现有的符号库来管理自己的符号。符号库中的符号可以用于原理图设计，并且可以与其他用户共享。KiCad 提供了多种方法来创建和管理符号库，包括使用符号库管理器、符号库编辑器等。用户可以通过创建新的符号库或加载现有的符号库来管理自己的符号。符号库中的符号可以用于原理图设计，并且可以与其他用户共享。KiCad 提供了多种方法来创建和管理符号库，包括使用符号库管理器、符号库编辑器等。



Note

只有 KiCad 知道如何打开的文件才会有

3.3 侧面工具条

窗口左侧的工具栏为常见的工程。

	创建一个新的工程。
 。	打开一个现有的工程。
	创建整个工程的压缩文件。这包括原理图文件、库、PCB 等。
	将工程压缩文件解压到一个目标目录中的文件将被覆目。
	刷新树状视图，以检测文件。
	在文件资源管理器中打开工程。

3.4 创建一个新工程

大多数 KiCad 设计都是从创建工程开工程管理器创建工程有两种新《工程 - 模板，工程模板》一节若要创建新工程，请使用 **文件**
菜单中的 **新建工程**... 命令，或单新建工程 按钮，或使用键盘快目默认为 Ctrl+N)。

系统将提示您输入工程名称。选 MyProject，KiCad 会在其中创建 MyProject 目录和目 MyProject/MyProject.kicad_pro。

如果您已经有一个存储工程文件 **新建工程** 对话框中取消选中 为复选框。

Note

强烈建议您将每个 KiCad 工程保存在変择工程名称后，KiCad 将在工程目。

example.kicad_pro	KiCad 工程文件。
example.kicad_sch	主原理图文件。
example.kicad_pcb	印刷电路板文件。

3.5 EDA

KiCad 可以导入由其他一些软件包创

*.sch, *.brd	Eagle 6.x 或更高版本 (XML 格式)
*.csa, *.cpa	CADSTAR 档案格式

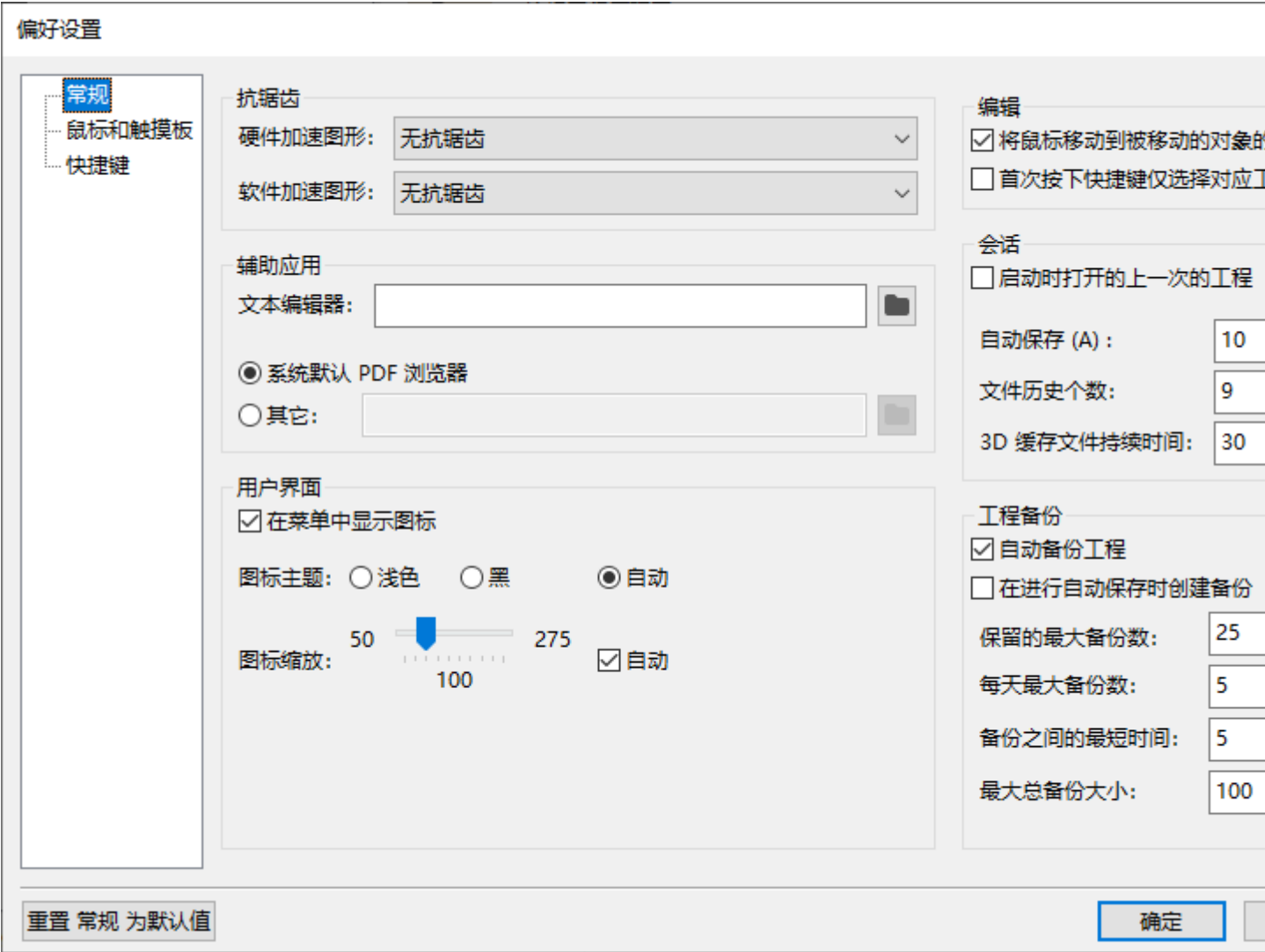
要从这些工具中导入工程，请W
文件菜单的导入非 KiCad 工程子菜&#x
系统将提示您在导入文件浏览V
(例如，project.sch 和 project.brd)。一旦选择了请&
KiCad 工程。

