

Schematic Editor

The KiCad Team

Table of Contents

Introduction to the KiCad Schematic Editor	2
Descripción	2
Resumen técnico	2
Generic Schematic Editor commands	3
Comandos del Raton	4
Teclas rápidas	5
Grid	9
Selección del Zoom	9
Mostrando las coordenadas del cursor	9
Barra de menú superior	10
Barra de herramientas superior	10
Iconos de la barra de herramientas derecha.	11
Iconos de la barra de herramientas izquierda	12
Menús emergentes y de edición rápida	13
Menú principal superior	14
Menú archivo	14
Menú Preferencias	16
Menú ayuda	22
Barra de Herramientas de uso general	23
Gestión de hoja	23
Herramienta de Búsqueda	23
Herramienta de Netlist	24
Herramienta de Anotado	25
Herramienta de Comprobación de Reglas Eléctricas	27
Herramienta Lista de Materiales	29
Edit Fields tool	32
Import tool for footprint assignment	34
Manage Symbol Libraries	35
Symbol Library Table	35
Creación y Edición de Esquemas	40
Introducción	40
Consideraciones generales	40
Symbol placement and editing	40
Electrical Connections	44
Complementos de Dibujo	52
Rescuing cached symbols	54
Esquemas Jerarquicos	56
Introducción	56
Navegando en la Jerarquía	56
Etiquetas locales, jerárquicas y globales	57
Summary of hierarchy creation	57
Símbolo de la hoja	57

Conexion - Pines de Jerarquia	58
Conexiones - Etiquetas de Jerarquía	58
Jerarquías Complejas	60
Jerarquías Planas	61
Symbol Annotation Tool	64
Introducción	64
Algunos ejemplos	65
Verificación del diseño con el Comprobador de Reglas Eléctricas.	68
Introducción	68
Como usar el ERC	68
Ejemplo de ERC	69
Mostrando diagnósticos	69
Pines de potencia e indicadores de potencia	70
Configuración	71
Archivo de reporte ERC	72
Creación del Netlist	73
Resumen	73
Formatos de Netlist	73
Ejemplos de Netlist	74
Notas en los Netlists	76
Otros Formatos	78
Plot e Imprimir	80
Introducción	80
Comandos comunes de impresión	80
Imprimir en Postscript	80
Imprimir en PDF	81
Imprimir en SVG	82
Imprimir en DXF	82
Imprimir en HPGL	82
Imprimir en papel	84
Symbol Editor	85
General Information About Symbol Libraries	85
Symbol Library Overview	85
Symbol Library Editor Overview	85
Selección y mantenimiento de bibliotecas	89
Creating Library Symbols	89
Elementos gráficos	96
Multiple Units per Symbol and Alternate Body Styles	98
Edición y Creación de Pines	101
Symbol Fields	108
Power Ports	51
Symbol Library Browser	114
Introducción	114
Viewlib - pantalla principal	115

Symbol Library Browser Top Toolbar	115
Creando Netlists y Ficheros BOM Personalizados	117
Ficheros Netlist Intermedios	117
Conversión a un nuevo formato de netlist	119
Introduccion de XSLT	119
Formato de Comandos: ejemplos para scripts de python	128
Estructura del Netlist Intermedio	128
Mas sobre xsltproc	133
Simulator	137
Assigning models	137
Spice directives	142
Simulation	142

Manual de referencia

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.

Copyright

This document is Copyright © 2010-2022 by its contributors as listed below. You may distribute it and/or modify it under the terms of either the GNU General Public License (<http://www.gnu.org/licenses/gpl.html>), version 3 or later, or the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0/>), version 3.0 or later.

Todas las marcas mencionadas en esta guía pertenecen a sus legítimos propietarios.

Contribuidores

Jean-Pierre Charras, Fabrizio Tappero, Graham Keeth

Traducción

Antonio Morales <antonio1010.mr@gmail.com>, 2015

Realimentación

Por favor dirija cualquier reporte de fallo, sugerencia o nuevas versiones a:

- About KiCad documentation: <https://gitlab.com/kicad/services/kicad-doc/issues>
- Acerca del software KiCad: <https://gitlab.com/kicad/code/kicad/issues>

Introduction to the KiCad Schematic Editor

Descripción

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

- Linux
- Apple macOS
- 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:

- Comprobador de reglas eléctricas (ERC) para el control automático de conexiones incorrectas y/o perdidas.
- Exportación de ficheros de plano en múltiples formatos (Postscript, PDF, HPGL y SVG)
- Bill of Materials generation (via Python or XSLT scripts, which allow many flexible formats).

Resumen técnico

The Schematic Editor is limited only by the available memory. There is thus no real limitation to the number of components, component pins, connections or sheets. In the case of multi-sheet schematics, the representation is hierarchical.

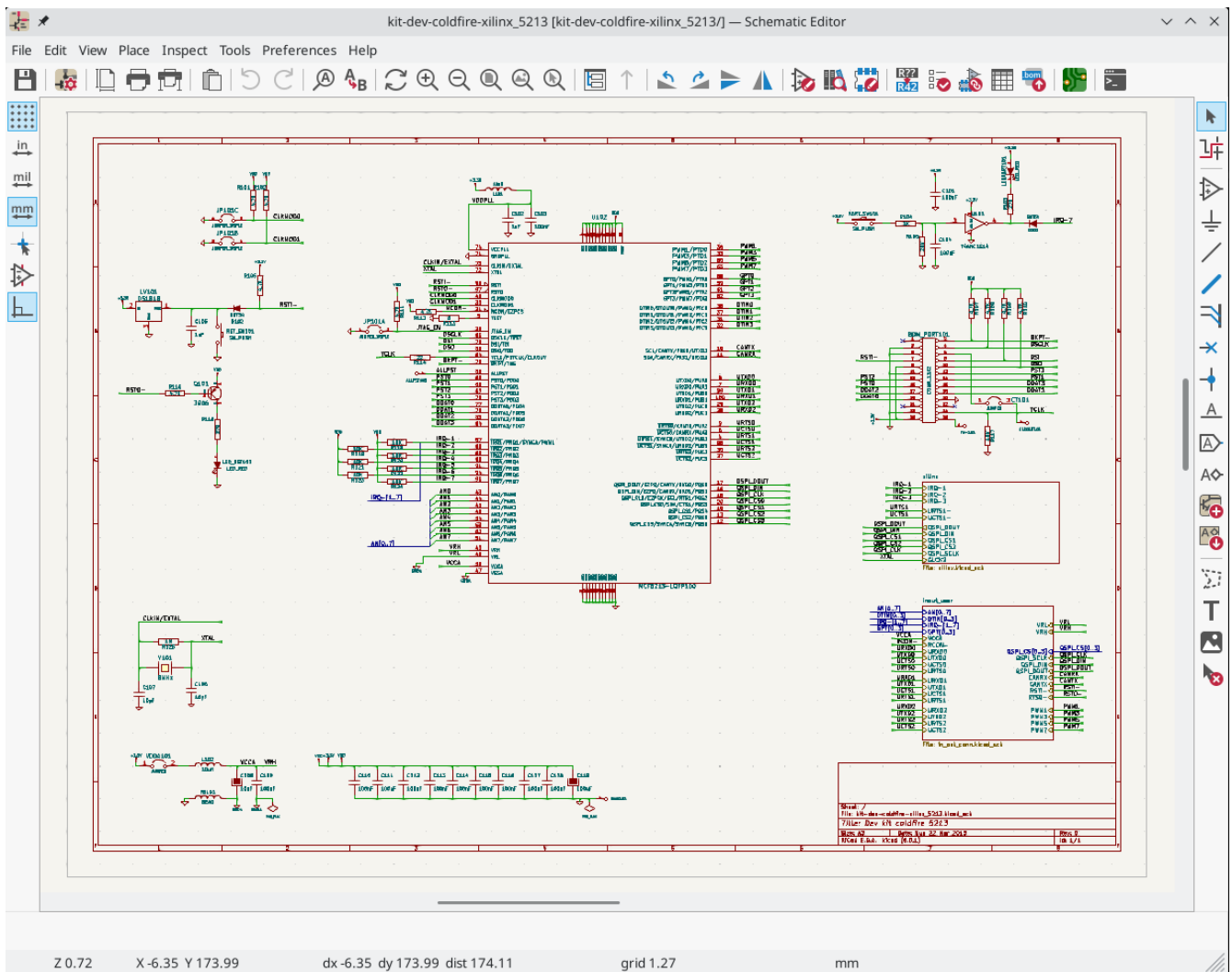
The Schematic Editor can use multi-sheet schematics in a few ways:

- Jerarquías simples (cada esquemas es usado solo una vez)
- Jerarquías complejas (algunos esquemas son usados mas de una vez con múltiples instancias)
- Jerarquías planas (los esquemas no están explícitamente conectados en el esquema principal)

Generic Schematic Editor commands

Commands can be executed by:

- Haciendo clic en la barra de menú (parte superior de la pantalla)
- Haciendo clic sobre los iconos de la parte superior de la pantalla (comandos generales)
- Haciendo clic sobre los iconos de la parte derecha de la pantalla (comandos particulares o "herramientas")
- Haciendo clic sobre los iconos de la parte izquierda de la pantalla (opciones de visualización)
- Presionando los botones del ratón (importantes comandos complementarios). En particular un clic derecho del ratón abre el menú contextual del elemento bajo el cursor (Zoom, rejilla y edición de elementos)
- Function keys (`F1`, `F2`, `F3`, `F4`, `Insert` and `Space`). Specifically: `Escape` cancels the command in progress. `Insert` allows the duplication of the last element created.
- Pressing hotkeys. For a list of hotkeys, see the **Help** → **List Hotkeys** menu entry or press `Ctrl` + `F1`. Many hotkeys select a tool but do not perform the tool's action until the canvas is clicked. This behavior can be changed by unchecking **First hotkey selects tool** in the **Common** Preferences pane. With this option unchecked, pressing a hotkey will select the tool and immediately perform the tool's action at the current cursor location.



Comandos del Ratón

Comandos Básicos

Botón Izquierdo

- Single click: Selects the item under the cursor and displays the item's characteristics in the status bar.
- Double click: edits the item if it is editable.

Botón derecho

- Opens a pop-up menu. If an item is selected, the items in the menu are related to the selected item. If an item is under the cursor when the right mouse button is clicked, the item is selected.

Selection operations

Schematic editor items can be selected by clicking on them. Multiple items can be selected at once. Add items to the selection with **Shift** + click, and remove items from the selection with **Ctrl** + **Shift** + click.

NOTE | On macOS, **Cmd** is used instead of **Ctrl**.

left mouse button	Select item.
Shift + left mouse button	Add item to selection.
Ctrl + Shift + left mouse button	Remove item from selection.
Ctrl + left mouse button	Highlight net.

Items can also be selected by drawing a box around them using the left mouse button.

Dragging from left to right includes all items fully enclosed by the box. Dragging from right to left includes all items touched by the box, even if they are not fully enclosed.

The **Shift** and **Ctrl** + **Shift** modifiers also work with drag selections to add and remove items from the selection, respectively.

Teclas rápidas






















- The **Ctrl** + **F1** displays the current hotkey list.
- All hotkeys can be redefined using the hotkey editor (**Preferences** → **Preferences...** → **Hotkeys**).

The default hotkey list is below. Many additional actions do not have hotkeys by default, but hotkeys can be assigned to them with the hotkey editor.

Action	Default Hotkey	Description
Click	Return	Performs left mouse button click
Double-click	End	Performs left mouse button double-click
Cursor Down	Down	
Cursor Down Fast	Ctrl + Down	
Cursor Left	Left	
Cursor Left Fast	Ctrl + Left	
Cursor Right	Right	
Cursor Right Fast	Ctrl + Right	
Cursor Up	Up	
Cursor Up Fast	Ctrl + Up	
Switch to Fast Grid 1	Alt + 1	
Switch to Fast Grid 2	Alt + 2	

Action	Default Hotkey	Description
Switch to Next Grid		
Switch to Previous Grid	+	
Reset Grid Origin		
Grid Origin		Set the grid origin point
New...	+	Create a new document in the editor
Open...	+	Open existing document
Pan Down	+	
Pan Left	+	
Pan Right	+	
Pan Up	+	
Print...	+	Print
Reset Local Coordinates		
Save	+	Save changes
Save As...	+ +	Save current document to another location
Always Show Cursor	+ +	Display crosshairs even in selection tool
Switch units	+	Switch between imperial and metric units
Update PCB from Schematic...		Update PCB with changes made to schematic
Center		Center
Zoom to Objects	+	Zoom to Objects
Zoom to Fit		Zoom to Fit
Zoom In at Cursor		Zoom In at Cursor
Zoom Out at Cursor		Zoom Out at Cursor
Refresh		Refresh

Action	Default Hotkey	Description
Delete		Deletes selected item(s)
Duplicate		Duplicates the selected item(s)
Find		Find text
Find and Replace		Find and replace text
Find Next		Find next match
Find Next Marker		
Paste		Paste item(s) from clipboard
Redo		Redo last edit
Select All		Select all items on screen
Undo		Undo last edit
List Hotkeys...		Displays current hotkeys table and corresponding commands
Preferences...		Show preferences for all open tools
Clear Net Highlighting		Clear any existing net highlighting
Edit Library Symbol...		Open the library symbol in the Symbol Editor
Edit with Symbol Editor		Open the selected symbol in the Symbol Editor
Highlight Net		Highlight net under cursor
Show Datasheet		Opens the datasheet in a browser
Add Sheet		Add a hierarchical sheet
Add Wire to Bus Entry		Add a wire entry to a bus
Add Global Label		Add a global label
Add Hierarchical Label		Add a hierarchical label
Add Junction		Add a junction
Add Label		Add a net label

Action	Default Hotkey	Description
Add Text		Add text
Add Symbol		Add a symbol
Add Bus		Add a bus
Add Lines		Add connected graphic lines
Add Wire		Add a wire
Finish Wire or Bus		Complete drawing at current segment
Unfold from Bus		Break a wire out of a bus
Autoplace Fields		Runs the automatic placement algorithm on the symbol or sheet's fields
Edit Footprint...		Displays footprint field dialog
Edit Reference Designator...		Displays reference designator dialog
Edit Value...		Displays value field dialog
Mirror Horizontally		Flips selected item(s) from left to right
Mirror Vertically		Flips selected item(s) from top to bottom
Properties...		Displays item properties dialog
Repeat Last Item		Duplicates the last drawn item
Rotate Counterclockwise		Rotates selected item(s) counter-clockwise
Drag		Drags the selected item(s)
Move		Moves the selected item(s)
Select Connection		Select a complete connection
Select Node		Select a connection item under the cursor
Leave Sheet		Display the parent sheet in the schematic editor

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`

It is possible to import hotkey settings from a `user.hotkeys` file using menu **Preferences** → **Preferences...** → **Hotkeys** → **Import Hotkeys...**

Grid

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 preferred grid to place symbols and wires in a schematic, and to place pins when designing a symbol in the Symbol Editor.

One can also work with a smaller grid from 25 mil to 10 mil. This is only intended for designing the symbol body or placing text and comments and not recommended for placing pins and wires.

Selección del Zoom

Para cambiar el nivel de zoom:

- El botón derecho abre el menú contextual y permite seleccionar el zoom deseado.
- Or use hotkeys:
 - **F1**: Zoom in
 - **F2**: Zoom out
 - **F4**: Center the view around the cursor pointer position
 - **Home**: Zoom and center the view to fit the entire schematic sheet
 - **Ctrl** + **Home**: Zoom and center the view to fit all of the objects in the schematic
 - **Ctrl** + **F5**: Activate the Zoom to Selection tool
- Zoom sobre la ventana:
 - Mouse wheel: Zoom in/out
 - Shift+Mouse wheel: Pan up/down
 - Ctrl+Mouse wheel: Pan left/right

Mouse scroll gestures are configurable in the **Mouse and Touchpad** page of the **Preferences** dialog.

Mostrando las coordenadas del cursor

The display units are in inches, mils, or millimeters.

La siguiente información se muestra en la parte inferior derecha de la pantalla.

- El factor de zoom

La posición absoluta del cursor

- La posición relativa del cursor
- The grid size
- The active unit system
- The active tool

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

Z 2.59 X 447.04 Y 212.09 dx 447.04 dy 212.09 dist 494.80 grid 1.27 mm Add Wire

Barra de menú superior

The top menu bar allows the opening and saving of schematics, program configuration and viewing the documentation.

