Schematic Editor

The KiCad Team

Table of Contents

NOTE This manual is in the process of being revised to cover the latest stable release version of KiCad. It contains some sections that have not yet been completed. We ask for your patience while our volunteer technical writers work on this task, and we welcome new contributors who would like to help make KiCad’s documentation better than ever.

版权

贡献者

Jean-Pierre Charras, Fabrizio Tappero, Wayne Stambaugh, Graham Keeth

翻译人员

Liu HanCheng <buaa_cnlhc@buaa.edu.cn>, 2018.

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

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

反馈

The KiCad project welcomes feedback, bug reports, and suggestions related to the software or its documentation. For more information on how to sumbit feedback or report an issue, please see the instructions at https://www.kicad.org/help/report-an-issue/

Introduction to the KiCad Schematic Editor

描述

The KiCad Schematic Editor is a schematic capture software distributed as a part of KiCad and available under the following operating systems:

- Linux
- Apple OS X
- Windows

Regardless of the OS, all KiCad files are 100% compatible from one OS to another.

The Schematic Editor is an integrated application where all functions of drawing, control, layout, library management and access to the PCB design software are carried out within the editor itself.

The KiCad Schematic Editor is intended to cooperate with the KiCad PCB Editor, which is KiCad's printed circuit design software. It can also export netlist files, which lists all the electrical connections, for other packages.

The Schematic Editor includes a symbol library editor, which can create and edit symbols and manage libraries. It also integrates the following additional but essential functions needed for modern schematic capture software:

- 电气规则检查（ERC），用于自动控制错误和缺失的连接
- 以多种格式导出绘图文件（Postscript，PDF，HPGL和SVG）
- 物料清单生成（通过 Python 或 XSLT 脚本，允许许多灵活的格式）。

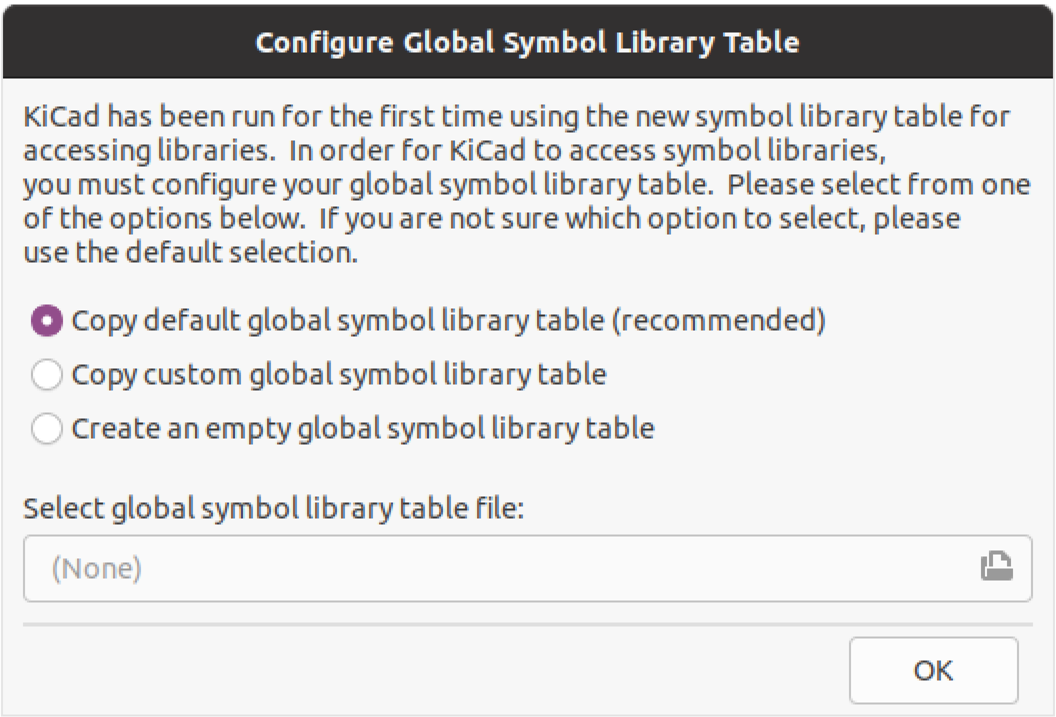
The Schematic Editor supports multi-sheet schematics in several ways:

- Flat hierarchies (schematic sheets are not explicitly connected in a master diagram).
- Simple hierarchies (each schematic sheet is used only once).
- Complex hierarchies (some schematic sheets are used multiple times).

Hierarchical schematics are described in detail later in the manual.

初始配置

When the Schematic Editor is run for the first time, if the the global symbol library table file `sym-lib-table` is not found in the KiCad configuration folder then KiCad will ask how to create this file:



Configure Global Symbol Library Table

KiCad has been run for the first time using the new symbol library table for accessing libraries. In order for KiCad to access symbol libraries, you must configure your global symbol library table. Please select from one of the options below. If you are not sure which option to select, please use the default selection.

● Copy default global symbol library table (recommended)

○ Copy custom global symbol library table

○ Create an empty global symbol library table

Select global symbol library table file:

(None)

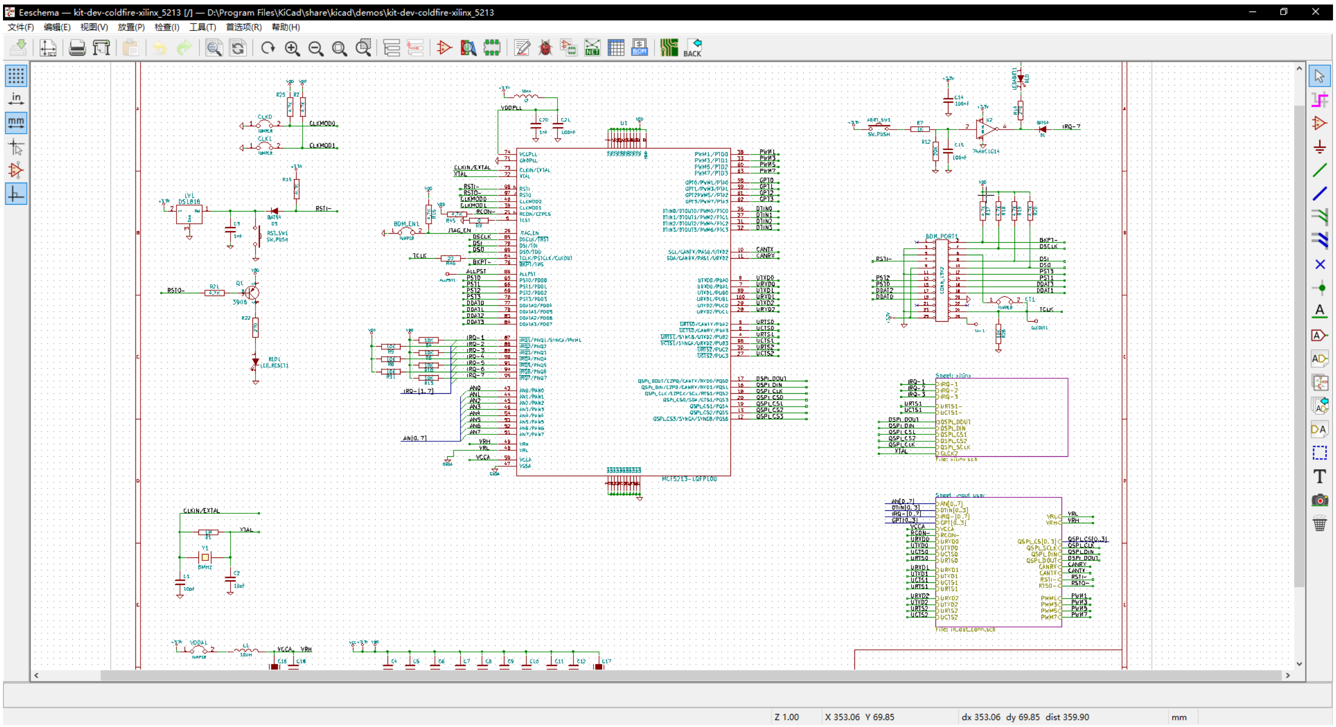
The first option is recommended (**Copy default global symbol library table (recommended)**). The default symbol library table includes all of the standard symbol libraries that are installed as part of KiCad.

If this option is disabled, KiCad was unable to find the default global symbol library table. This probably means you did not install the standard symbol libraries with KiCad, or they are not installed where KiCad expects to find them. On some systems the KiCad libraries are installed as a separate package.

- If you have installed the standard KiCad symbol libraries and want to use them, but the first option is disabled, select the second option and browse to the `sym-lib-table` file in the directory where the KiCad libraries were installed.
- If you already have a custom symbol library table that you would like to use, select the second option and browse to your `sym-lib-table` file.
- If you want to construct a new symbol library table from scratch, select the third option.

Symbol library management is described in more detail later.

The Schematic Editor User Interface


Eeschema — kit-dev-coldfire-xilinx_5213 [/] — D:\Program Files\KiCad\share\kicad\demos\kit-dev-coldfire-xilinx_5213
文件(F) 编辑(E) 视图(V) 放置(P) 检查(I) 工具(T) 首选项(R) 帮助(H)
BACK
in
mm
MCF5213-LQFP100
Z 1.00
X 353.06 Y 69.85
dx 353.06 dy 69.85 dist 359.90
mm

The main Schematic Editor user interface is shown above. The center contains the main editing canvas, which is surrounded by:

- Top toolbars (file management, zoom tools, editing tools)
- Left toolbar (display options)
- Message panel and status bar at bottom
- Right panel (drawing and design tools)

Navigating the editing canvas

The editing canvas displays the schematic being designed. You can pan and zoom to different parts of the schematic and open any schematic sheet in the design.

By default, dragging with the middle or right mouse button will pan the canvas view and scrolling the mouse wheel will zoom the view in or out. You can change this behavior in the Mouse and Touchpad section of the preferences (see Configuration and Customization for details).

Several other zoom tools are available in the top toolbar:

- zooms in on the center of the viewport.
- zooms out from the center of the viewport.
- zooms to fit the frame around the drawing sheet.
- zooms to fit the items within the drawing sheet.
- allows you to draw a box to determine the zoomed area.

The cursor's current position is displayed at the bottom of the window (X and Y), along with the current zoom factor (Z), the cursor's relative position (dx, dy, and dist), the grid setting, and the display units.

The relative coordinates can be reset to zero by pressing `Space`. This is useful for measuring distance between two points or aligning objects.

热键

The `Ctrl`+`F1` shortcut displays the current hotkey list. The default hotkey list is included in the Actions Reference section of the manual.

The hotkeys described in this manual use the key labels that appear on a standard PC keyboard. On an Apple keyboard layout, use the `Cmd` key in place of `Ctrl`, and the `Option` key in place of `Alt`.

Many actions do not have hotkeys assigned by default, but hotkeys can be assigned or redefined using the hotkey editor (**Preferences** → **Preferences...** → **Hotkeys**).

> **NOTE** Many of the actions available through hotkeys are also available in context menus. To access the context menu, right-click in the editing canvas. Different actions will be available depending on what is selected or what tool is active.

Hotkeys are stored in the file `user.hotkeys` in KiCad's configuration directory. The location is platform-specific:

- Windows: `%APPDATA%\kicad\6.0\user.hotkeys`
- Linux: `~/.config/kicad/6.0/user.hotkeys`
- macOS: `~/Library/Preferences/kicad/6.0/user.hotkeys`

KiCad can import hotkey settings from a `user.hotkeys` file using the **Import Hotkeys** button in the hotkey editor.

Mouse operations and selection

Selecting items in the editing canvas is done with the left mouse button. Single-clicking on an object will select it and dragging will perform a box selection. A box selection from left to right will only select items that are fully inside the box. A box selection from right to left will select any items that touch the box. A left-to-right selection box is drawn in yellow, with a cursor that indicates exclusive selection, and a right-to-left selection box is drawn in blue with a cursor that indicates inclusive selection.

The selection action can be modified by holding modifier keys while clicking or dragging. The following modifier keys apply when clicking to select single items:

Modifier Keys (Windows)	Modifier Keys (Linux)	Modifier Keys (macOS)	Selection Effect
Shift	Shift	Shift	Add the item to the existing selection.
Ctrl + Shift	Ctrl + Shift	Cmd + Shift	Remove the item from the existing selection.
long click	long click or Alt	long click or Option	Clarify selection from a pop-up menu.
Ctrl	Ctrl	Cmd	Highlight the net of the selected copper item.

The following modifier keys apply when dragging to perform a box selection:

Modifier Keys (Windows)	Modifier Keys (Linux)	Modifier Keys (macOS)	Selection Effect
Shift or Ctrl	Shift or Ctrl	Shift or Ctrl	Add item(s) to the existing selection.
Ctrl + Shift	Ctrl + Shift	Cmd + Shift	Remove item(s) from the existing selection.

Selecting an object displays information about the object in the message panel at the bottom of the window. Double-clicking an object opens a window to edit the object's properties.

Pressing Esc will always cancel the current tool or operation and return to the selection tool. Pressing Esc while the selection tool is active will clear the current selection.

Left toolbar display controls

The left toolbar provides options to change the display of items in the Schematic Editor.

⁙	Turns grid display on/off. **Note:** by default, hiding the grid will disable grid snapping. This behavior can be changed in the Display Options section of Preferences.
in mil mm	Display/entry of coordinates and dimensions in inches, mils, or millimeters.
✛	Switches between full-screen and small editing cursor (crosshairs).
▷	Turns invisible pin display on/off.
⊾	Switches between free angle and horizontal/vertical placement of new wires, buses, and graphical lines.

原理图创建和编辑

简介

A schematic designed with KiCad is more than a simple graphic representation of an electronic device. It is normally the entry point of a development chain that allows for:

- 验证一组规则（ERC，电气规则检查）以检测错误和遗漏。
- Automatically generating a bill of materials.
- 用于仿真软件（如 SPICE）的（创建 - 定制 - 网表和文件 - 文件，生成网表）。
- Defining a circuit for transferring to PCB layout.

原理图主要由符号，电线，标签，连接点，总线和电源端口组成。 为了清晰起见，您可以放置纯粹的图形元素，如总线条目，注释和折线。

Symbols are added to the schematic from symbol libraries. After the schematic is made, the set of connections and footprints is imported into the PCB editor for designing a board.

Schematics can be contained in a single sheet or split among multiple sheets. In KiCad, multi-sheet schematics are organized hierarchically, with a root sheet and sub-sheet(s). Each sheet is its own `.kicad_sch` file and is itself a complete KiCad schematic. Working with hierarchical schematics is described in the Hierarchical Schematics chapter.

Schematic editing operations

Schematic editing tools are located in the right toolbar. When a tool is activated, it stays active until a different tool is selected or the tool is canceled with the Esc key. The selection tool is always activated when any other tool is canceled.

↖	Selection tool (the default tool)
⌗	Highlight a net by marking its wires and net labels with a different color. If the PCB Editor is also open then copper corresponding to the selected net will be highlighted as well. Net highlighting can be cleared by clicking with the highlight tool in an empty space, or by using the Clear Net Highlighting hotkey (~).
▷	Display the symbol selector dialog to place a new symbol.
⏚	Display the power symbol selector dialog to place a new power symbol.
╱	Draw a wire.
╱	Draw a bus.
⋮	Draw wire-to-bus entry points. These elements are only graphical and do not create a connection, thus they should not be used to connect wires together.
✕	Place a "No Connect" flag. These flags should be placed on symbol pins which are meant to be left unconnected. "No connect" flags indicate to the Electrical Rule Checker that the pin is intentionally unconnected and not an error.
●	Place a junction. This connects two crossing wires or a wire and a pin, which can sometimes be ambiguous without a junction (i.e. if a wire end or a pin is not directly connected to another wire end).
A	Place a local label. Local labels connect items located **in the same sheet**. For connections between two different sheets, use global or hierarchical labels.
A	Place a global label. All global labels with the same name are connected, even when located on different sheets.
A◇	Place a hierarchical label. Hierarchical labels are used to create a connection between a subsheet and the sheet’s parent sheet. See the Hierarchical Schematics section for more information about hierarchical labels, sheets, and pins.
⊕	Place a hierarchical subsheet. You must specify the file name for this subsheet.
A◇	Import a hierarchical pin from a subsheet. This command can be executed only on hierarchical subsheets. It will create hierarchical pins corresponding to hierarchical labels placed in the target subsheet.
⋰	Draw lines. **Note:** Lines are graphical objects and are not the same as wires placed with the Wire tool. They do not connect anything.
T	Place a text comment.
🖼	Place a bitmap image.
⊗	Delete clicked items.

Grids

In the Schematic Editor the cursor always moves over a grid. The grid can be customized:

- Size can be changed using the right click menu or using **View** → **Grid Properties…**.
- Color can be changed in the **Colors** page of the **Preferences** dialog (menu **Preferences** → **General Options**).
- Visibility can be switched using the left-hand toolbar button.

The default grid size is 50 mil (0.050") or 1.27 millimeters.

This is the recommended grid for placing symbols and wires in a schematic, and for placing pins when designing a symbol in the Symbol Editor.

NOTE

Wires connect with other wires or pins only if their ends coincide **exactly**. Therefore it is very important to keep symbol pins and wires aligned to the grid. It is recommended to always use a 50 mil grid when placing symbols and drawing wires because the KiCad standard symbol library and all libraries that follow its style also use a 50 mil grid. **Using a grid size other than 50 mil will result in schematics without proper connectivity!**

Smaller grids can also be used, but this is intended only for text and symbol graphics, and not recommended for placing pins and wires.

NOTE

Symbols, wires, and other elements that are not aligned to the grid can be snapped back to the grid by selecting them, right clicking, and clicking **Align Elements to Grid**.

Snapping

Schematic elements such as symbols, wires, text, and graphic lines are snapped to the grid when moving, dragging, and drawing them. Additionally, the wire tool snaps to pins even when grid snapping is disabled. Both grid and pin snapping can be disabled while moving the mouse by using the modifier keys in the table below.

NOTE

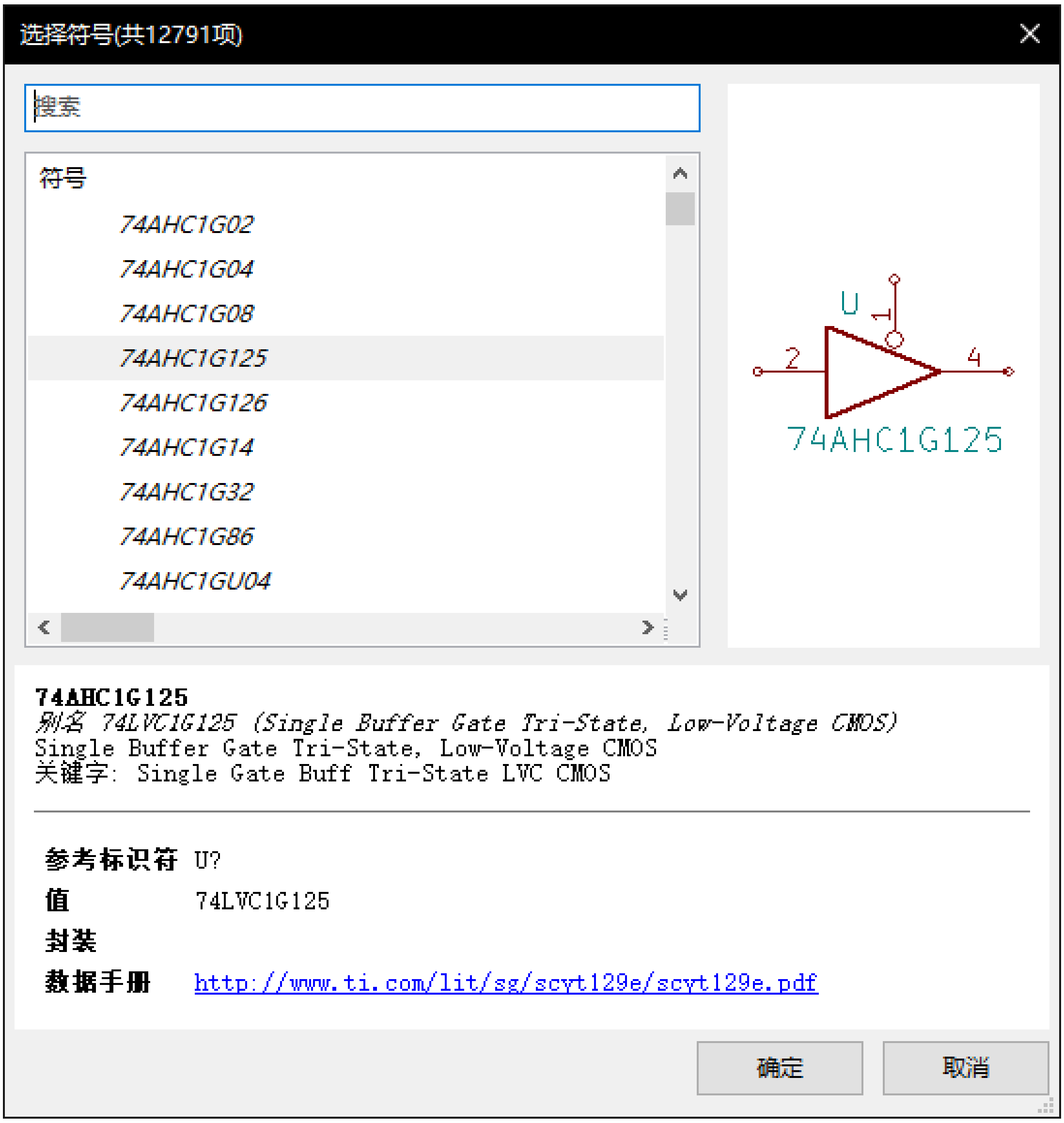
On Apple keyboards, use the `Cmd` key instead of `Ctrl`.

Modifier Key	**Effect**
`Ctrl`	Disable grid snapping.
`Shift`	Disable snapping wires to pins.

Working with symbols

Placing symbols

To load a symbol into your schematic you can use the icon . A dialog box allows you to type the name of the symbol to load.



选择符号(共12791项)

搜索

符号

- 74AHC1G02
- 74AHC1G04
- 74AHC1G08
- 74AHC1G125
- 74AHC1G126
- 74AHC1G14
- 74AHC1G32
- 74AHC1G86
- 74AHC1GU04

74AHC1G125
别名 74LVC1G125 (Single Buffer Gate Tri-State, Low-Voltage CMOS)
Single Buffer Gate Tri-State, Low-Voltage CMOS
关键字: Single Gate Buff Tri-State LVC CMOS

参考标识符	U?
值	74LVC1G125
封装	
数据手册	http://www.ti.com/lit/sg/scyt129e/scyt129e.pdf

The Choose Symbols dialog will filter symbols by name, keywords, and description according to what you type into the search field.

Some advanced filters are available:

- **Wildcards:** use the characters `?` and `*` respectively to mean "any single character or no characters" and "any number of any characters, including none".
- **Key-value pairs:** if a library part's description or keywords contain a tag of the format "Key:123", you can match relative to that by typing "Key>123" (greater than), "Key<123" (less than), etc. Numbers may include one of the following case-insensitive suffixes:

p	n	u	m	k	meg	g	t
10^{-12}	10^{-9}	10^{-6}	10^{-3}	10^{3}	10^{6}	10^{9}	10^{12}

ki	mi	gi	ti
2^{10}	2^{20}	2^{30}	2^{40}

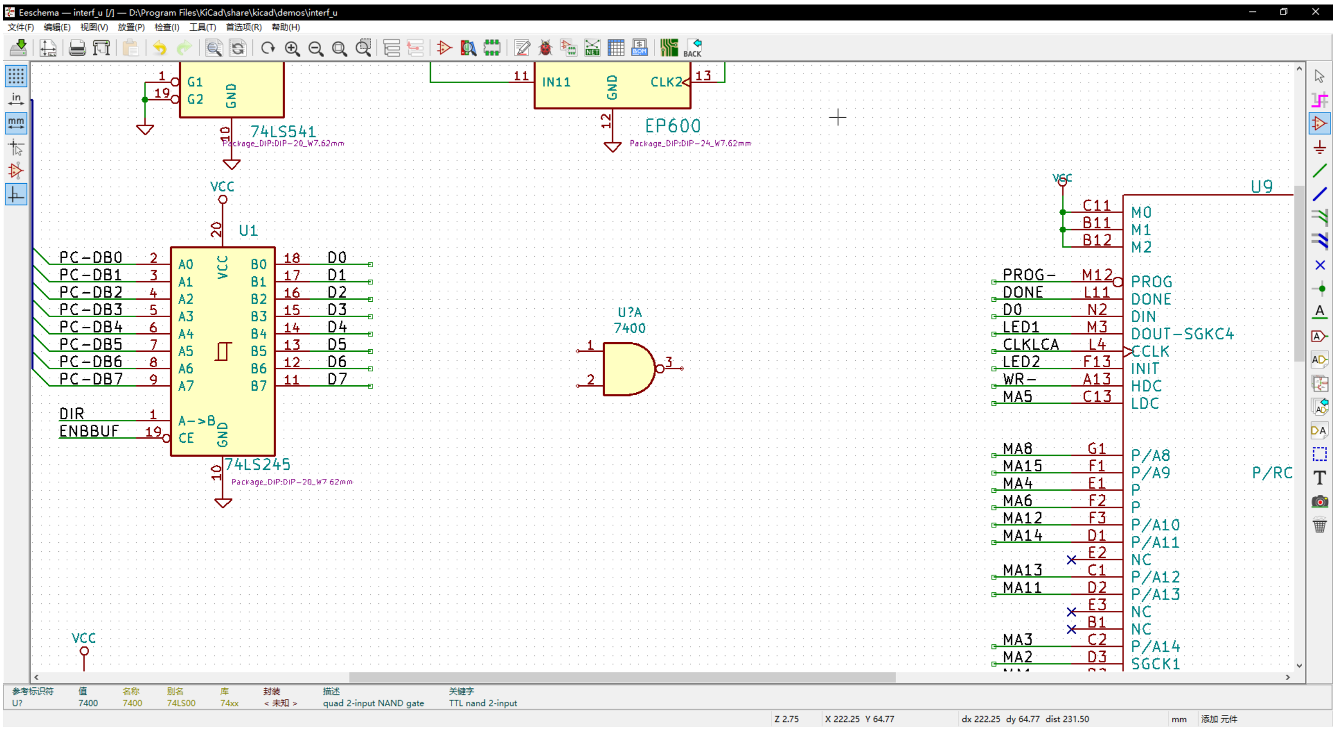
-

Regular expressions: if you're familiar with regular expressions, these can be used too. The regular expression flavor used is the wxWidgets Advanced Regular Expression style, which is similar to Perl regular expressions.

If the symbol specifies a default footprint, this footprint will be previewed in the lower right. If the symbol includes footprint filters, alternate footprints that satisfy the footprint filters can be selected in the footprint dropdown menu at right.

After selecting a symbol to place, the symbol will be attached to the cursor. Left clicking the desired location in the schematic places the symbol into the schematic. Before placing the symbol in the schematic, you can rotate it, mirror it, and edit its fields, by either using the hotkeys or the right-click context menu. These actions can also be performed after placement.

这是放置期间的符号：



参考标识符	值	名称	别名	库	封装	描述	关键字
U?	7400	7400	74LS00	74xx	< 未知 >	quad 2-input NAND gate	TTL nand 2-input

Z 2.75 X 222.25 Y 64.77 dx 222.25 dy 64.77 dist 231.50 mm 添加 元件

If the **Place repeated copies** option is checked, after placing a symbol KiCad will start placing another copy of the symbol. This process continues until the user presses Esc.

For symbols with multiple units, if the **Place all units** option is checked, after placing the symbol KiCad will start placing the next unit in the symbol. This continues until the last unit has been placed or the user presses Esc.

Placing power ports

A power port symbol is a symbol representing a connection to a power net. The symbols are grouped in the `power` library, so they can be placed using the symbol chooser. However, as power placements are frequent, the ⏚ tool is available. This tool is similar, except that the search is done directly in the `power` library and any other library that contains power symbols.

Moving symbols

Symbols can be moved using the Move (M) or Drag (G) tools. These tools act on the selected symbol, or if no symbol is selected they act on the symbol under the cursor.

The **Move** tool moves the symbol itself without maintaining wired connections to the symbol pins.