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 抗锯齿设置



Accelerated graphics antialiasing: KiCad can use different methods to prevent aliasing (jagged lines) when rendering using a graphics card. Different methods may look better on different hardware, so you may want to experiment to find the one that looks best to you.

Fallback graphics antialiasing: KiCad can also apply antialiasing when using the fallback graphics mode. Enabling this feature may result in poor performance on some hardware.

Text editor: Choose a text editor to use when opening text files from the project tree view.

PDF viewer: Choose a program to use when opening PDF files.

Show icons in menus: Enables icons in drop-down menus throughout the KiCad user interface.

Note

抗锯齿设置

Icon theme: Sets whether to use the icon theme designed for light window backgrounds or dark window backgrounds. The default setting of Automatic will choose the theme based on the lightness of the operating system window theme.

Icon scale: Sets the size of the icons used in menus and buttons throughout KiCad. Choose *Automatic* to pick an appropriate icon scale automatically based on your operating system settings.

Canvas scale: Sets the scale of the drawing canvas used in the KiCad editors. Choose *Automatic* to pick an appropriate canvas scale automatically based on your operating system settings.

Apply icon scaling to fonts: This setting will scale fonts used in the UI according to the icon scale setting. This is not needed for most users, but may improve the look of KiCad on certain Linux platforms when using a high-DPI display.

Warp mouse to origin of moved object: When enabled, the mouse cursor will be repositioned (warped) to the origin of an object when you start a move command on that object.

First hotkey selects tool: When disabled, pressing the hotkey for a command such as *Add Wire* will immediately start the command at the current cursor location. When enabled, pressing the hotkey the first time will just select the *Add Wire* tool but will not immediately begin a wire.

Remember open files for next project launch: When enabled, KiCad will automatically re-open any files that were previously open when a project is re-opened.

Auto save: When editing schematics and board files, KiCad can automatically save your work periodically. Set to 0 to disable this feature.

File history size: Configure the number of entries in the list of recently-opened files

3D cache file duration: KiCad creates a cache of 3D models in order to speed up the 3D viewer. You can configure how long to keep this cache before deleting old files.

Automatically backup projects: When enabled, KiCad projects will be archived to ZIP files automatically according to the settings below. The archives will be stored in a subfolder of the project folder. Backups are created when saving files in the project.

Create backups when auto save occurs: When enabled, a backup will be created every time an automatic file save occurs (if the backup is permitted by the settings below). This setting has no effect if the auto save interval is set to 0 (disabled).