File Edit View Place Inspect Tools Preferences Help












Barra de herramientas superior
























This toolbar gives access to the main functions of the Schematic Editor.

If the Schematic Editor is run in standalone mode, this is the available tool set:



Note that when KiCad runs in project mode, the first two icons are not available as they work with individual files.

	Create a new schematic (only in standalone mode).
	Open a schematic (only in standalone mode).
	Save complete schematic project.
	Set the schematic-specific options.
	Select the sheet size and edit the title block.
	Open print dialog.
	Open plot dialog.
	Paste a copied/cut item or block to the current sheet.
	Undo: Revert the last change.
	Redo: Revert the last undo operation.
	Show the dialog to search symbols and texts in the schematic.



















	Show the dialog to search and replace texts in the schematic.
	Refresh screen.
	Zoom in.
	Zoom out.
	Zoom to fit the entire schematic sheet.
	Zoom to fit all objects in the schematic.
	Zoom to fit selected items.
	View and navigate the hierarchy tree.
	Leave the current sheet and go up in the hierarchy.
	Rotate selected items counter-clockwise.
	Rotate selected items clockwise.
	Mirror selected items vertically.
	Mirror selected items horizontally.
	Call the symbol library editor to view and modify libraries and symbols.
	Browse symbol libraries.
	Open the footprint library editor to view and modify libraries and footprints.
	Annotate symbols.
	Electrical Rules Checker (ERC), automatically validate electrical connections.
	Open the footprint assignment tool to assign footprints to symbols.
	Bulk edit symbol fields in a spreadsheet interface.
	Generate the Bill of Materials (BOM).
	Open the PCB editor.
	Open the Python scripting console.

Iconos de la barra de herramientas derecha.

Esta barra de herramientas contiene herramientas para:








- Place symbols, wires, buses, junctions, labels, text, etc.

- Create hierarchical subsheets and connection symbols.

	Cancel the active command or tool and go into selection mode.
	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.
	Display the symbol selector dialog to select a new symbol to be placed.
	Display the power symbol selector dialog to select a power symbol to be placed.
	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. It is done to notify the Electrical Rules Checker that lack of connection for a particular pin is intentional and should not be reported.
	Place a junction. This connects two crossing wires or a wire and a pin, when it can be ambiguous (i.e. if a wire end or a pin is not directly connected to another wire end).
	Place a local label. Local label connects items located in the same sheet . For connections between two different sheets, you have to use global or hierarchical labels.
	Place a global label. All global labels with the same name are connected, even when located on different sheets.
	Place a hierarchical label. Hierarchical labels are used to create a connection between a subsheet and the parent sheet that contains it.
	Place a hierarchical subsheet. You must specify the file name for this subsheet.
	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 a line. These are only graphical and do not connect anything.
	Place a text comment.
	Place a bitmap image.
	Delete clicked items.

Iconos de la barra de herramientas izquierda

Esta barra de herramientas gestiona las opciones de visualización:

	Toggle grid visibility.
	Switch units to inches.
	Switch units to mils (0.001 inches).
	Switch units to millimeters.
	Choose the cursor shape (full screen/small).
	Toggle visibility of "invisible" pins.
	Toggle free angle/90 degrees wires and buses placement.

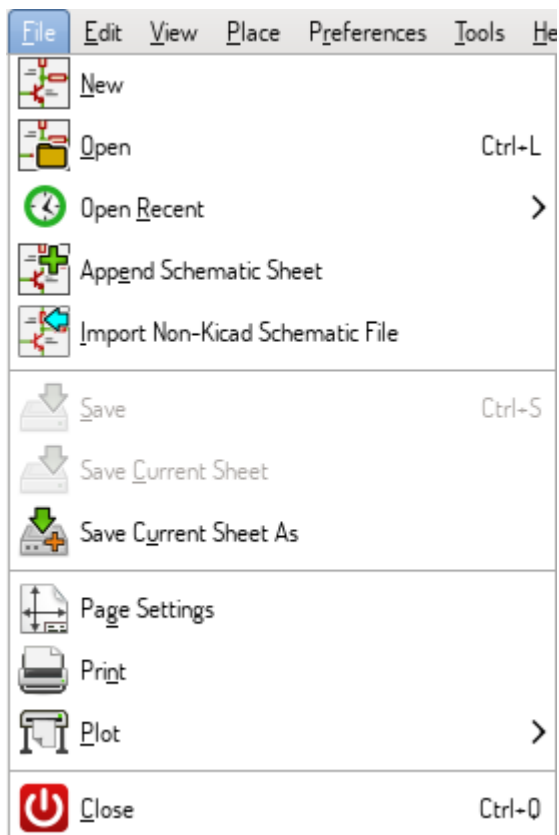
Menús emergentes y de edición rápida

Un clic con el botón derecho abre un menú contextual para el elemento seleccionado. Este contiene:

- Factor de zoom.
- Ajuste de la rejilla.
- Copy/Paste/Delete commands.
- Add Wire/Bus.
- Parámetros comúnmente editados del elemento seleccionado.

Menú principal superior

Menú archivo

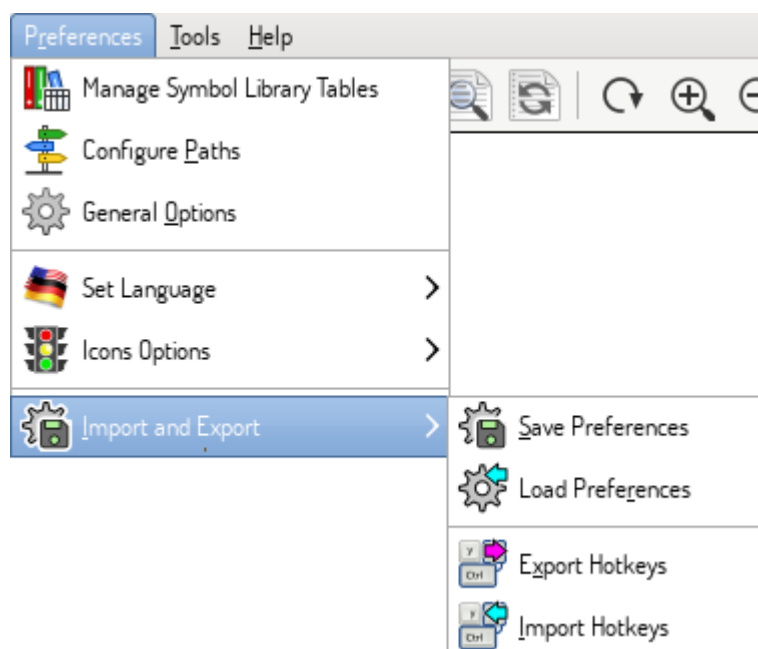


New	Close current schematic and start a new one (only in standalone mode).
Open	Load a schematic project (only in standalone mode).
Open Recent	Open a schematic project from the list of recently opened files (only in standalone mode).
Save	Save current sheet and all its subsheets.
Save As...	Save the current sheet under a new name (only in standalone mode).
Save Current Sheet Copy As...	Save a copy of the current sheet under a new name (only in project mode).
Insert Schematic Sheet Content...	Insert the contents of another schematic sheet into the current sheet (only in standalone mode).
Import	Import a non-KiCad schematic or a footprint assignment file.
Export	Export a netlist or a drawing of the schematic to the clipboard.
Schematic Setup...	Set up schematic formatting, electrical rules, net classes, and text variables.
Page Settings...	Configure page dimensions and title block.
Print	Print schematic project (See also chapter Plot and Print).
Plot	Export to PDF, PostScript, HPGL or SVG format (See chapter Plot and Print).
Quit	Terminate the application.

Schematic Setup

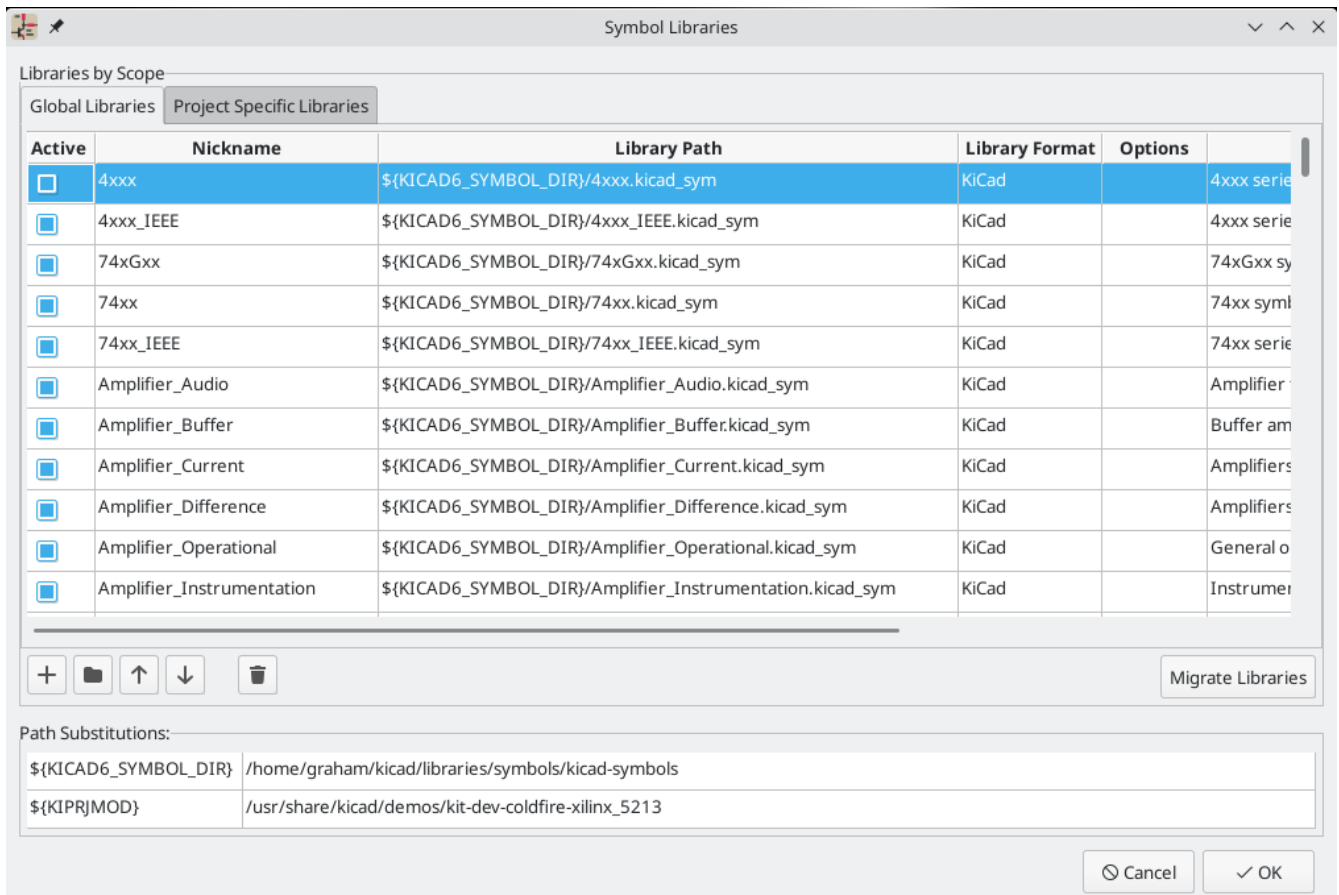
The Schematic Setup window is used to set schematic options that are specific to the currently active schematic. For example, the Schematic Setup window contains formatting options, electrical rule configuration, netclass setup, and schematic text variable setup.

Menú Preferencias



Configure Paths...	Set the default search paths.
Manage Symbol Library Tables...	Add/remove symbol libraries.
Preferences...	Preferences (units, grid size, field names, etc.).
Set Language	Select interface language.

Manage Symbol Library Tables



KiCad uses two library tables to store the list of available symbol libraries, which differ by the scope:

Global Libraries

Libraries listed in the Global Library table are available to every project. They are saved in the `sym-lib-table` in the KiCad configuration directory, which is system-dependent:


- Windows: `%APPDATA%\kicad\6.0\sym-lib-table`
- Linux: `~/.config/kicad/6.0/sym-lib-table`
- macOS: `~/Library/Preferences/kicad/6.0/sym-lib-table`

Project Specific Libraries

Libraries listed in Project Specific Libraries table are available to the currently opened project. They are saved in a `sym-lib-table` file in the project directory.

Both library tables are visible by clicking on **Global Libraries** or **Project Specific Libraries** tab in the Manage Library Tables window.

Add a new library

Add a library either by clicking the  button and selecting a file or clicking the **+** button and typing a path to a library file. The selected library will be added to the currently opened library table (Global/Project Specific).

Remove a library

Remove a library by selecting one or more libraries and clicking the  button.

Library properties

Each row in the table stores several fields describing a library:

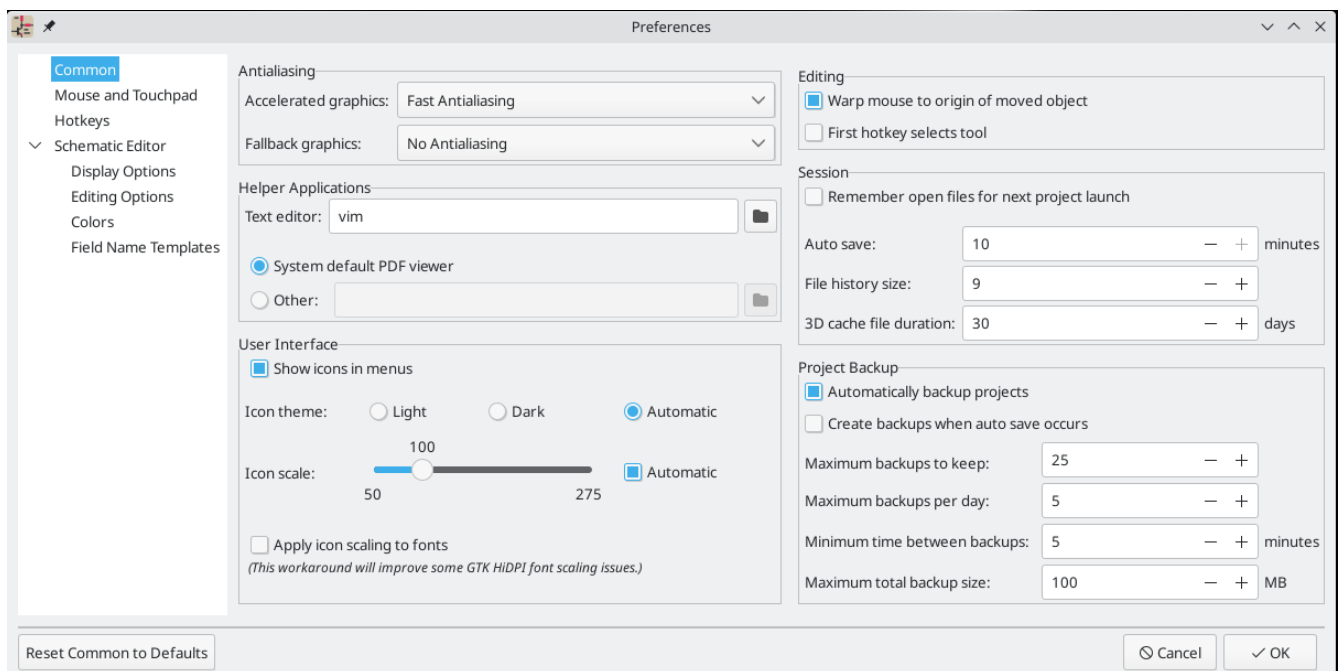
Active	Enables/disables the library. It is useful to temporarily reduce the loaded library set.
Nickname	Nickname is a short, unique identifier used for assigning symbols to components. Symbols are represented by '<Library Nickname>:<Symbol Name>' strings.
Library Path	Path points to the library location.
Plugin Type	Determines the library file format. KiCad 6.0 libraries use the "KiCad" format, while KiCad 5.x libraries use the "Legacy" format. Legacy libraries are read-only.
Options	Stores library specific options, if used by plugin.
Description	Briefly characterizes the library contents.