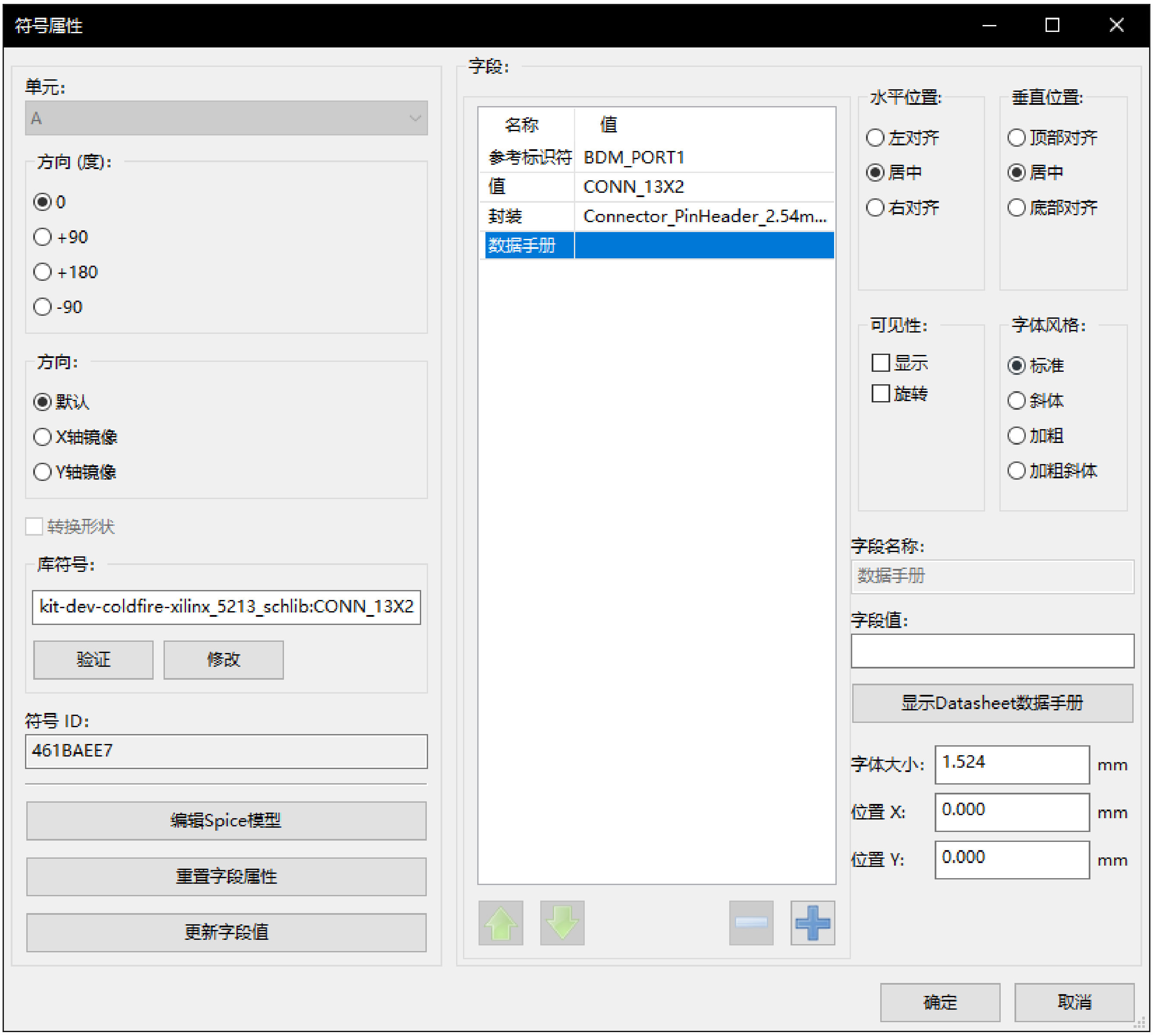
The **Drag** tool moves the symbol without breaking wired connections to its pins, and therefore moves the connected wires as well.

You can also Drag symbols by clicking and dragging them with the mouse, depending on the **Left button drag gesture** setting in the **Mouse and Touchpad** section of Preferences.

Symbols can also be rotated (R) or mirrored in the X (X) or Y (Y) directions.

Editing symbol properties

A symbol's fields can be edited in the symbol's Properties window. Open the Symbol Properties window for a symbol with the E hotkey or by double-clicking on the symbol.


符号属性

单元: A

方向 (度):
● 0
○ +90
○ +180
○ -90

方向:
● 默认
○ X轴镜像
○ Y轴镜像

□ 转换形状

库符号: kit-dev-coldfire-xilinx_5213_schlib:CONN_13X2

符号 ID: 461BAEE7

字段:

名称	值
参考标识符	BDM_PORT1
值	CONN_13X2
封装	Connector_PinHeader_2.54m...
数据手册	

水平位置:
○ 左对齐
● 居中
○ 右对齐

垂直位置:
○ 顶部对齐
● 居中
○ 底部对齐

可见性:
□ 显示
□ 旋转

字体风格:
● 标准
○ 斜体
○ 加粗
○ 加粗斜体

字段名称: 数据手册

字段值:

字体大小: 1.524 mm

位置 X: 0.000 mm

位置 Y: 0.000 mm

The Symbol Properties window displays all the fields of a symbol in a table. New fields can be added, and existing fields can be deleted, edited, reordered, moved, or resized.

每个字段都可以是可见的或隐藏的，并且可以水平或垂直显示。 始终为正常显示的符号（无旋转或镜像）指示显示的位置，并且相对于符号的锚点。

The position and orientation properties of each field may be hidden in this dialog. They can be shown by right-clicking on the column header of the fields table and enabling the "Orientation", "X Position", and/or "Y Position" columns. Other columns can be shown or hidden as desired.

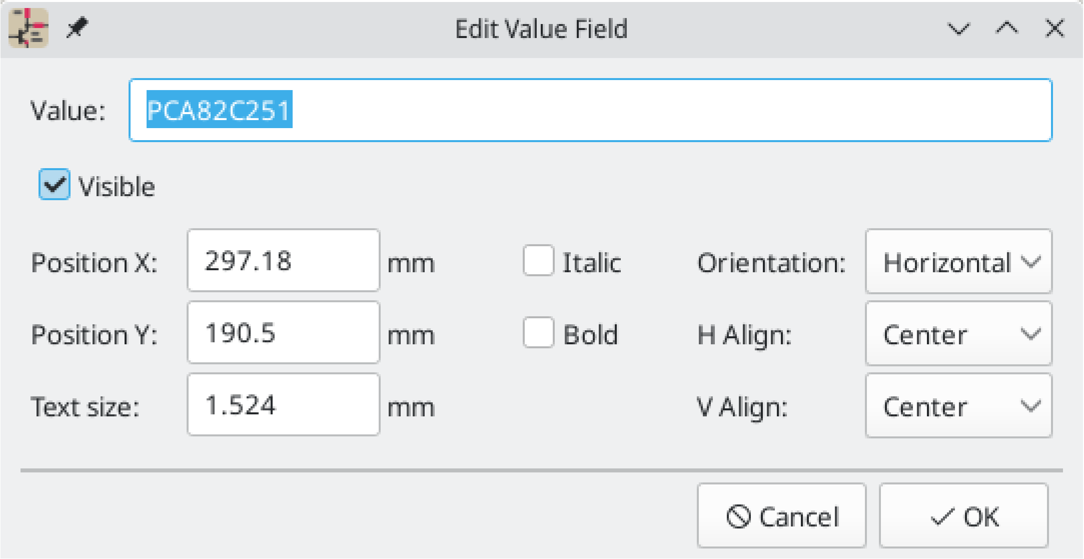
The "Update Symbol from Library…" button is used to update the schematic's copy of the symbol to match the copy in the library. The "Change Symbol…" button is used to swap the current symbol to a different symbol in the library.

"Edit Symbol…" opens the Symbol Editor to edit the copy of the symbol in the schematic. Note that the original symbol in the library will not be modified. The "Edit Library Symbol…" button opens the Symbol Editor to edit the original symbol in the library. In this case, the symbol in the schematic will not be modified until the user clicks the "Update Symbol from Library…" button.

Editing symbol fields individually

An individual symbol text field can be edited directly with the E hotkey (with a field selected instead of a symbol) or by double-clicking on the field.

Some symbol fields have their own hotkey to edit them directly. With the symbol selected, the Reference, Value, and Footprint fields can be edited with the U, V, or F hotkeys, respectively.



Value: PCA82C251

☑ Visible

Position X: 297.18 mm   □ Italic   Orientation: Horizontal

Position Y: 190.5 mm   □ Bold   H Align: Center

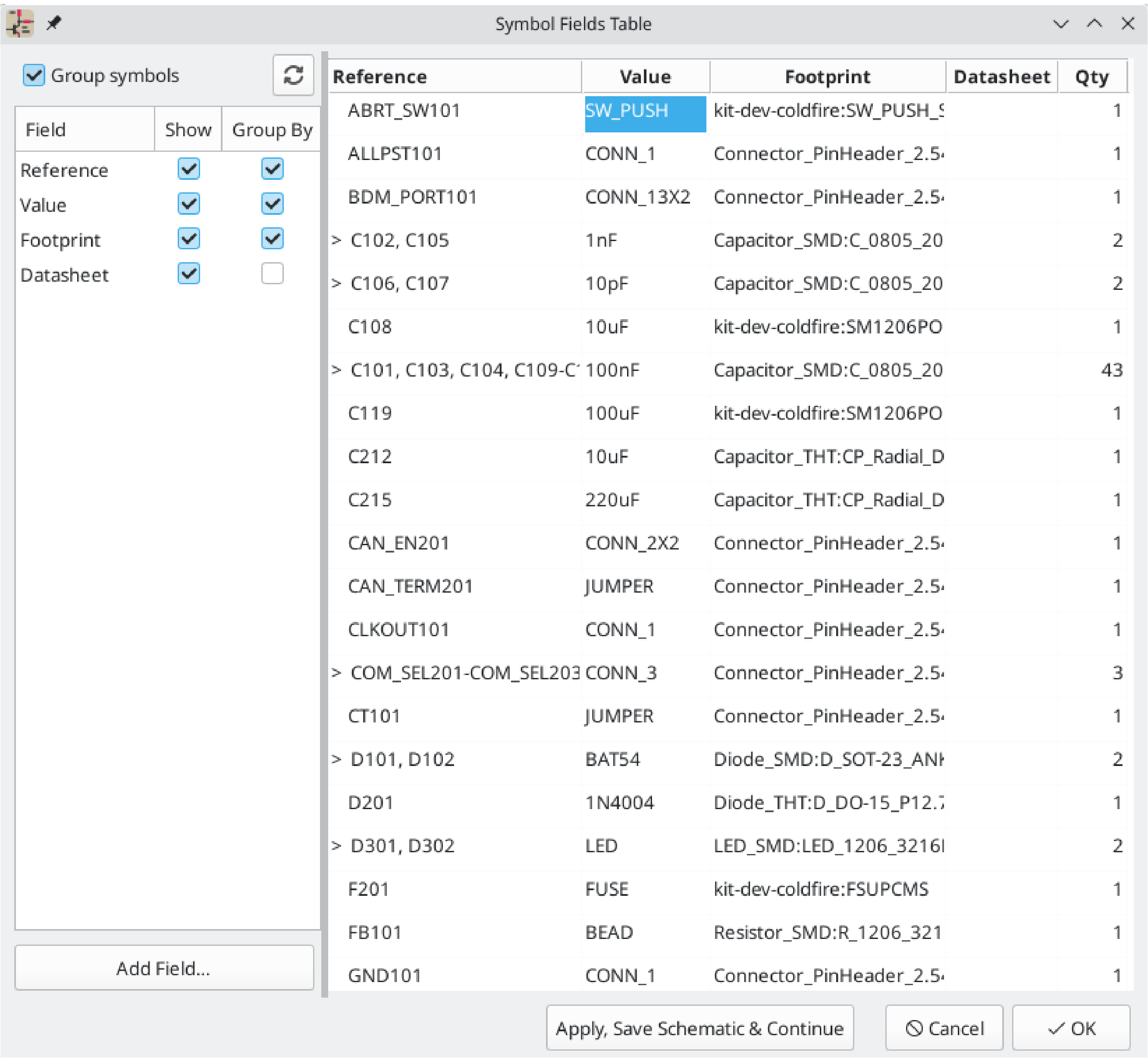
Text size: 1.524 mm   V Align: Center

The options in this dialog are the same as those in the full Symbol Properties dialog, but are specific to a single field.

Symbol fields can be automatically moved to an appropriate location with the Autoplace Fields action (select a symbol and press O). Field autoplacement is configurable in the Schematic Editor's Editing Options, including a setting to always autoplace fields.

Symbol Fields Table

The Symbol Fields Table allows you to view and modify field values for all symbols in a spreadsheet interface. You can open the Symbol Fields Table with the ▦ button.



Symbol Fields Table

☑ Group symbols

Field	Show	Group By
Reference	☑	☑
Value	☑	☑
Footprint	☑	☑
Datasheet	☑	□

Reference	Value	Footprint	Datasheet	Qty
ABRT_SW101	SW_PUSH	kit-dev-coldfire:SW_PUSH_S		1
ALLPST101	CONN_1	Connector_PinHeader_2.54		1
BDM_PORT101	CONN_13X2	Connector_PinHeader_2.54		1
> C102, C105	1nF	Capacitor_SMD:C_0805_20		2
> C106, C107	10pF	Capacitor_SMD:C_0805_20		2
C108	10uF	kit-dev-coldfire:SM1206PO		1
> C101, C103, C104, C109-C1	100nF	Capacitor_SMD:C_0805_20		43
C119	100uF	kit-dev-coldfire:SM1206PO		1
C212	10uF	Capacitor_THT:CP_Radial_D		1
C215	220uF	Capacitor_THT:CP_Radial_D		1
CAN_EN201	CONN_2X2	Connector_PinHeader_2.54		1
CAN_TERM201	JUMPER	Connector_PinHeader_2.54		1
CLKOUT101	CONN_1	Connector_PinHeader_2.54		1
> COM_SEL201-COM_SEL203	CONN_3	Connector_PinHeader_2.54		3
CT101	JUMPER	Connector_PinHeader_2.54		1
> D101, D102	BAT54	Diode_SMD:D_SOT-23_ANK		2
D201	1N4004	Diode_THT:D_DO-15_P12.7		1
> D301, D302	LED	LED_SMD:LED_1206_3216I		2
F201	FUSE	kit-dev-coldfire:FSUPCMS		1
FB101	BEAD	Resistor_SMD:R_1206_321		1
GND101	CONN_1	Connector_PinHeader_2.54		1

Cells are navigated with the arrow keys, or with Tab / Shift + Tab to move right / left and Enter / Shift + Enter to move down / up, respectively.

A range of cells can be selected by clicking and dragging. The whole range of selected cells will be copied (Ctrl + C) or pasted into (Ctrl + V) on a copy or paste action. Copying a range of cells from the table can be useful for creating a BOM. More details of copying and pasting cells are described below.

Any symbol field can be shown or hidden using the **Show** checkboxes on the left, or by right-clicking on the header of the table. New symbol fields can be added using the **Add Field...** button.

Similar symbols can optionally be grouped by any symbol field using the **Group By** checkboxes. Grouped symbols are shown in a single row in the table. The grouped row can be expanded to show the individual symbols by clicking the arrow at the left of the row.

Tricks to simplify filling fields

There are several special copy/paste methods in the spreadsheet for pasting values into larger regions, including auto-incrementing pasted cells. These features may be useful when pasting values that are shared in several symbols.

这些方法如下所示。

1. Copy (Ctrl + C)			2. Select target cells			3. Paste (Ctrl + V)		
abc			abc			abc		
						abc	abc	
						abc	abc	
11	12	13	11	12	13	11	12	13
						11	12	13
						11	12	13
						11	12	13
11			11			11	11	11
21			21			21	21	21
31			31			31	31	31
41			41			41	41	41
51			51			51	51	51
11	12		11	12		11	12	
21	22		21	22		21	22	
						11	12	11
						21	22	21
						11	12	11
11	12	13	11	12	13	11	12	13
21	22	23	21	22	23	21	22	23
						11	12	

NOTE 这些技术也可以在具有网格控制元素的其他对话框中使用。

Reference Designators and Symbol Annotation

Reference designators are unique identifiers for components in a design. They are often printed on a PCB and in assembly diagrams, and allow you to match symbols in a schematic to the corresponding components on a board.

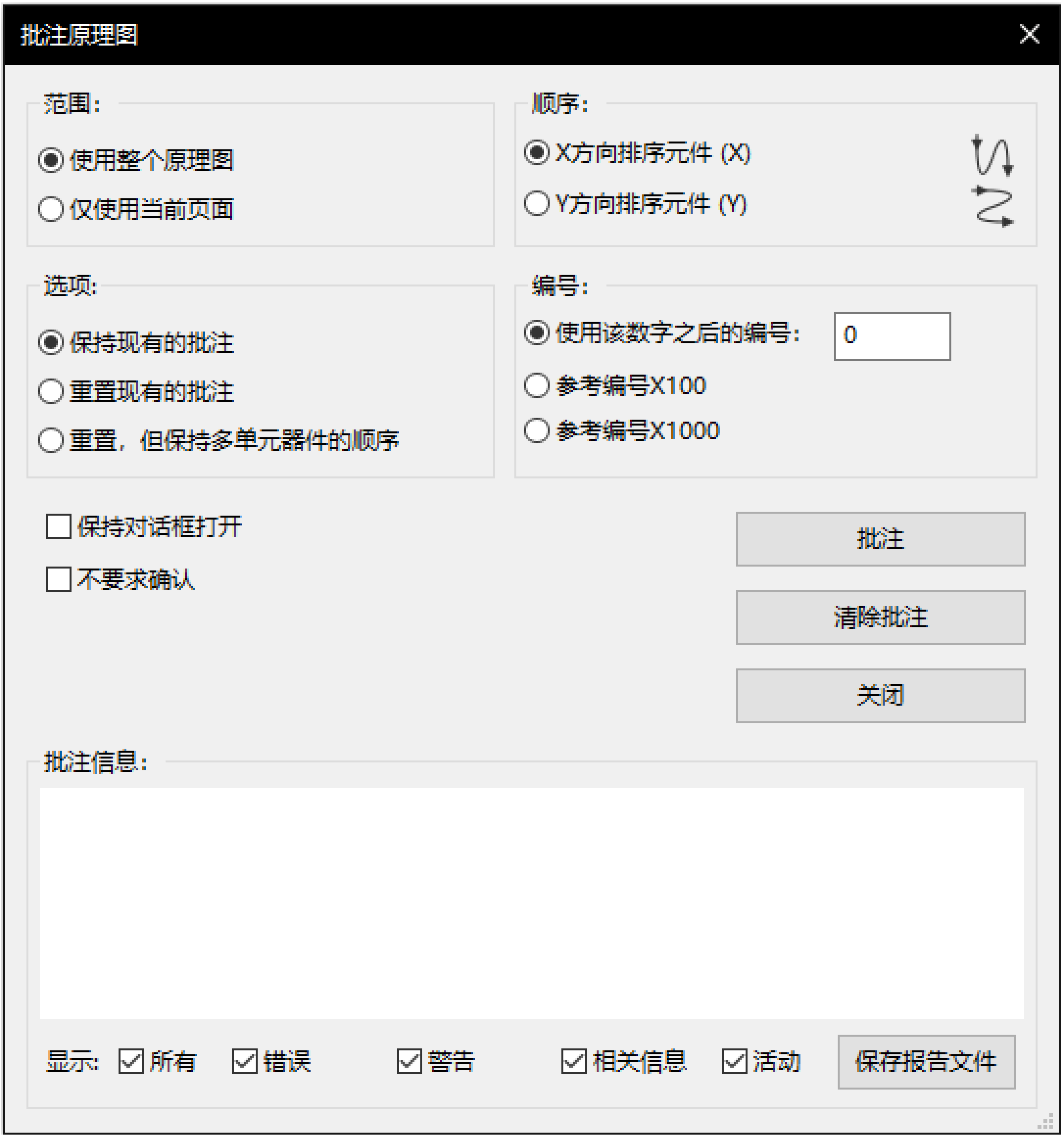
In KiCad, reference designators consist of a letter indicating the type of component (`R` for resistor, `C` for capacitor, `U` for IC, etc.) followed by a number. If the symbol has multiple units then the reference designator will also have a trailing letter indicating the unit. Symbols that don't have a reference designator set have a `?` character instead of the number. Reference designators must be unique.

Reference designators can set manually by editing a symbol's reference designator field, or automatically using the Annotation tool.

> **NOTE** The process of setting a symbol's reference designator is called **annotation**.

批注工具

The Annotation tool automatically assigns reference designators to symbols in the schematic. To launch the Annotation tool, click the R?? R42 button in the top toolbar.



批注原理图

范围:
● 使用整个原理图
○ 仅使用当前页面

顺序:
● X方向排序元件 (X)
○ Y方向排序元件 (Y)

选项:
● 保持现有的批注
○ 重置现有的批注
○ 重置，但保持多单元器件的顺序

编号:
● 使用该数字之后的编号: 0
○ 参考编号X100
○ 参考编号X1000

□ 保持对话框打开
□ 不要求确认

批注信息:

显示: ☑ 所有 ☑ 错误 ☑ 警告 ☑ 相关信息 ☑ 活动

The tool provides several options to control how symbols are annotated.

Scope: Selects whether annotation is applied to the entire schematic, to only the current sheet, or to only the selected symbols.

Options: Selects whether annotation should apply to all symbols and reset *existing reference designators, or apply only to unannotated symbols.

Order: Chooses the direction of numbering. If symbols are sorted by X position, all symbols on the left side of a schematic sheet will be lower numbered than symbols on the right side of the sheet. If symbols are sorted by Y position, all symbols on the top of a sheet will be lower numbered than symbols at the bottom of the sheet.

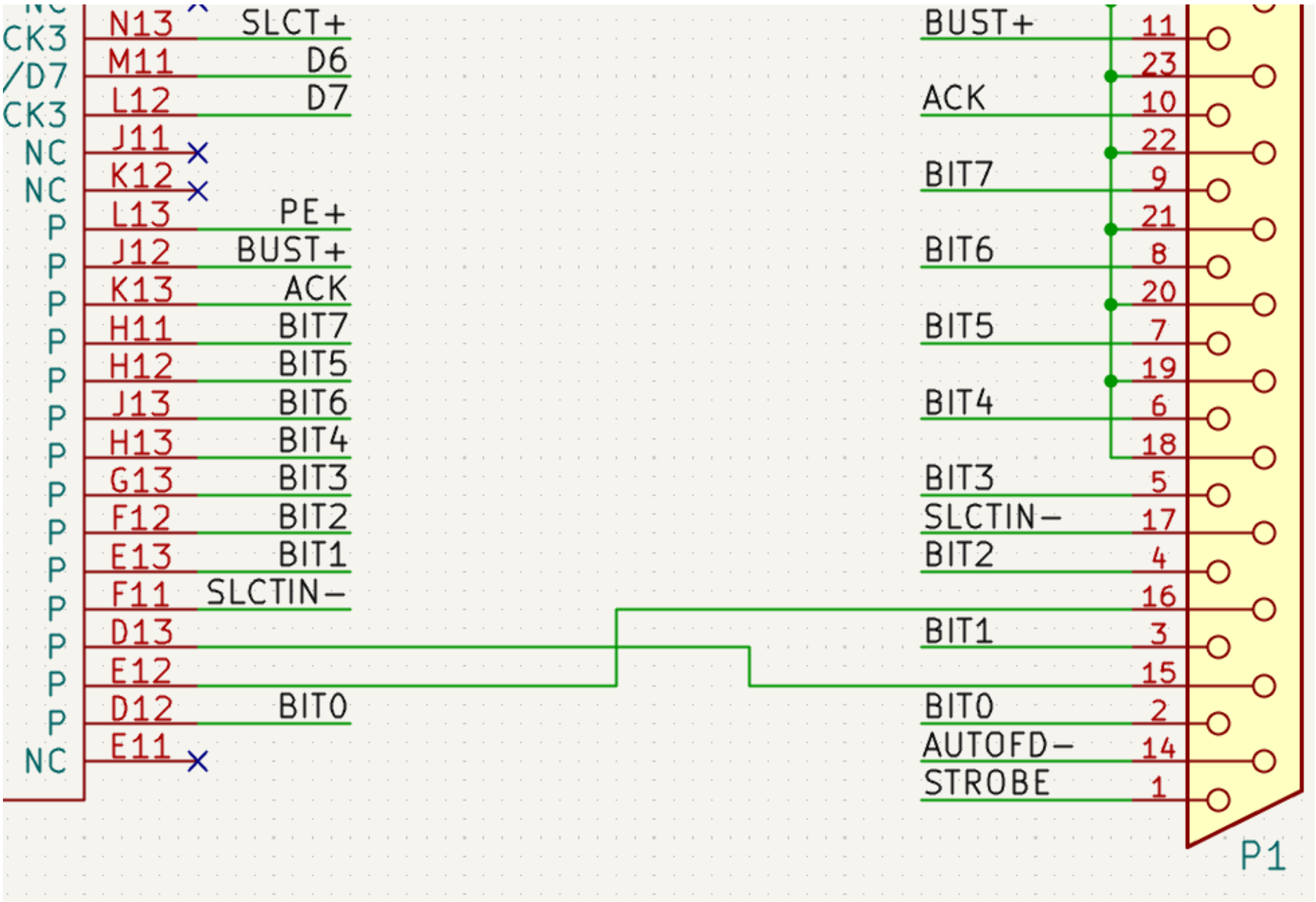
Numbering: Selects the starting point for numbering reference designators. The lowest unused number above the starting point is picked for each reference designator. The starting point can be an arbitrary number (typically zero), or it can be the sheet number multiplied by 100 or 1000 so that each part's reference designator corresponds to the schematic page it is on.

The **Clear Annotation** button clears all reference designators in the selected scope.

Annotation messages can be filtered with the checkboxes at the bottom or saved to a report using the **Save…** button.

Electrical Connections

There are two primary ways to establish connections: wires and labels. Both are shown in the schematic below.



Connections can also be made with buses and with implicit connections via hidden power pins.

This section will also discuss two special types of symbols that can be added with the "Power port" button on the right toolbar:

- **Power ports**: symbols for connecting wires to a power or ground net.
- **PWR_FLAG**: a specific symbol for indicating that a net is powered when it is not connected to a power output pin (for example, a power net that is supplied by an off-board connector).

Label Connections

Labels are used to assign net names to wires and pins. Wires with the same net name are considered to be connected. A net can only have one name. If two different labels are placed on the same net, an ERC violation will be generated. Only one of the net names will be used in the netlist. The final net name is determined according to the rules described below.

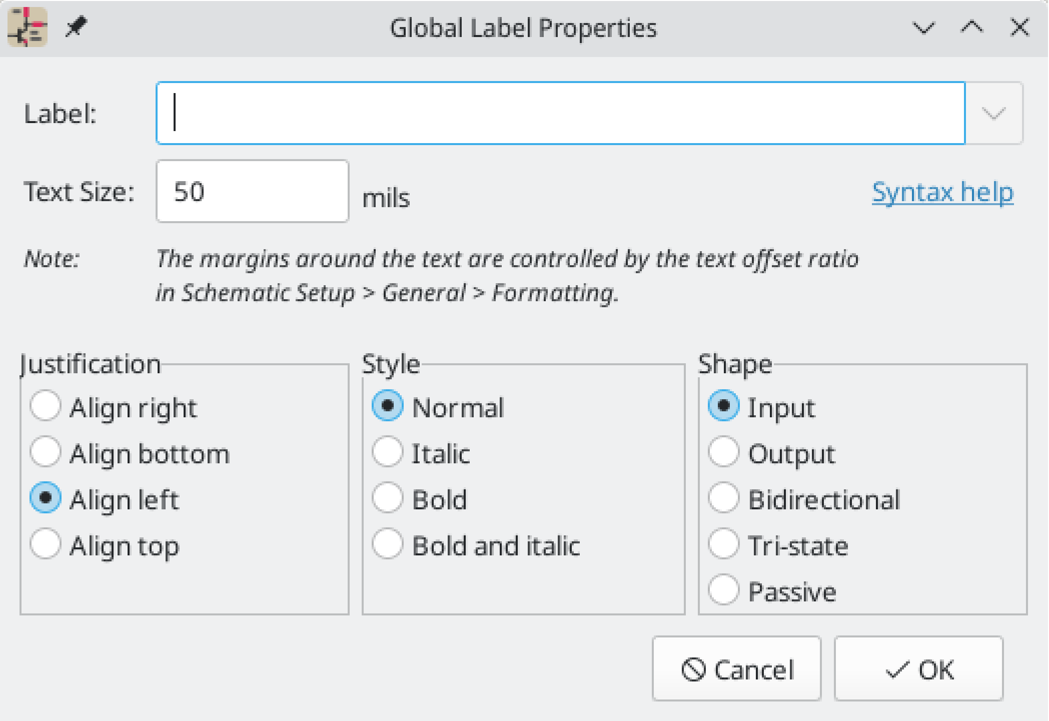
There are three types of labels, each with a different connection scope.

- **Local labels**, also referred to simply as labels, only make connections within a sheet. Add a local label with the A button in the right toolbar.
- **Global labels** make connections anywhere in a schematic, regardless of sheet. Add a global label with the A button in the right toolbar.
- **Hierarchical labels** connect to hierarchical sheet pins and are used in hierarchical schematics for connecting child sheets to their parent sheet. Add a hierarchical label with the A◇ button in the right toolbar.

NOTE

Labels that have the same name will connect, regardless of the label type, if they are in the same sheet.