Maximum backups to keep: When creating a new backup, the oldest backup file will be deleted to keep the total number of backup files below this limit.

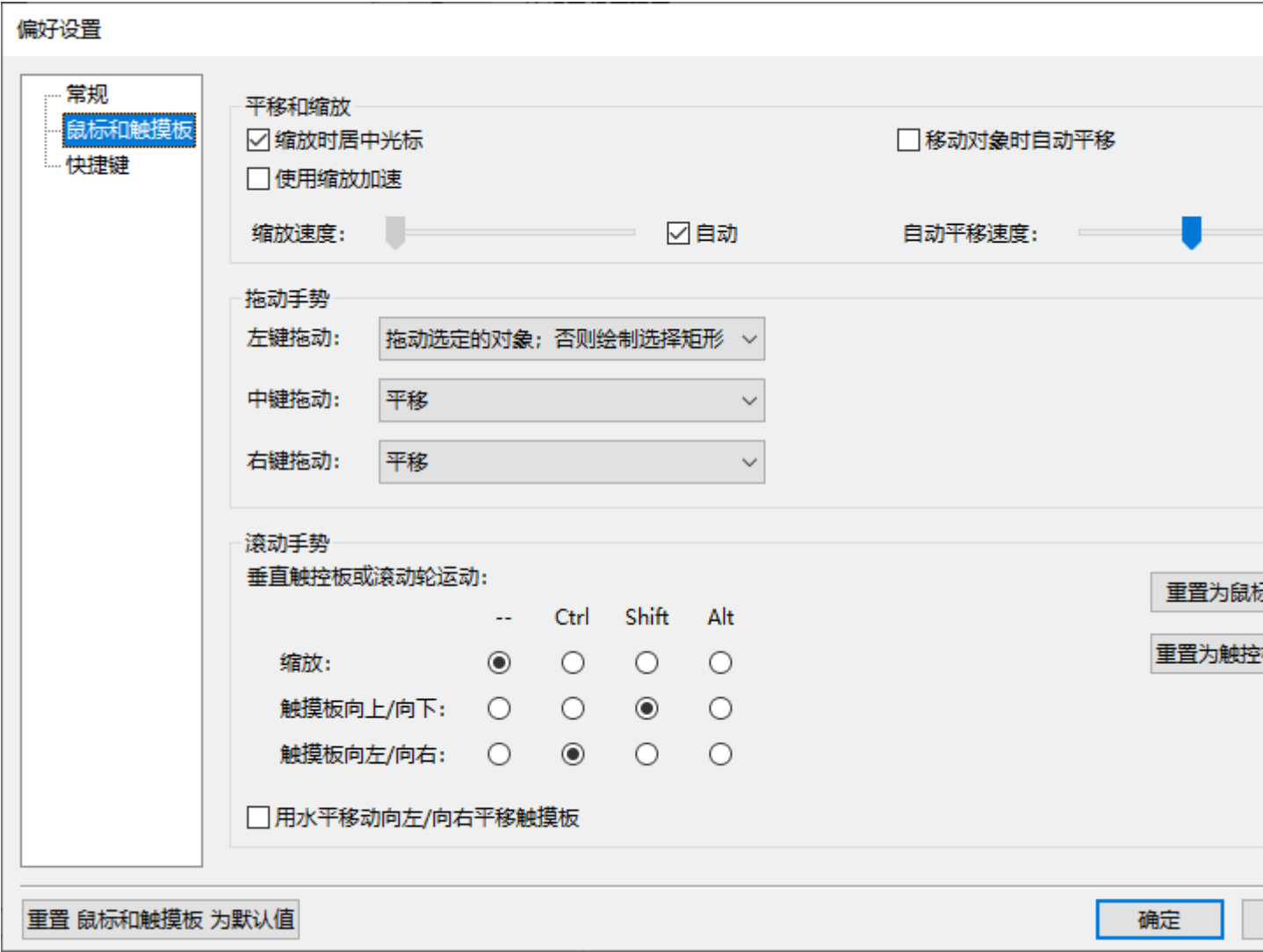
Maximum backups per day: When creating a new backup, the oldest backup file created on the current day will be deleted to stay below this limit.

Minimum time between backups: If backup is triggered (for example, by saving a board file), the backup will not be created if an existing backup file is newer than this limit.