Preferences

Common Preferences

NOTE

TODO: write this section

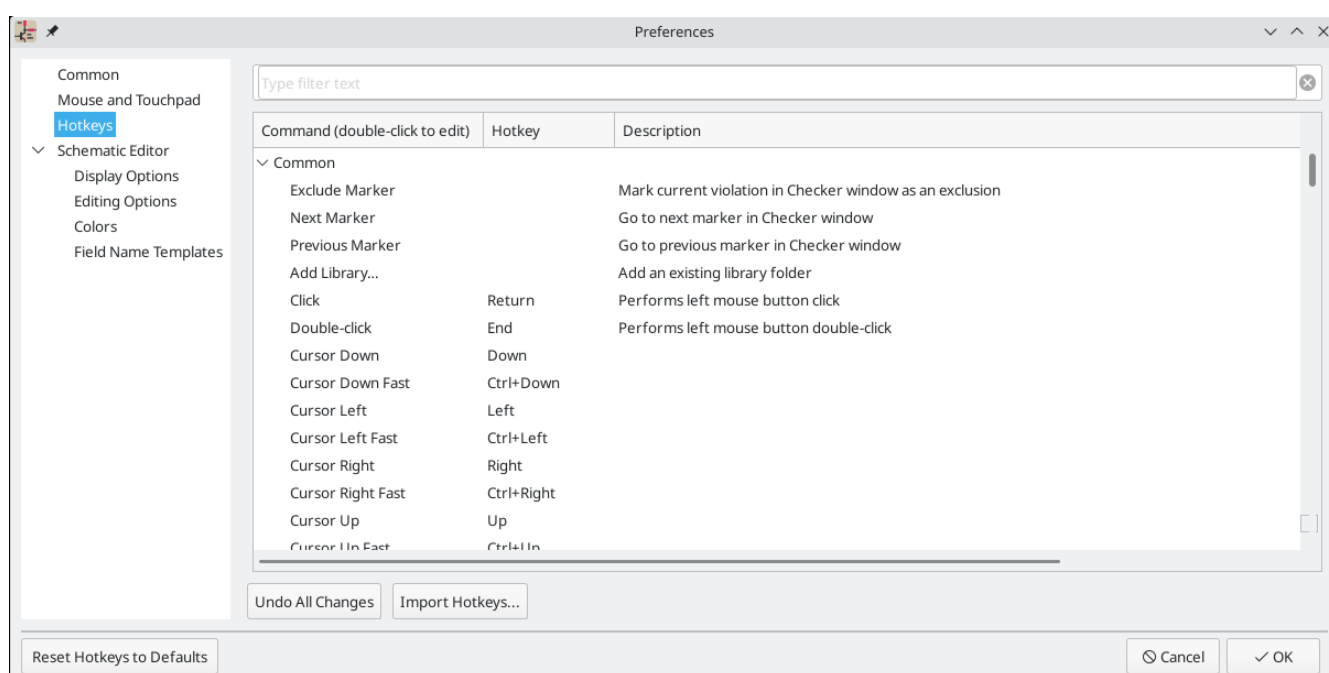


Mouse and Touchpad

Center and warp cursor on zoom	If checked, the pointed location is warped to the screen center when zooming in/out.
Use touchpad to pan	When enabled, view is panned using scroll wheels (or touchpad gestures) and to zoom one needs to hold Ctrl . Otherwise scroll wheels zoom in/out and Ctrl / Shift are the panning modifiers.
Pan while moving object	If checked, automatically pans the window if the cursor leaves the window during drawing or moving.

Teclas rápidas

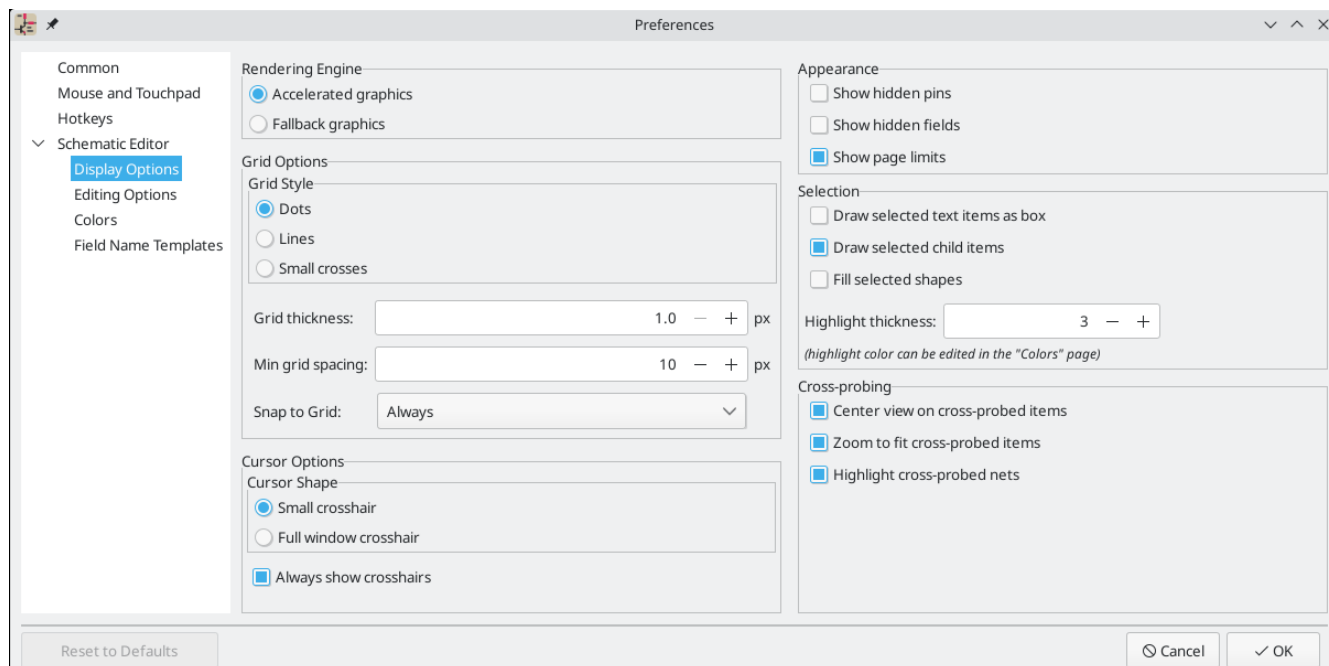
Redefine hotkeys.



Select a new hotkey by double clicking an action or right click on an action to show a popup menu:

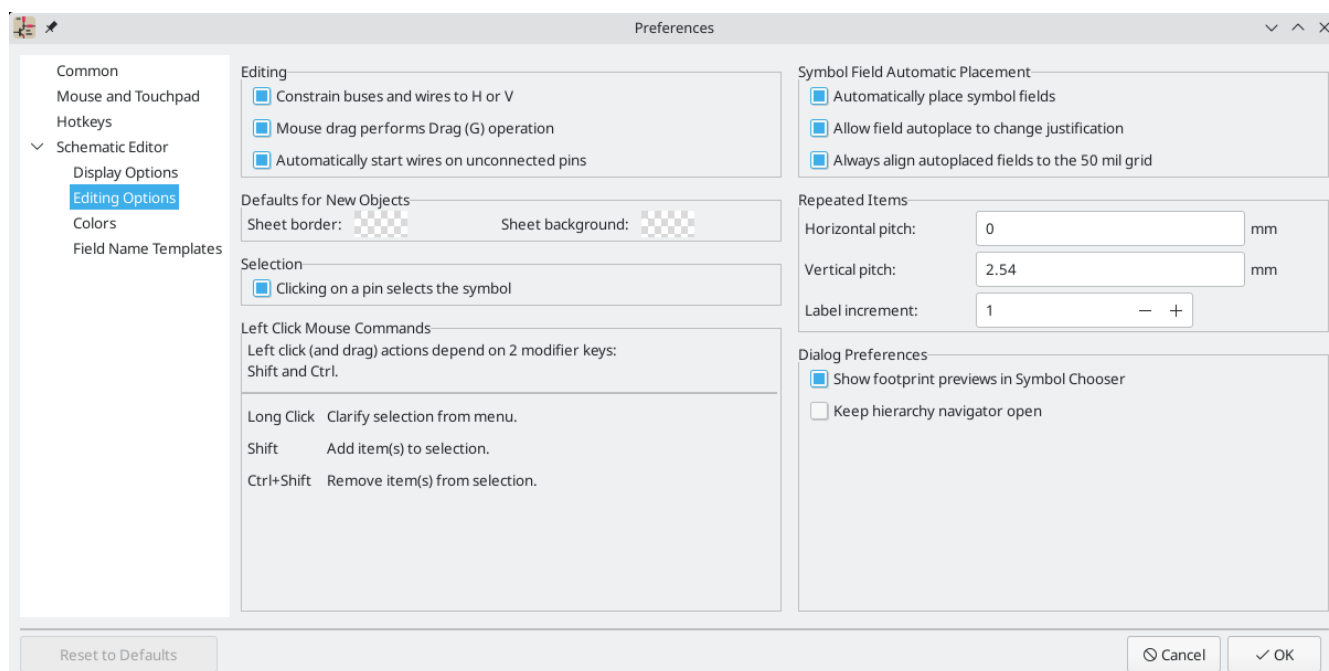
Edit	Define a new hotkey for the action (same as double click).
Undo Changes	Reverts the recent hotkey changes for the action.
Clear Assigned Hotkey	
Restore Default	Sets the action hotkey to its default value.

Display Options



Grid Size	<p>Grid size selection.</p> <p>It is recommended to work with normal grid (0.050 inches or 1,27 mm). Smaller grids are used for component building.</p>
Bus thickness	Pen size used to draw buses.
Line thickness	Pen size used to draw objects that do not have a specified pen size.
Part ID notation	Style of suffix that is used to denote symbol units (U1A, U1.A, U1-1, etc.)
Icon scale	Adjust toolbar icons size.
Show Grid	Grid visibility setting.
Restrict buses and wires to H and V orientation	If checked, buses and wires are drawn only with vertical or horizontal lines. Otherwise buses and wires can be placed at any orientation.
Show hidden pins:	Display invisible (or <i>hidden</i>) pins, typically power pins.
Show page limits	If checked, shows the page boundaries on screen.
Footprint previews in symbol chooser	<p>Displays a footprint preview frame and footprint selector when placing a new symbol.</p> <p>Note: it may cause problems or delays, use at your own risk.</p>

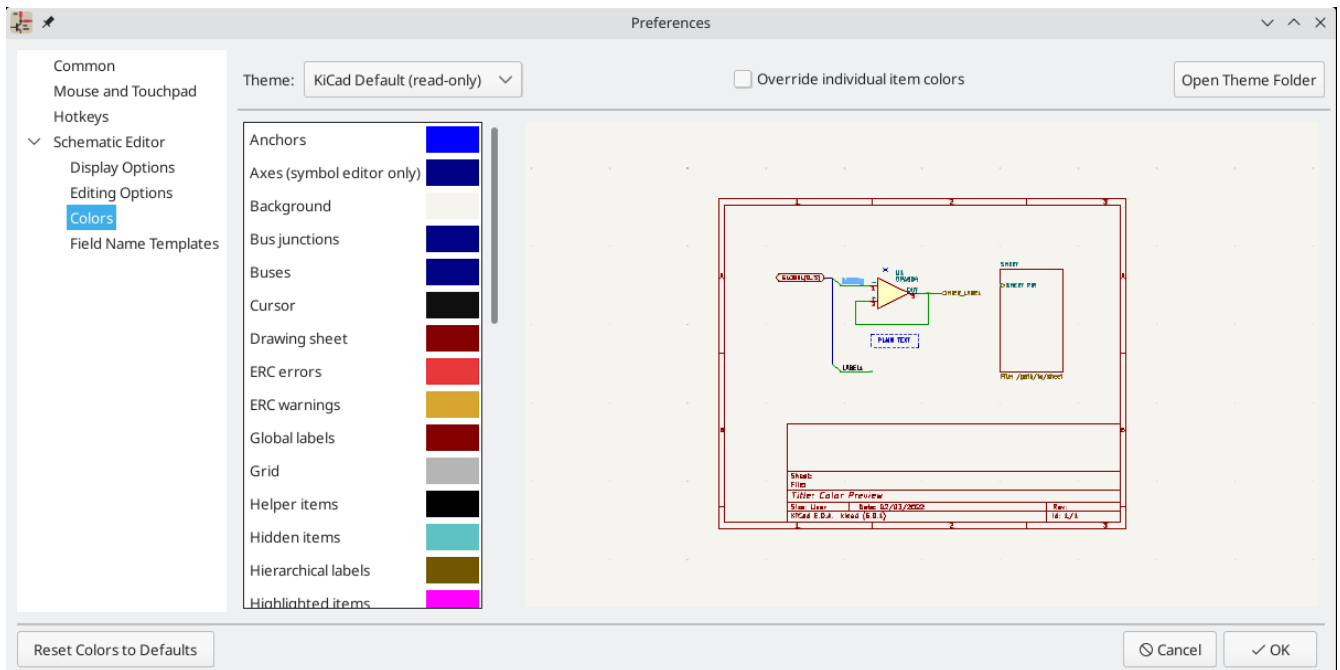
Editing Options



Measurement units	Select the display and the cursor coordinate units (inches or millimeters).
Horizontal pitch of repeated items	Increment on X axis during element duplication (default: 0) (after placing an item like a symbol, label or wire, a duplication is made by the Insert key)
Vertical pitch of repeated items	Increment on Y axis during element duplication (default: 0.100 inches or 2,54 mm).
Increment of repeated labels	Increment of label value during duplication of texts ending in a number, such as bus members (usual value 1 or -1).
Default text size	Text size used when creating new text items or labels.
Auto-save time interval	Time in minutes between saving backups.
Automatically place symbol fields	If checked, symbol fields (e.g. value and reference) in newly placed symbols might be moved to avoid collisions with other items.
Allow field autoplace to change justification	Extension of 'Automatically place symbol fields' option. Enable text justification adjustment for symbol fields when placing a new part.
Always align autoplaced fields to the 50 mil grid	Extension of 'Automatically place symbol fields' option. If checked, fields are autoplaced using 50 mils grid, otherwise they are placed freely.

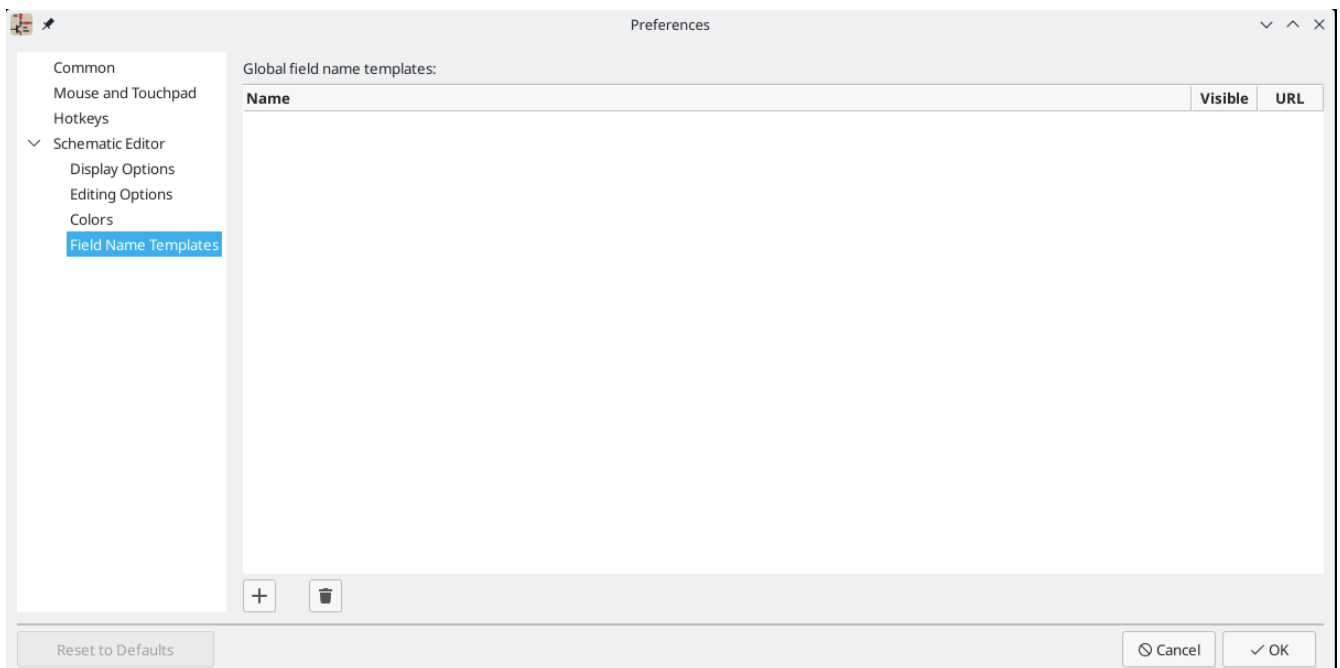
Colors

Color scheme for various graphic elements. Click on any of the color swatches to select a new color for a particular element.