After using the appropriate button or hotkey to create a label, the Label Properties dialog appears.



Global Label Properties

Label:

Text Size: 50 mils

Syntax help

Note: *The margins around the text are controlled by the text offset ratio in Schematic Setup > General > Formatting.*

Justification
- ○ Align right
- ○ Align bottom
- ● Align left
- ○ Align top

Style
- ● Normal
- ○ Italic
- ○ Bold
- ○ Bold and italic

Shape
- ● Input
- ○ Output
- ○ Bidirectional
- ○ Tri-state
- ○ Passive

The **Label** field sets the label's text, which determines the net that the label assigns to its attached wire. Label text supports markup for overbars, subscripts, etc., as well as variable substitution. Use the **Syntax help** link in the dialog for a summary.

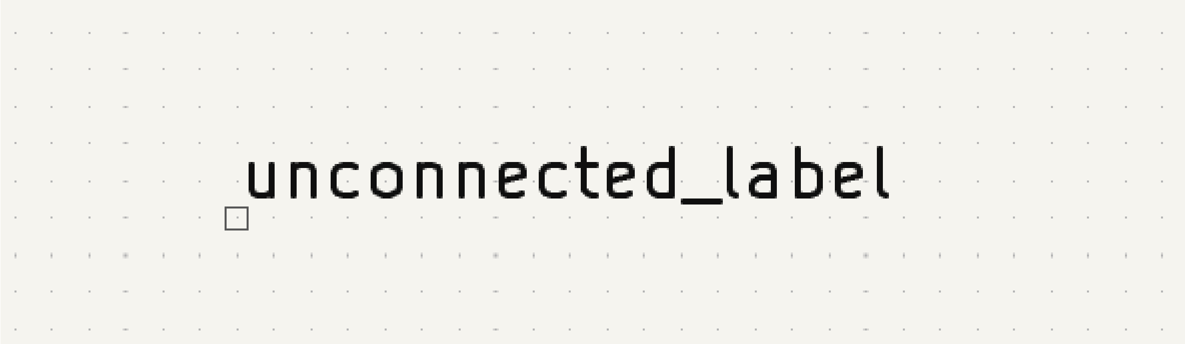
Justification sets the position of the label's connection point relative to the label's text. For example, when **Align right** is selected the connection point will be to the right of the text.

Text size and **Style** control the appearance of the label's text. **Shape** controls the shape of the outline around the label; this is purely visual and has no electrical consequence. Local labels do not have an outline, and therefore do not have **Shape** options.

NOTE

Global labels have additional settings to control margins around the label text in the Schematic Setup dialog.

After accepting the label properties, the label is attached to the cursor for placement. The connection point for a label is the small square in the corner of the label. The square disappears when the label is connected to a wire or the end of a pin.



The connection point's position relative to the label text can be changed by choosing a different label orientation in the label's properties, or by mirroring/rotating the label.

The Label Properties dialog can be accessed at any time by selecting a label and using the E hotkey, double-clicking on the label, or with **Properties...** in the right-click context menu.

Wire Connections

To establish a connection, a segment of wire must be connected by its end to another segment or to a pin. Only wire ends create connections; if a wire crosses the middle of another wire, a connection will not be made.

Unconnected wire ends have a small square that indicates the connection point. The square disappears when a connection is made to the wire end. Unconnected pins have a circle, which also disappears when a connection is made.

NOTE

Wires connect with other wires or pins only if their ends coincide exactly. Therefore it is important to keep symbol pins and wires aligned to the grid. It is recommended to always use a 50 mil grid when placing symbols and drawing wires because the KiCad standard symbol library and all libraries that follow its style also use a 50 mil grid.

NOTE

Symbols, wires, and other elements that are not aligned to the grid can be snapped back to the grid by selecting them, right clicking, and selecting **Align Elements to Grid**.

Drawing and editing wires

To begin connecting elements with wire, use the Wire tool in the right toolbar (W). Wires can also be automatically started by clicking on an unconnected symbol pin or wire end.

Wires can be moved using the Move (M) or Drag (G) tools. As with symbols, the **Move** tool moves only the selected segment, without maintaining existing connections to other segments. The **Drag** tool maintains existing connections.

If a segment is selected or the cursor is over the middle of a wire, the move/drag action will move the entire segment. If the cursor is over a corner or wire end, the move/drag action will act on one end of the segment.

Wire Junctions

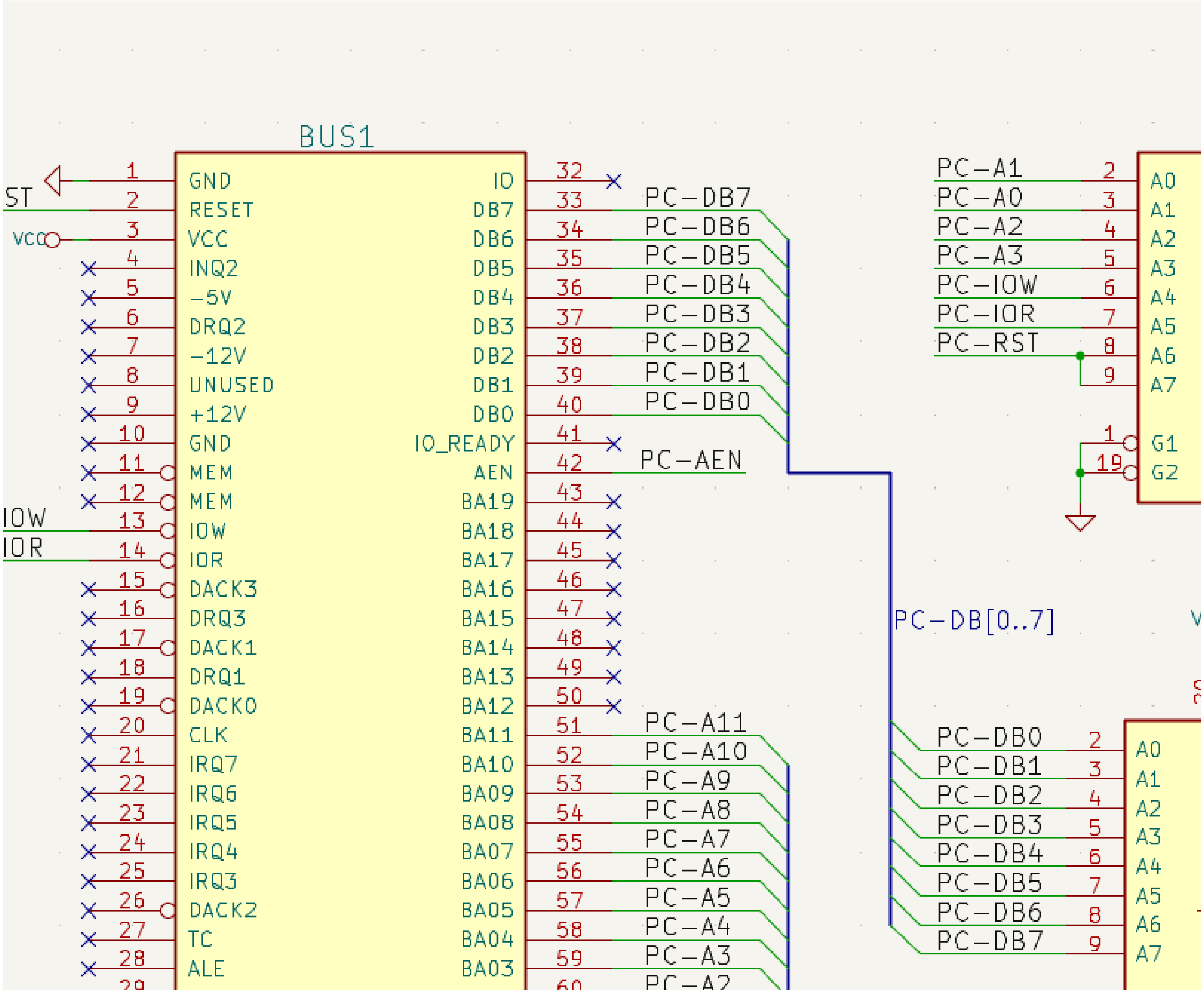
Wires that cross are not implicitly connected. It is necessary to join them by explicitly adding a junction dot if a connection is desired (button in the right toolbar). Junction dots will be automatically added to wires that start or end on top of an existing wire.

Junction dots are used in the schematic figure above on the wires connected to `P1` pins 18, 19, 20, 21, 22, and 23.

Bus Connections

Buses are a way to group related signals in the schematic in order to simplify complicated designs. Buses can be drawn like wires using the bus tool , and are named using labels the same way signal wires are.

In the following schematic, many pins are connected to buses, which are the thick blue lines in the center.



总线编号

There are two types of bus in KiCad 6.0 and later: vector buses and group buses.

一个 **向量总线** 是以公共前缀开头并以数字结尾的信号集合。 向量总线命名为‘<PREFIX> [M..N]’，其中‘PREFIX’是任何有效的信号名称，‘M’是第一个后缀号，‘N’是最后一个后缀号。 例如，总线‘DATA [0..7]’包含信号‘DATA0’，‘DATA1’，依此类推，直到‘DATA7’。 指定‘M’和‘N’的顺序无关紧要，但两者都必须是非负的。

一个 **组总线** 是一个或多个信号和/或矢量总线的集合。 组总线可用于将相关信号捆绑在一起，即使它们具有不同的名称。 组总线使用特殊标签语法：

‘<OPTIONAL_NAME>{SIGNAL1 SIGNAL2 SIGNAL3}’

该组的成员列在由空格字符分隔的花括号（‘{}’）内。 该组的可选名称位于左大括号之前。 如果组总线未命名，则 PCB 上生成的网络将只是组内的信号名称。 如果组总线具有名称，则生成的网络将具有名称作为前缀，其中句点（‘.’）将前缀与信号名称分开。

例如，总线‘{SCL SDA}’有两个信号成员，在网表中这些信号将是‘SCL’和‘SDA’。 总线“USB1 {DP DM}”将生成名为“USB1.DP”和“USB1.DM”的网络。 对于在几个类似电路上重复使用较大总线的设计，使用这种技术可以节省时间。

组总线还可以包含矢量总线。 例如，总线‘MEMORY {A [7..0] D [7..0] OE WE}’包含矢量总线和普通信号，并将产生诸如“MEMORY.A7”和“MEMORY.OE”之类的网络在 PCB 上的。

Bus wires can be drawn and connected in the same manner as signal wires, including using junctions to create connections between crossing wires. Like signals, buses cannot have more than one name — if two conflicting labels are attached to the same bus, an ERC violation will be generated.

总线成员之间的连接

Pins connected between the same members of a bus must be connected by labels. It is not possible to connect a pin directly to a bus; this type of connection will be ignored by KiCad.

在上面的示例中，连接是通过放置在连接到引脚的导线上的标签进行的。 到总线的总线入口（45度线段）仅是图形化的，并不是形成逻辑连接所必需的。

In fact, using the repetition command (`Insert`), connections can be very quickly made in the following way, if component pins are aligned in increasing order (a common case in practice on components such as memories, microprocessors…):

- Place the first label (for example `PCA0`)
- Use the repetition command as much as needed to place members. KiCad will automatically create the next labels (`PCA1` , `PCA2` …) vertically aligned, theoretically on the position of the other pins.
- 在第一个标签下画线。 然后使用重复命令将其他导线放在标签下。
- 如果需要，以相同的方式放置总线条目（放置第一个条目，然后使用重复命令）。

NOTE

In the **Schematic Editor** → **Editing Options** section of the Preferences menu, you can set the repetition parameters:

- Horizontal pitch
- Vertical pitch
- Label increment (labels can be incremented or decremented by 1, 2, 3, etc.)

总线正在展开

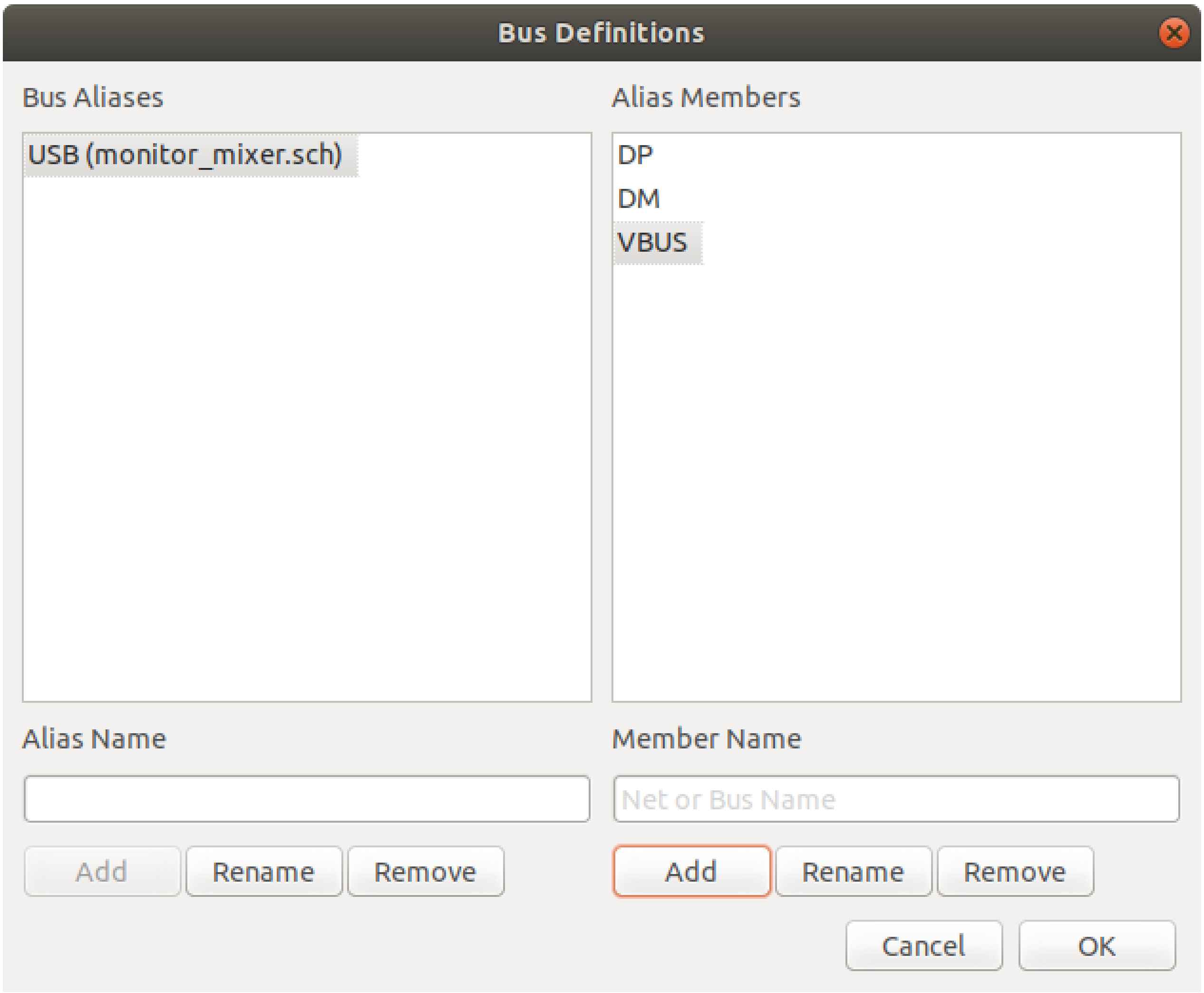
The unfold tool allows you to quickly break out signals from a bus. To unfold a signal, right-click on a bus object (a bus wire, etc) and choose **Unfold from Bus**. Alternatively, use the **Unfold Bus** hotkey (default: `C`) when the cursor is over a bus object. The menu allows you to select which bus member to unfold.

选择总线成员后，下一次单击将把总线成员标签放在所需位置。 该工具自动生成总线入口和导线，通向标签位置。放置标签后，您可以继续放置其他线段（例如，连接到组件引脚）并以任何正常方式完成线缆。

总线别名

总线别名是一种快捷方式，可让您更有效地使用大型组总线。 它们允许您定义组总线并为其指定一个简短的名称，然后可以在原理图中使用该名称而不是完整的组名。

To create bus aliases, open the **Bus Definitions** dialog in the **Tools** menu.



USB (monitor_mixer.sch)

DP
DM
VBUS

别名可以被命名为任何有效的信号名称。 使用该对话框，您可以向别名添加信号或矢量总线。 作为一种快捷方式，您可以键入或粘贴由空格分隔的信号和/或总线列表，并将它们全部添加到别名定义中。 在这个例子中，我们定义了一个名为'USB'的别名，其成员为"DP"，"DM"和"VBUS"。

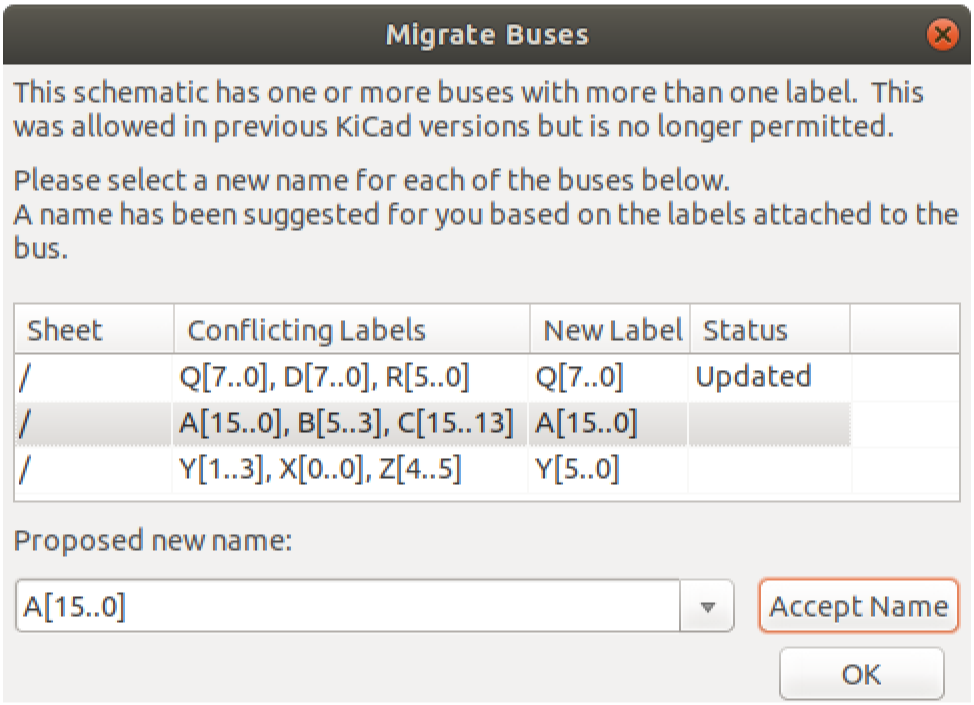
定义别名后，可以通过将别名放在组总线的大括号内来用于组总线标签：'{USB}'。 这与标记总线"{DP DM VBUS}"具有相同的效果。 您还可以为组添加前缀名称，例如"USB1 {USB}"，这会产生如上所述的"USB1.DP"等网络。 对于复杂的总线，使用别名可以使原理图上的标签更短。 请记住，别名只是一个快捷方式，别名的名称不包含在网表中。

总线别名保存在原理图文件中。 在给定的原理图工作表中创建的任何别名都可用于同一层次结构设计中的任何其他原理图工作表。

有多个标签的总线

KiCad 5.0 及更早版本允许将具有不同标签的总线连接在一起，并且在网络列表期间将加入这些总线的成员。 此行为已在 KiCad 6.0 中删除，因为它与组总线不兼容，并且还导致令人困惑的网表，因为不容易预测给定信号将接收的名称。

如果您在现代版本的 KiCad 中打开使用此功能的设计，您将看到"迁移总线"对话框，该对话框将指导您更新原理图，以便在任何给定的总线线路上只存在一个标签。



Migrate Buses

This schematic has one or more buses with more than one label. This was allowed in previous KiCad versions but is no longer permitted.

Please select a new name for each of the buses below.
A name has been suggested for you based on the labels attached to the bus.

Sheet	Conflicting Labels	New Label	Status
/	Q[7..0], D[7..0], R[5..0]	Q[7..0]	Updated
/	A[15..0], B[5..3], C[15..13]	A[15..0]	
/	Y[1..3], X[0..0], Z[4..5]	Y[5..0]	

Proposed new name: A[15..0]

对于具有多个标签的每组总线，您必须选择要保留的标签。 下拉名称框允许您在设计中存在的标签之间进行选择，或者您可以通过手动将其输入新名称字段来选择其他名称。

Hidden Power Pins

When the power pins of a symbol are visible, they must be connected, as with any other signal. However, symbols such as gates and flip-flops are sometimes drawn with hidden power input pins which are connected implicitly.

KiCad automatically connects invisible pins with type "power input" to a global net with the same name as the pin. For example, if a symbol has a hidden power input pin named `VCC`, this pin will be globally connected to the `VCC` net on all sheets.

NOTE	Hidden pins can be shown in the schematic by checking the **Show hidden pins** option in the **Schematic Editor** → **Display Options** section of the preferences, or by selecting **View** → **Show hidden pins**. There is also a toggle icon on the left toolbar.

It may be necessary to join power nets of different names (for example, `GND` in TTL components and `VSS` in MOS components). To accomplish this, add a power port symbol for each net and connect them with a wire.

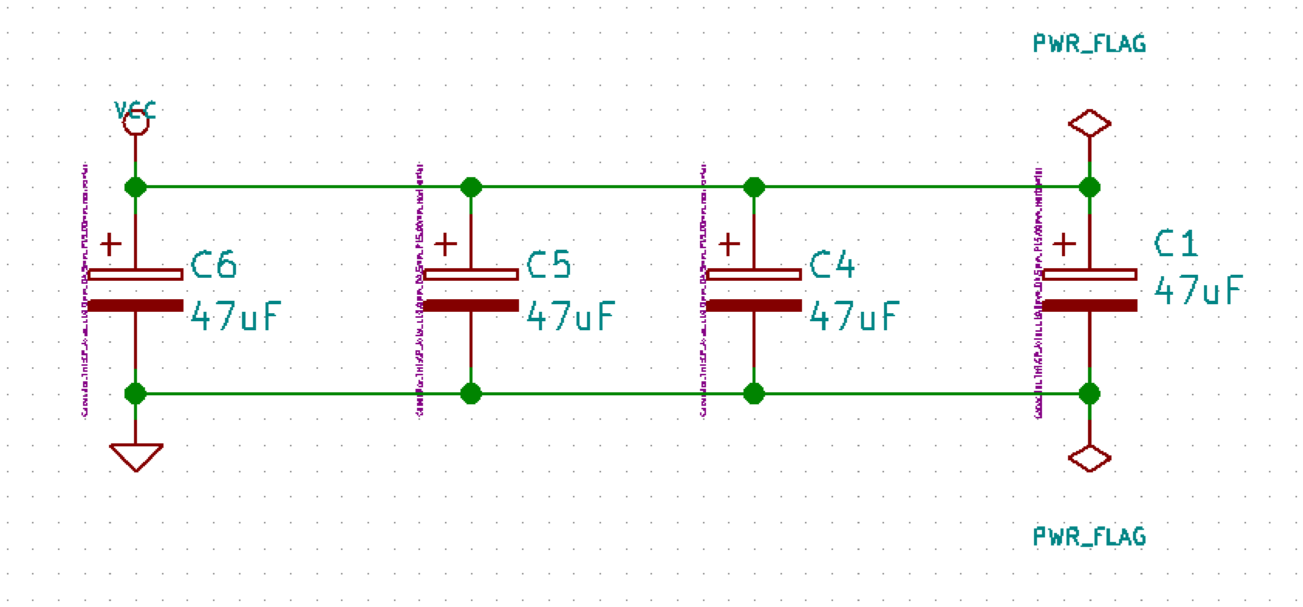
If hidden power pins are used, it is not recommended to use local labels for power connection, as they will not connect to hidden power pins on other sheets.

NOTE	Care must be taken with hidden power input pins because they can create unintentional connections. By nature, hidden pins are invisible and do not display their pin name. This makes it easy to accidentally connect two power pins to the same net. For this reason, **using invisible power pins in symbols is not recommended** outside of power port symbols, and is only supported for compatibility with legacy designs and symbols.

Power Ports

Power ports are symbols that are conventionally used to represent a connection to a power net, such as `VCC` or `GND`. In addition to being a visual indicator that the attached net is a power rail, power ports make global connections: two power ports with the same pin name connect to each other anywhere in the schematic, regardless of sheet.

In the figure below, power port symbols are used to connect the positive and negative terminals of the capacitors to the `VCC` and `GND` nets, respectively.



In the KiCad standard library, power ports are found in the `power` library, but power port symbols can be created in any library. To create a custom power port, make a new symbol with a power input pin that is set to be invisible. Name the pin according to the desired power net. In addition, set the "Define as power symbol" symbol property. As described in the hidden power pins section, invisible power input pins make global connections based on the hidden power pin's name. The process of creating a power port is described in more detail in the Symbol Editor section.

NOTE The connected net name is determined by the power port's **pin name**, not the name or value of the symbol. This means that power port net names can only be changed in the symbol editor, not in the schematic.

Net name assignment rules

Every net in the schematic is assigned a name, whether that name is specified by the user or automatically generated by KiCad.

When multiple labels are attached to the same net, the final net name is determined in the following order, from highest priority to lowest:

1. 全局标签
2. Power ports
3. Local labels
4. 分层标签
5. Hierarchical sheet pins

If there are multiple labels of one type attached to a net, the names are sorted alphabetically and the first is used.

If a net travels through multiple sheets in a hierarchy, and has no global label or power port, it will take its name from the highest level of the hierarchy where it has a hierarchical label or local label. As described above, local labels take priority over hierarchical labels.

If none of the labels above are attached to a net, the net's name is automatically generated based on the connected symbol pins.

PWR_FLAG

Two `PWR_FLAG` symbols are visible in the screenshot above. They indicate to ERC that the two power nets `VCC` and `GND` are actually connected to a power source, as there is no explicit power source such as a voltage regulator output attached to either net.

Without these two flags, the ERC tool would diagnose: *Error: Input Power pin not driven by any Output Power pins.*

The `PWR_FLAG` symbol is found in the `power` symbol library. The same effect can be achieved by connecting any power output pin to the net.

No-connection flag

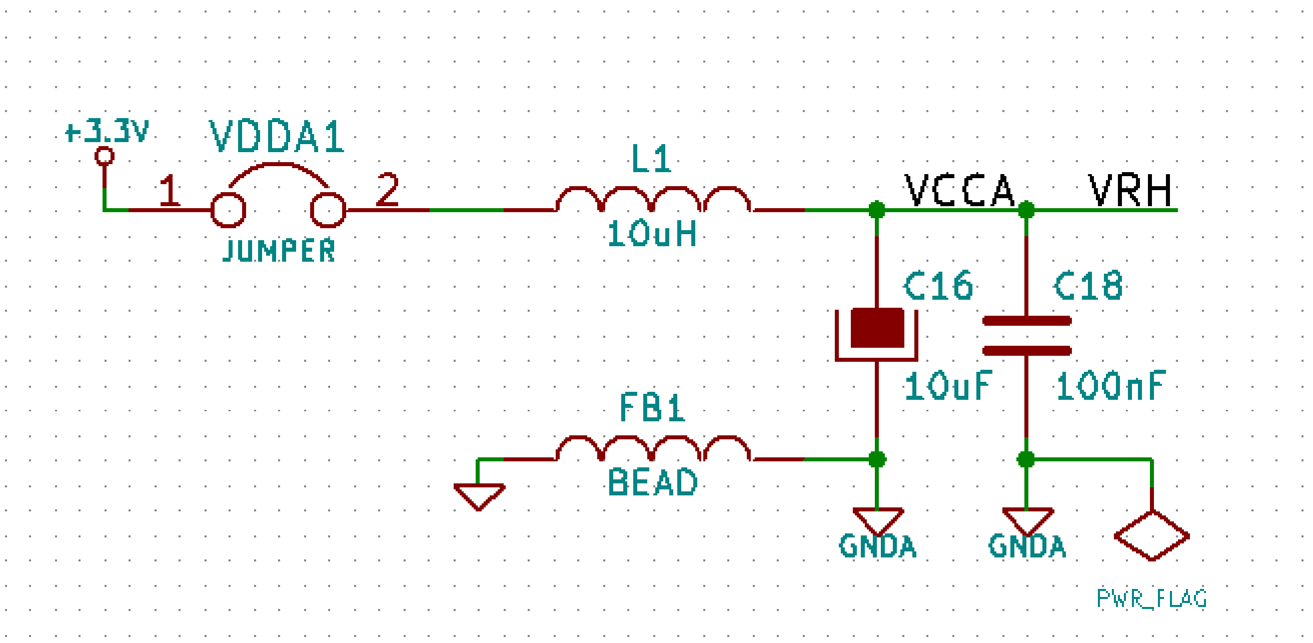
No-connection flags (✕) are used to indicate that a pin is intentionally unconnected. These flags do not have any effect on the schematic's connectivity, but they prevent "unconnected pin" ERC warnings for pins that are intentionally unconnected.