Maximum total backup size: When creating a new backup file, the oldest backup files will be deleted to keep the total size of the backup files directory below this limit.

Remember open files for next project launch: When checked, KiCad will re-open the schematic and board editor if they were open the last time you closed the project manager.

4.2 鼠标和触摸板



Center and warp cursor on zoom: When enabled, zooming using the hotkeys or mouse wheel will cause the view to be centered on the cursor location.

Use zoom acceleration: When enabled, scrolling the mouse wheel or touchpad faster will cause the zoom to change faster.

Zoom speed: Controls how much the zoom changes for a given amount of scrolling the mouse wheel or touchpad. Use *Automatic* to set a default value depending on your operating system.

Automatically pan while moving object: When enabled, the view can be panned while moving an object by moving close to the edge of the canvas.

Auto pan speed: Controls how fast the canvas pans while moving an object.

Mouse buttons: You can set the behavior of dragging the middle and right mouse buttons to zoom the view, pan the view, or have no effect. You can also set the behavior of dragging the left mouse button depending on whether or not any objects are already selected in the editing canvas.

Note

鼠标和触摸板

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񓀊设置，设置》一节)。如果您在 KiCad 快捷键，则可以通过导入适当在 .hotkeys 文件将该配置传输到另一台目.

4.4 路径配置

在 KiCad 中，可以使用 *environment* 变量 定义设一些环境变量由 KiCad 在内部定义，形状等的路径。

当绝对路径未知或可能发生变变请考虑以下可能安装在不同位目.

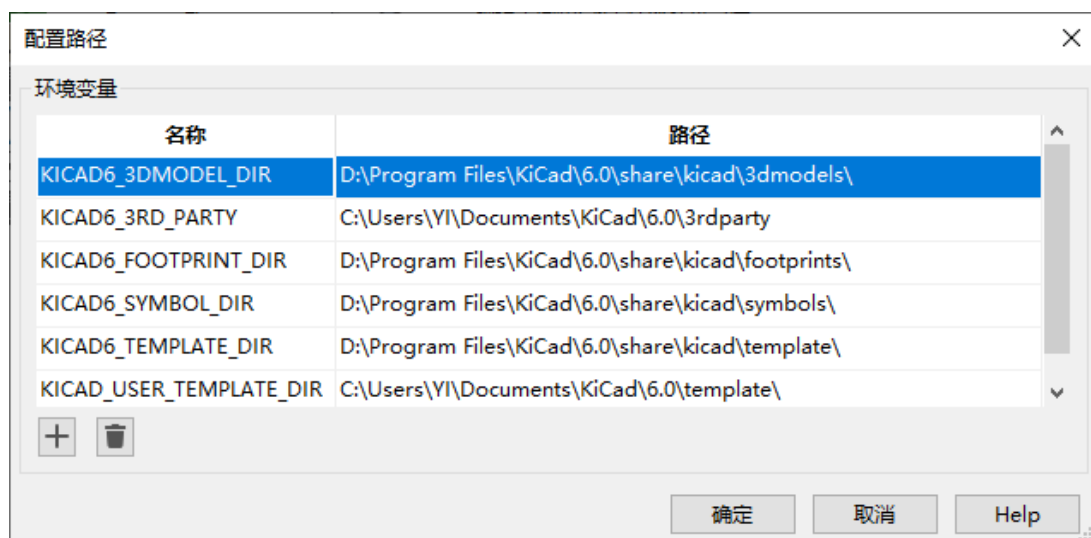
- 原理图符号库
- 封装库
- 封装定义中使用的 3D 形状文件

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`.

在 偏好设置 → 配置路径... 菜单中， KiCad 环境变量定义路径，如果需要

KiCad environment variables:

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.

Additional environment variables:

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 库配置

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.