Default Fields

Define additional custom fields and corresponding values that will appear in newly placed symbols.



Menú ayuda

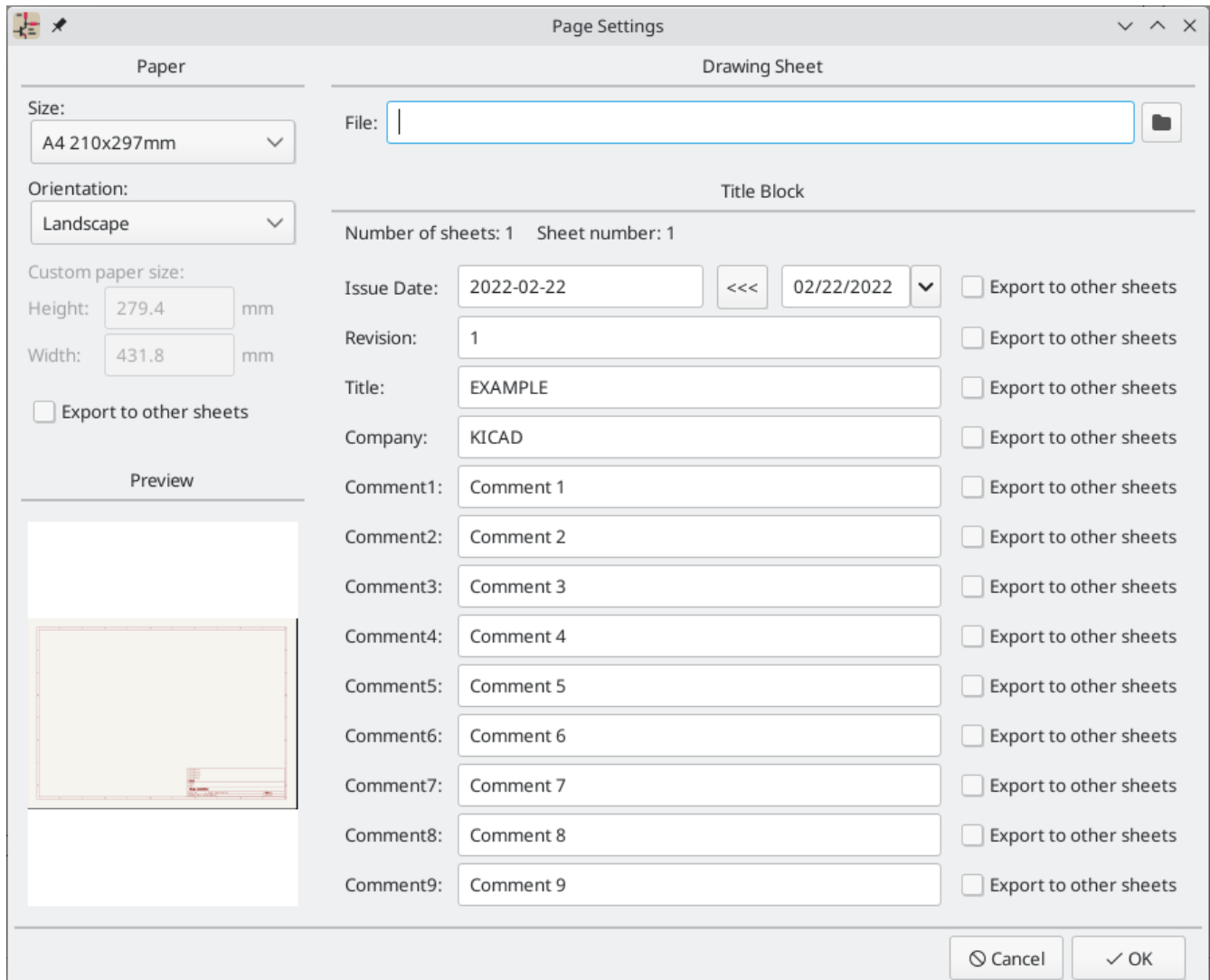
Access to on-line help (this document) for an extensive tutorial about KiCad.

Use the **Report a Bug** item to report a bug online. Full KiCad version and user system information is available via the **Copy Version Info** button in the **About KiCad** window.

Barra de Herramientas de uso general

Gestión de hoja

The Sheet Settings icon () allows you to define the sheet size and the contents of the title block.

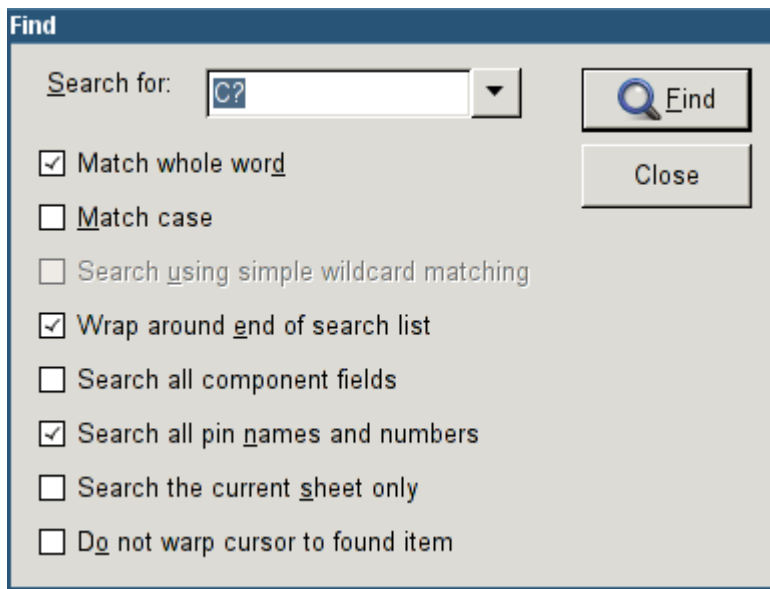


The screenshot shows the 'Page Settings' dialog box with two main tabs: 'Paper' and 'Drawing Sheet'. The 'Paper' tab is active, showing options for 'Size' (A4 210x297mm), 'Orientation' (Landscape), and 'Custom paper size' (Height: 279.4 mm, Width: 431.8 mm). There is a checkbox for 'Export to other sheets'. The 'Drawing Sheet' tab is also visible, showing a 'File' field and a 'Title Block' section. The 'Title Block' section includes fields for 'Issue Date' (2022-02-22), 'Revision' (1), 'Title' (EXAMPLE), 'Company' (KICAD), and nine 'Comment' fields (Comment 1 to Comment 9). Each comment field has an 'Export to other sheets' checkbox. The dialog box has a 'Preview' section at the bottom left showing a drawing sheet template. At the bottom right, there are 'Cancel' and 'OK' buttons.

El numerado de las hojas se actualiza automáticamente. Puede ajustar la fecha actual presionando el botón a la derecha de "Issue Date", pero esto no se actualizará automáticamente.

Herramienta de Búsqueda

The Find icon () can be used to access the search tool.



You can search for a reference, a value or a text string in the current sheet or in the whole hierarchy. Once found, the cursor will be positioned on the found element in the relevant sub-sheet.

Herramienta de Netlist

The Netlist icon () opens the netlist generation tool.

The tool creates a file which describe all connections in the entire hierarchy.

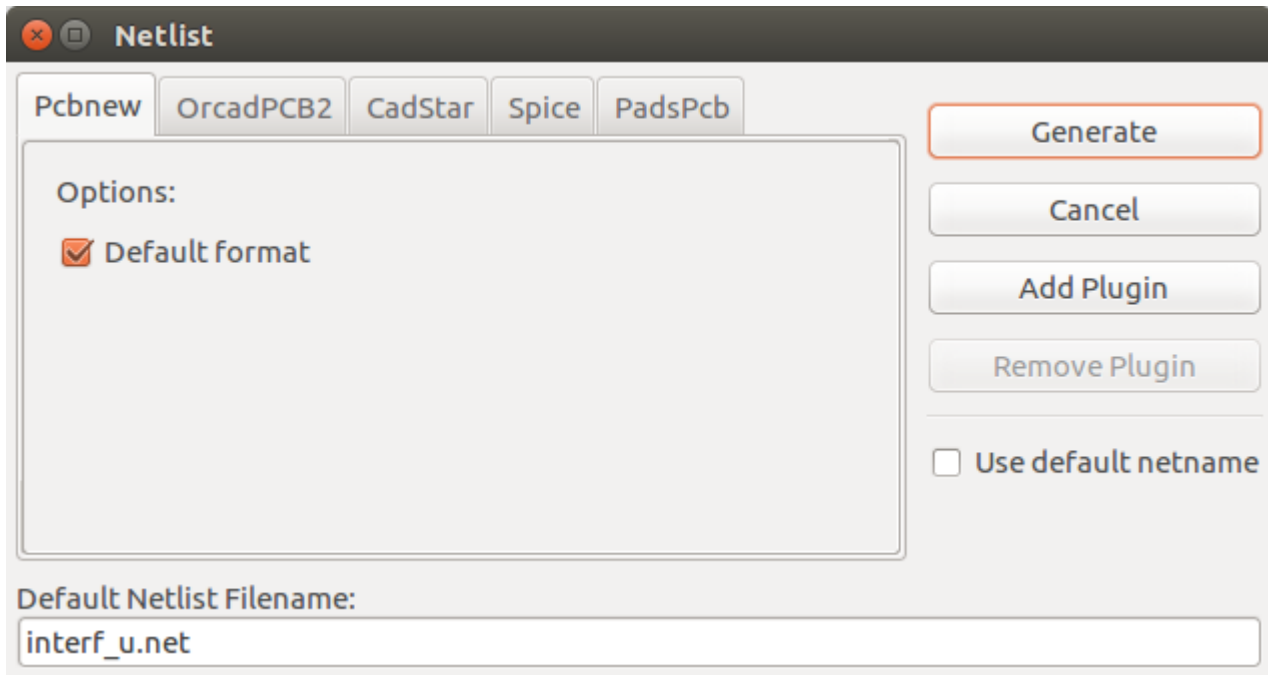
In a multisheet hierarchy, any local label is visible only inside the sheet to which it belongs. For example: the label LABEL1 of sheet 3 is different from the label LABEL1 of sheet 5 (if no connection has been intentionally introduced to connect them). This is due to the fact that the sheet name path is internally associated with the local label.

NOTE

Even though there is no text length limit for labels in KiCad, please take into account that other programs reading the generated netlist may have such constraints.

NOTE

Avoid spaces in labels, because they will appear as separated words in the generated file. It is not a limitation of KiCad, but of many netlist formats, which often assume that a label has no spaces.



Opciones:

Default Format	Check to select Pcbnew as the default format.
----------------	---


También se pueden generar otros formatos:

- Orcad PCB2
- CadStar
- Spice (simulators)

External plugins can be added to extend the netlist formats list (PadsPcb Plugin was added in the picture above).

There is more information about creating netlists in [Create a Netlist](#) chapter.

Herramienta de Anotado

The icon  launches the annotation tool. This tool assigns references to components.

Para componentes multi-parte (como el 7400 TTL que contiene 4 puertas), un sufijo multi-parte es añadido (así un 7400 TTL designado como U3 será dividido en U3A, U3B, U3C y U3D).

You can unconditionally annotate all the components or only the new components, i.e. those which were not previously annotated.

Annotate Schematic

Scope

☒ Use the entire schematic

☐ Use the current page only

☒ Keeep existing annotation


☐ Reset existing annotation

☐ Reset, but do not swap any annotated multi-unit parts

Annotation Order

☐ Sort components by X position

☒ Sort components by Y position



Annotation Choice

☒ Use first free number in schematic

☐ Start to sheet number*100 and use first free number

☐ Start to sheet number*1000 and use first free number

Dialog

☐ Automatically close this dialog

☐ Silent mode

Close

Clear Annotation

Annotate

Alcance

Use the entire schematic	All sheets are re-annotated (default).
Use the current page only	Only the current sheet is re-annotated (this option is to be used only in special cases, for example to evaluate the amount of resistors in the current sheet.).
Keep existing annotation	Conditional annotation, only the new components will be re-annotated (default).
Reset existing annotation	Unconditional annotation, all the components will be re-annotated (this option is to be used when there are duplicated references).
Reset, but do not swap any annotated multi-unit parts	Keeps all groups of multiple units (e.g. U2A, U2B) together when reannotating.

Orden de Anotado

Selects the order in which components will be numbered (either horizontally or vertically).

Tipo de Anotado

Selects the assigned reference format.

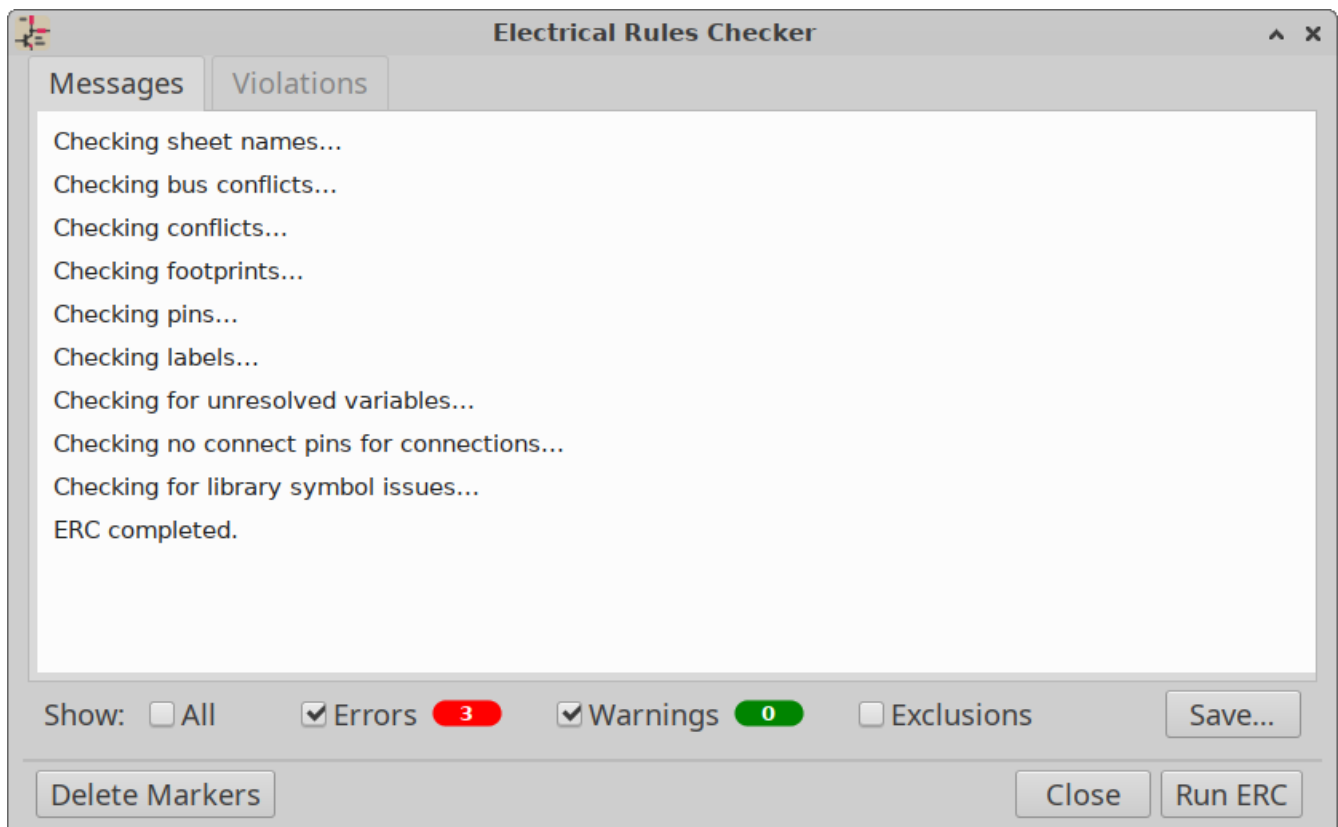
Herramienta de Comprobación de Reglas Eléctricas

The icon  launches the electrical rules check (ERC) tool.