Graphical items

Text and graphic lines

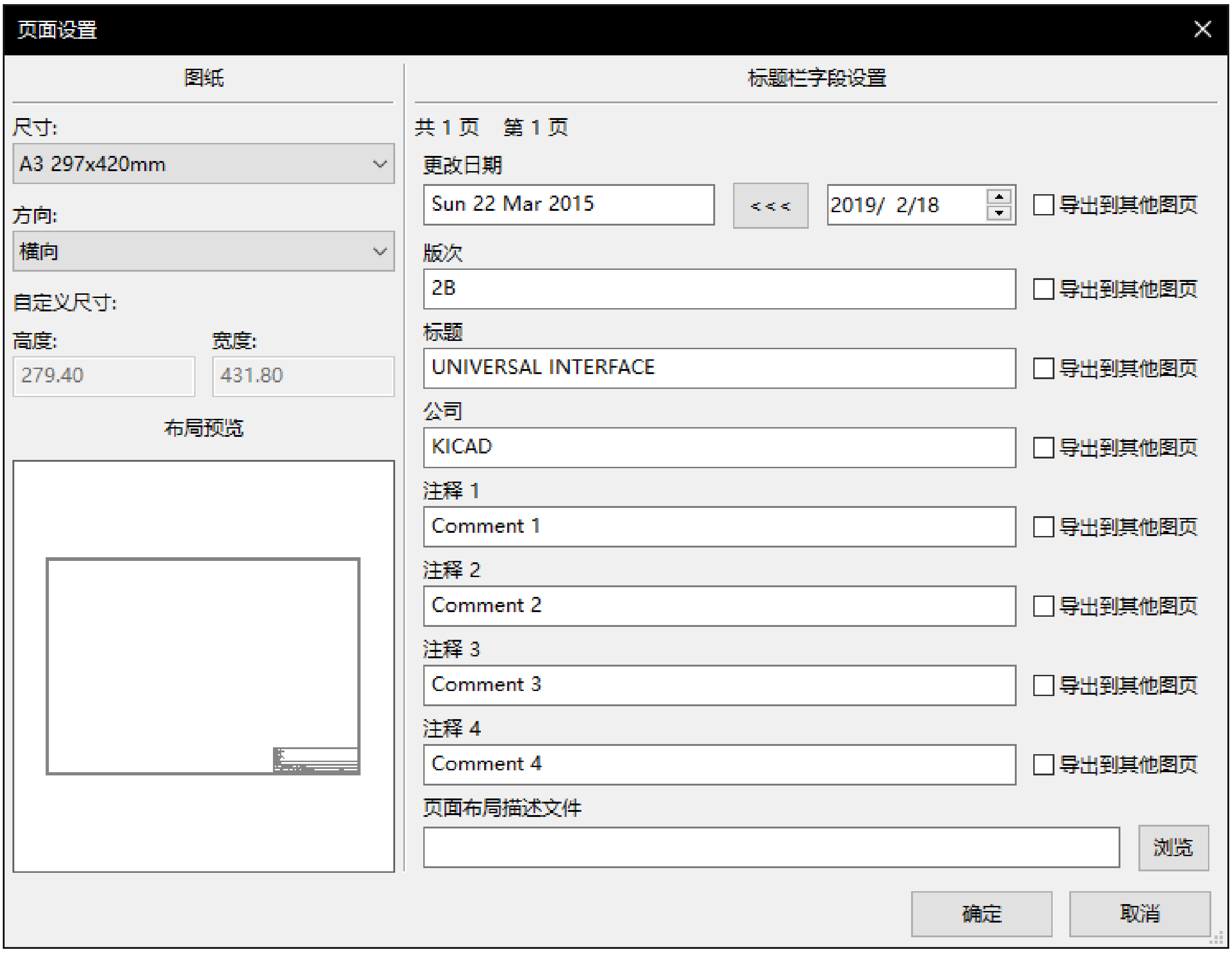
It can be useful to place annotations such as text fields and frames to aid in understanding the schematic. Text fields (T) and graphic lines ( ) are intended for this use, as opposed to labels and wires, which are connection elements.

The image below shows graphic lines and text in addition to wires, local labels, and hierarchical labels.



表格标题栏

The title block is edited with the Page Settings tool (🗋).



页面设置

图纸

尺寸:
A3 297x420mm

方向:
橫向

自定义尺寸:

高度: 279.40
宽度: 431.80

布局预览

标题栏字段设置

共 1 页　第 1 页

更改日期
Sun 22 Mar 2015　<<<　2019/ 2/18　☐ 导出到其他图页

版次
2B　☐ 导出到其他图页

标题
UNIVERSAL INTERFACE　☐ 导出到其他图页

公司
KICAD　☐ 导出到其他图页

注释 1
Comment 1　☐ 导出到其他图页

注释 2
Comment 2　☐ 导出到其他图页

注释 3
Comment 3　☐ 导出到其他图页

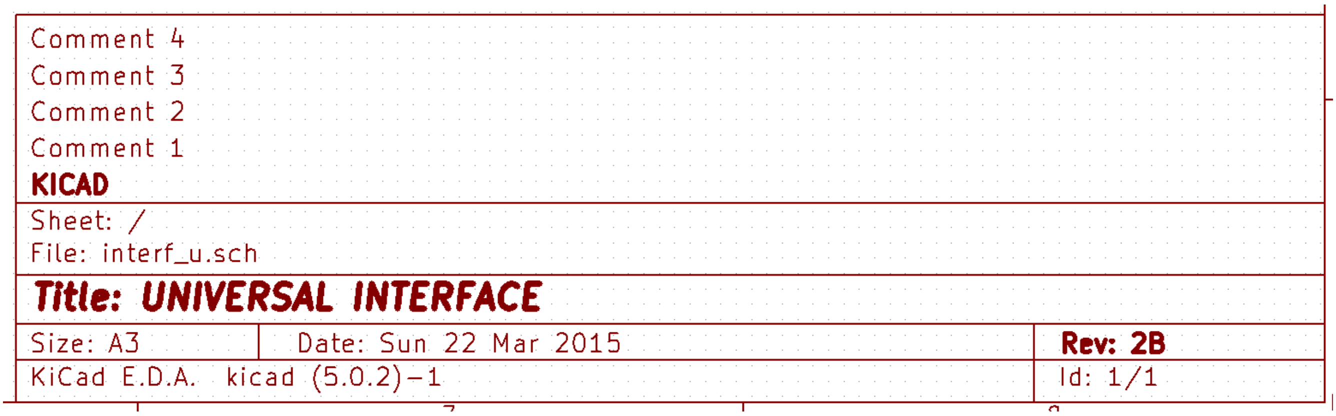
注释 4
Comment 4　☐ 导出到其他图页

页面布局描述文件

Each field in the title block can be edited, as well as the paper size and orientation. If the "Export to other sheets" option is checked for a field, that field will be updated in the title block of all sheets, rather than only the current sheet.

You can set the date to today’s or any other date by pressing the left arrow button by "Issue Date", but the date in the schematic will not be automatically updated.

A drawing sheet template file can also be selected.



Comment 4
Comment 3
Comment 2
Comment 1
KICAD
Sheet: /
File: interf_u.sch
Title: UNIVERSAL INTERFACE
Size: A3 Date: Sun 22 Mar 2015 Rev: 2B
KiCad E.D.A. kicad (5.0.2)-1 Id: 1/1

The sheet number (Sheet X/Y) is automatically updated, but sheet page numbers can also be manually set using **Edit** → **Edit Sheet Page Number…**.

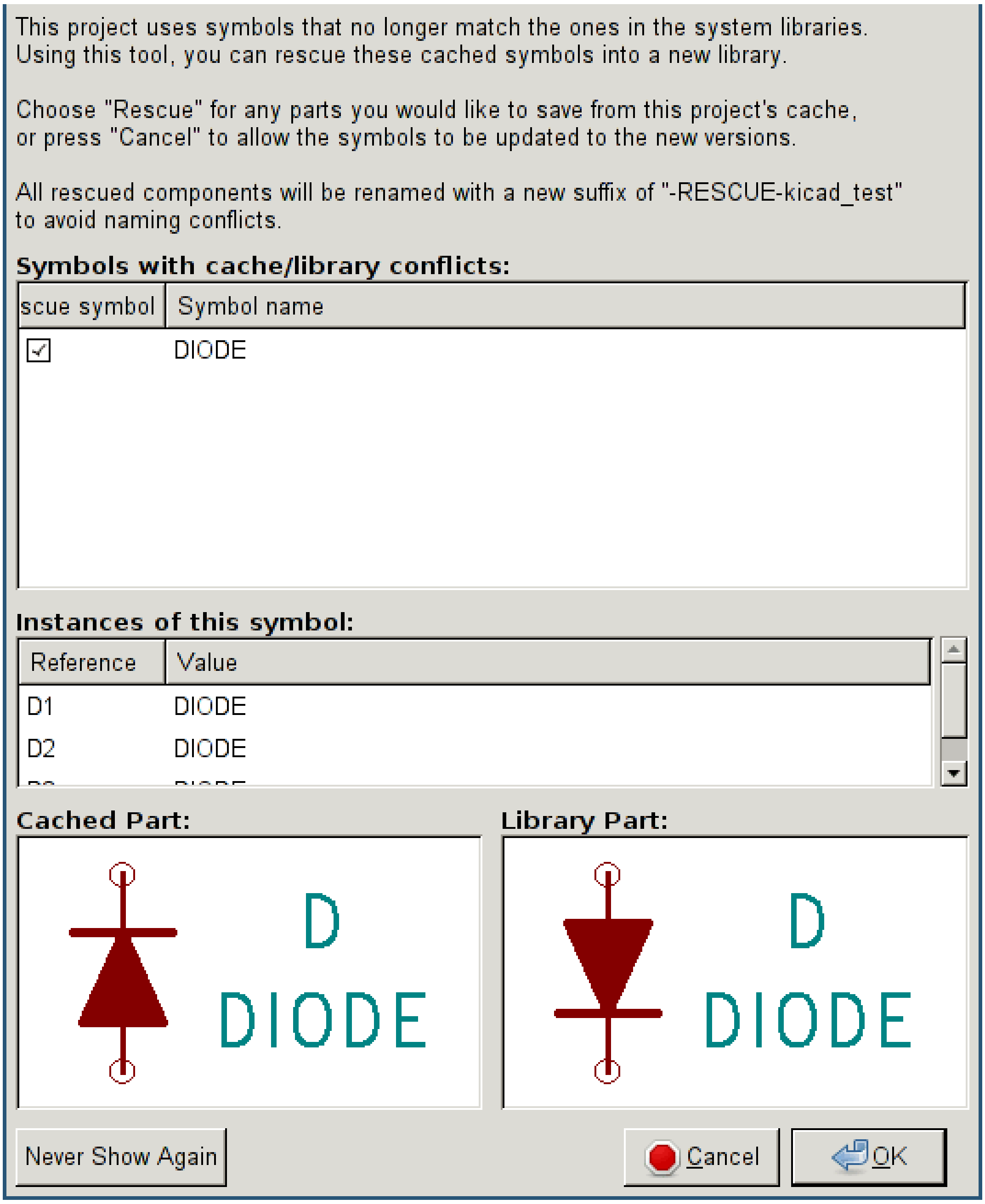
Schematic Setup

The Schematic Setup window is used to set schematic options that are specific to the currently active schematic. For example, the Schematic Setup window contains formatting options, electrical rule configuration, netclass setup, and schematic text variable setup.

抢救缓存的符号

By default, KiCad loads symbols from the project libraries according to the set paths and library order. This can cause a problem when loading a very old project: if the symbols in the library have changed or have been removed or the library no longer exists since they were used in the project, the ones in the project would be automatically replaced with the new versions. The new versions might not line up correctly or might be oriented differently leading to a broken schematic.

When a project is saved, a cache library with the contents of the current library symbols is saved along with the schematic. This allows the project to be distributed without the full libraries. If you load a project where symbols are present both in its cache and in the system libraries, KiCad will scan the libraries for conflicts. Any conflicts found will be listed in the following dialog:



This project uses symbols that no longer match the ones in the system libraries.
Using this tool, you can rescue these cached symbols into a new library.

Choose "Rescue" for any parts you would like to save from this project's cache,
or press "Cancel" to allow the symbols to be updated to the new versions.

All rescued components will be renamed with a new suffix of "-RESCUE-kicad_test"
to avoid naming conflicts.

Symbols with cache/library conflicts:

scue symbol	Symbol name
☑	DIODE

Instances of this symbol:

Reference	Value
D1	DIODE
D2	DIODE

You can see in this example that the project originally used a diode with the cathode facing up, but the library now contains one with the cathode facing down. This change would break the schematic! Pressing OK here will cause the symbol cache library to be saved into a special `rescue` library and all the symbols are renamed to avoid naming conflicts.

If you press Cancel, no rescues will be made, so KiCad will load all the new components by default. If you save the schematic at this point, your cache will be overwritten and the old symbols will not be recoverable. If you have saved the schematic, you can still go back and run the rescue function again by selecting "Rescue Cached Components" in the "Tools" menu to call up the rescue dialog again.

如果您不想看到此对话框，可以按 *从不再显示* 。 默认设置是不执行任何操作并允许加载新元件。 可以在 *库* 首选项中更改此选项。

分层原理图

简介

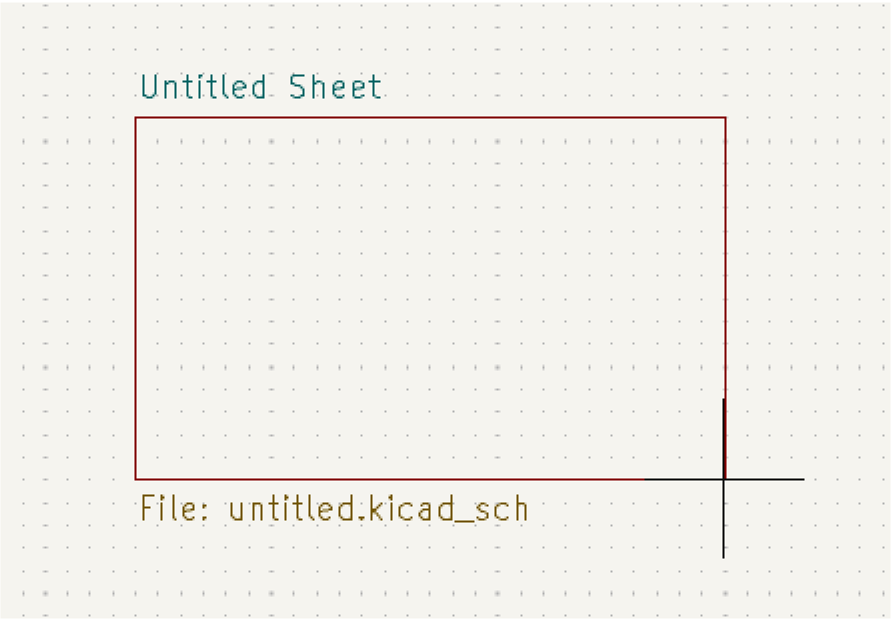
In KiCad, multi-sheet schematics are hierarchical: there is a single root sheet, and additional sheets are created as subsheets of either the root sheet or another subsheet. Sheets can be included in a hierarchy multiple times, if desired.

Carefully drawing a schematic as a hierarchical design improves schematic legibility and reduces repetitive drawing.

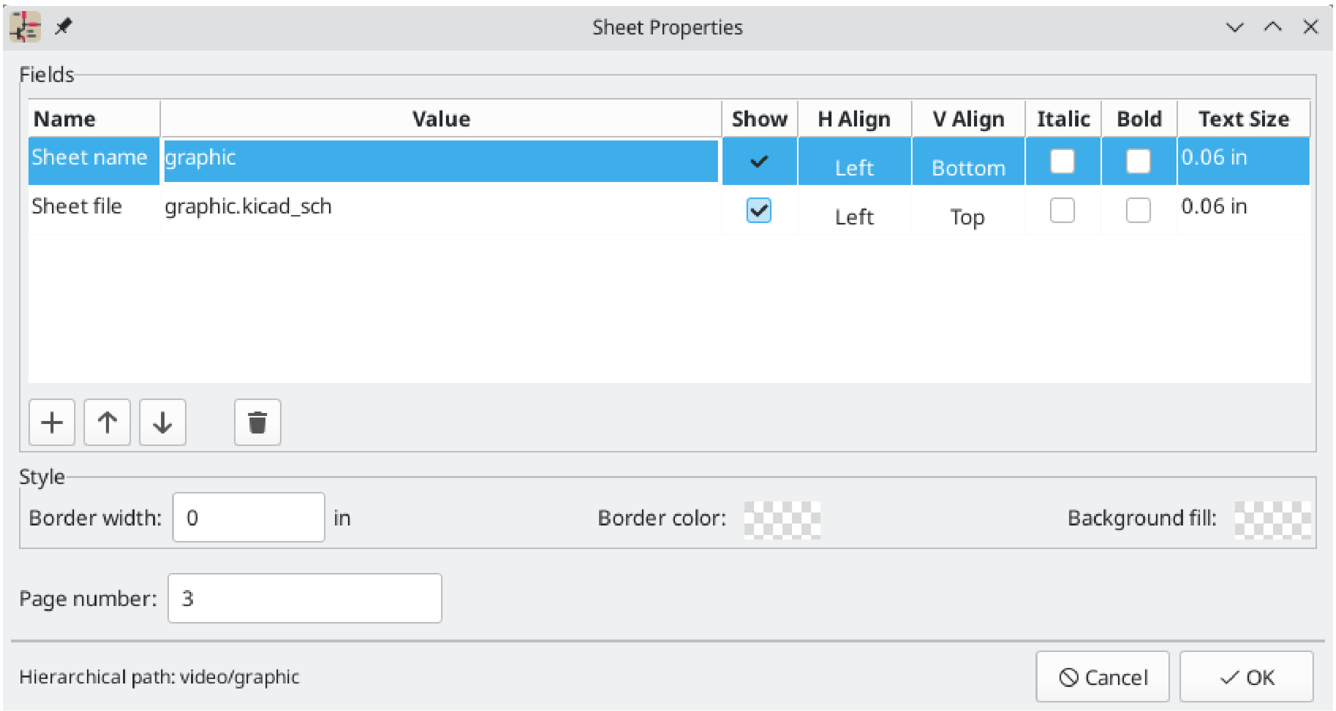
Creating a hierarchical schematic starts from the root sheet. The process is to create a subsheet, then draw the circuit in the subsheet and make the necessary electrical connections between sheets. Connections can be made between nets in a subsheet and nets in the parent sheet using hierarchical pins and labels, or between any two nets in the hierarchy using global labels.

Adding sheets to a design

You can add a subsheet to a design with the Add Hierarchical Sheet tool (S hotkey, or the button in the right toolbar). Launch the tool, then click twice in the canvas to draw the upper left and lower right corners of the subsheet symbol. Make the sheet outline large enough to fit the hierarchical pins you will add later.



The Sheet Properties dialog will appear and prompt you for a sheet name and filename.



Name	Value	Show	H Align	V Align	Italic	Bold	Text Size
Sheet name	graphic	✓	Left	Bottom	□	□	0.06 in
Sheet file	graphic.kicad_sch	☑	Left	Top	□	□	0.06 in

Border width: 0 in Border color: ▨ Background fill: ▨

Page number: 3

Hierarchical path: video/graphic

The **sheet name** must be unique, as it is used in the full net name for any nets in the subsheet. For example, a net with the local label `net1` in the sheet `sheet1` would have a full net name of `/sheet1/net1`. The sheet name is also used to refer to the sheet in various places in the GUI, including the title block and the hierarchy navigator.

The **sheet filename** specifies the file that the new sheet will be saved to or loaded from. The path to the sheet file can be relative or absolute. It is usually preferable to save subsheet files in the project directory and use a relative path so that the project is portable.

A single sheet file can be used more than once in a project by specifying the same filename for each repeated sheet; the circuit drawn in the sheet will be instantiated once per usage, and any edits in once instance will be reflected in the other instances.

NOTE	Sheet files can be shared between multiple projects to allow design reuse between projects. However, this is not recommended due to path portability concerns and the risk of unintentionally changing other projects while editing a shared sheet.

The sheet's **page number** is configurable here. The page number is displayed in the sheet title block and the hierarchy navigator, and sheets are sorted by page number in the hierarchy navigator and when printing or plotting.

Several graphical options are also available. **Border width** sets the width of the border around the sheet shape. **Border color** and **Background fill** set the color for the border and fill of the sheet shape, respectively. If no color is set, a checkerboard swatch is shown and the default values from the color theme are used.

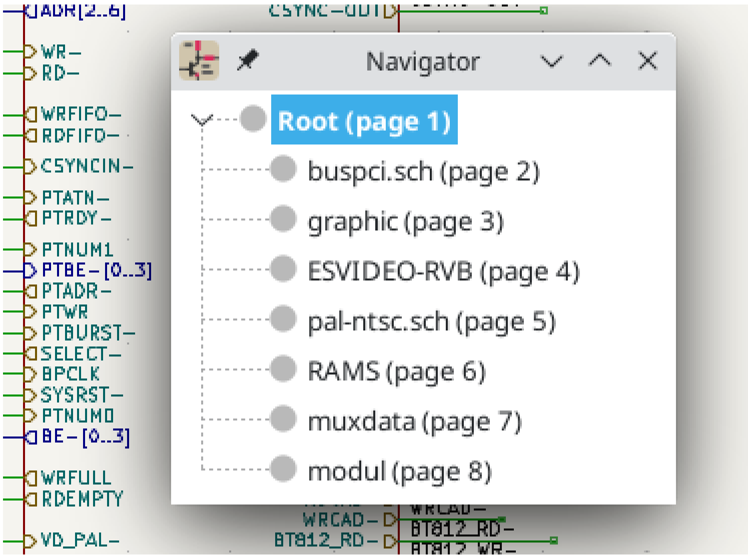
The Sheet Properties dialog can be accessed at any time by selecting a sheet symbol and using the E hotkey, or by right-clicking on a sheet symbol and selecting **Properties…**.

Navigating between sheets

You can enter a hierarchical sheet from the parent sheet by double-clicking the child sheet's shape, or right-clicking the child sheet and selecting **Enter Sheet**.

Return to the parent sheet by using the ↑ button in the top toolbar, or by right-clicking in an empty part of the schematic and clicking **Leave Sheet**.

Alternatively, you can jump to any sheet with the hierarchy navigator. To open the hierarchy navigator, click the button in the top toolbar. Each sheet in the design is displayed as an item in the tree. Clicking a sheet name opens that sheet in the editing canvas.



Navigator

- Root (page 1)
  - buspci.sch (page 2)
  - graphic (page 3)
  - ESVIDEO-RVB (page 4)
  - pal-ntsc.sch (page 5)
  - RAMS (page 6)
  - muxdata (page 7)
  - modul (page 8)

By default, the hierarchy navigator closes after a new sheet is opened. It can be configured to always remain open by selecting the **Keep hierarchy navigator open** option in the Editing Options section of the Schematic Editor preferences.

Electrical connections between sheets

Label overview

Electrical connections between sheets are made with labels. There are several kinds of labels in KiCad, each with a different connection scope.

- **Local labels** only make connections within a sheet. Therefore local labels cannot be used to connect between sheets. Local labels are added with the A button.
- **Global labels** make connections anywhere in a schematic, regardless of sheet. Global labels are added with the A button.
- **Hierarchical labels** connect to **hierarchical sheet pins** accessible in the parent sheet. Hierarchical designs rely on hierarchical labels and pins to make connections between parent sheets and child sheets; you can think of hierarchical pins as defining the interface for a sheet. Hierarchical labels are added with the A button.

NOTE	Labels that have the same name will connect, regardless of the label type, if they are in the same sheet.

NOTE

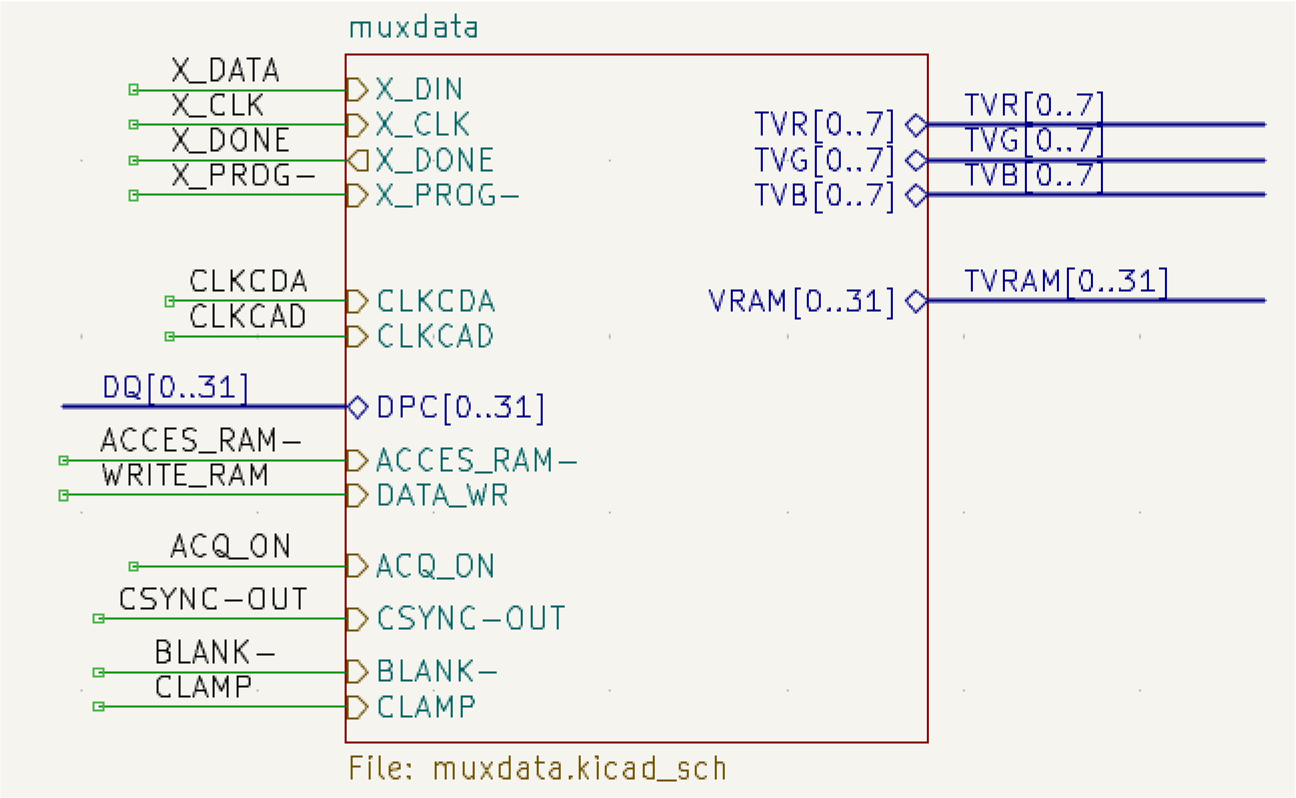
Hidden power pins can also be considered global labels, because they connect anywhere in the schematic hierarchy.

Hierarchical sheet pins

After placing hierarchical labels within the subsheet, matching **hierarchical pins** can be added to the subsheet symbol in the parent sheet. You can then make connections to the hierarchical pins with wires, labels, and buses. Hierarchical pins in a subsheet symbol are connected to the matching hierarchical labels in the subsheet itself.

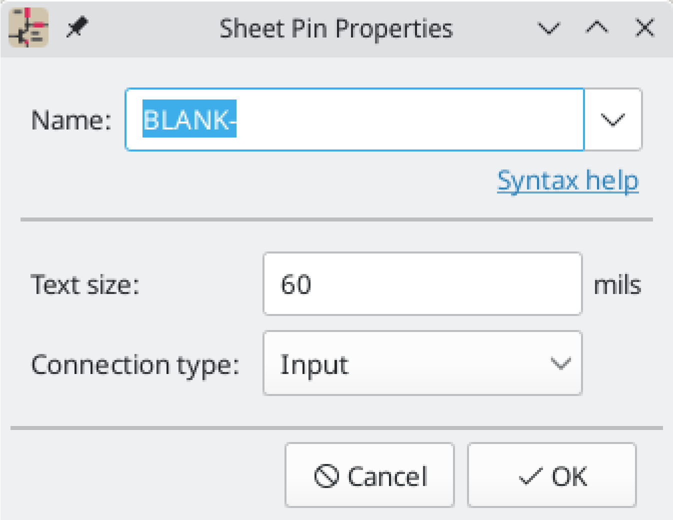
NOTE

Hierarchical labels must be defined in the subsheet before the corresponding hierarchical sheet pin can be imported in the sheet symbol.