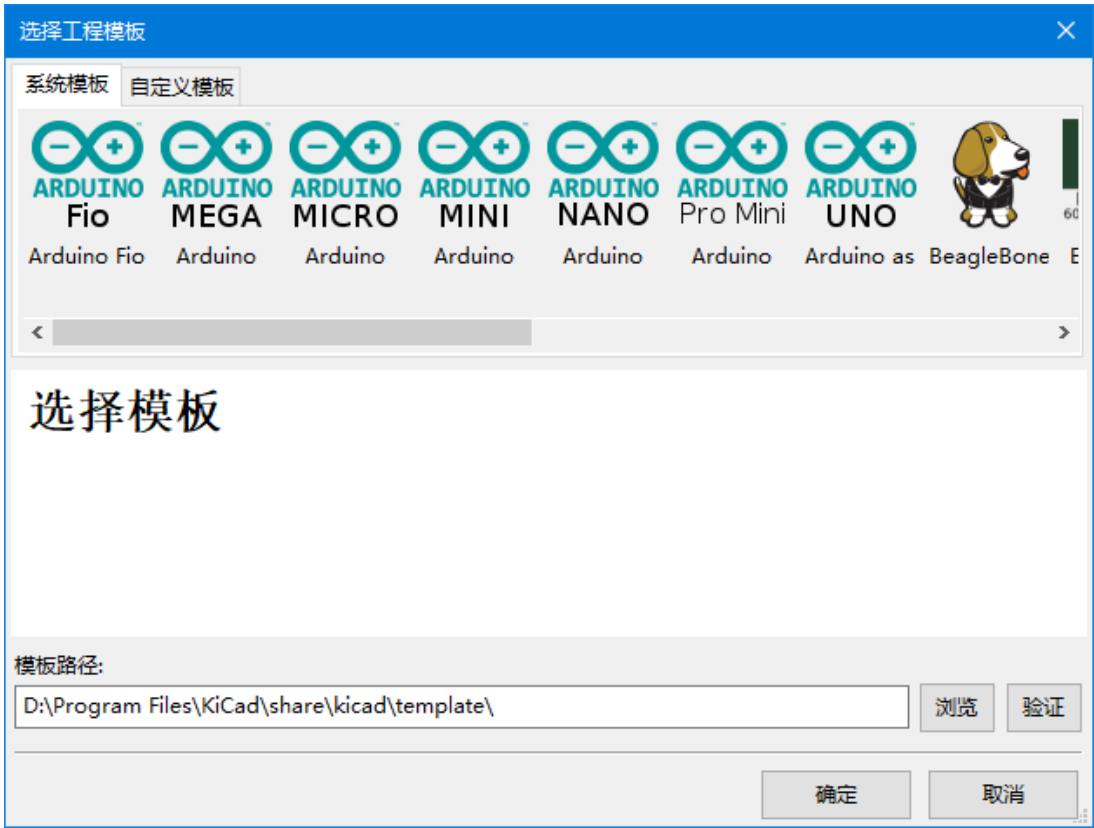
Chapter 5

5. 工程模板

工程模板是 KiCad 中用于快速创建新工程的功能。它允许用户选择预定义的模板，如 Arduino Fio、MEGA、MICRO、MINI、NANO、Pro Mini 和 UNO，或者自定义模板。选择模板后，KiCad 会自动生成项目文件并打开 PCB 编辑器。