This tool performs a design verification and is able to detect forgotten connections, and inconsistencies.

Once you have run the ERC, KiCad places markers to highlight problems. The error description is displayed after left clicking on the marker. An error report file can also be generated.

Ventana principal del ERC



Errors are displayed in the Electrical Rules Checker dialog:

- Número total de errores y avisos.
- Número de errores.
- Número de avisos.

Opciones:

Create ERC file report	Check this option to generate an ERC report file.
------------------------	---

Comandos:

Delete Markers	Remove all ERC error/warnings markers.
Run	Start an Electrical Rules Check.
Close	Close the dialog.

- Haciendo clic sobre un mensaje de error se salta a la marca correspondiente en el esquema.

Ventana de opciones del ERC

Electrical Rules Checker

ERC Options

Initialize to Default

Pin to pin connections

	Input Pin	Output Pin	Bidirectional Pin	Tri-State Pin	Passive Pin	Unspecified Pin	Power Input Pin	Power Output Pin	Open Collector	Open Emitter	No Connection
Input Pin	Green	Green	Green	Green	Green	Green	Green	Green	Green	Green	Green
Output Pin	Green	Red E	Green	Green	Green	Green	Green	Green	Green	Green	Green
Bidirectional Pin	Green	Green	Green	Green	Green	Green	Green	Green	Green	Green	Green
Tri-State Pin	Green	Yellow W	Green	Green	Green	Green	Green	Green	Green	Green	Green
Passive Pin	Green	Green	Green	Green	Green	Green	Green	Green	Green	Green	Green
Unspecified Pin	Yellow W	Yellow W	Yellow W	Yellow W	Yellow W	Yellow W	Yellow W	Yellow W	Yellow W	Yellow W	Yellow W
Power Input Pin	Green	Green	Green	Yellow W	Green	Yellow W	Green	Green	Green	Green	Green
Power Output Pin	Green	Red E	Yellow W	Red E	Green	Yellow W	Green	Red E	Green	Green	Green
Open Collector	Green	Red E	Green	Yellow W	Green	Yellow W	Green	Red E	Green	Green	Green
Open Emitter	Green	Red E	Yellow W	Yellow W	Green	Yellow W	Green	Red E	Green	Green	Green
No Connection	Red E	Red E	Red E	Red E	Red E	Red E	Red E	Red E	Red E	Red E	Red E

Label to label connections

☒ Test similar labels

☒ Test unique global labels

This tab allows you to define the connectivity rules between pins; you can choose between 3 options for each case:

- Sin error
- Aviso

Error

Cada cuadro de la matriz puede ser modificado haciendo clic en el.


Opciones:

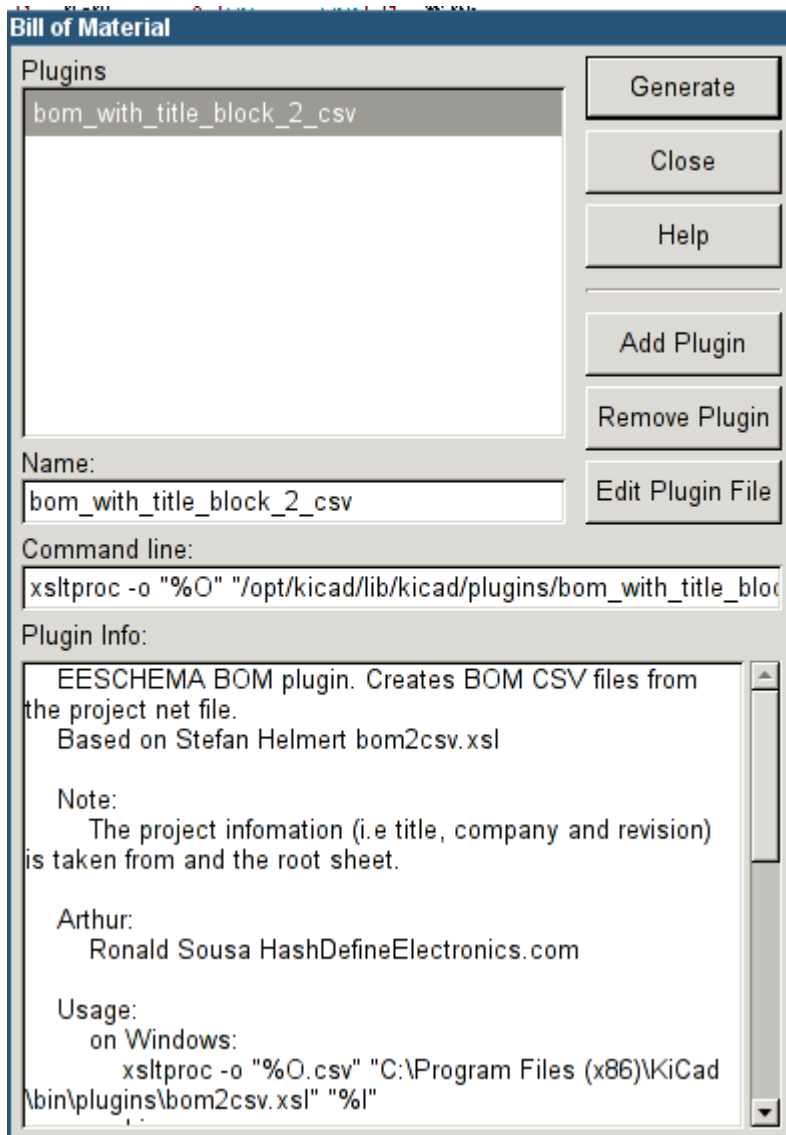
Test similar labels	Report labels that differ only by letter case (e.g. label/Label/LaBeL). Net names are case-sensitive therefore such labels are treated as separate nets.
Test unique global labels	Report global lables that occur only once for a particular net. Normally it is required to have at least two make a connection.

Comandos:

Initialize to Default	Restores the original settings.
-----------------------	---------------------------------

Herramienta Lista de Materiales

The icon  launches the bill of materials (BOM) generator. This tool generates a file listing the components and/or hierarchical connections (global labels).

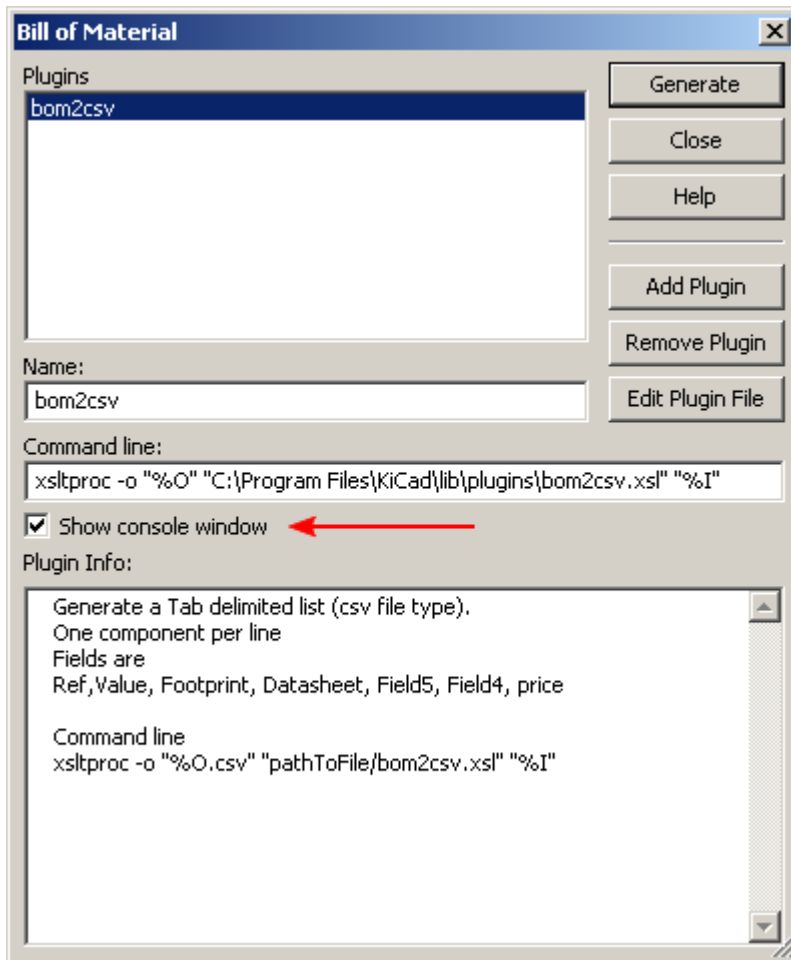


The Schematic Editor's BOM generator makes use of external plugins, either as XSLT or Python scripts. There are a few examples installed inside the KiCad program files directory.

Un conjunto de propiedades de componente útil para usar en un listado BOM son:

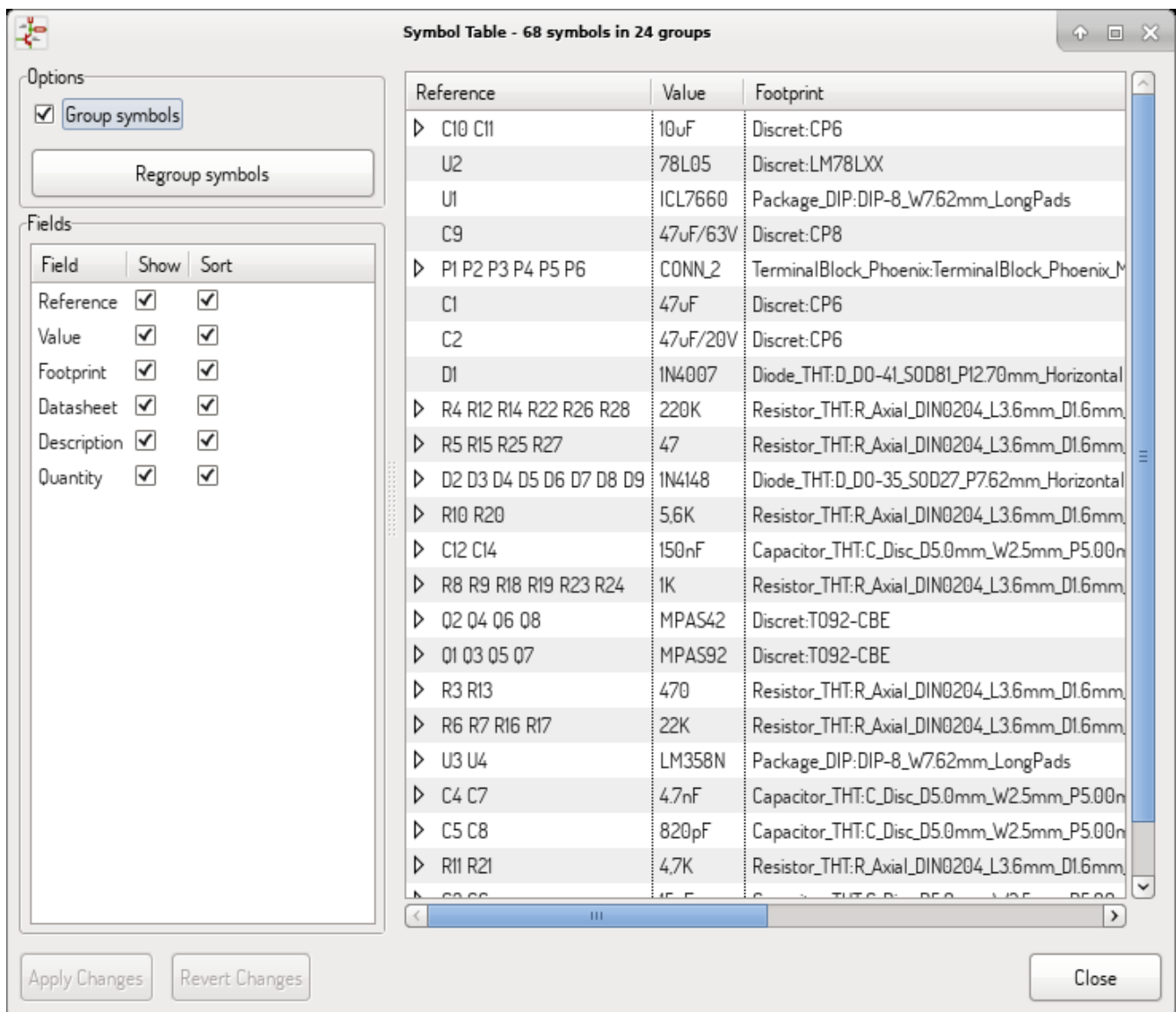
- Valor - nombre único para cada componente usado.
- Huella - tanto asignada manualmente como anotada de vuelta (ver abajo)
- Campo1 - Nombre del fabricante
- Campo2 - Número de referencia del Fabricante.
- Campo3 - Número de referencia del Distribuidor.

Por ejemplo:



Edit Fields tool

The icon  opens a spreadsheet to view and modify field values for all symbols.



Once you modify field values, you need to either accept changes by clicking on 'Apply' button or undo them by clicking on 'Revert' button.

Tricks to simplify fields filling

There are several special copy/paste methods in spreadsheet. They may be useful when entering field values that are repeated in a few components.

These methods are illustrated below.

Copy (Ctrl+C)	Selection	Paste (Ctrl+V)																																													
<table> <tr><td>abc</td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>	abc															<table> <tr><td>abc</td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>	abc															<table> <tr><td>abc</td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>	abc														
abc																																															
abc																																															
abc																																															
<table> <tr><td>11</td><td>12</td><td>13</td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>	11	12	13													<table> <tr><td>11</td><td>12</td><td>13</td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>	11	12	13													<table> <tr><td>11</td><td>12</td><td>13</td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>	11	12	13												
11	12	13																																													
11	12	13																																													
11	12	13																																													
<table> <tr><td>11</td><td></td><td></td></tr> <tr><td>21</td><td></td><td></td></tr> <tr><td>31</td><td></td><td></td></tr> <tr><td>41</td><td></td><td></td></tr> <tr><td>51</td><td></td><td></td></tr> </table>	11			21			31			41			51			<table> <tr><td>11</td><td></td><td></td></tr> <tr><td>21</td><td></td><td></td></tr> <tr><td>31</td><td></td><td></td></tr> <tr><td>41</td><td></td><td></td></tr> <tr><td>51</td><td></td><td></td></tr> </table>	11			21			31			41			51			<table> <tr><td>11</td><td>11</td><td>11</td></tr> <tr><td>21</td><td>21</td><td>21</td></tr> <tr><td>31</td><td>31</td><td>31</td></tr> <tr><td>41</td><td>41</td><td>41</td></tr> <tr><td>51</td><td>51</td><td>51</td></tr> </table>	11	11	11	21	21	21	31	31	31	41	41	41	51	51	51
11																																															
21																																															
31																																															
41																																															
51																																															
11																																															
21																																															
31																																															
41																																															
51																																															
11	11	11																																													
21	21	21																																													
31	31	31																																													
41	41	41																																													
51	51	51																																													
<table> <tr><td>11</td><td>12</td><td></td></tr> <tr><td>21</td><td>22</td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>	11	12		21	22											<table> <tr><td>11</td><td>12</td><td></td></tr> <tr><td>21</td><td>22</td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>	11	12		21	22											<table> <tr><td>11</td><td>12</td><td></td></tr> <tr><td>21</td><td>22</td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>	11	12		21	22										
11	12																																														
21	22																																														
11	12																																														
21	22																																														
11	12																																														
21	22																																														
<table> <tr><td>11</td><td>12</td><td>13</td></tr> <tr><td>21</td><td>22</td><td>23</td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>	11	12	13	21	22	23										<table> <tr><td>11</td><td>12</td><td>13</td></tr> <tr><td>21</td><td>22</td><td>23</td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>	11	12	13	21	22	23										<table> <tr><td>11</td><td>12</td><td>13</td></tr> <tr><td>21</td><td>22</td><td>23</td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>	11	12	13	21	22	23									
11	12	13																																													
21	22	23																																													
11	12	13																																													
21	22	23																																													
11	12	13																																													
21	22	23																																													

NOTE

These techniques are also available in other dialogs with a grid control element.