For every hierarchical label in the subsheet, import the corresponding hierarchical pin into the sheet symbol by clicking the button in the right toolbar, then clicking on the sheet symbol. A sheet pin for the first unmatched hierarchical label will be attached to the cursor, where it can be placed anywhere along the border of the sheet symbol. Clicking again with the tool will continue to import additional sheet pins until there are no more hierarchical pins to import from the subsheet. Sheet pins can also be imported by selecting **Import Sheet Pin** in a sheet symbol's right-click context menu.

You can edit the properties of a sheet pin in the Sheet Pin Properties dialog. Open this dialog by double-clicking a sheet pin, selecting a sheet pin and using the E hotkey, or right-clicking a sheet pin and selecting **Properties...**.


Name: BLANK-

Text size: 60 mils

Connection type: Input

The sheet pin's **name** can be edited in the textbox or by selecting from the dropdown list of hierarchical labels in the subsheet. A sheet pin's name has to match the corresponding hierarchical label in the subsheet, so if a pin name is changed the label must change as well.

The **connection type** changes the shape of the sheet pin, and has no electrical effect. It can be set to Input, Output, Bidirectional, Tri-state, or Passive. The pin's **text size** can also be changed.

Hierarchical design examples

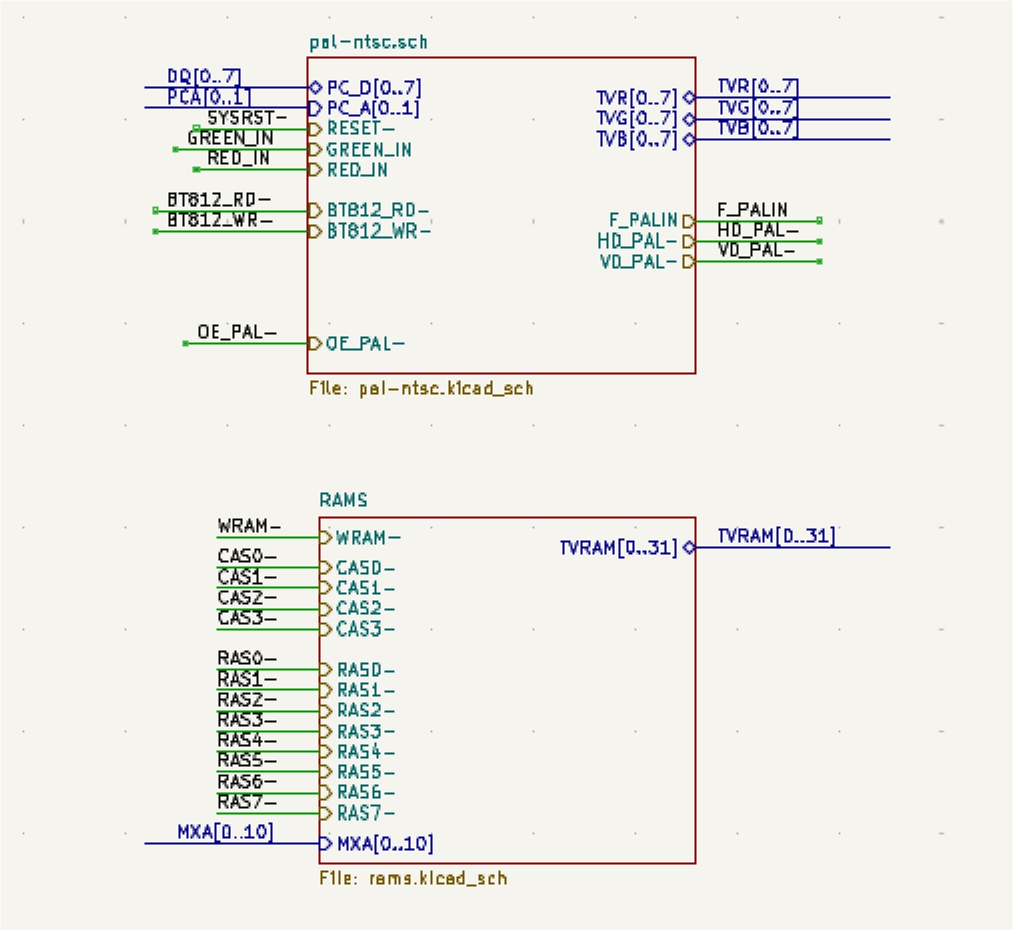
Hierarchical designs can be put into one of several categories:

- **Simple:** each sheet is used only once.
- **Complex:** some sheets are instantiated multiple times.
- **Flat:** a sub-case of a **simple** hierarchy, without connections between subsheets and their parent. Flat hierarchies can be used to represent a non-hierarchical design.

Each hierarchy model can be useful; the most appropriate one depends on the design.

Simple hierarchy

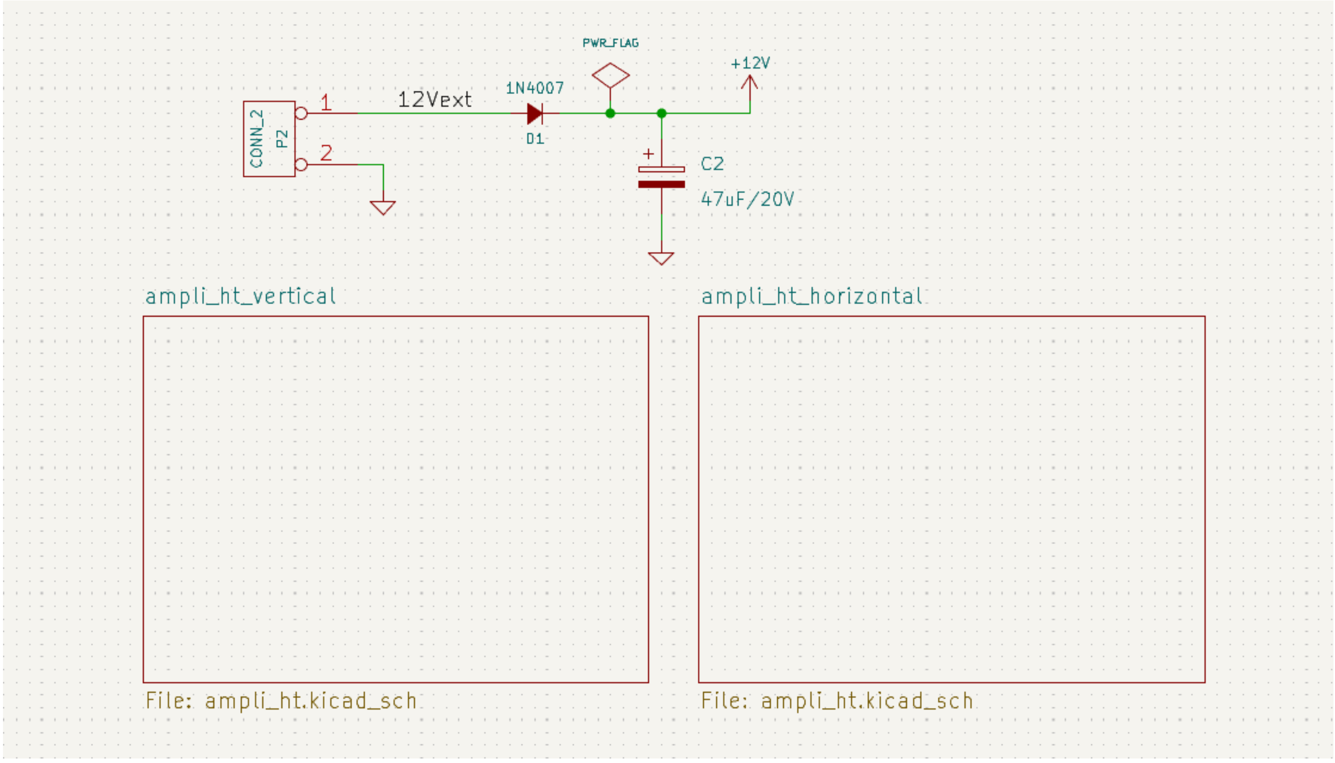
An example of a simple hierarchy is the `video` demo project included with KiCad. The root sheet contains seven unique subsheets, each with hierarchical labels and sheet pins linking the sheets to each other in the root sheet. Two of the subsheet symbols are shown below.



复杂层次结构

The `complex_hierarchy` demo project is an example of a complex hierarchy. The root sheet contains two subsheet symbols, which both refer to the same sheet file (`ampli_ht.kicad_sch`). This allows the design to include two copies of the same amplifier circuit. Although the two sheet symbols refer to the same filename, the sheet names are unique (`ampli_ht_vertical` and `ampli_ht_horizontal`). Inside each subsheet the circuits are identical except for the reference designators, which as always are unique.

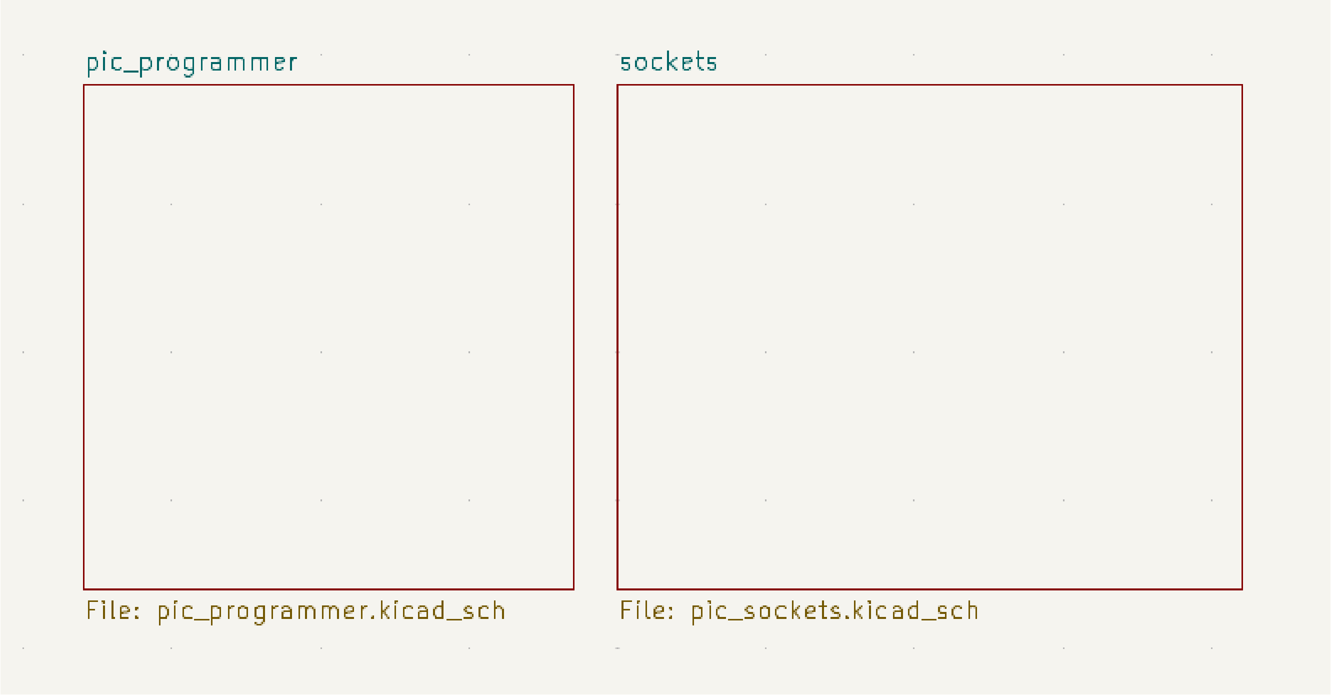
This project contains no sheet pin connections. The only connections between the root sheet and the subsheets are global power connections made with power port symbols. However, sheets in a complex hierarchy could include sheet pin connections if appropriate for the design.



平面层次结构

The `flat_hierarchy` demo project is an example of a flat hierarchy. The root sheet contains two unique subsheet symbols with no hierarchical sheet pins. The root sheet in this project does nothing except hold the subsheets, and the subsheets are used only as additional pages in the schematic.

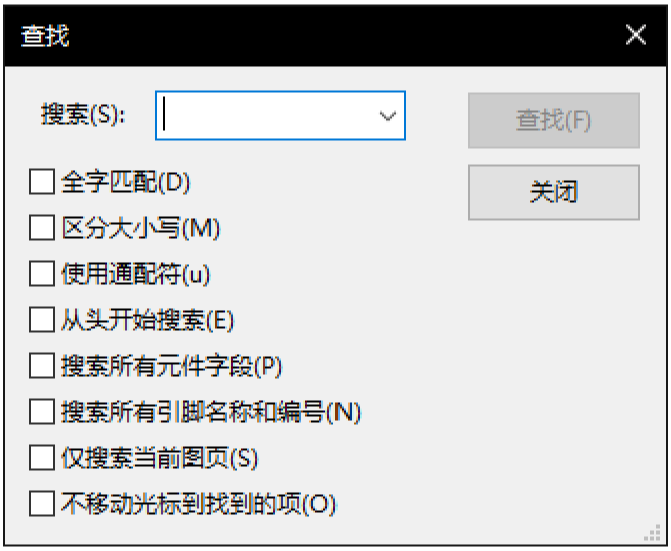
NOTE This is the simplest way to create multi-page schematics in KiCad.



Inspecting a schematic

Find tool

The Find tool searches for text in the schematic, including reference designators, pin names, symbol fields, and graphic text. When the tool finds a match, the canvas is zoomed and centered on the match and the text is highlighted. Launch the tool using the button in the top toolbar.



搜索(S):

□ 全字匹配(D)
□ 区分大小写(M)
□ 使用通配符(u)
□ 从头开始搜索(E)
□ 搜索所有元件字段(P)
□ 搜索所有引脚名称和编号(N)
□ 仅搜索当前图页(S)
□ 不移动光标到找到的项(O)

The Find tool has several options:

Match case: Selects whether the search is case-sensitive.

Words: When selected, the search will only match the search term with complete words in the schematic. When unselected, the search will match if the search term is part of a larger word in the schematic.

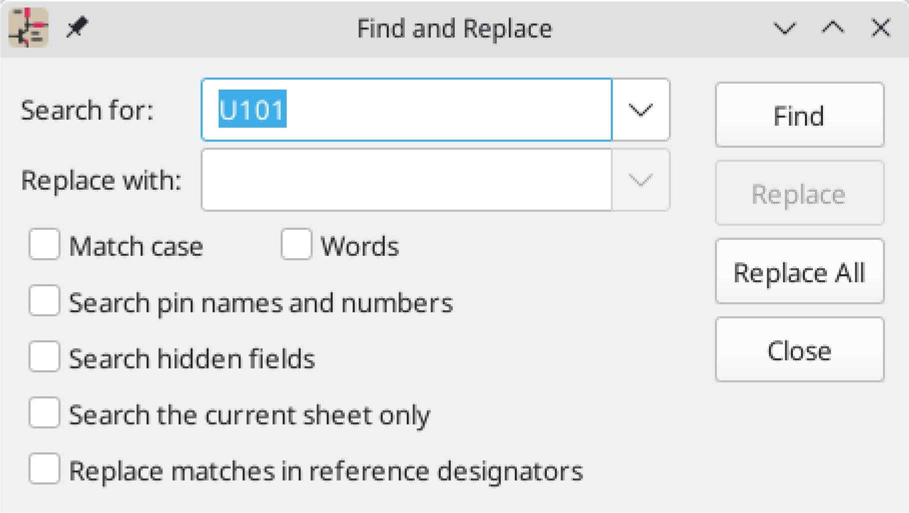
Wildcards: When selected, wildcards can be used in the search terms. `?` matches any single character, and `*` matches any number of characters. Note that when this option is selected, partial matches are not returned: searching for `abc*` will match the string `abcd`, but searching for `abc` will not.

Search pin names and numbers: Selects whether the search should apply to pin names and numbers.

Search hidden fields: Selects whether the search should apply only to visible fields or if it should include hidden symbol fields.

Search the current sheet only: Selects whether the search should be limited to the current schematic sheet or to the entire schematic.

There is also a Find and Replace tool which is activated with the button in the top toolbar. This tool behaves the same as the Find tool, but additionally can replace some or all matches with different text.



Search for: U101

Replace with:

☐ Match case ☐ Words

☐ Search pin names and numbers

☐ Search hidden fields

☐ Search the current sheet only

☐ Replace matches in reference designators

If the **Replace matches in reference designators** option is checked, reference designators will be modified if they contain matching text. Otherwise reference designators will not be affected.

Net highlighting

An electrical net can be highlighted in the schematic editor to visualize all of the places it appears in the schematic. Net highlighting can be activated in the Schematic Editor or by highlighting the corresponding net in the PCB editor when cross-probe highlighting is enabled (see below). When net highlighting is active, the highlighted net will be shown in a different color. By default this color is pink, but it is configurable in the Color section of the Preferences dialog.

Nets can be highlighted by clicking on a wire or pin using the Highlight Net tool in the right toolbar (). Alternatively, the Highlight Net hotkey (`) highlights the net under the cursor. If there are no nets or pins under the cursor, any existing highlighting will be cleared. The highlighting can also be cleared by using the Clear Net Highlight action (hotkey ~).

Cross-probing from the PCB

KiCad allows bi-directional cross-probing between the schematic and the PCB. There are several different types of cross-probing.

Selection cross-probing allows you to select a symbol or pin in the schematic to select the corresponding footprint or pad in the PCB (if one exists) and vice-versa. By default, cross-probing will result in the display centering on the cross-probed item and zooming to fit. This behavior can be disabled in the Display Options section of the Preferences dialog.

Highlight cross-probing allows you to highlight a net in the schematic and PCB at the same time. If the option "Highlight cross-probed nets" is enabled in the Display Options section of the Preferences dialog, highlighting a net or bus in the schematic editor will cause the corresponding net or nets to be highlighted in the PCB editor, and vice versa.

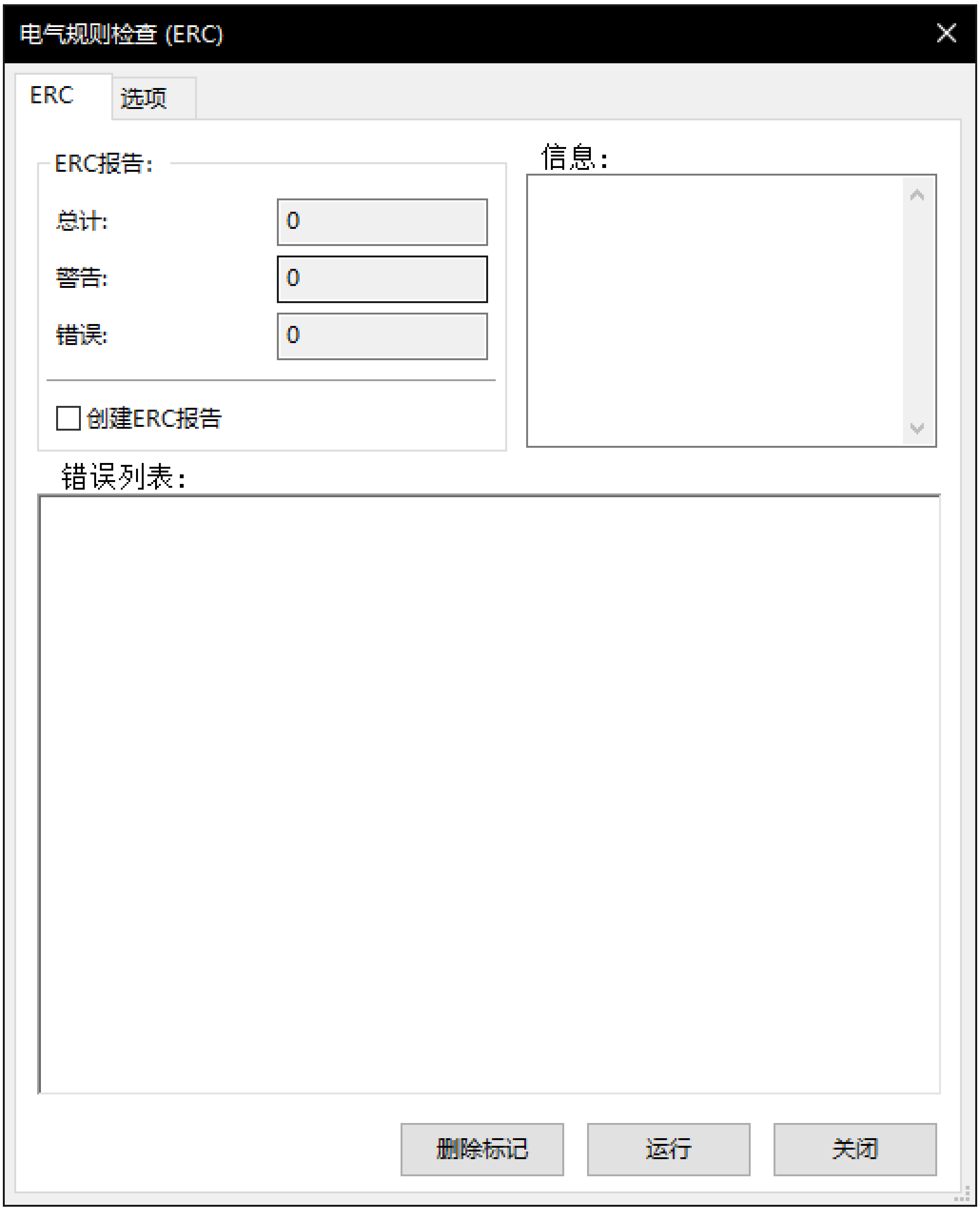
使用电气规则检查进行设计验证

The Electrical Rules Check (ERC) tool performs an automatic check of your schematic. The ERC checks for any errors in your sheet, such as unconnected pins, unconnected hierarchical symbols, shorted outputs, etc. ERC output is reported as errors or warnings depending on the severity of the issue detected.

Naturally, an automatic check is not infallible, and it is not possible to detect all design errors. Such a check is still very useful, because it allows you to detect many oversights and small errors. All detected issues should be checked and addressed before proceeding.

The quality of the ERC is directly related to the care taken in declaring electrical pin properties during symbol library creation.

ERC can be started by clicking on the ☑ button in the top toolbar and clicking the **Run ERC** button.



ERC报告：

总计: 0

警告: 0

错误: 0

□ 创建ERC报告

信息：

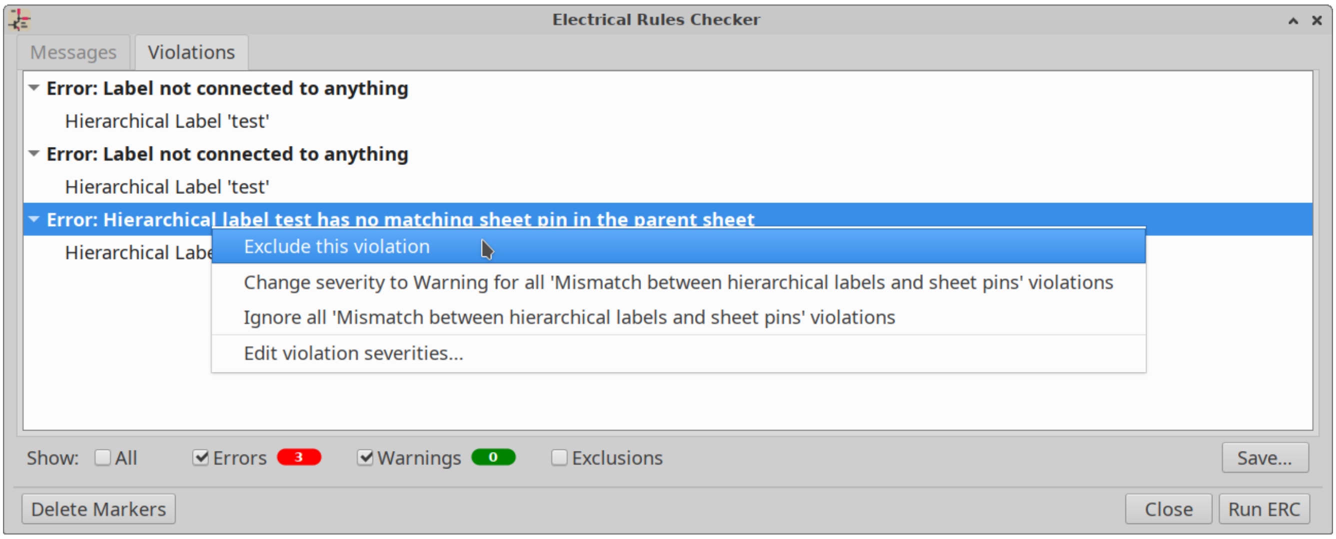
错误列表：

Any warnings or errors are reported in the **Violations** tab, and markers for each violation are placed in the schematic so that they point to the relevant part of the schematic. Warnings are indicated by yellow arrows, and errors have red arrows. Excluded violations are shown as green arrows.

NOTE	Selecting a violation in the ERC window jumps to the selected violation marker in the schematic.

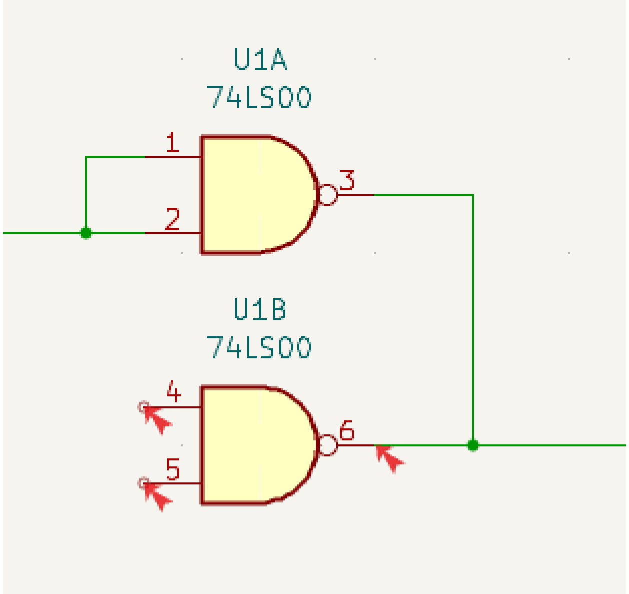
The numbers at the bottom of the window show the number of errors, warnings, and exclusions. Each type of violation can be filtered from the list using the respective checkboxes. Clicking **Delete Markers** will clear all violations until ERC is run again.

Violations can be right-clicked in the dialog to ignore them or change their severity:


Electrical Rules Checker
Messages
Violations
Error: Label not connected to anything
Hierarchical Label 'test'
Error: Label not connected to anything
Hierarchical Label 'test'
Error: Hierarchical label test has no matching sheet pin in the parent sheet
Hierarchical Labe
Exclude this violation
Change severity to Warning for all 'Mismatch between hierarchical labels and sheet pins' violations
Ignore all 'Mismatch between hierarchical labels and sheet pins' violations
Edit violation severities...
Show: □ All
☑ Errors 3
☑ Warnings 0
□ Exclusions
Save...
Delete Markers
Close
Run ERC

- **Exclude this violation:** ignores this particular violation, but does not affect any other violations.
- **Change severity:** changes a type of violation from warning to error, or error to warning. This affects all violations of a given type.
- **Ignore all:** ignores all violations of a given type.

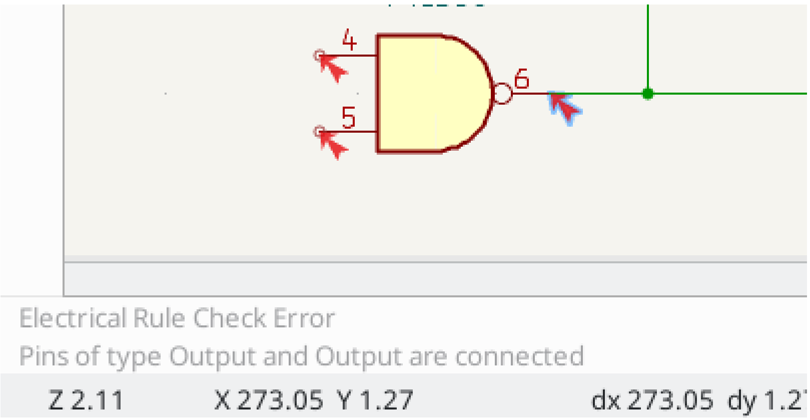
ERC example



Here you can see three errors:

- Two outputs have been connected together (red arrow at right).
- Two inputs have been left unconnected (red arrows at left).

Selecting an ERC marker displays a description of the violation in the message pane at the bottom of the window.