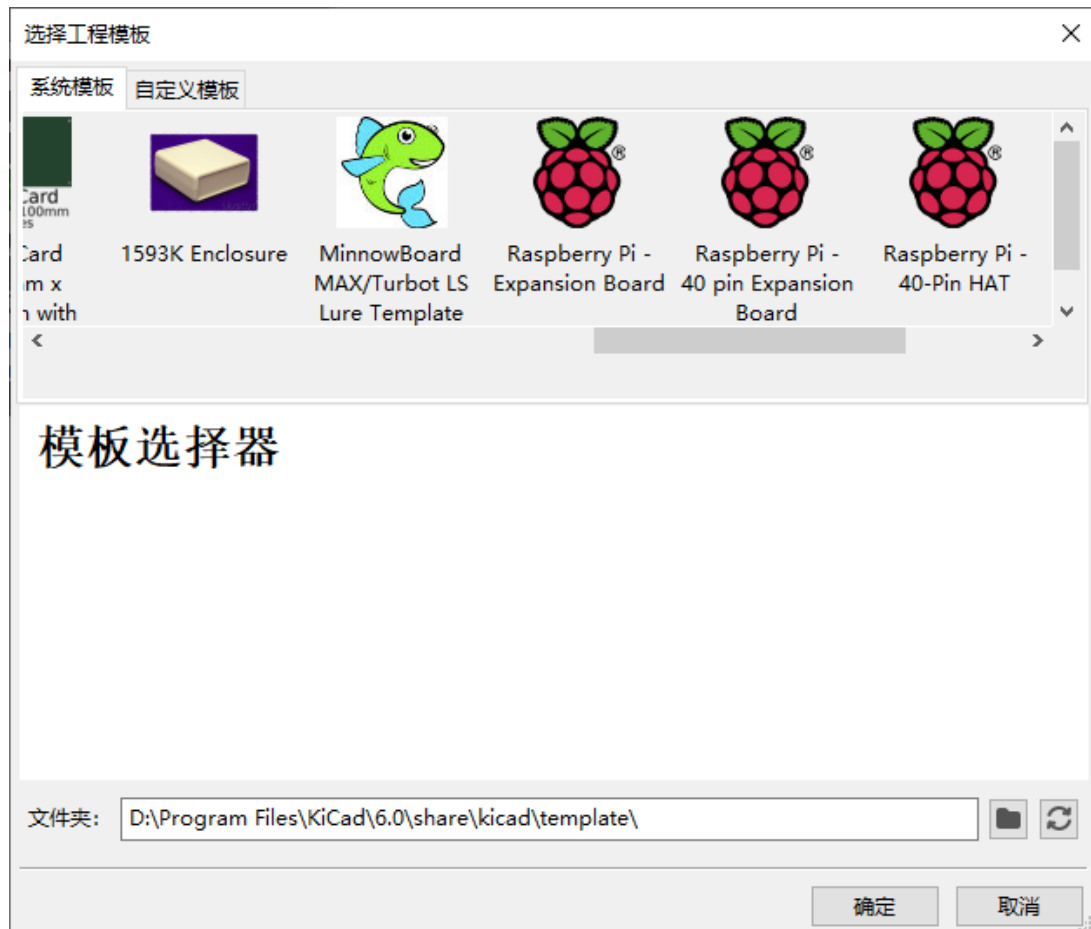
5.1 工程模板

在 KiCad 中选择工程模板的步骤如下：



在 KiCad 中选择工程模板的步骤如下：

1. 打开 KiCad 主界面。
2. 点击 "File" 菜单，选择 "New" 子菜单，然后点击 "Project"。
3. 在弹出的 "Select Project Template" 对话框中，选择所需的模板（如 Arduino Fio、MEGA、MICRO、MINI、NANO、Pro Mini 或 UNO）。
4. 点击 "OK" 按钮，KiCad 将自动创建项目文件并打开 PCB 编辑器。



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** 的模板创建工新工程 的工程：

模板 &#x793a;&#x4f8b; 目录中的文件名称的文	在工程 &#x65b0;&#x5de5;&#x7a0b; 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	&#x65b0;&#x5de5;&#x7a0b;.kicad_pro &#x65b0;&#x5de5;&#x7a0b;.kicad_sch &#x65b0;&#x5de5;&#x7a0b;.kicad_pcb &#x65b0;&#x5de5;&#x7a0b;- &#x7b2c;&#x4e00;&#x6b21;.kicad_sch &#x7b2c;&#x4e8c;&#x6b21;- &#x65b0;&#x5de5;&#x7a0b;.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。

模板 &#x793a;&#x4f8b; 目录中的文件名称的文	在 &#x65b0;&#x5de5;&#x7a0b; &#x65b0;&#x5de5;&#x7a0b; 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	&#x65b0;&#x5de5;&#x7a0b;.kicad_sch &#x7b2c;&#x4e00;&#x6b21;- &#x65b0;&#x5de5;&#x7a0b;.kicad_sch &#x7b2c;&#x4e00;&#x6b21;- &#x65b0;&#x5de5;&#x7a0b;.kicad_pcb &#x7b2c;&#x4e8c;&#x6b21;- &#x65b0;&#x5de5;&#x7a0b;.kicad_sch &#x7b2c;&#x4e8c;&#x6b21;- &#x65b0;&#x5de5;&#x7a0b;.kicad_pcb &#x65b0;&#x5de5;&#x7a0b;.kicad_pro (默认) &#x65b0;&#x5de5;&#x7a0b;.kicad_pcb (默认)