Import tool for footprint assignment

Accesso:

The icon  launches the back-annotate tool.

This tool allows footprint changes made in the PCB Editor to be imported back into the footprint fields in the Schematic Editor.

Manage Symbol Libraries

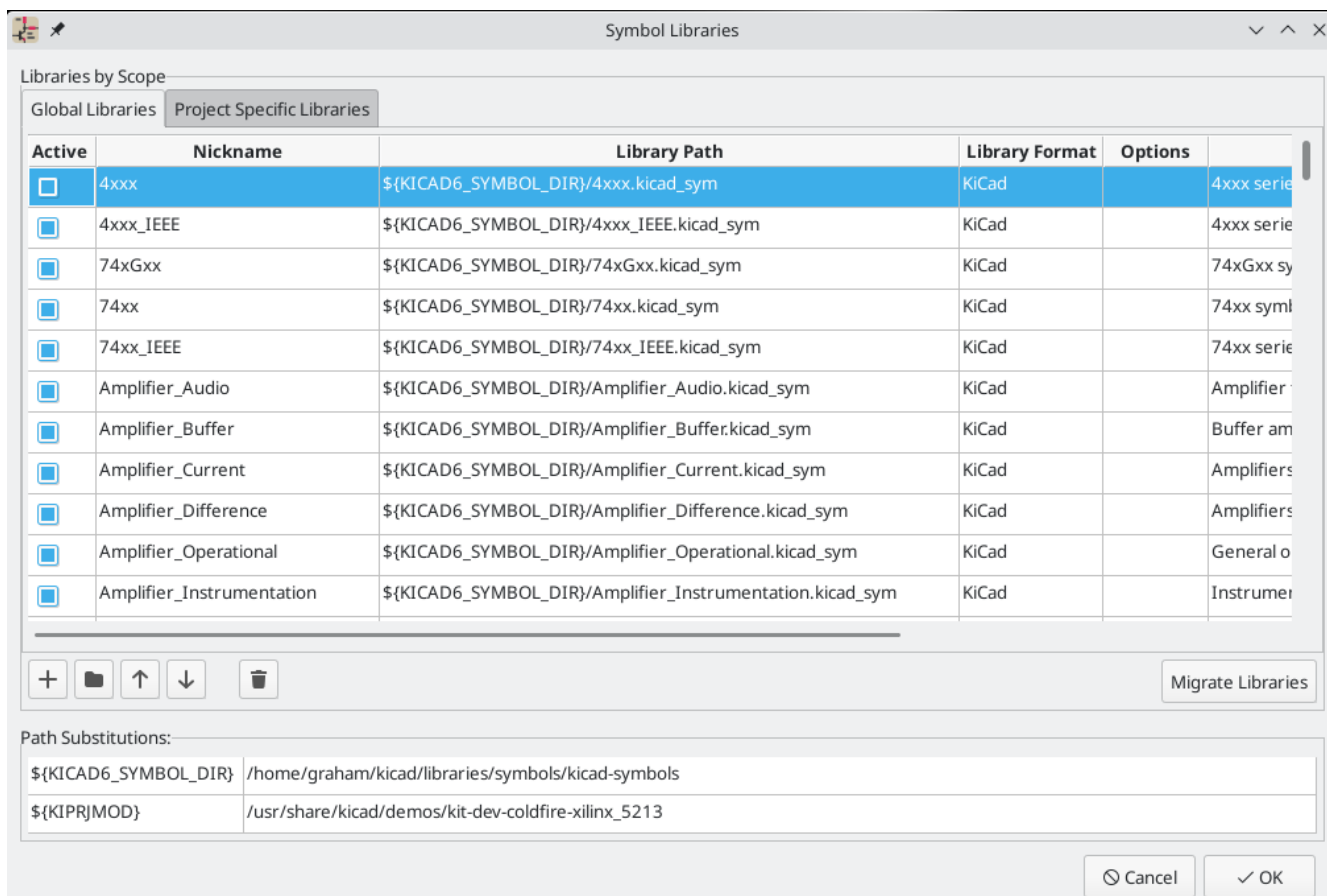
Symbol libraries hold collections of symbols used when creating schematics. Each symbol in a schematic is uniquely identified by a full name that is composed of a library nickname and a symbol name. An example is `Audio:AD1853`.

Symbol Library Table

The symbol library table holds a list of all library files KiCad knows about. The symbol library table is constructed from the global symbol library table file and the project specific symbol library table file.

When a symbol is loaded, KiCad uses the library nickname, `Audio` in our example, to lookup the library location in the symbol library table.

The image below shows the symbol library table editing dialog which can be opened by invoking the **Manage Symbol Libraries...** entry in the **Preferences** menu.



Global Symbol Library Table

The global symbol library table contains the list of libraries that are always available regardless of the currently loaded project file. The table is saved in the file `sym-lib-table` in the user's KiCad configuration folder. The [location of this folder](#) is dependent upon the operating system being used.

Project Specific Symbol Library Table

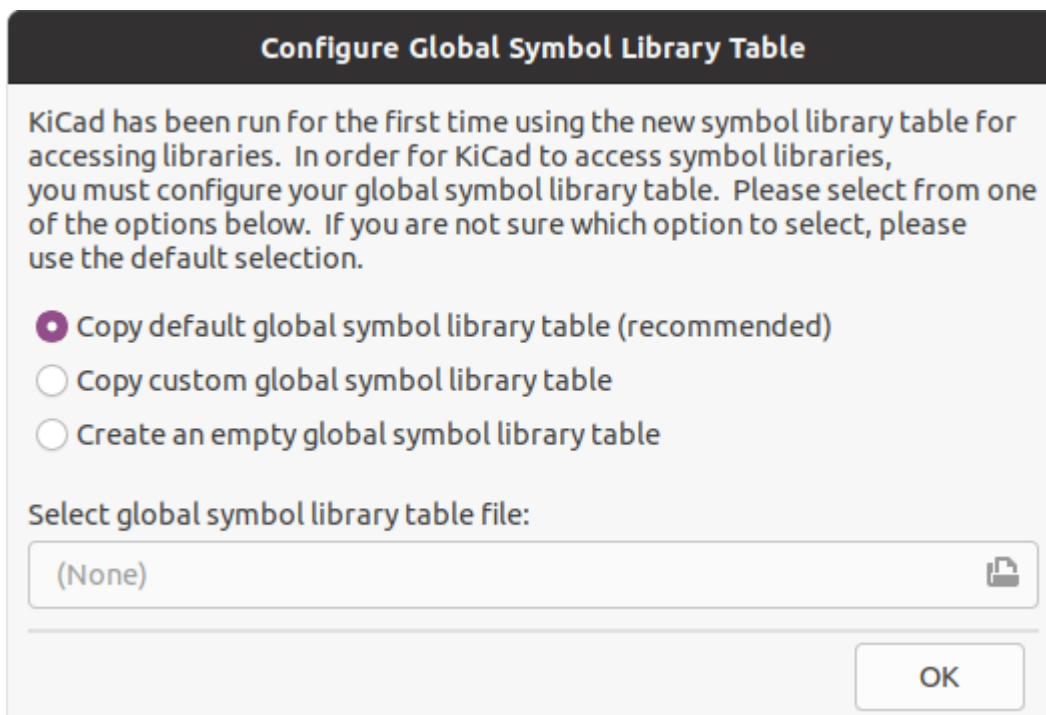
The project specific symbol library table contains the list of libraries that are available specifically for the currently loaded project file. The project specific symbol library table can only be edited when it is loaded

along with the project file. If no project file is loaded or there is no symbol library table file in the current project path, an empty table is created which can be edited and later saved along with the project file.

Initial Configuration

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 present the "Configure Global Symbol Library Table" dialog to the user. The dialog presents the user with three options.

- **Copy default global symbol library table (recommended).** If this option is selected, KiCad will copy the default symbol library table file stored in the system's Kicad template folder to the file `sym-lib-table` in the user's KiCad configuration folder. If the default template `sym-lib-table` file cannot be found, this option will be grayed out. The missing default table is usually caused by the KiCad default libraries not being installed (on some systems they are installed by a separate package). If the libraries are installed in a non-standard location, use the second option and browse to the library table location manually.
- **Copy custom global symbol library table.** If this option is selected, the user must browse to the desired symbol library table file, which will be copied to the user's KiCad configuration directory.
- **Create an empty global symbol library table.** An empty symbol library table file will be created in the user's KiCad configuration directory. The user must add libraries to the table manually.



NOTE

The default symbol library table includes all of the symbol libraries that are installed as part of KiCad. This may or may not be desirable depending on usages and the speed of the system. The amount of time required to load the symbol libraries is proportional to the number of libraries in the symbol library table. If symbol library load times are excessive, remove rarely and/or never used libraries from the global library table and add them to the project library table as required.

Adding Table Entries

In order to use a symbol library, it must first be added to either the global table or the project specific table. The project specific table is only applicable when you have a project file open.

NOTE | Each library entry must have a unique nickname.

The library nickname does not have to be related in any way to the actual library file name or path. The colon `:` and `\` characters cannot be used anywhere in the library nickname. Each library entry must have a valid path and/or file name depending on the type of library. Paths can be defined as absolute, relative, or by environment variable substitution (see section below).

The appropriate library format must be selected in order for the library to be properly read. "KiCad" format is used for KiCad version 6 libraries (`.kicad_sym` files), while "Legacy" format is used for libraries from older versions of KiCad (`.lib` files). Legacy libraries are read-only, but can be migrated to KiCad format libraries using the **Migrate Libraries** button (see section [Migrating Legacy Libraries](#)).

There is also a description field to add a description of the library entry. The option field is not used at this time so adding options will have no effect when loading libraries.

- Please note that you cannot have duplicate library nicknames in the same table. However, you can have duplicate library nicknames in both the global and project specific symbol library table.
- The project specific table entry will take precedence over the global table entry when duplicate nicknames occur.
- When entries are defined in the project specific table, a `sym-lib-table` file containing the entries will be written into the folder of the currently open project file.

Environment Variable Substitution

One of the most powerful features of the symbol library table is environment variable substitution. This allows for definition of custom paths to where symbol libraries are stored in environment variables. Environment variable substitution is supported by using the syntax `${ENV_VAR_NAME}` in the library path.

By default, at run time KiCad defines two environment variables relevant for locating symbol libraries:

- the `$KIPRJMOD` environment variable that always points to the currently open project directory. `$KIPRJMOD` cannot be modified.
- the `$KICAD6_SYMBOL_DIR` environment variable. This points to the path where the default symbol libraries that were installed with KiCad.

You can override `$KICAD6_SYMBOL_DIR` by redefining it in **Preferences** → **Configure Paths...** This is useful for using libraries installed in a nonstandard location.

`$KIPRJMOD` allows you to store libraries in the project path without having to define the absolute path (which is not always known) to the library in the project specific symbol library table.

Usage Patterns

Symbol libraries can be defined either globally or specifically to the currently loaded project. Symbol libraries defined in the user's global table are always available and are stored in the `sym-lib-table` file in

the user's KiCad configuration folder. The project-specific symbol library table is active only for the currently open project file.

There are advantages and disadvantages to each method. Defining all libraries in the global table means they will always be available when needed. The disadvantage of this is that load time will increase.

Defining all symbol libraries on a project specific basis means that you only have the libraries required for the project which decreases symbol library load times. The disadvantage is that you always have to remember to add each symbol library that you need for every project.

One usage pattern would be to define commonly used libraries globally and the libraries only required for the project in the project specific library table. There is no restriction on how to define libraries.

Migrating Legacy Libraries

Legacy libraries (`.lib` files) are read-only, but they can be migrated to KiCad version 6 libraries (`.kicad_sym`). KiCad version 6 libraries cannot be viewed or edited by KiCad versions older than 6.0.0.

Legacy libraries can be converted to KiCad 6 libraries by selecting them in the symbol library table and clicking the **Migrate Libraries** button. Multiple libraries can be selected and migrated at once by `Ctrl`-clicking or `shift`-clicking.

Libraries can also be converted one at a time by opening them in the Symbol Editor and saving them as a new library.

Legacy Project Remapping

When loading a schematic created prior to the symbol library table implementation, KiCad will attempt to remap the symbol library links in the schematic to the appropriate library table symbols. The success of this process is dependent on several factors:

- the original libraries used in the schematic are still available and unchanged from when the symbol was added to the schematic.
- all rescue operations were performed when detected to create a rescue library or keep the existing rescue library up to date.
- the integrity of the project symbol cache library has not been corrupted.

WARNING

The remapping will make a back up of all the files that are changed during remapping in the rescue-backup folder in the project folder. Always make a back up of your project before remapping just in case something goes wrong.

WARNING

The rescue operation is performed even if it has been disabled to ensure the correct symbols are available for remapping. Do not cancel this operation or the remapping will fail to correctly remap schematics symbols. Any broken symbol links will have to be fixed manually.

NOTE

If the original libraries have been removed and the rescue was not performed, the cache library can be used as a recovery library as a last resort. Copy the cache library to a new file name and add the new library file to the top of the library list using a version of KiCad prior to the symbol library table implementation.

Creación y Edición de Esquemas

Introducción

Un esquema puede ser representado por una única hoja, pero, si fuera suficientemente grande, podría requerir varias hojas.

A schematic represented by several sheets is hierarchical, and all its sheets (each one represented by its own file) constitute a complete KiCad schematic. The manipulation of hierarchical schematics will be described in the [Hierarchical Schematics](#) chapter.

Consideraciones generales

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:


- Validar una serie de reglas ([Comprobador de Reglas Eléctricas](#)) para detectar errores y omisiones.
- Generar automáticamente listas de materiales ([BOM](#))
- [Generando netlist](#) para software de simulación como SPICE
- [Defining a circuit](#) for transferring to PCB layout.

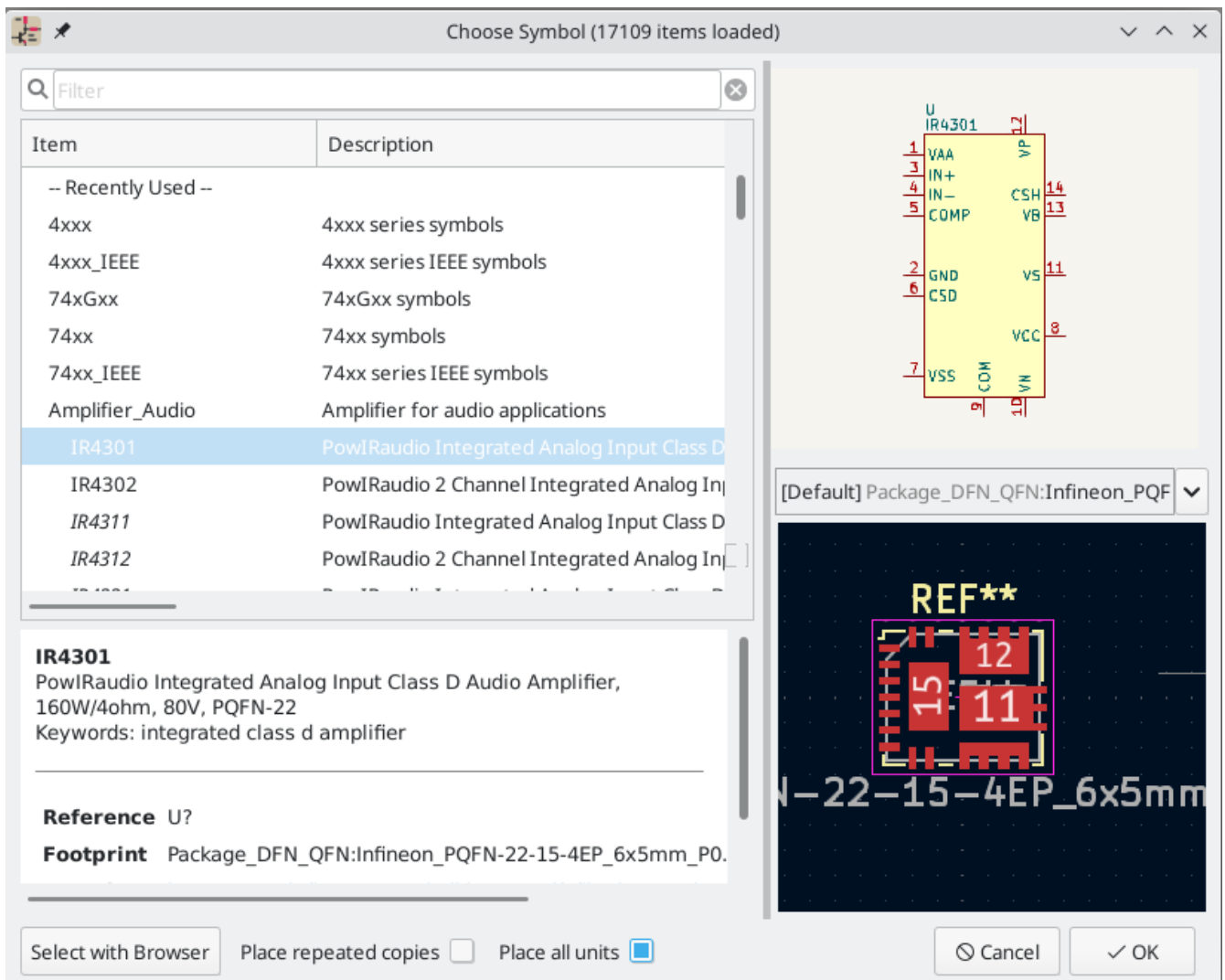
A schematic mainly consists of symbols, wires, labels, junctions, buses and power ports. For clarity in the schematic, you can place purely graphical elements like bus entries, comments, and polylines.