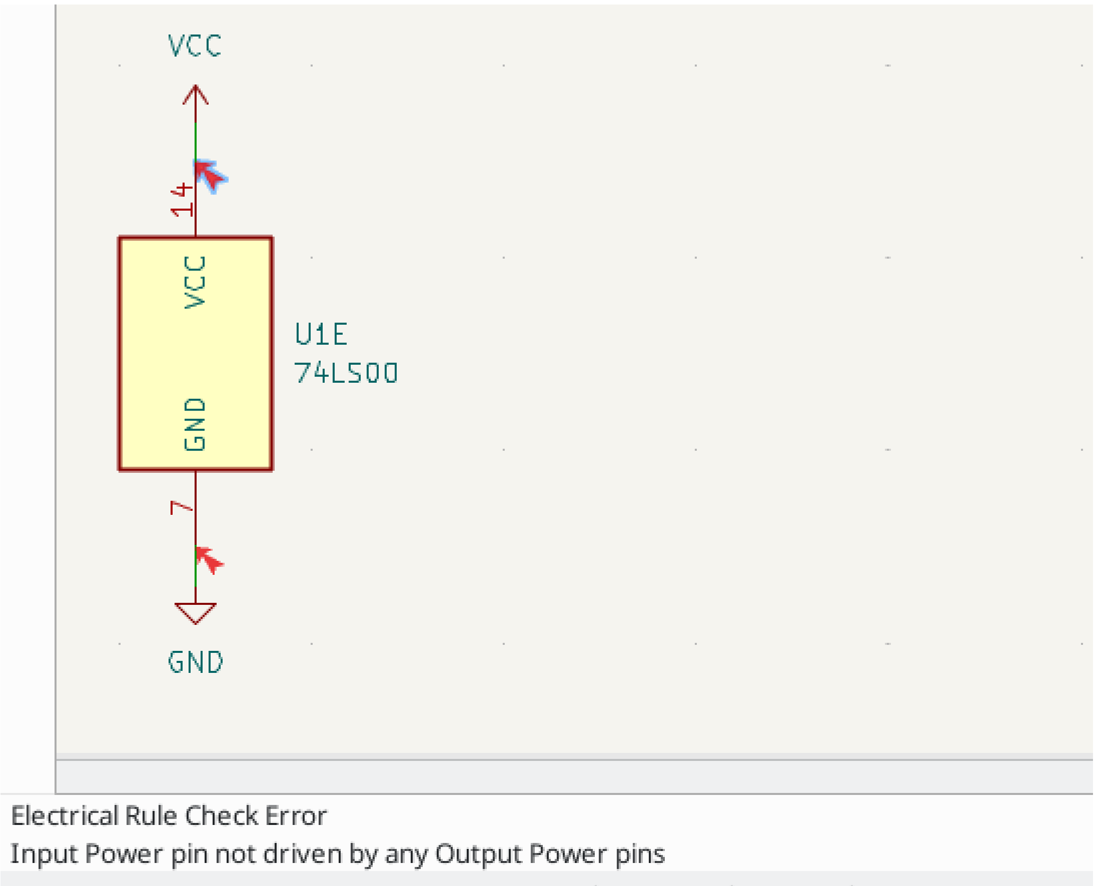


Electrical Rule Check Error
Pins of type Output and Output are connected

Z 2.11 X 273.05 Y 1.27 dx 273.05 dy 1.2

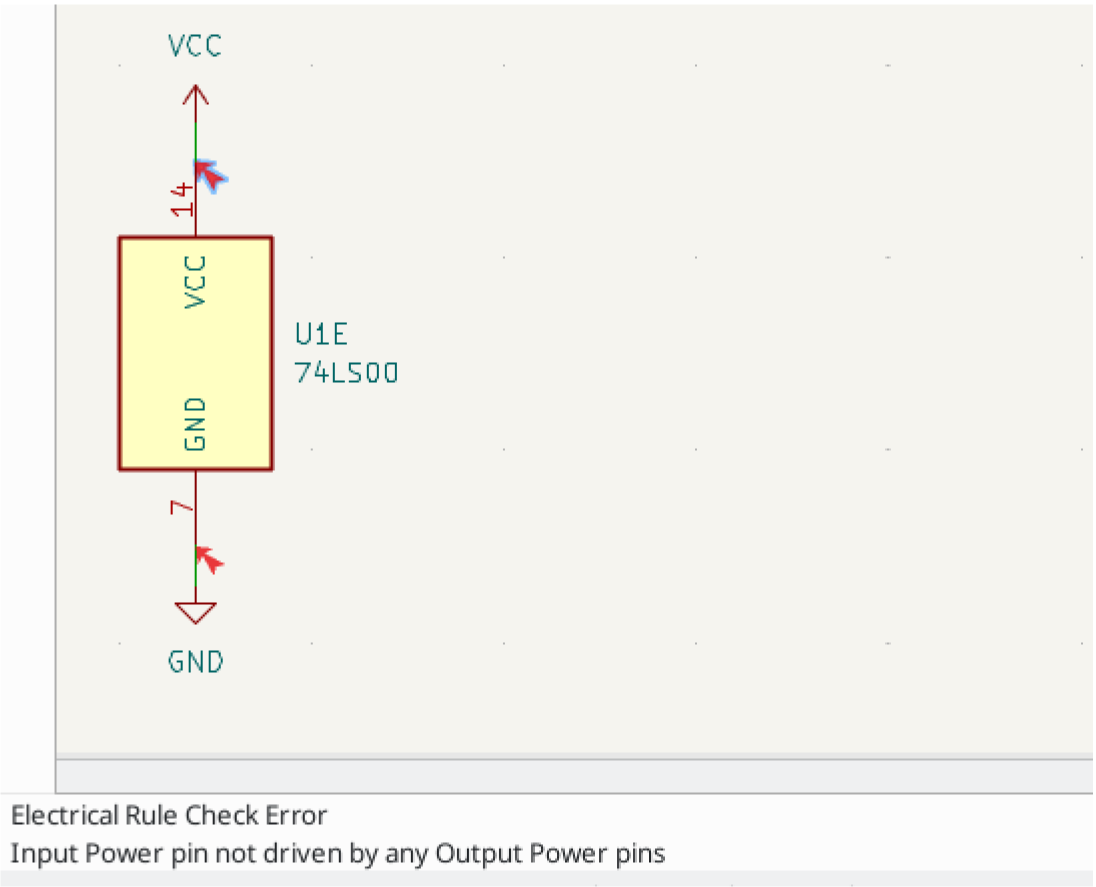
Power pins and power flags

It is common to have an "Input Power pin not driven by any Output Power pins" error on power pins, as shown in the example below, even though the power pins seem to be properly connected to a power rail. This happens in designs where the power is provided through connectors or other components that are not marked as power outputs. In these cases ERC won't detect any Output Power pins connected to the net and will determine the Input Power pin is not driven by a power source.



Electrical Rule Check Error
Input Power pin not driven by any Output Power pins

To avoid this warning, connect the net to `PWR_FLAG` symbol on such a power net as shown in the following example. The `PWR_FLAG` symbol is found in the `power` symbol library. Alternatively, connect any power output pin to the net; `PWR_FLAG` is simply a symbol with a single power output pin.



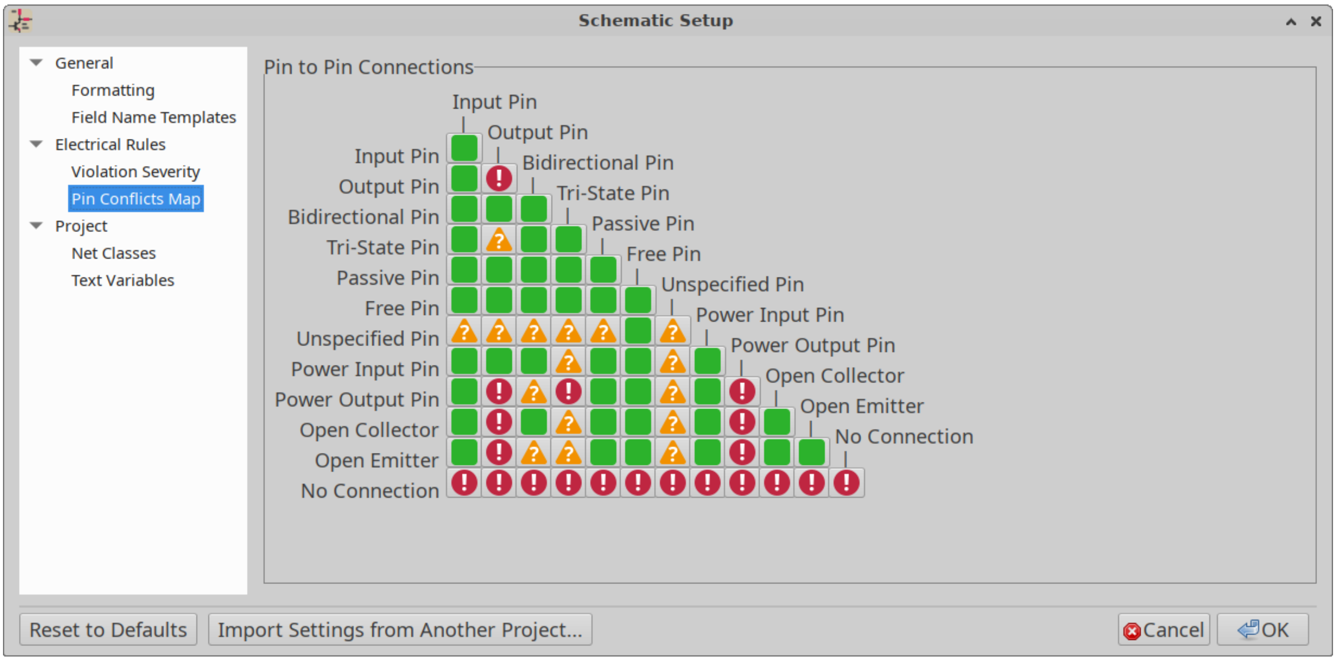
Electrical Rule Check Error
Input Power pin not driven by any Output Power pins

Ground nets often need a `PWR_FLAG` as well, because voltage regulators have outputs declared as power outputs, but their ground pins are typically marked as power inputs. Therefore grounds can appear unconnected to a source unless a `PWR_FLAG` symbol is used.

For more information about power pins and power flags, see the `PWR_FLAG` documentation.

ERC Configuration

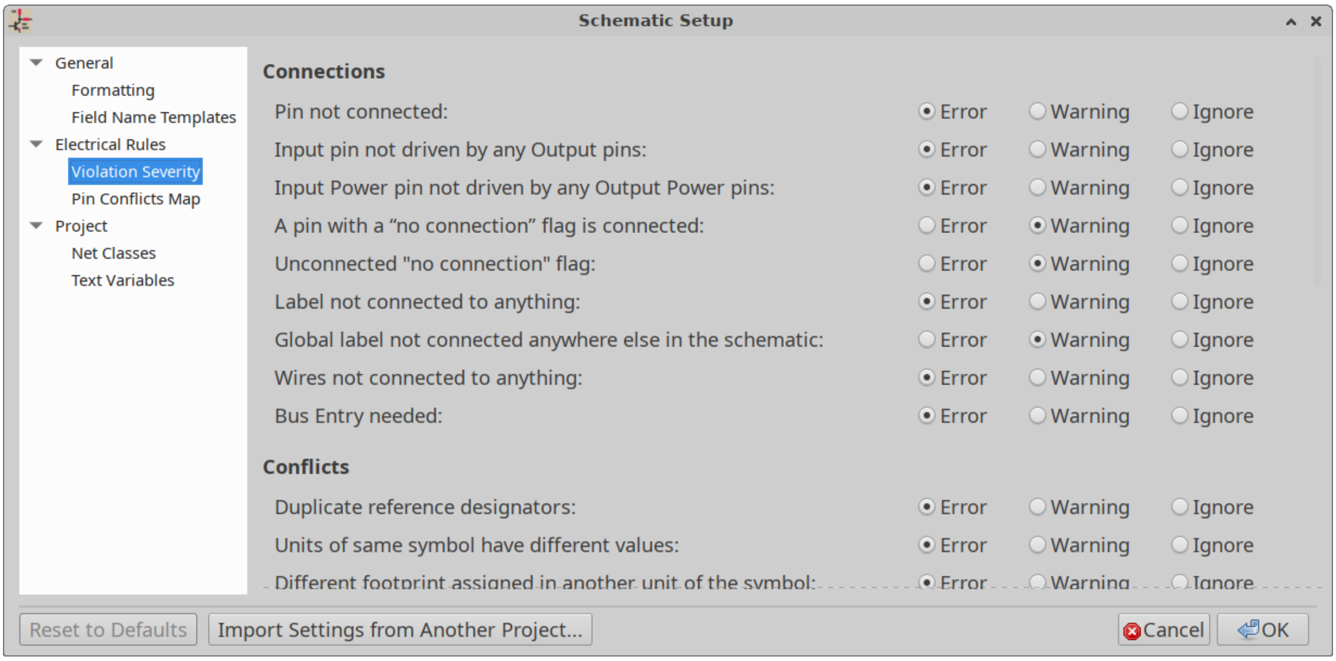
The *Pin Conflicts Map* panel in Schematic Setup allows you to configure connectivity rules to define electrical conditions for errors and warnings based on what types of pins are connected to each other.



Pin to Pin Connections

	Input Pin	Output Pin	Bidirectional Pin	Tri-State Pin	Passive Pin	Free Pin	Unspecified Pin	Power Input Pin	Power Output Pin	Open Collector	Open Emitter	No Connection
Input Pin	🟩											
Output Pin	🟩	❗										
Bidirectional Pin	🟩	🟩	🟩									
Tri-State Pin	🟩	⚠	🟩	🟩								
Passive Pin	🟩	🟩	🟩	🟩	🟩							
Free Pin	🟩	🟩	🟩	🟩	🟩	🟩						
Unspecified Pin	⚠	⚠	⚠	⚠	⚠	🟩	⚠					
Power Input Pin	🟩	🟩	🟩	⚠	🟩	🟩	⚠	🟩				
Power Output Pin	🟩	❗	⚠	❗	🟩	🟩	⚠	🟩	❗			
Open Collector	🟩	❗	🟩	⚠	🟩	🟩	⚠	🟩	❗	🟩		
Open Emitter	🟩	❗	⚠	⚠	🟩	🟩	⚠	🟩	❗	🟩	🟩	
No Connection	❗	❗	❗	❗	❗	❗	❗	❗	❗	❗	❗	❗

单击所需的单元格方块可以更改规则，使其循环选择：正常，警告，错误。



Connections

Pin not connected:	● Error	○ Warning	○ Ignore
Input pin not driven by any Output pins:	● Error	○ Warning	○ Ignore
Input Power pin not driven by any Output Power pins:	● Error	○ Warning	○ Ignore
A pin with a “no connection” flag is connected:	○ Error	● Warning	○ Ignore
Unconnected "no connection" flag:	○ Error	● Warning	○ Ignore
Label not connected to anything:	● Error	○ Warning	○ Ignore
Global label not connected anywhere else in the schematic:	○ Error	● Warning	○ Ignore
Wires not connected to anything:	● Error	○ Warning	○ Ignore
Bus Entry needed:	● Error	○ Warning	○ Ignore

Conflicts

Duplicate reference designators:	● Error	○ Warning	○ Ignore
Units of same symbol have different values:	● Error	○ Warning	○ Ignore
Different footprint assigned in another unit of the symbol:	● Error	○ Warning	○ Ignore

The *Violation Severity* panel in Schematic Setup lets you configure what types of ERC messages should be reported as Errors, Warnings or ignored.

ERC 报告文件

An ERC report file can be generated and saved by clicking the **Save...** button in the ERC dialog. The file extension for ERC report files is `.rpt`. Here is an example ERC report file.

```
ERC report (Fri 21 Oct 2022 02:07:05 PM EDT, Encoding UTF8)

***** Sheet /
[pin_not_driven]: Input pin not driven by any Output pins
    ; Severity: error
    @(149.86 mm, 60.96 mm): Symbol U1B [74LS00] Pin 4 [, Input, Line]
[pin_not_connected]: Pin not connected
    ; Severity: error
    @(149.86 mm, 60.96 mm): Symbol U1B [74LS00] Pin 4 [, Input, Line]
[pin_not_connected]: Pin not connected
    ; Severity: error
    @(149.86 mm, 66.04 mm): Symbol U1B [74LS00] Pin 5 [, Input, Line]
[pin_to_pin]: Pins of type Output and Output are connected
    ; Severity: error
    @(165.10 mm, 63.50 mm): Symbol U1B [74LS00] Pin 6 [, Output, Inverted]
    @(165.10 mm, 46.99 mm): Symbol U1A [74LS00] Pin 3 [, Output, Inverted]
[pin_not_driven]: Input pin not driven by any Output pins
    ; Severity: error
    @(149.86 mm, 66.04 mm): Symbol U1B [74LS00] Pin 5 [, Input, Line]

 ** ERC messages: 5  Errors 5  Warnings 0
```

Assigning Footprints

Before routing a PCB, footprints need to be selected for every component that will be assembled on the board. Footprints define the copper connections between physical components and the routed traces on a circuit board.

Some symbols come with footprints pre-assigned, but for many symbols there are multiple possible footprints, so the user needs to select the appropriate one.

KiCad offers several ways to assign footprints:

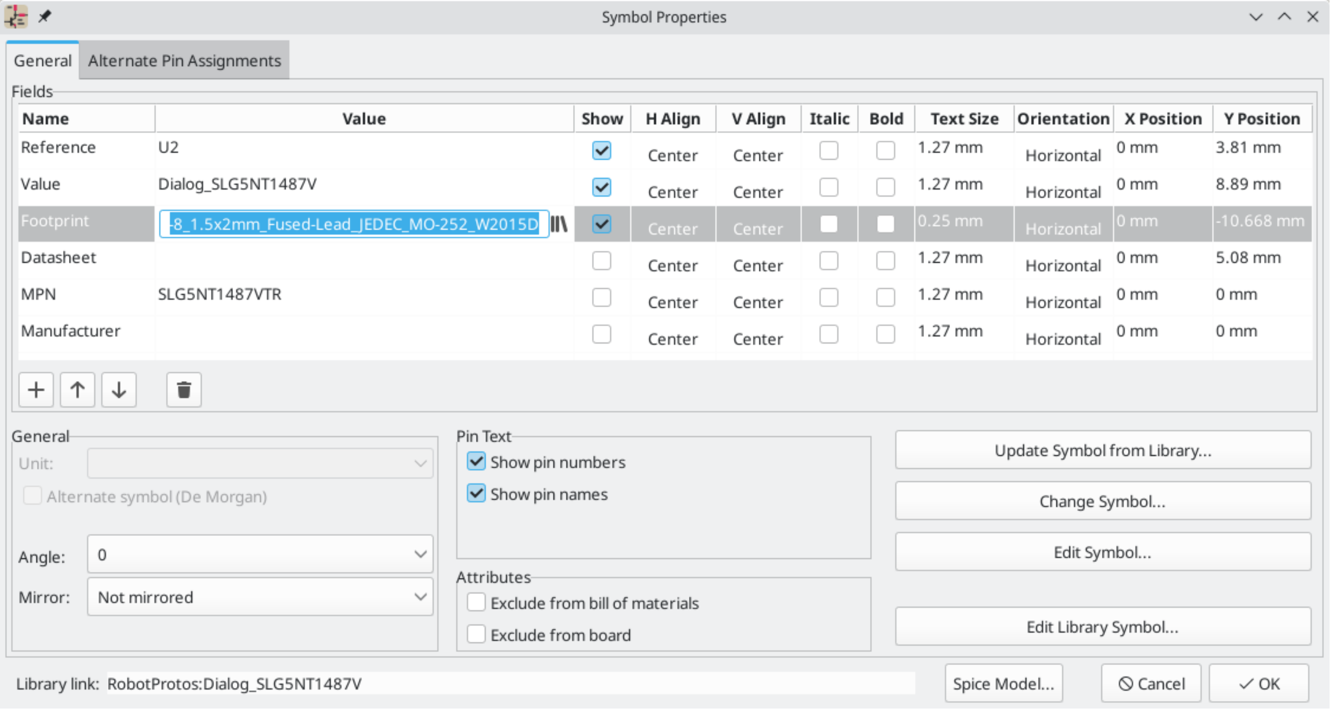
- 符号属性
  - Symbol Properties Dialog
  - Symbol Fields Table
- While placing symbols
- Footprint Assignment Tool

Each method will be explained below. Which to use is a matter of preference; one method may be more convenient depending on the situation. All of these methods are equivalent in that they store the name of the selected footprint in the symbol's `Footprint` field.

NOTE The Footprint Library Table needs to be configured before footprints can be assigned. For information on configuring the Footprint Library Table, please see the PCB Editor manual.

Assigning Footprints in Symbol Properties

A symbol's `Footprint` field can be edited directly in the symbol's Properties window.



Name	Value	Show	H Align	V Align	Italic	Bold	Text Size	Orientation	X Position	Y Position
Reference	U2	☑	Center	Center	□	□	1.27 mm	Horizontal	0 mm	3.81 mm
Value	Dialog_SLG5NT1487V	☑	Center	Center	□	□	1.27 mm	Horizontal	0 mm	8.89 mm
Footprint	-8_1.5x2mm_Fused-Lead_JEDEC_MO-252_W2015D	☑	Center	Center	□	□	0.25 mm	Horizontal	0 mm	-10.668 mm
Datasheet		□	Center	Center	□	□	1.27 mm	Horizontal	0 mm	5.08 mm
MPN	SLG5NT1487VTR	□	Center	Center	□	□	1.27 mm	Horizontal	0 mm	0 mm
Manufacturer		□	Center	Center	□	□	1.27 mm	Horizontal	0 mm	0 mm

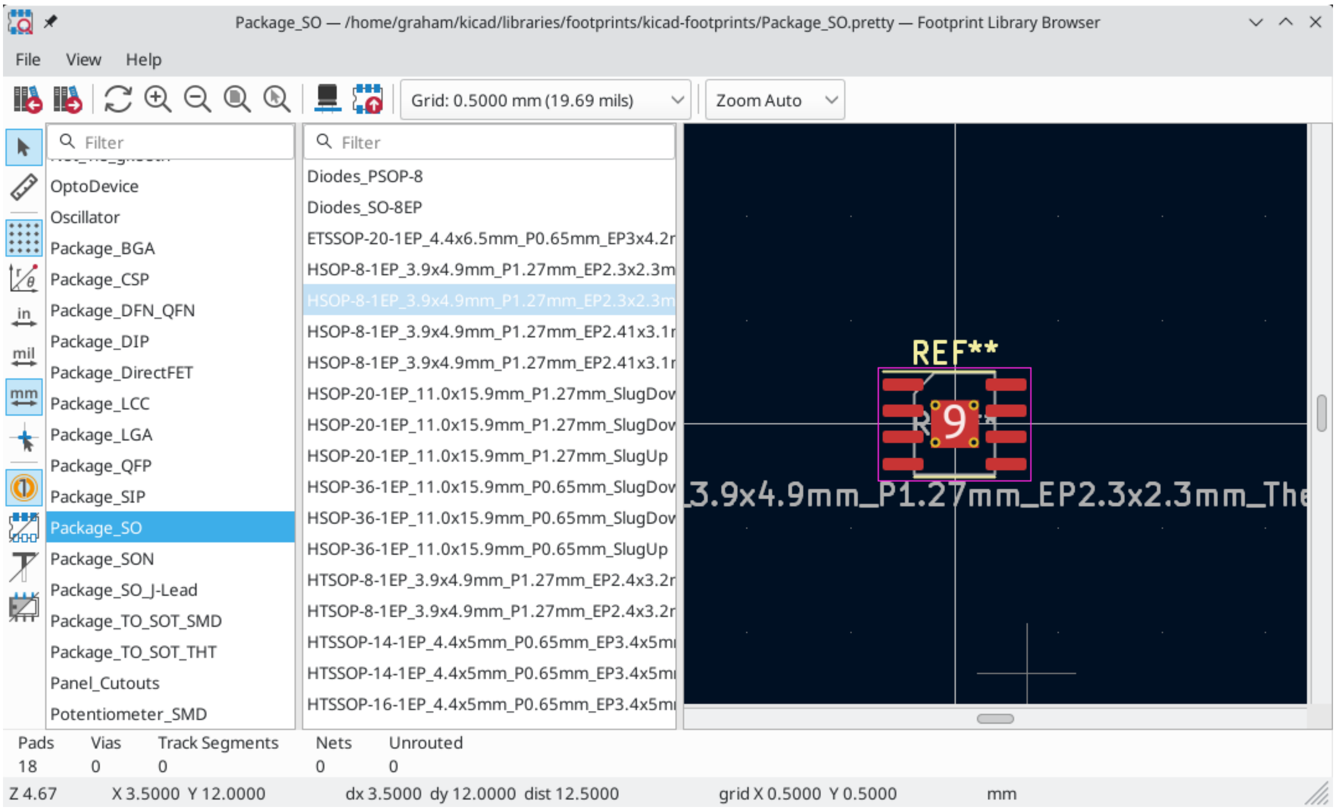
General
Unit:
□ Alternate symbol (De Morgan)
Angle: 0
Mirror: Not mirrored

Pin Text
☑ Show pin numbers
☑ Show pin names

Attributes
□ Exclude from bill of materials
□ Exclude from board

Library link: RobotProtos:Dialog_SLG5NT1487V

Clicking the 📚 button in the `Footprint` field opens the Footprint Library Browser, which shows the available footprints and footprint libraries. Single clicking a footprint name selects the footprint and displays it in the preview pane on the right, while double clicking on a footprint closes the browser and sets the symbol's `Footprint` field to the selected footprint.

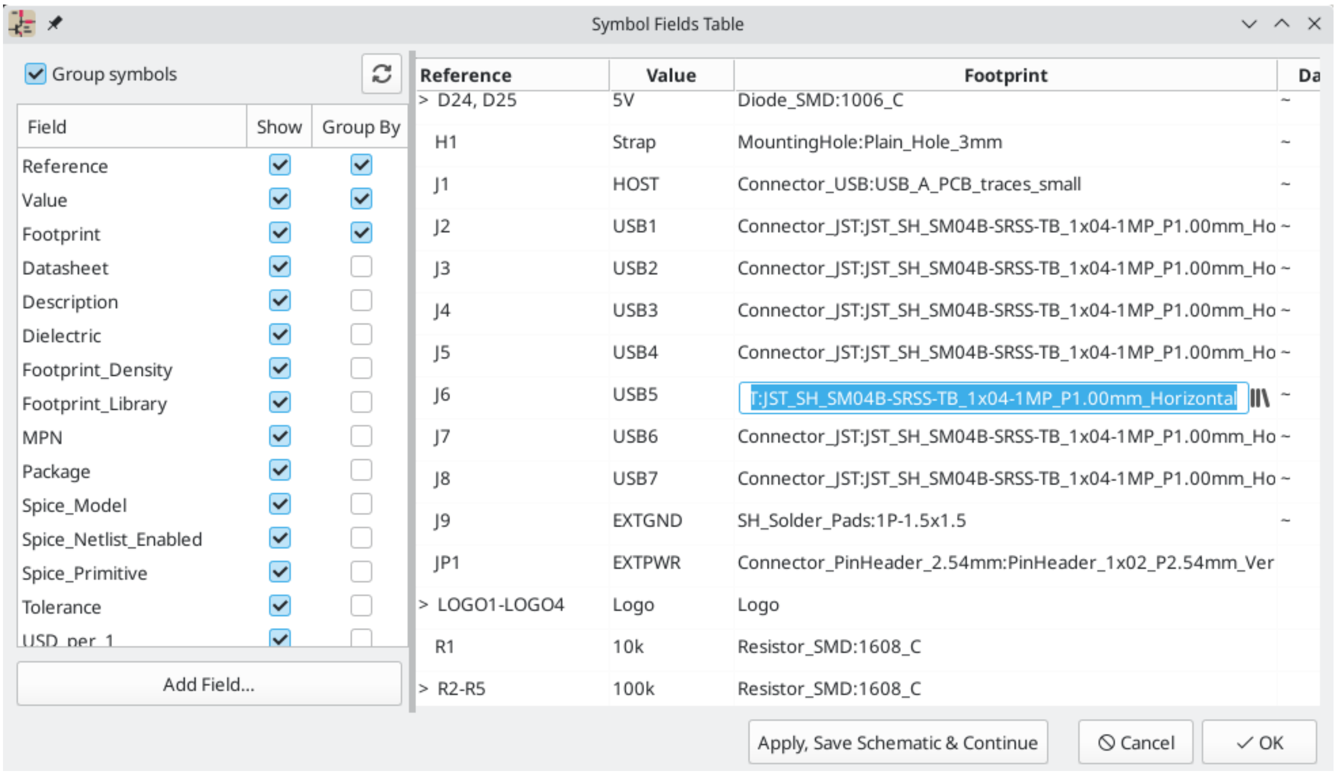

Package_SO — /home/graham/kicad/libraries/footprints/kicad-footprints/Package_SO.pretty — Footprint Library Browser
File
View
Help
Grid: 0.5000 mm (19.69 mils)
Zoom Auto
Filter
OptoDevice
Oscillator
Package_BGA
Package_CSP
Package_DFN_QFN
Package_DIP
Package_DirectFET
Package_LCC
Package_LGA
Package_QFP
Package_SIP
Package_SO
Package_SON
Package_SO_J-Lead
Package_TO_SOT_SMD
Package_TO_SOT_THT
Panel_Cutouts
Potentiometer_SMD
Filter
Diodes_PSOP-8
Diodes_SO-8EP
ETSSOP-20-1EP_4.4x6.5mm_P0.65mm_EP3x4.2r
HSOP-8-1EP_3.9x4.9mm_P1.27mm_EP2.3x2.3m
HSOP-8-1EP_3.9x4.9mm_P1.27mm_EP2.3x2.3m
HSOP-8-1EP_3.9x4.9mm_P1.27mm_EP2.41x3.1r
HSOP-8-1EP_3.9x4.9mm_P1.27mm_EP2.41x3.1r
HSOP-20-1EP_11.0x15.9mm_P1.27mm_SlugDov
HSOP-20-1EP_11.0x15.9mm_P1.27mm_SlugDov
HSOP-20-1EP_11.0x15.9mm_P1.27mm_SlugUp
HSOP-36-1EP_11.0x15.9mm_P0.65mm_SlugDov
HSOP-36-1EP_11.0x15.9mm_P0.65mm_SlugDov
HSOP-36-1EP_11.0x15.9mm_P0.65mm_SlugUp
HTSOP-8-1EP_3.9x4.9mm_P1.27mm_EP2.4x3.2r
HTSOP-8-1EP_3.9x4.9mm_P1.27mm_EP2.4x3.2r
HTSSOP-14-1EP_4.4x5mm_P0.65mm_EP3.4x5mi
HTSSOP-14-1EP_4.4x5mm_P0.65mm_EP3.4x5mi
HTSSOP-16-1EP_4.4x5mm_P0.65mm_EP3.4x5mi
REF**
9
3.9x4.9mm_P1.27mm_EP2.3x2.3mm_The
Pads 18
Vias 0
Track Segments 0
Nets 0
Unrouted 0
Z 4.67
X 3.5000 Y 12.0000
dx 3.5000 dy 12.0000 dist 12.5000
grid X 0.5000 Y 0.5000
mm

Assigning Footprints with the Symbol Fields Table

Rather than editing the properties of each symbol individually, the Symbol Fields Table can be used to view and edit the properties of all symbols in the design in one place. This includes assigning footprints by editing the `Footprint` field of each symbol.