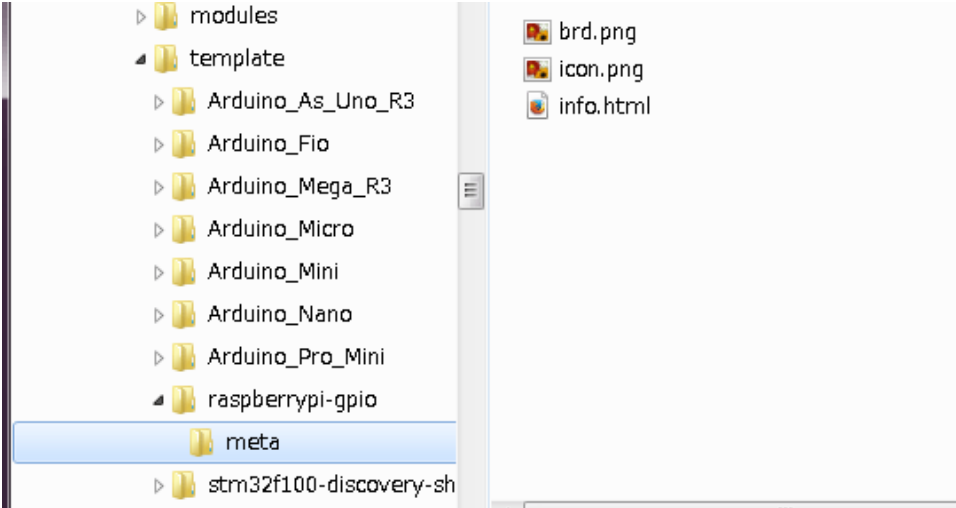
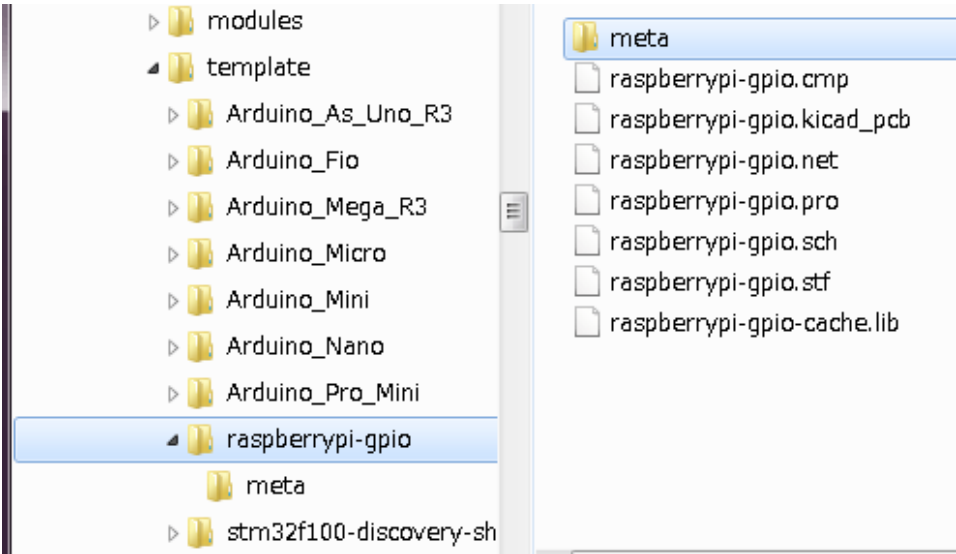
作为模板名称重命名规则的例外。
KiCad 将基于该工程文件名称进行重

模板 &#x793a;&#x4f8b; 目录中的文件名称的文	在 &#x65b0;&#x5de5;&#x7a0b; &#x65b0;&#x5de5;&#x7a0b; 目录中创建的文
示例.kicad_sch 示例.kicad_pcb &#x7b2c;&#x4e00;&#x6b21;- &#x793a;&#x4f8b;.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 &#x65b0;&#x5de5;&#x7a0b;.kicad_pro &#x65b0;&#x5de5;&#x7a0b;.kicad_sch &#x65b0;&#x5de5;&#x7a0b;.kicad_pcb 第二次- 示例.kicad_sch 第二次- 示例.kicad_pcb

Note

5.3.1

raspberrypi-gpio



5.3.2

meta/info.html	HTML
----------------	------

<title> 标记确定向用户公开以供选拢
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;&#x677f;
-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 $\frac{1}{2} \times \frac{1}{2} \times \frac{1}{2}$

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

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