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.

Symbol placement and editing

Find and place a symbol

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.



The Choose Symbols dialog will filter symbols by name, keywords, and description according to what you type into the search field. Advanced filters can be used just by typing them:

- **Wildcards:** use the characters `?` and `*` respectively to mean "any character" and "any number of characters".
- **Relational:** 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}

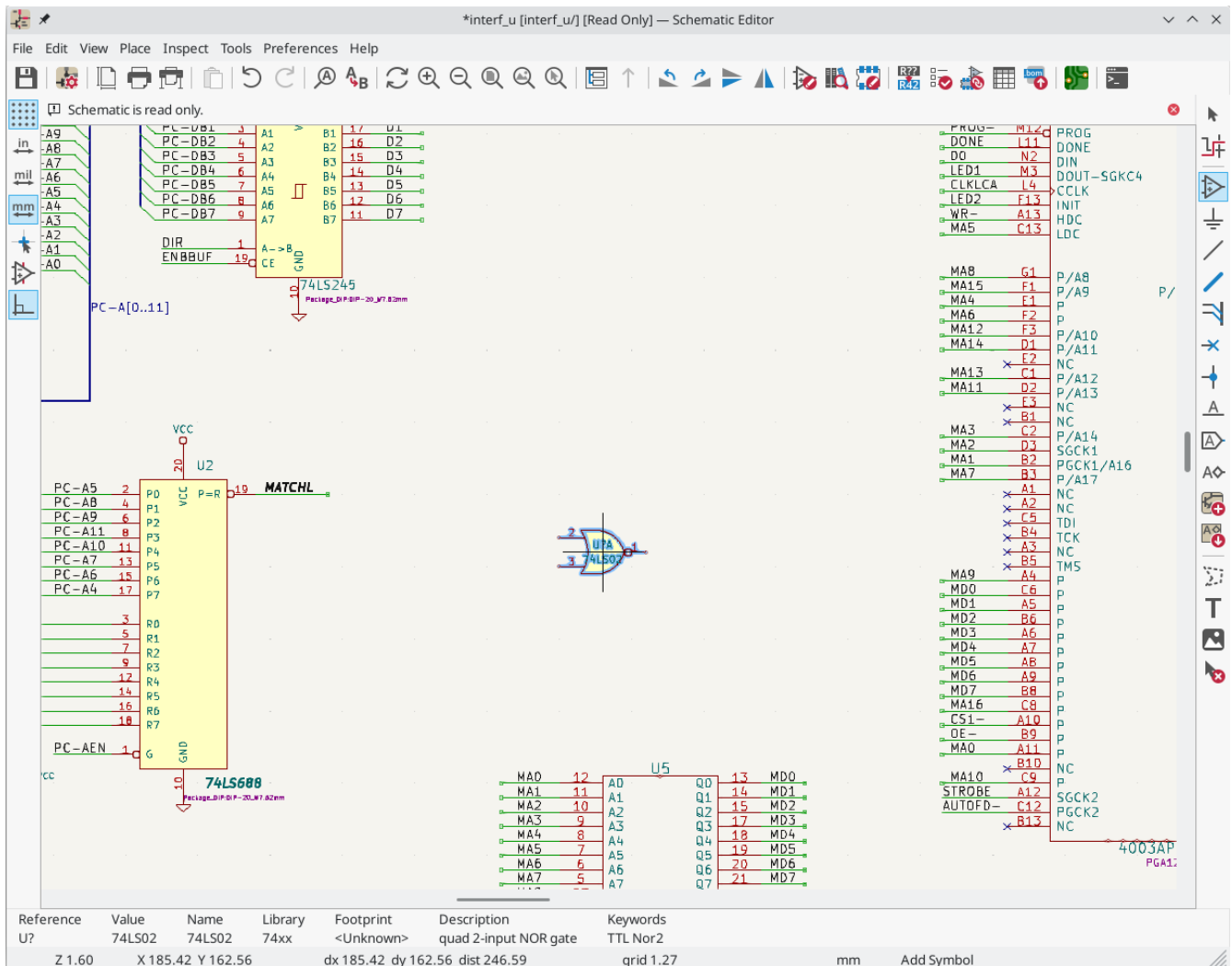
ki	mi	gi	ti
2^{10}	2^{20}	2^{30}	2^{40}

- **Regular expression:** 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.


Here is a symbol during placement:



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.

Symbol Editing and Modification (already placed component)

There are two ways to edit a symbol:

- Modification of the symbol itself: position, orientation, unit selection on a multi-unit symbol.
- Modification of one of the fields of the symbol: reference, value, footprint, etc.

When a symbol has just been placed, you may have to modify its value (particularly for resistors, capacitors, etc.), but it is useless to assign to it a reference number right away, or to select the unit (except for components with locked units, which you have to assign manually). This can be done automatically by the annotation function.

Symbol modification

To modify some feature of a symbol, position the cursor on the symbol, and then either:

- Double-click on the symbol to open the full editing dialog.
- Haga clic derecho para abrir el menú contextual y use uno de los comandos: Mover, Orientar, Editar, Borrar, etc,
- Use a hotkey to perform an action on the symbol (**E** to open the properties dialog, **R** to rotate, etc.). Note that hotkeys act on the selected symbol; if no symbol is selected hotkeys act on the symbol under the cursor.

Symbols can also be selected by clicking on them or drag-selecting them. Selected symbols can be modified by clicking relevant buttons in the top toolbar or using a hotkey.

Modificación de los campos de texto

Puede modificar la referencia, valor, posición, orientación, tamaño del texto y visibilidad de los campos:

- Haga doble clic sobre el campo de texto para modificarlo.
- Haga clic derecho para abrir el menú contextual y use uno de los comandos: Mover, Rotar, Editar, Borrar, etc.
- Position the cursor over the field (if nothing is selected) or select the field and press **E** to edit the field.
- Position the cursor over the symbol (if nothing is selected) or select the symbol and press **V**, **U**, or **F** hotkeys to directly edit the symbol's value, reference designator, or footprint fields, respectively.

For more options, or in order to create fields, double-click on the symbol to open the Symbol Properties dialog.

Symbol Properties

General Alternate Pin Assignments

Fields

Name	Value	Show	H Align	V Align	Italic	Bold	Text Size	Orientation	X Position	Y Position
Reference	U2	<input type="checkbox"/>	Center	Center	<input type="checkbox"/>	<input type="checkbox"/>	1.524 mm	Horizontal	3.81 mm	27.94 mm
Value	74LS688	<input checked="" type="checkbox"/>	Center	Center	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1.524 mm	Horizontal	8.89 mm	-27.94 mm
Footprint	Package_DIP:DIP-20	<input checked="" type="checkbox"/>	Center	Center	<input type="checkbox"/>	<input type="checkbox"/>	0.762 mm	Horizontal	8.89 mm	-30.48 mm
Datasheet		<input type="checkbox"/>	Center	Center	<input type="checkbox"/>	<input type="checkbox"/>	1.524 mm	Horizontal	0 mm	0 mm

General
 Unit:
☐ Alternate symbol (De Morgan)
 Angle:
 Mirror:

Pin Text
☒ Show pin numbers
☒ Show pin names

Attributes
☐ Exclude from bill of materials
☐ Exclude from board

Library link:

Each field can be visible or hidden, and displayed horizontally or vertically. The displayed position is always indicated for a normally displayed symbol (no rotation or mirroring) and is relative to the anchor point of the symbol.

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.

Electrical Connections

Introducción

There are a number of elements that can be added to a schematic to electrically connect components. All of these elements can be placed with the buttons on the vertical right toolbar or using hotkeys.

Estos elementos son:

- **Wires:** direct connection between pins.
- **Buses:** connections for a group of signals.

Bus entries: connections between wires and buses.

- **No-connection flags:** terminations for pins or wires that are intentionally unconnected. These flags prevent ERC violations for unconnected pins.
- **Junctions:** connections between crossing wires or buses.
- **Net labels:** local name for a signal. Signals within a sheet that have the same net label are connected.
- **Global labels:** global name for a signal. Signals with the same global label are connected even if they are not in the same sheet.
- **Hierarchical labels:** a label for a signal in a subsheet that enables the signal to be accessed in a parent sheet. See the [Hierarchical Schematics](#) section for more information about hierarchical labels, sheets, and pins.
- **Hierarchical sheets:** an instantiation of a subsheet within a parent sheet. The parent sheet can connect to the subsheet through the subsheet's hierarchical pins.
- **Hierarchical pins:** connection points between a parent sheet and a subsheet. Hierarchical pins appear at the parent sheet's level and correspond to hierarchical labels in the subsheet.

Several other types of items can be placed on the schematic but do not affect connectivity:

- **Graphical lines:** graphical lines for presentation.
- **Text:** textual comments and annotations.
- **Bitmap images:** raster graphics from an external file.

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

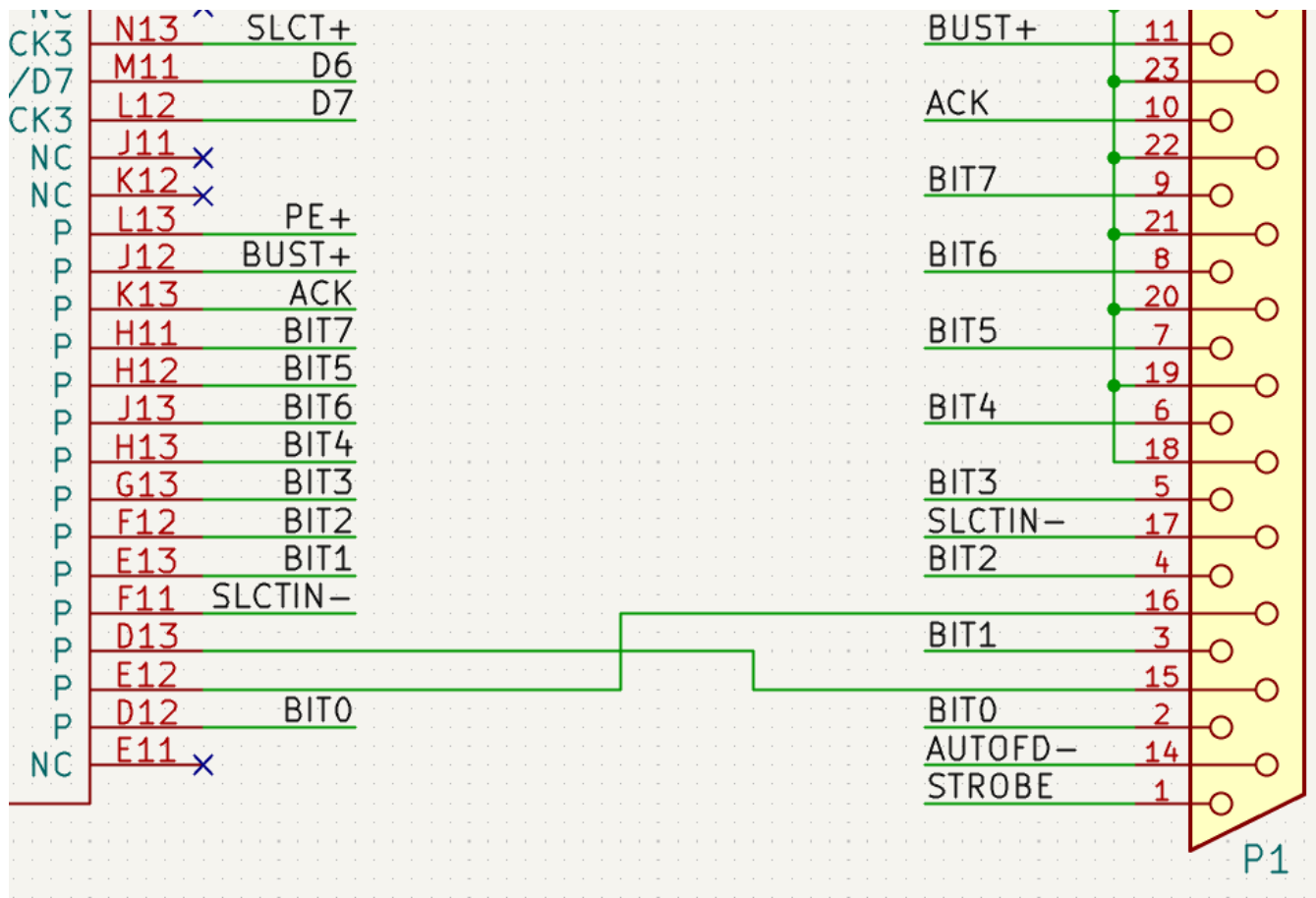
- **Power ports:** symbols for connecting wires to a power or ground net.
- **PWR_FLAG:** a specific symbol for indicating that a net is powered when it is not connected to a power output pin (for example, a power net that is supplied by an off-board connector).

Conexiones (Hilos y etiquetas)

Existen dos métodos de establecer conexiones:

- Hilos entre dos pines.
- Etiquetas.

La siguiente figura muestra los dos métodos:



Label Connections

The point of "contact" of a label is the small square in the corner of the label. The square disappears when the label is connected. The position of the connection point relative to the label text can be changed by choosing a different label orientation in the label properties, or by mirroring/rotating the label.

The label's connection point must be in contact with a wire or the end of a pin for the label to be connected.

Wire Connections

Para establecer una conexión, un segmento de hilo debe estar conectado en su extremo a otro segmento de hilo o a un pin.

Si existe superposición (si un hilo pasa sobre un pin, pero no es conectado al extremo del pin) no existe conexión.

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."

Wire Junctions

Wires that cross are not implicitly connected. It is necessary to join them with a junction dot if a connection is desired. Junction dots will be automatically added to wires that start or end on top of an existing wire.

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

Nets with Multiple Names

A signal 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.

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 automatically be connected to the global VCC net.

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, the use of invisible power pins in symbols is not recommended outside of power port symbols, and is only supported for compatibility with legacy designs and symbols.

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 (options) 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-ports,power port symbol> for each net and connect them with a wire.

It is not recommended to use labels for power connection. These only have a "local" connection scope, and will not connect to invisible power pins.

Wiring

To begin connecting elements, you may either use the 'Wire' or 'Bus' tools from the right-hand toolbar, or you can auto-start a new wire from any existing pin or unconnected wire.

The wire drag action will drag the entire wire if you start dragging from the middle of the wire. Alternatively, it will drag just one corner if you start the drag action over a corner where two wires connect

Conexiones (Buses)

En el siguiente esquema, múltiples pines están conectados a buses.

that are repeated across several similar circuits, using this technique can save time.

Group buses can also contain vector buses. For example, the bus `MEMORY{A[7..0] D[7..0] OE WE}` contains both vector buses and plain signals, and will result in nets such as `MEMORY.A7` and `MEMORY.OE` on the PCB.

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.

Conexión entre miembros de bus

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.

En el ejemplo anterior, las conexiones son realizadas por las etiquetas colocadas sobre los hilos conectados a los pines. Las entradas de bus (segmentos de hilo a 45 grados) son solo representaciones gráficas, y no forman necesariamente conexiones lógicas.