The Symbol Fields Table is accessed with **Tools** → **Edit Symbol Fields...**, or with the ▦ button on the top toolbar.

The `Footprint` field behaves the same here as in the Symbol Properties window: it can be edited directly, or footprints can be selected visually with the Footprint Library Browser.



Field	Show	Group By
Reference	☑	☑
Value	☑	☑
Footprint	☑	☑
Datasheet	☑	□
Description	☑	□
Dielectric	☑	□
Footprint_Density	☑	□
Footprint_Library	☑	□
MPN	☑	□
Package	☑	□
Spice_Model	☑	□
Spice_Netlist_Enabled	☑	□
Spice_Primitive	☑	□
Tolerance	☑	□
USD_per_1	☑	□

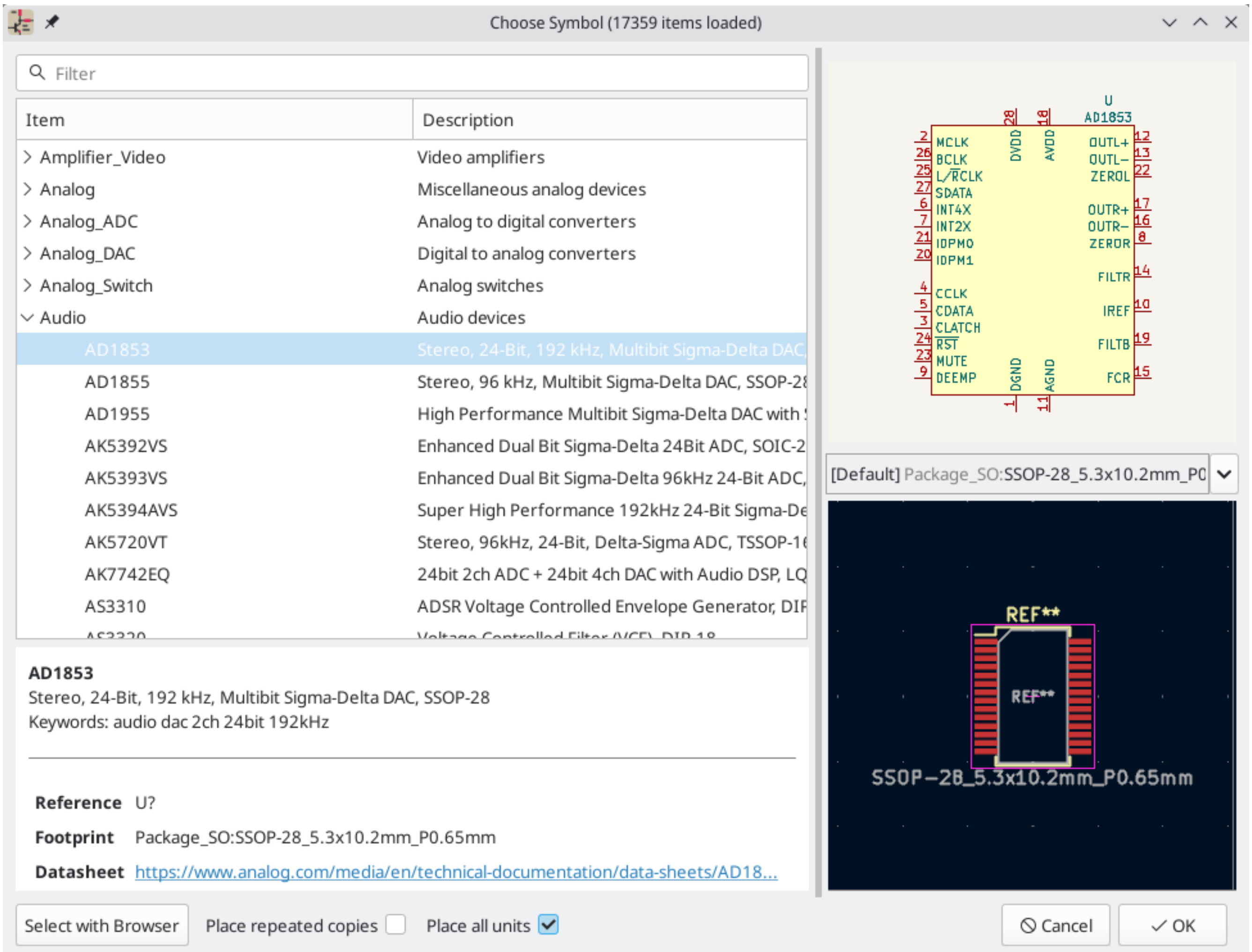
Reference	Value	Footprint	Da
> D24, D25	5V	Diode_SMD:1006_C	~
H1	Strap	MountingHole:Plain_Hole_3mm	~
J1	HOST	Connector_USB:USB_A_PCB_traces_small	~
J2	USB1	Connector_JST:JST_SH_SM04B-SRSS-TB_1x04-1MP_P1.00mm_Ho	~
J3	USB2	Connector_JST:JST_SH_SM04B-SRSS-TB_1x04-1MP_P1.00mm_Ho	~
J4	USB3	Connector_JST:JST_SH_SM04B-SRSS-TB_1x04-1MP_P1.00mm_Ho	~
J5	USB4	Connector_JST:JST_SH_SM04B-SRSS-TB_1x04-1MP_P1.00mm_Ho	~
J6	USB5	T:JST_SH_SM04B-SRSS-TB_1x04-1MP_P1.00mm_Horizontal	~
J7	USB6	Connector_JST:JST_SH_SM04B-SRSS-TB_1x04-1MP_P1.00mm_Ho	~
J8	USB7	Connector_JST:JST_SH_SM04B-SRSS-TB_1x04-1MP_P1.00mm_Ho	~
J9	EXTGND	SH_Solder_Pads:1P-1.5x1.5	~
JP1	EXTPWR	Connector_PinHeader_2.54mm:PinHeader_1x02_P2.54mm_Ver	
> LOGO1-LOGO4	Logo	Logo	
R1	10k	Resistor_SMD:1608_C	
> R2-R5	100k	Resistor_SMD:1608_C	

For more information on the Symbol Fields Table, see the section on editing symbol properties.

Assigning Footprints While Placing Symbols

Footprints can be assigned to symbols when the symbol is first added to the schematic.

Some symbols are defined with a default footprint. These symbols will have this footprint preassigned when they are added to the schematic. The default footprint is shown in the Add Symbol dialog. For symbols without a default symbol defined, the footprint dropdown will say "No default footprint", and the footprint preview canvas will say "No footprint specified".



Choose Symbol (17359 items loaded)

Filter

Item	Description
Amplifier_Video	Video amplifiers
Analog	Miscellaneous analog devices
Analog_ADC	Analog to digital converters
Analog_DAC	Digital to analog converters
Analog_Switch	Analog switches
Audio	Audio devices
AD1853	Stereo, 24-Bit, 192 kHz, Multibit Sigma-Delta DAC,
AD1855	Stereo, 96 kHz, Multibit Sigma-Delta DAC, SSOP-28
AD1955	High Performance Multibit Sigma-Delta DAC with
AK5392VS	Enhanced Dual Bit Sigma-Delta 24Bit ADC, SOIC-2
AK5393VS	Enhanced Dual Bit Sigma-Delta 96kHz 24-Bit ADC,
AK5394AVS	Super High Performance 192kHz 24-Bit Sigma-De
AK5720VT	Stereo, 96kHz, 24-Bit, Delta-Sigma ADC, TSSOP-16
AK7742EQ	24bit 2ch ADC + 24bit 4ch DAC with Audio DSP, LQ
AS3310	ADSR Voltage Controlled Envelope Generator, DIP

AD1853
Stereo, 24-Bit, 192 kHz, Multibit Sigma-Delta DAC, SSOP-28
Keywords: audio dac 2ch 24bit 192kHz

Reference U?
Footprint Package_SO:SSOP-28_5.3x10.2mm_P0.65mm
Datasheet https://www.analog.com/media/en/technical-documentation/data-sheets/AD18...

[Default] Package_SO:SSOP-28_5.3x10.2mm_P0

Symbols can have footprint filters that specify which footprints are appropriate to use with that symbol. If footprint filters are defined for the selected symbol, all footprints that match the footprint filters will appear as options in the footprint dropdown. The selected footprint will be displayed in the preview canvas and will be assigned to the symbol when the symbol is added to the schematic.

NOTE Footprint options will not appear in the footprint dropdown unless the footprint libraries are loaded. Footprint libraries are loaded the first time the Footprint Editor or Footprint Library Browser are opened in a session.

For more information on footprint filters, see the Symbol Editor Documentation.

Assigning Footprints with the Footprint Assignment Tool

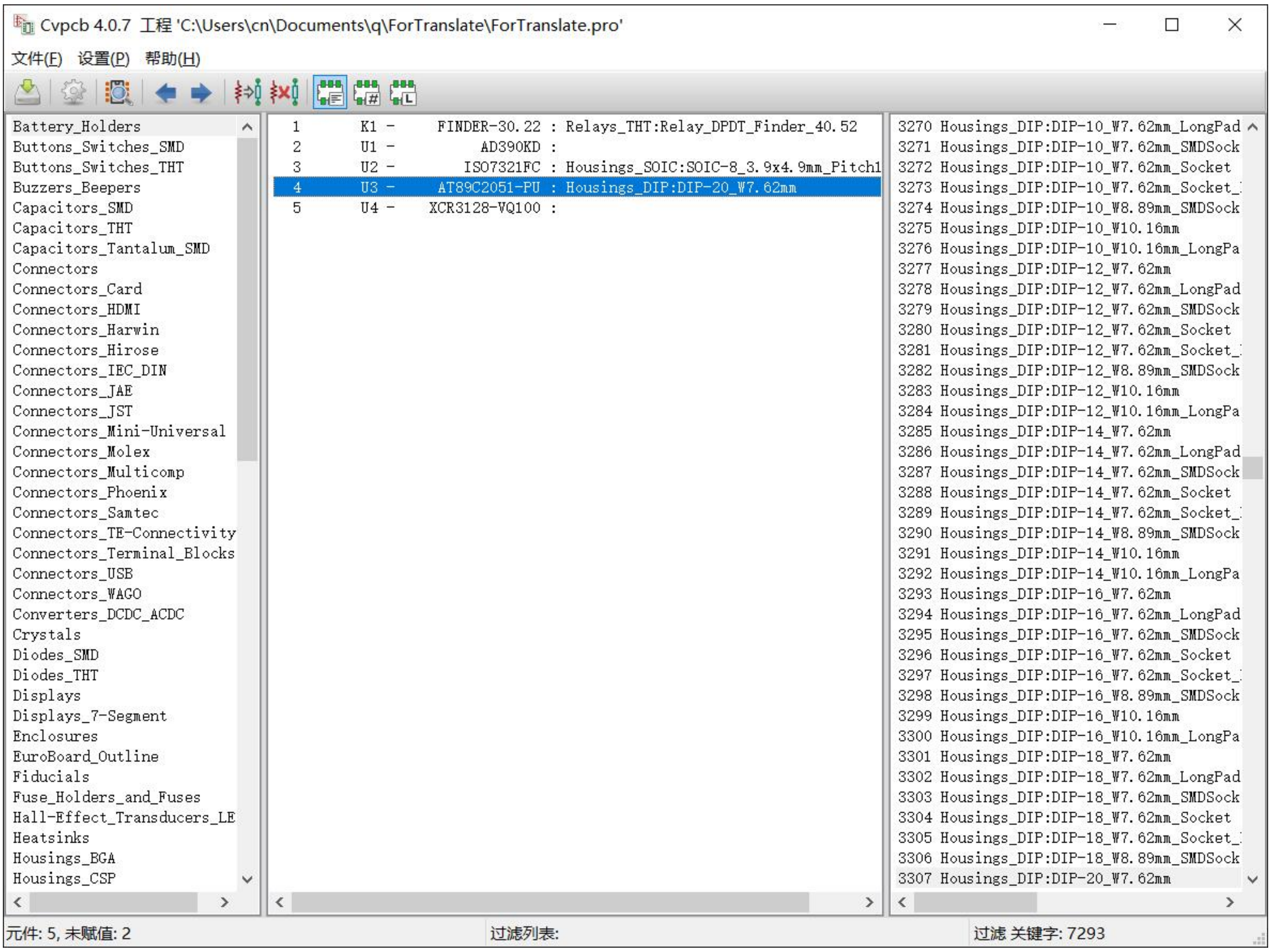
The Footprint Assignment Tool allows you to associate symbols in your schematic to footprints used when laying out the printed circuit board. It provides footprint list filtering, footprint viewing, and 3D component model viewing to help ensure the correct footprint is associated with each component.

Components can be assigned to their corresponding footprints manually or automatically by creating equivalence files (.equ files). Equivalence files are lookup tables associating each component with its footprint.

Run the tool with **Tools** → **Assign Footprints...**, or by clicking the icon in the top toolbar.

Footprint Assignment Tool Overview

The image below shows the main window of the Footprint Assignment Tool.



```
Cvpcb 4.0.7 工程 'C:\Users\cn\Documents\q\ForTranslate\ForTranslate.pro'
```

```
Battery_Holders
Buttons_Switches_SMD
Buttons_Switches_THT
Buzzers_Beepers
Capacitors_SMD
Capacitors_THT
Capacitors_Tantalum_SMD
Connectors
Connectors_Card
Connectors_HDMI
Connectors_Harwin
Connectors_Hirose
Connectors_IEC_DIN
Connectors_JAE
Connectors_JST
Connectors_Mini-Universal
Connectors_Molex
Connectors_Multicomp
Connectors_Phoenix
Connectors_Samtec
Connectors_TE-Connectivity
Connectors_Terminal_Blocks
Connectors_USB
Connectors_WAGO
Converters_DCDC_ACDC
Crystals
Diodes_SMD
Diodes_THT
Displays
Displays_7-Segment
Enclosures
EuroBoard_Outline
Fiducials
Fuse_Holders_and_Fuses
Hall-Effect_Transducers_LE
Heatsinks
Housings_BGA
Housings_CSP
```

```
1      K1 -     FINDER-30.22 : Relays_THT:Relay_DPDT_Finder_40.52
2      U1 -          AD390KD :
3      U2 -        ISO7321FC : Housings_SOIC:SOIC-8_3.9x4.9mm_Pitch1
4      U3 -     AT89C2051-PU : Housings_DIP:DIP-20_W7.62mm
5      U4 -    XCR3128-VQ100 :
```

```
3270 Housings_DIP:DIP-10_W7.62mm_LongPad
3271 Housings_DIP:DIP-10_W7.62mm_SMDSock
3272 Housings_DIP:DIP-10_W7.62mm_Socket
3273 Housings_DIP:DIP-10_W7.62mm_Socket_
3274 Housings_DIP:DIP-10_W8.89mm_SMDSock
3275 Housings_DIP:DIP-10_W10.16mm
3276 Housings_DIP:DIP-10_W10.16mm_LongPa
3277 Housings_DIP:DIP-12_W7.62mm
3278 Housings_DIP:DIP-12_W7.62mm_LongPad
3279 Housings_DIP:DIP-12_W7.62mm_SMDSock
3280 Housings_DIP:DIP-12_W7.62mm_Socket
3281 Housings_DIP:DIP-12_W7.62mm_Socket_
3282 Housings_DIP:DIP-12_W8.89mm_SMDSock
3283 Housings_DIP:DIP-12_W10.16mm
3284 Housings_DIP:DIP-12_W10.16mm_LongPa
3285 Housings_DIP:DIP-14_W7.62mm
3286 Housings_DIP:DIP-14_W7.62mm_LongPad
3287 Housings_DIP:DIP-14_W7.62mm_SMDSock
3288 Housings_DIP:DIP-14_W7.62mm_Socket
3289 Housings_DIP:DIP-14_W7.62mm_Socket_
3290 Housings_DIP:DIP-14_W8.89mm_SMDSock
3291 Housings_DIP:DIP-14_W10.16mm
3292 Housings_DIP:DIP-14_W10.16mm_LongPa
3293 Housings_DIP:DIP-16_W7.62mm
3294 Housings_DIP:DIP-16_W7.62mm_LongPad
3295 Housings_DIP:DIP-16_W7.62mm_SMDSock
3296 Housings_DIP:DIP-16_W7.62mm_Socket
3297 Housings_DIP:DIP-16_W7.62mm_Socket_
3298 Housings_DIP:DIP-16_W8.89mm_SMDSock
3299 Housings_DIP:DIP-16_W10.16mm
3300 Housings_DIP:DIP-16_W10.16mm_LongPa
3301 Housings_DIP:DIP-18_W7.62mm
3302 Housings_DIP:DIP-18_W7.62mm_LongPad
3303 Housings_DIP:DIP-18_W7.62mm_SMDSock
3304 Housings_DIP:DIP-18_W7.62mm_Socket
3305 Housings_DIP:DIP-18_W7.62mm_Socket_
3306 Housings_DIP:DIP-18_W8.89mm_SMDSock
3307 Housings_DIP:DIP-20_W7.62mm
```

```
元件: 5, 未赋值: 2          过滤列表:          过滤 关键字: 7293
```

- The left pane contains the list of available footprint libraries associated with the project.
- The center pane contains the list of symbols in the schematic.
- The right pane contains the list of available footprints loaded from the project footprint libraries.
- The bottom pane describes the filters that have been applied to the footprint list and prints information about the footprint selected in the rightmost pane.

The top toolbar contains the following commands:

💾	Transfer the current footprint associations to the schematic.
📚	Edit the global and project footprint library tables.
🔍	View the selected footprint in the footprint viewer.
←	Select the previous symbol without a footprint association.
→	Select the next symbol without a footprint association.
↶	Undo last edit.
↷	Redo last edit.
⟳	Perform automatic footprint association using an equivalence file.
✖	Delete all footprint assignments.
☰	Filter footprint list by footprint filters defined in the selected symbol.
#	Filter footprint list by pin count of the selected symbol.
L	Filter footprint list by selected library.

The following table lists the keyboard commands for the Footprint Assignment Tool:

Right Arrow / Tab	Activate the pane to the right of the currently activated pane. Wrap around to the first pane if the last pane is currently activated.
Left Arrow	Activate the pane to the left of the currently activated pane. Wrap around to the last pane if the first pane is currently activated.
Up Arrow	Select the previous item of the currently selected list.
Down Arrow	Select the next item of the currently selected list.
Page Up	Select the item one full page upwards of the currently selected item.
Page Down	Select the item one full page downwards of the currently selected item.
Home	Select the first item of the currently selected list.
End	Select the last item of the currently selected list.

Manually Assigning Footprints with the Footprint Assignment Tool

To manually associate a footprint with a component, first select a component in the component (middle) pane. Then select a footprint in the footprint (right) pane by double-clicking on the name of the desired footprint. The footprint will be assigned to the selected component, and the next component without an assigned footprint is automatically selected.

> **NOTE** If no footprints appear in the footprint pane, check that the footprint filter options are correctly applied.

When all components have footprints assigned to them, click the **OK** button to save the assignments and exit the tool. Alternatively, click **Cancel** to discard the updated assignments, or **Apply, Save Schematic & Continue** to save the new assignments without exiting the tool.

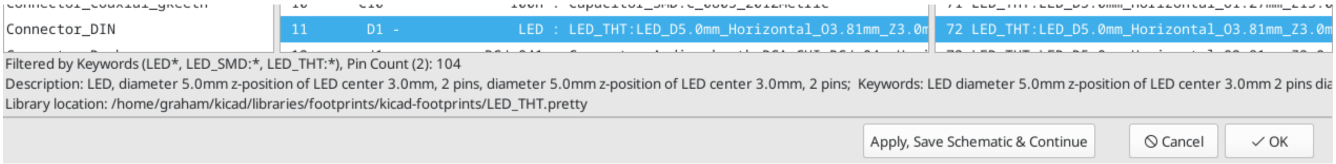
过滤封装列表

There are four filtering options which restrict which footprints are displayed in the footprint pane. The filtering options are enabled and disabled with three buttons and a textbox in the top toolbar.

- ≡: Activate filters that can be defined in each symbol. For example, an opamp symbol might define filters that show only SOIC and DIP footprints.
- #: Only show footprints that match the selected symbol's pin count.
- L: Only show footprints from the library selected in the left pane.
- Entering text in the textbox hides footprints that do not match the text. This filter is disabled when the box is empty.

When all filters are disabled, the full footprint list is shown.

The applied filters are described in the bottom pane of the window, along with the number of footprints that meet the selected filters. For example, when the symbol's footprint filters and pin count filters are enabled, the bottom pane prints the footprint filters and pin count:



```
Connector_DIN    11    D1 -    LED : LED_THT:LED_D5.0mm_Horizontal_O3.81mm_Z3.0m    72 LED_THT:LED_D5.0mm_Horizontal_O3.81mm_Z3.0m
```

Filtered by Keywords (LED*, LED_SMD:*, LED_THT:*), Pin Count (2): 104
Description: LED, diameter 5.0mm z-position of LED center 3.0mm, 2 pins, diameter 5.0mm z-position of LED center 3.0mm, 2 pins; Keywords: LED diameter 5.0mm z-position of LED center 3.0mm 2 pins dia
Library location: /home/graham/kicad/libraries/footprints/kicad-footprints/LED_THT.pretty

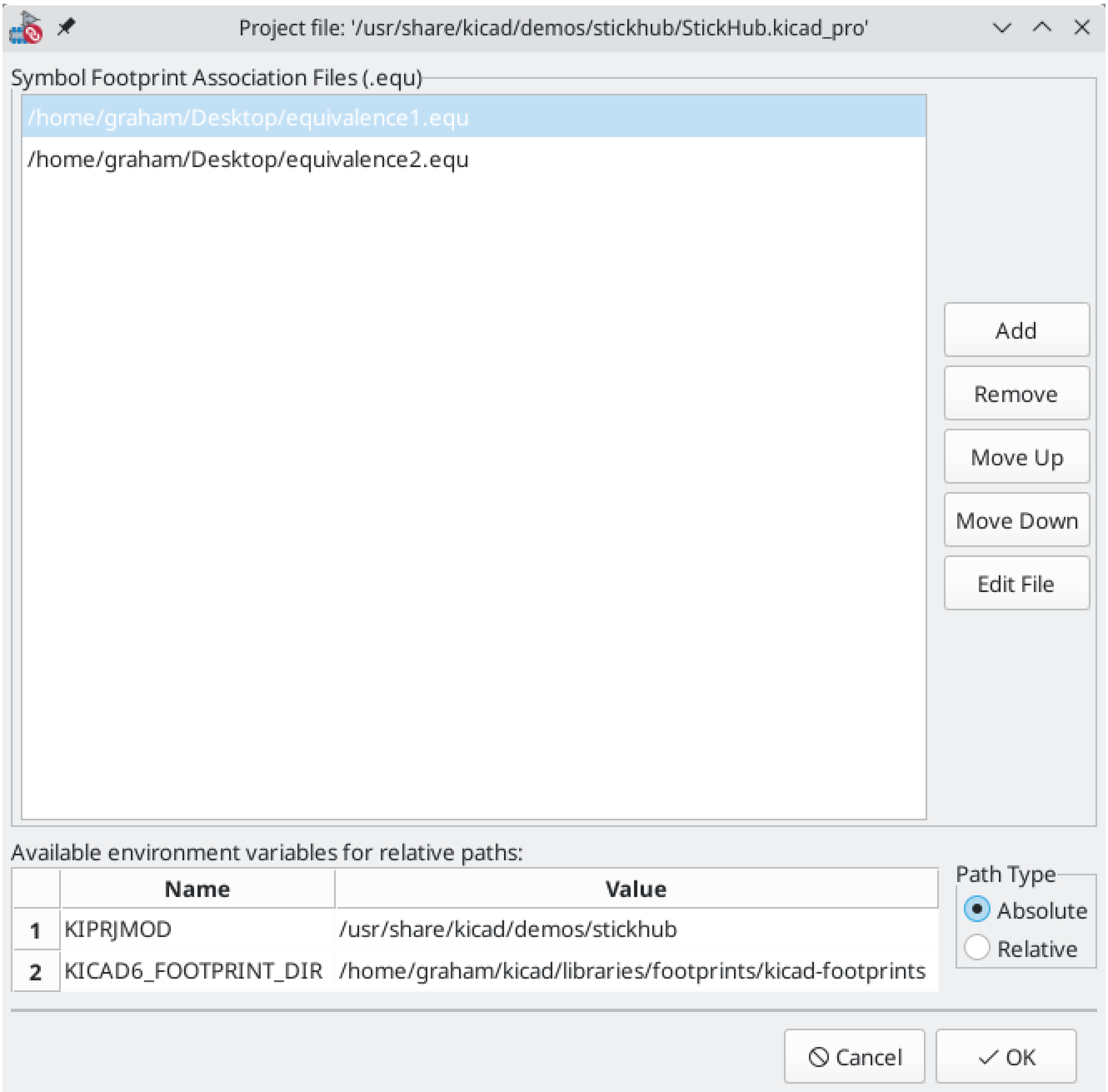
Multiple filters can be used at once to help narrow down the list of possibly appropriate footprints in the footprint pane. The symbols in KiCad's standard library define footprint filters that are designed to be used in combination with the pin count filter.

Automatically Assigning Footprints with the Footprint Assignment Tool

The Footprint Assignment Tool allows you to store footprint assignments in an external file and load the assignments later, even in a different project. This allows you to automatically associate symbols with the appropriate footprints.

The external file is referred to as an equivalence file, and it stores a mapping of a symbol value to a corresponding footprint. Equivalence files typically use the `.equ` file extension. Equivalence files are plain text files with a simple syntax, and must be created by the user using a text editor. The syntax is described below.

You can select which equivalence files to use by clicking **Preferences** → **Manage Footprint Association Files** in the Footprint Assignment Tool.



Project file: '/usr/share/kicad/demos/stickhub/StickHub.kicad_pro'

Symbol Footprint Association Files (.equ)

/home/graham/Desktop/equivalence1.equ

/home/graham/Desktop/equivalence2.equ

Available environment variables for relative paths:

	Name	Value
1	KIPRJMOD	/usr/share/kicad/demos/stickhub
2	KICAD6_FOOTPRINT_DIR	/home/graham/kicad/libraries/footprints/kicad-footprints

Path Type

● Absolute

○ Relative

- Add new equivalence files by clicking the **Add** button.
- Remove the selected equivalence file by clicking the **Remove** button.
- Change the priority of equivalence files by clicking the **Move Up** and **Move Down** buttons. If a symbol's value is found in multiple equivalence files, the footprint from the last matching equivalence file will override earlier equivalence files.
- Open the selected equivalence file by clicking the **Edit File** button.

Relevant environment variables are shown at the bottom of the window. When the **Relative** path option is checked, these environment variables will automatically be used to make paths to selected equivalence files relative to the project or footprint libraries.

Once the desired equivalence files have been loaded in the correct order, automatic footprint association can be performed by clicking the button in the top toolbar of the Footprint Assignment Tool.

All symbols with a value found in a loaded equivalence file will have their footprints automatically assigned. However, symbols that already have footprints assigned will not be updated.

Equivalence 文件格式

Equivalence files consist of one line for each symbol value. Each line has the following structure:

```
'<symbol value>' '<footprint library>:<footprint name>'
```

Each name/value must be surrounded by single quotes (`'`) and separated by one or more spaces. Lines starting with `#` are comments.

For example, if you want all symbols with the value `LM4562` to be assigned the footprint `Package_SO:SOIC-8_3.9x4.9_P1.27mm`, the line in the equivalence file should be:

```
'LM4562' 'Package_SO:SOIC-8_3.9x4.9_P1.27mm'
```

Equivalence 文件示例:

```
#regulators 'LP2985LV' 'Package_TO_SOT_SMD:SOT-23-5_HandSoldering' ```

==== 查看当前封装

The Footprint Assignment Tool contains a footprint viewer. Clicking the
image:images/icons/icon_footprint_browser_24.png[footprint viewer icon] button in the top
toolbar launches the footprint viewer and shows the selected footprint.

image::images/zh/footprint_view.png[scaledwidth="90%", alt="查看封装"]

The top toolbar contains the following commands:

[width="90%", cols="10%,90%"]
|=======================================================================
|image:images/icons/refresh_24.png[]
|Refresh view
|image:images/icons/zoom_in_24.png[]
|Zoom in

|image:images/icons/zoom_out_24.png[]
|Zoom out

|image:images/icons/zoom_fit_in_page_24.png[]
|Zoom to fit drawing in display area

|image:images/icons/shape_3d_24.png[]
|Show 3D viewer
|=======================================================================

The left toolbar contains the following commands:

[width="90%", cols="10%,90%"]
|=======================================================================
|image:images/icons/cursor_24.png[]
|Use the select tool

|image:images/icons/measurement_24.png[]
|Interactively measure between two points

|image:images/icons/grid_24.png[]
|Display grid dots or lines

|image:images/icons/polar_coord_24.png[]
|Switch between polar and cartesian coordinate systems

|image:images/icons/unit_inch_24.png[]
|Use inches

|image:images/icons/unit_mil_24.png[]
|Display coordinates in mils (1/1000 of an inch)

|image:images/icons/unit_mm_24.png[]
|Display coordinates in millimeters

|image:images/icons/cursor_shape_24.png[]
|Toggle display of full-window crosshairs

|image:images/icons/pad_number_24.png[]
|Toggle between drawing pads in sketch or normal mode
```

(export (version "E") (design (source "/usr/share/kicad/demos/simulation/sallen_key/sallen_key.kicad_sch") (date "Sun 01 May 2022 03:14:05 PM EDT") (tool "Eeschema (6.0.4)") (sheet (number "1") (name "/") (tstamps "/") (title_block (title) (company) (rev) (date) (source "sallen_key.kicad_sch") (comment (number "1") (value "")) (comment (number "2") (value "")) (comment (number "3") (value "")) (comment (number "4") (value "")) (comment (number "5") (value "")) (comment (number "6") (value "")) (comment (number "7") (value "")) (comment (number "8") (value "")) (comment (number "9") (value "")))))) (components (comp (ref "C1") (value "100n") (libsource (lib "sallen_key_schlib") (part "C") (description "")) (property (name "Sheetname") (value "")) (property (name "Sheetfile") (value "sallen_key.kicad_sch")) (sheetpath (names "/") (tstamps "/")) (tstamps "00000000-0000-0000-0000-00005789077d")) (comp (ref "C2") (value "100n") (fields (field (name "Fieldname") "Value") (field (name "SpiceMapping") "1 2") (field (name "Spice_Primitive") "C")) (libsource (lib "sallen_key_schlib") (part "C") (description "")) (property (name "Fieldname") (value "Value")) (property (name "Spice_Primitive") (value "C")) (property (name "SpiceMapping") (value "1 2")) (property (name "Sheetname") (value "")) (property (name "Sheetfile") (value "sallen_key.kicad_sch")) (sheetpath (names "/") (tstamps "/")) (tstamps "00000000-0000-0000-0000-00005789085b")) (comp (ref "R1") (value "1k") (fields (field (name "Fieldname") "Value") (field (name "SpiceMapping") "1 2") (field (name "Spice_Primitive") "R")) (libsource (lib "sallen_key_schlib") (part "R") (description "")) (property (name "Fieldname") (value "Value")) (property (name "SpiceMapping") (value "1 2")) (property (name "Spice_Primitive") (value "R")) (property (name "Sheetname") (value "")) (property (name "Sheetfile") (value "sallen_key.kicad_sch")) (sheetpath (names "/") (tstamps "/")) (tstamps "00000000-0000-0000-0000-0000578906ff")) (comp (ref "R2") (value "1k") (fields (field (name "Fieldname") "Value") (field (name "SpiceMapping") "1 2") (field (name "Spice_Primitive") "R")) (libsource (lib "sallen_key_schlib") (part "R") (description "")) (property (name "Fieldname") (value "Value")) (property (name "SpiceMapping") (value "1 2")) (property (name "Spice_Primitive") (value "R")) (property (name "Sheetname") (value "")) (property (name "Sheetfile") (value "sallen_key.kicad_sch")) (sheetpath (names "/") (tstamps "/")) (tstamps "00000000-0000-0000-0000-000057890691")) (comp (ref "U1") (value "AD8051") (fields (field (name "Spice_Lib_File") "ad8051.lib") (field (name "Spice_Model") "AD8051") (field (name "Spice_Netlist_Enabled") "Y") (field (name "Spice_Primitive") "X")) (libsource (lib "sallen_key_schlib") (part "Generic_Opamp") (description "")) (property (name "Spice_Primitive") (value "X")) (property (name "Spice_Model") (value "AD8051")) (property (name "Spice_Lib_File") (value "ad8051.lib")) (property (name "Spice_Netlist_Enabled") (value "Y")) (property (name "Sheetname") (value "")) (property (name "Sheetfile") (value "sallen_key.kicad_sch")) (sheetpath (names "/") (tstamps "/")) (tstamps "00000000-0000-0000-0000-00005788ff9f")) (comp (ref "V1") (value "AC 1") (libsource (lib "sallen_key_schlib") (part "VSOURCE") (description "")) (property (name "Sheetname") (value "")) (property (name "Sheetfile") (value "sallen_key.kicad_sch")) (sheetpath (names "/") (tstamps "/")) (tstamps "00000000-0000-0000-0000-000057336052")) (comp (ref "V2") (value "DC 10") (fields (field (name "Fieldname") "Value") (field (name "Spice_Node_Sequence") "1 2") (field (name "Spice_Primitive") "V")) (libsource (lib "sallen_key_schlib") (part "VSOURCE") (description "")) (property (name "Fieldname") (value "Value")) (property (name "Spice_Primitive") (value "V")) (property (name "Spice_Node_Sequence") (value "1 2")) (property (name "Sheetname") (value "")) (property (name "Sheetfile") (value "sallen_key.kicad_sch")) (sheetpath (names "/") (tstamps "/")) (tstamps "00000000-0000-0000-0000-0000578900ba")) (comp (ref "V3") (value "DC 10") (fields (field (name "Fieldname") "Value") (field (name "Spice_Node_Sequence") "1 2") (field (name "Spice_Primitive") "V")) (libsource (lib "sallen_key_schlib") (part "VSOURCE") (description "")) (property (name "Fieldname") (value "Value")) (property (name "Spice_Primitive") (value "V")) (property (name "Spice_Node_Sequence") (value "1 2")) (property (name "Sheetname") (value "")) (property (name "Sheetfile") (value "sallen_key.kicad_sch")) (sheetpath (names "/") (tstamps "/")) (tstamps "00000000-0000-0000-0000-000057890232"))) (libparts (libpart (lib "sallen_key_schlib") (part "C") (footprints (fp "C?") (fp "C_????*") (fp "C????") (fp "SMD*c") (fp "Capacitor*")) (fields (field (name "Reference") "C") (field (name "Value") "C")) (pins (pin (num "1") (name "") (type "passive")) (pin (num "2") (name "") (type "passive")))) (libpart (lib

*"sallen_key_schlib") (part "Generic_Opamp") (fields (field (name "Reference") "U") (field (name "Value") "Generic_Opamp")) (pins (pin (num "1") (name "") (type "input")) (pin (num "2") (name "-") (type "input")) (pin (num "3") (name "V") (type "power_in")) (pin (num "4") (name "V-") (type "power_in")) (pin (num "5") (name "") (type "output")))) (libpart (lib "sallen_key_schlib") (part "R") (footprints (fp "R**") (fp "Resistor_*")) (fields (field (name "Reference") "R") (field (name "Value") "R")) (pins (pin (num "1") (name "") (type "passive")) (pin (num "2") (name "") (type "passive")))) (libpart (lib "sallen_key_schlib") (part "VSOURCE") (fields (field (name "Reference") "V") (field (name "Value") "VSOURCE") (field (name "Fieldname") "Value") (field (name "Spice_Primitive") "V") (field (name "Spice_Node_Sequence") "1 2")) (pins (pin (num "1") (name "") (type "input")) (pin (num "2") (name "") (type "input"))))) (libraries (library (logical "sallen_key_schlib") (uri "/usr/share/kicad/demos/simulation/sallen_key/sallen_key_schlib.kicad_sym"))) (nets (net (code "1") (name "/lowpass") (node (ref "C1") (pin "1") (pintype "passive")) (node (ref "U1") (pin "2") (pinfunction "-") (pintype "input")) (node (ref "U1") (pin "5") (pintype "output"))) (net (code "2") (name "GND") (node (ref "C2") (pin "2") (pintype "passive")) (node (ref "V1") (pin "2") (pintype "input")) (node (ref "V2") (pin "2") (pintype "input")) (node (ref "V3") (pin "1") (pintype "input"))) (net (code "3") (name "Net-(C1-Pad2)") (node (ref "C1") (pin "2") (pintype "passive")) (node (ref "R1") (pin "1") (pintype "passive")) (node (ref "R2") (pin "2") (pintype "passive"))) (net (code "4") (name "Net-(C2-Pad1)") (node (ref "C2") (pin "1") (pintype "passive")) (node (ref "R2") (pin "1") (pintype "passive")) (node (ref "U1") (pin "1") (pinfunction "") (pintype "input"))) (net (code "5") (name "Net-(R1-Pad2)") (node (ref "R1") (pin "2") (pintype "passive")) (node (ref "V1") (pin "1") (pintype "input"))) (net (code "6") (name "VDD") (node (ref "U1") (pin "3") (pinfunction "V") (pintype "power_in")) (node (ref "V2") (pin "1") (pintype "input"))) (net (code "7") (name "VSS") (node (ref "U1") (pin "4") (pinfunction "V-") (pintype "power_in")) (node (ref "V3") (pin "2") (pintype "input")))))

```
In Spice format, the netlist is as follows:
```

include "ad8051.lib"

XU1 Net-*C2-Pad1* /lowpass VDD VSS /lowpass AD8051 C2 Net-*C2-Pad1* GND 100n C1 /lowpass Net-*C1-Pad2* 100n R2 Net-*C2-Pad1* Net-*C1-Pad2* 1k R1 Net-*C1-Pad2* Net-*R1-Pad2* 1k V1 Net-*R1-Pad2* GND AC 1 V2 VDD GND DC 10 V3 GND VSS DC 10 .ac dec 10 1 1Meg .end

```
:experimental:

[[managing-symbol-libraries]]
== Managing Symbol Libraries

符号库包含创建原理图时使用的符号集合。 原理图中的每个符号由一个全名唯一标识，该全名由库昵称和符号名称组成。 一个例子
是 “音频：AD1853”。

=== 符号库表

KiCad uses a table of symbol libraries to map symbol libraries to a library nickname. Kicad
uses a global symbol library table as well as a table specific to each project. To edit
either symbol library table, use **Preferences** -> **Manage Symbol Libraries...**.

image::images/zh/options_symbol_lib.png[scaledwidth="80%", alt="符号列表文件对话框"]

The global symbol library table contains the list of libraries that are always available
regardless of the currently loaded project. The table is saved in the file `sym-lib-table`
in the KiCad configuration folder. xref:../kicad/kicad.adoc#config-file-location[The
location of this folder] depends on the operating system being used.

The project specific symbol library table contains the list of libraries that are available
specifically for the currently loaded project. If there are any project-specific symbol
libraries, the table is saved in the file `sym-lib-table` in the project folder.

==== 初始配置

The first time the KiCad Schematic Editor is run and the global symbol table file `sym-lib-
table` is not found in the KiCad configuration folder, KiCad will guide the user through
setting up a new symbol library table. This process is described <<initial-
configuration,above>>.

==== Managing Table Entries

Symbol libraries can only be used if they have been added to either the global or project-
specific symbol library table.

Add a library either by clicking the image:images/icons/small_folder_16.png[Folder icon]
button and selecting a library or clicking the image:images/icons/small_plus_16.png[Plus
icon] button and typing the path to a library file. The selected library will be added to
the currently opened library table (Global or Project Specific). Libraries can be removed
by selecting desired library entries and clicking the
image:images/icons/small_trash_16.png[Delete icon] button.

The image:images/icons/small_up_16.png[Up icon] and
image:images/icons/small_down_16.png[Down icon] buttons move the selected library up and
down in the library table. This does not affect the display order of libraries in the
Symbol Library Browser, Symbol Editor, or Add Symbol tool.

Libraries can be made inactive by unchecking the *Active* checkbox in the first column.
Inactive libraries are still in the library table but do not appear in any library browsers
and are not loaded from disk, which can reduce loading times.

A range of libraries can be selected by clicking the first library in the range and then
kbd:[Shift]-clicking the last library in the range.

Each library must have a unique nickname: duplicate library nicknames are not allowed in
the same table. However, nicknames can be duplicated between the global and project library
tables. Libraries in the project table take precedence over libraries with the same name in
the global table.
```

一般网表文件结构

中间网表占五个部分。

- “标题” 部分。
- “元件” 部分。
- “库元件” 部分。
- “库” 部分。
- “网” 部分。

The file content has the delimiter `<export>`

~~~
 ...  ```" data-lang="xml  ...  ```">[[the-header-section]]
===== “标题” 部分

The header has the delimiter `<design>`

```xml <design> <source>F:\kicad_aux\netlist_test\netlist_test.sch</source>
<date>21/08/2010 08:12:08</date> <tool>eeschema (2010-08-09 BZR 2439)-unstable</tool>
</design> ```

此部分可被视为批注部分。

[[the-components-section]]
===== “元件” 部分

The component section has the delimiter `<components>`

```xml <components> <comp ref="P1"> <value>CONN_4</value> <libsource lib="conn"
part="CONN_4"/> <sheetpath names="/" tstamps="/"/> <tstamps>4C6E2141</tstamps> </comp>
</components> ``` This section contains the list of components in your schematic. Each
component is described like this:

```xml <comp ref="P1"> <value>CONN_4</value> <libsource lib="conn" part="CONN_4"/>
<sheetpath names="/" tstamps="/"/> <tstamps>4C6E2141</tstamps> </comp> ```

[width="100%", cols="37%,63%"]
|=======================================================================
|Element name |Element description

|`libsource` |name of the lib where this component was found.
|`part` |component name inside this library.
|`sheetpath` |path of the sheet inside the hierarchy: identify the sheet
within the full schematic hierarchy.
|`tstamps` |timestamp of the component.
|=======================================================================

[[note-about-time-stamps-for-components]]
====== Note about time stamps for components

To identify a component in a netlist and therefore on a board, the timestamp reference is
used as unique for each component. However KiCad provides an auxiliary way to identify a
component which is the corresponding footprint on the board. This allows the re-annotation
of components in a schematic project and does not lose the link between the component and
its footprint.

时间戳是原理图项目中每个元件或工作表的唯一标识符。但是，在复杂的层次结构中，同一工作表多次使用，因此此工作表包含具
有相同时间戳的元件。

A given sheet inside a complex hierarchy has an unique identifier: its sheetpath. A given
component (inside a complex hierarchy) has a unique identifier: the sheetpath and its
timestamp.

[[the-libparts-section]]
===== “库部件” 部分

The libparts section has the delimiter `<libparts>`, and the content of this section is
defined in the schematic libraries.

```xml
<libparts>
<libpart lib="device" part="CP">
~~~

Element name	Element description
`<footprints>`	The symbol’s footprint filters. Each footprint filter is in a separate `<fp>` tag.
`<fields>`	The symbol’s fields. Each field’s name and value is given in a separate `<field name="fieldname">… </field> tag.
`<pins>`	The symbol’s pins. Each pin is given in a separate `<pin num="pinnum" type="pintype"/>` tag. Possible pintypes are described below.

Possible electrical pin types are:

Pintype	Description
Input	Usual input pin
Output	Usual output
Bidirectional	Input or Output
Tri-state	Bus input/output
Passive	Usual ends of passive components
Unspecified	Unknown electrical type
Power input	Power input of a component
Power output	Power output like a regulator output
Open collector	Open collector often found in analog comparators
Open emitter	Open emitter sometimes found in logic
Not connected	Must be left open in schematic

“库” 部分

The libraries section has the delimiter `<libraries>`. This section contains the list of schematic libraries used in the project.

```
<libraries>
  <library logical="device">
    <uri>F:\kicad\share\library\device.lib</uri>
  </library>
  <library logical="conn">
    <uri>F:\kicad\share\library\conn.lib</uri>
  </library>
</libraries>
```

“网” 部分

The nets section has the delimiter `<nets>`. This section describes the connectivity of the schematic by listing all nets and the pins connected to each net.

```
<nets>
  <net code="1" name="GND">
    <node ref="U1" pin="7"/>
    <node ref="C1" pin="2"/>
    <node ref="U2" pin="7"/>
    <node ref="P1" pin="4"/>
  </net>
  <net code="2" name="VCC">
    <node ref="R1" pin="1"/>
    <node ref="U1" pin="14"/>
    <node ref="U2" pin="4"/>
    <node ref="U2" pin="1"/>
    <node ref="U2" pin="14"/>
    <node ref="P1" pin="1"/>
  </net>
</nets>
```

可能的网络包含以下内容。

```
<net code="1" name="GND">
  <node ref="U1" pin="7"/>
  <node ref="C1" pin="2"/>
  <node ref="U2" pin="7"/>
  <node ref="P1" pin="4"/>
</net>
```

Element name	Element Description
`net code`	an internal identifier for this net
`name`	the net name
`node`	the pin (identified by `pin`) of a symbol (identified by `ref`) which is connected to the net

Example netlist exporters

Some example netlist exporters using XSLT are included below.

XSLT itself is an XML language very suitable for XML transformations. The `xsltproc` program can be used to read the Intermediate XML netlist input file, apply a style-sheet to transform the input, and save the results in an output file. Use of `xsltproc` requires a style-sheet file using XSLT conventions. The full conversion process is handled by KiCad, after it is configured once to run `xsltproc` in a specific way.

The document that describes XSL Transformations (XSLT) is available here: http://www.w3.org/TR/xslt

NOTE

When writing a new netlist exporter, consider using Python or another tool rather than XSLT.

PADS netlist example using XSLT

The following example shows how to create an exporter for the PADS netlist format using `xlstproc`.

The PADS netlist format is comprised of two sections:

- A list of footprints
- A list of nets, together with the pads connected to each net.

Below is an XSL style-sheet which converts the intermediate netlist file to the PADS netlist format.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!--XSL style sheet to Eeschema Generic Netlist Format to PADS netlist format
    Copyright (C) 2010, SoftPLC Corporation.
    GPL v2.

    如何使用:
        https://lists.launchpad.net/kicad-developers/msg05157.html
-->

<!DOCTYPE xsl:stylesheet [
  <!ENTITY nl  "&#xd;&#xa;"> <!--new line CR, LF -->
]>

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"> <xsl:output
method="text" omit-xml-declaration="yes" indent="no"/>

<xsl:template match="/export">
    <xsl:text>*PADS-PCB*&nl;*PART*&nl;</xsl:text>
    <xsl:apply-templates select="components/comp"/>
    <xsl:text>&nl;*NET*&nl;</xsl:text>
    <xsl:apply-templates select="nets/net"/>
    <xsl:text>*END*&nl;</xsl:text>
</xsl:template>

<!-- for each component -->
<xsl:template match="comp">
    <xsl:text> </xsl:text>
    <xsl:value-of select="@ref"/>
    <xsl:text> </xsl:text>
    <xsl:choose>
        <xsl:when test = "footprint != '' ">
            <xsl:apply-templates select="footprint"/>
        </xsl:when>
        <xsl:otherwise>
            <xsl:text>unknown</xsl:text>
        </xsl:otherwise>
    </xsl:choose>
    <xsl:text>&nl;</xsl:text>
</xsl:template>

<!-- for each net -->
<xsl:template match="net">
    <!-- nets are output only if there is more than one pin in net -->
    <xsl:if test="count(node)>1">
        <xsl:text>*SIGNAL* </xsl:text>
        <xsl:choose>
            <xsl:when test = "@name != '' ">
                <xsl:value-of select="@name"/>
            </xsl:when>
            <xsl:otherwise>
                <xsl:text>N-</xsl:text>
                <xsl:value-of select="@code"/>
            </xsl:otherwise>
        </xsl:choose>
        <xsl:text>&nl;</xsl:text>
        <xsl:apply-templates select="node"/>
    </xsl:if>
</xsl:template>

<!-- for each node -->
```

进行此转换的命令行是：

```
kicad\\bin\\xsltproc.exe -o test.net kicad\\bin\\plugins\\netlist_form_pads-pcb.xsl test.tmp
```

Cadstar netlist example using XSLT

The following example shows how to create an exporter for the Cadstar netlist format using `xlstproc`.

The Cadstar format is comprised of two sections:

- The footprint list
- The Nets list: grouping pads references by nets

Below is an XSL style-sheet which converts the intermediate netlist file to the Cadstar netlist format.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!--XSL style sheet to Eeschema Generic Netlist Format to CADSTAR netlist format
    Copyright (C) 2010, Jean-Pierre Charras.
    Copyright (C) 2010, SoftPLC Corporation.
    GPL v2. -->

<!DOCTYPE xsl:stylesheet [
  <!ENTITY nl  "&#xd;&#xa;"> <!--new line CR, LF -->
]>

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"> <xsl:output
method="text" omit-xml-declaration="yes" indent="no"/>

<!-- Netlist header -->
<xsl:template match="/export">
    <xsl:text>.HEA&nl;</xsl:text>
    <xsl:apply-templates select="design/date"/>  <!-- Generate line .TIM <time> -->
    <xsl:apply-templates select="design/tool"/>  <!-- Generate line .APP <eeschema version>
-->
    <xsl:apply-templates select="components/comp"/>  <!-- Generate list of components -->
    <xsl:text>&nl;&nl;</xsl:text>
    <xsl:apply-templates select="nets/net"/>          <!-- Generate list of nets and
connections -->
    <xsl:text>&nl;.END&nl;</xsl:text>
</xsl:template>

 <!-- Generate line .TIM 20/08/2010 10:45:33 -->
<xsl:template match="tool">
    <xsl:text>.APP "</xsl:text>
    <xsl:apply-templates/>
    <xsl:text>"&nl;</xsl:text>
</xsl:template>

 <!-- Generate line .APP "eeschema (2010-08-17 BZR 2450)-unstable" -->
<xsl:template match="date">
    <xsl:text>.TIM </xsl:text>
    <xsl:apply-templates/>
    <xsl:text>&nl;</xsl:text>
</xsl:template>

<!-- for each component -->
<xsl:template match="comp">
    <xsl:text>.ADD_COM </xsl:text>
    <xsl:value-of select="@ref"/>
    <xsl:text> </xsl:text>
    <xsl:choose>
        <xsl:when test = "value != '' ">
            <xsl:text>"</xsl:text> <xsl:apply-templates select="value"/> <xsl:text>"
</xsl:text>
        </xsl:when>
        <xsl:otherwise>
            <xsl:text>""</xsl:text>
        </xsl:otherwise>
    </xsl:choose>
    <xsl:text>&nl;</xsl:text>
</xsl:template>

<!-- for each net -->
<xsl:template match="net">
    <!-- nets are output only if there is more than one pin in net -->
```

TER C1.2

```
         U2.7
         P1.4
.ADD_TER R1.1 "VCC"
.TER     U1.14
         U2.4
         U2.1
         U2.14
         P1.1
.ADD_TER U1.2 "N-4"
.TER     U2.3
.ADD_TER P1.2 "/SIG_OUT"
.TER     U2.5
         U2.2
.ADD_TER R1.2 "/CLOCK_IN"
.TER     C1.1
         U1.1
         P1.3
```

END

~~~
===== OrcadPCB2 netlist example using XSLT

This format has only one section which is the footprint list. Each footprint includes a
list of its pads with reference to a net.

Below is an XSL style-sheet which converts the intermediate netlist file to the Orcad
netlist format.

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<!--XSL style sheet to Eeschema Generic Netlist Format to CADSTAR netlist format
    Copyright (C) 2010, SoftPLC Corporation.
    GPL v2.

    如何使用:
        https://lists.launchpad.net/kicad-developers/msg05157.html
-->

<!DOCTYPE xsl:stylesheet [
  <!ENTITY nl  "&#xd;&#xa;"> <!--new line CR, LF -->
]>

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"> <xsl:output
method="text" omit-xml-declaration="yes" indent="no"/>

<!--
    Netlist header
    Creates the entire netlist
    (can be seen as equivalent to main function in C
-->
<xsl:template match="/export">
    <xsl:text>( { Eeschema Netlist Version 1.1  </xsl:text>
    <!-- Generate line .TIM <time> -->
<xsl:apply-templates select="design/date"/>
<!-- Generate line eeschema version ... -->
<xsl:apply-templates select="design/tool"/>
<xsl:text>}&nl;</xsl:text>

<!-- Generate the list of components --> <xsl:apply-templates select="components/comp"/>
<!-- Generate list of components -->

<!-- end of file --> <xsl:text>)&nl;*&nl;</xsl:text> </xsl:template>

<!--
    Generate id in header like "eeschema (2010-08-17 BZR 2450)-unstable"
-->
<xsl:template match="tool">
    <xsl:apply-templates/>
</xsl:template>

<!--
    Generate date in header like "20/08/2010 10:45:33"
-->
<xsl:template match="date">
    <xsl:apply-templates/>
    <xsl:text>&nl;</xsl:text>
</xsl:template>

<!--
~~~

```
( { Eeschema Netlist Version 1.1 29/08/2010 21:07:51 eeschema (2010-08-28 BZR 2458)-unstable} ( 4C6E2141
$noname P1 CONN_4 ( 1 VCC ) ( 2 /SIG_OUT ) ( 3 /CLOCK_IN ) ( 4 GND ) ) ( 4C6E20BA $noname U2 74LS74 ( 1
VCC ) ( 2 /SIG_OUT ) ( 3 N-04 ) ( 4 VCC ) ( 5 /SIG_OUT ) ( 6 ? ) ( 7 GND ) ( 14 VCC ) ) ( 4C6E20A6 $noname U1
74LS04 ( 1 /CLOCK_IN ) ( 2 N-04 ) ( 7 GND ) ( 14 VCC ) ) ( 4C6E2094 $noname C1 CP ( 1 /CLOCK_IN ) ( 2 GND ) ) (
4C6E208A $noname R1 R ( 1 VCC ) ( 2 /CLOCK_IN ) ) ) *
```

```
:experimental:

[[eeschema-actions-reference]]
== Actions reference
Below is a list of every available *action* in the KiCad Schematic Editor: a command that
can be assigned to a hotkey.

////
Note to translators: you do not need to translate this table by hand.

It is generated from KiCad using the Dump Hotkeys button that is shown in the hotkeys
editor if you
add the line `HotkeysDumper=1` to your advanced config file (`kicad_advanced` file in the
config
directory)
////

=== Schematic Editor

// NOTE: this text between the section header and the table is *required* or
// asciidoctor-web-pdf will not insert page breaks in the table correctly and
// the PDF will be truncated.
The actions below are available in the Schematic Editor. Hotkeys can be assigned to any of
these actions in the **Hotkeys** section of the preferences.

[width="100%", options="header", cols="20%,15%,65%"]
|===
| Action | Default Hotkey | Description
| Align Elements to Grid
  |
  |
| Annotate Schematic...
  |
  | Fill in schematic symbol reference designators
| Assign Footprints...
  |
  | Run footprint assignment tool
| Clear Net Highlighting
  | kbd:[~]
  | Clear any existing net highlighting
| Export Drawing to Clipboard
  |
  | Export drawing of current sheet to clipboard
| Edit Library Symbol...
  | kbd:[Ctrl+Shift+E]
  | Open the library symbol in the Symbol Editor
| Edit Sheet Page Number...
  |
  | Edit the page number of the current or selected sheet
| Edit Symbol Fields...
  |
  | Bulk-edit fields of all symbols in schematic
| Edit Symbol Library Links...
  |
  | Edit links between schematic and library symbols
| Edit with Symbol Editor
  | kbd:[Ctrl+E]
  | Open the selected symbol in the Symbol Editor
| Highlight on PCB
  |
```