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.
- Dibuje el hilo bajo la primera etiqueta. Después utilice el comando repetir para colocar los otros hilos bajo el resto de etiquetas.
- Si fuera necesario, coloque las entradas al bus del mismo modo (coloque la primera entrada y utilice el comando repetir)

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.).

Bus unfolding

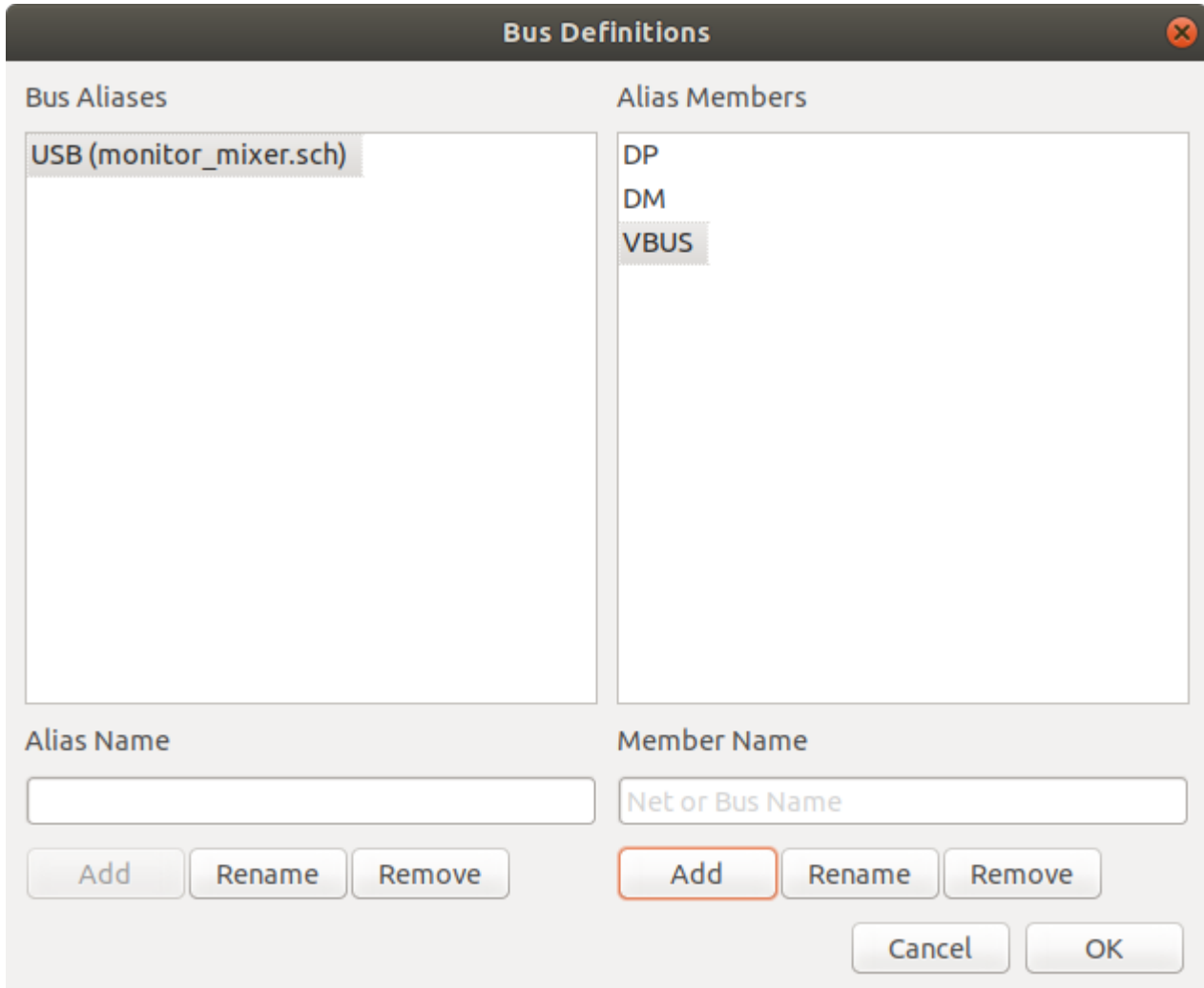
The unfold tool allows you to quickly break out signals from a bus. To unfold a signal, right-click on a bus object (a bus wire, etc) and choose **Unfold from Bus**. Alternatively, use the **Unfold Bus** hotkey (default: `C`) when the cursor is over a bus object. The menu allows you to select which bus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

After selecting the bus member, the next click will place the bus member label at the desired location. The tool automatically generates a bus entry and wire leading up to the label location. After placing the label, you can continue placing additional wire segments (for example, to connect to a component pin) and complete the wire in any of the normal ways.

Bus aliases

Bus aliases are shortcuts that allow you to work with large group buses more efficiently. They allow you to define a group bus and give it a short name that can then be used instead of the full group name across the schematic.

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



An alias may be named any valid signal name. Using the dialog, you can add signals or vector buses to the alias. As a shortcut, you can type or paste in a list of signals and/or buses separated by spaces, and they will all be added to the alias definition. In this example, we define an alias called **USB** with members **DP**, **DM**, and **VBUS**.

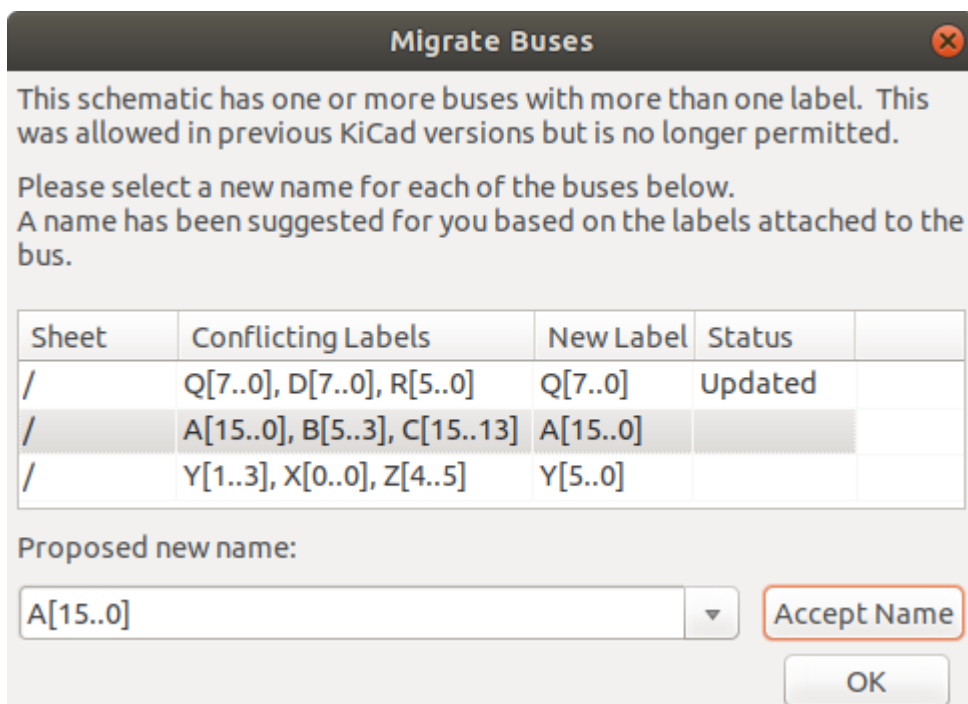
After defining an alias, it can be used in a group bus label by putting the alias name inside the curly braces of the group bus: **{USB}**. This has the same effect as labeling the bus **{DP DM VBUS}**. You can also add a prefix name to the group, such as **USB1{USB}**, which results in nets such as **USB1.DP** as described above. For complicated buses, using aliases can make the labels on your schematic much shorter. Keep in mind that the aliases are just a shortcut, and the name of the alias is not included in the netlist.

Bus aliases are saved in the schematic file. Any aliases created in a given schematic sheet are available to use in any other schematic sheet that is in the same hierarchical design.

Buses with more than one label

KiCad 5.0 and earlier allowed the connection of bus wires with different labels together, and would join the members of these buses during netlisting. This behavior has been removed in KiCad 6.0 because it is incompatible with group buses, and also leads to confusing netlists because the name that a given signal will receive is not easily predicted.

If you open a design that made use of this feature in a modern version of KiCad, you will see the Migrate Buses dialog which guides you through updating the schematic so that only one label exists on any given set of bus wires.



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:

▼ Accept Name

OK

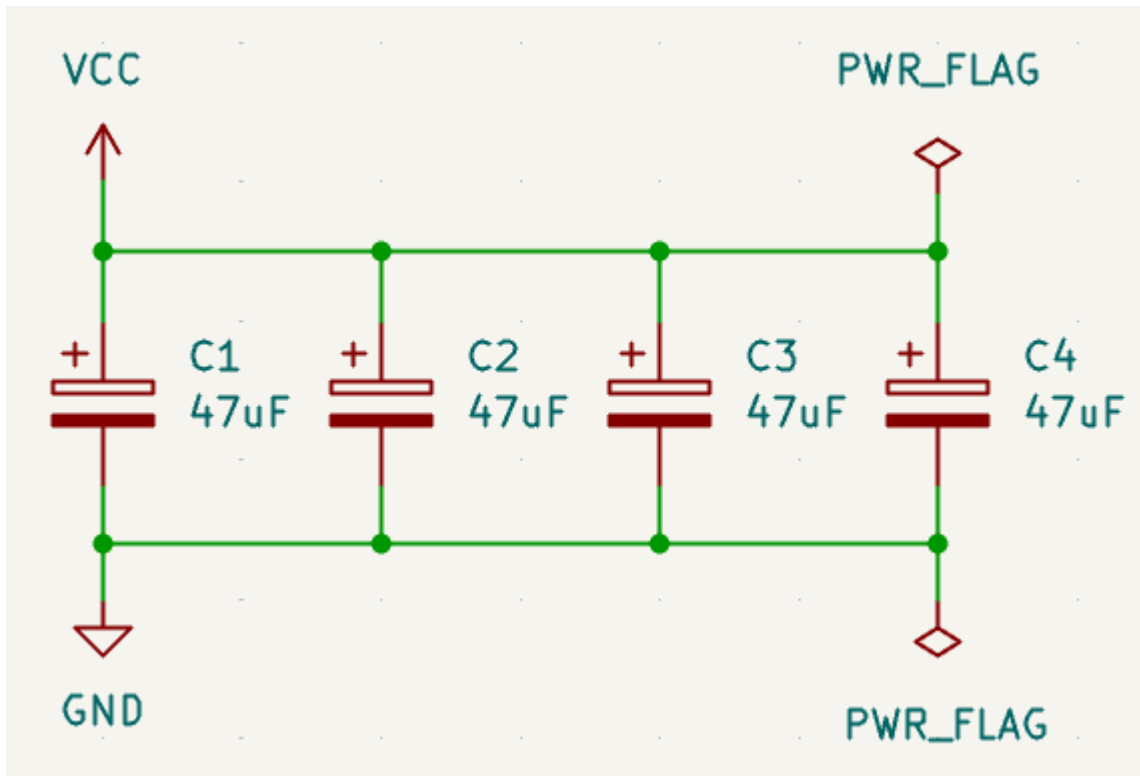
For each set of bus wires that has more than one label, you must choose the label to keep. The drop-down name box lets you choose between the labels that exist in the design, or you can choose a different name by manually entering it into the new name field.

Power Ports

Power port symbols are conventionally used to connect pins to power nets. Power port symbols have a single pin which is invisible and marked as a power input. As described in the [hidden power pins section](#), any wire connected to the pin of a power port is therefore automatically connected to the power net with the same name as the port's pin.

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 hidden pin marked as a power input. Name the pin according to the desired power net.

La siguiente figura muestra un ejemplo de conexiones con puertos de potencia.



In this example, power ports symbols are used to connect the positive and negative terminals of the capacitors to the `VCC` and `GND` nets, respectively.

Power port symbols are found in the `power` symbol library. They can also be created by drawing a symbol with a hidden "power input" pin that has the name of the desired power net.

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

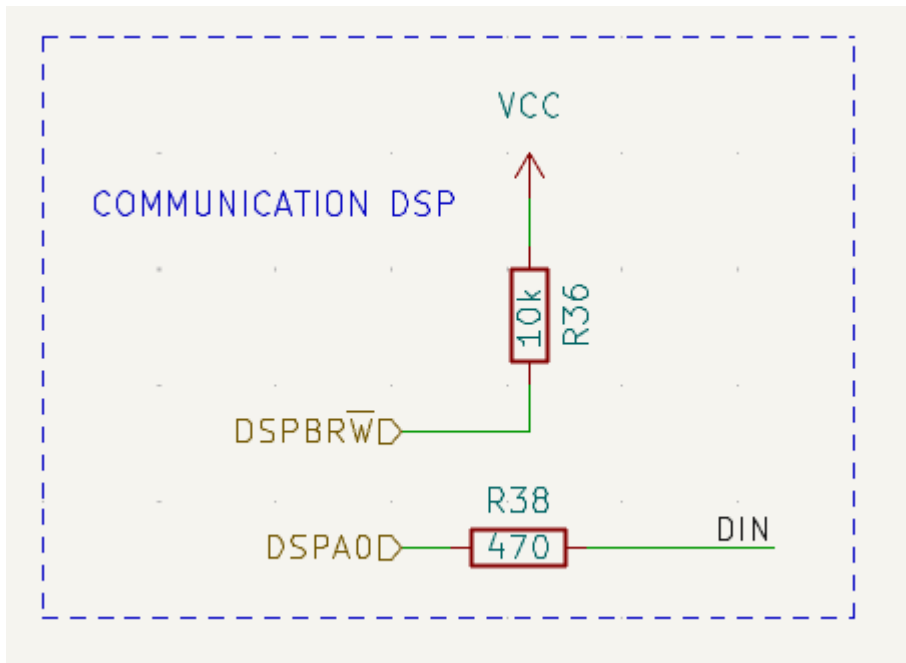
No-connection flags (→X) are used to indicate that a pin is intentionally unconnected. These flags do not have any effect on the schematic's connectivity, but they prevent "unconnected pin" ERC warnings for pins that are intentionally unconnected.

Complementos de Dibujo

Text comments 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 (Σ) 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.



Bloque de titulo de la hoja

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

Page Settings

Paper

Size: A4 210x297mm

Orientation: Landscape

Custom paper size:

Height: 279.4 mm

Width: 431.8 mm

☐ Export to other sheets

Preview

Drawing Sheet

File:

Title Block

Number of sheets: 1 Sheet number: 1

Issue Date: 2022-02-22 <<< 02/22/2022 ☐ Export to other sheets

Revision: 1 ☐ Export to other sheets

Title: EXAMPLE ☐ Export to other sheets

Company: KICAD ☐ Export to other sheets

Comment1: Comment 1 ☐ Export to other sheets

Comment2: Comment 2 ☐ Export to other sheets

Comment3: Comment 3 ☐ Export to other sheets

Comment4: Comment 4 ☐ Export to other sheets

Comment5: Comment 5 ☐ Export to other sheets

Comment6: Comment 6 ☐ Export to other sheets

Comment7: Comment 7 ☐ Export to other sheets

Comment8: Comment 8 ☐ Export to other sheets

Comment9: Comment 9 ☐ Export to other sheets

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.

A drawing sheet template file can also be selected.

Comment 4		
Comment 3		
Comment 2		
Comment 1		
KICAD		
Sheet: /		
File: title_block_example.kicad_sch		
Title: EXAMPLE		
Size: A4	Date: 2022-02-22	Rev: 1
KiCad E.D.A. kicad (6.0.1)		Id: 1/1
4	5	6

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

Rescuing cached symbols

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.

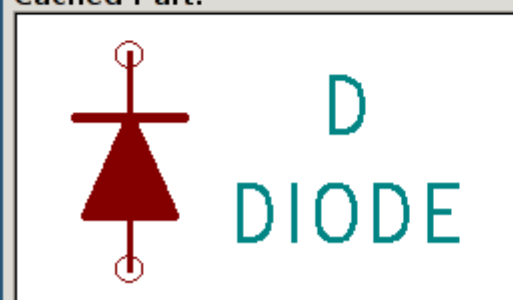
Symbols with cache/library conflicts:

scue symbol	Symbol name
<input checked="" type="checkbox"/>	DIODE

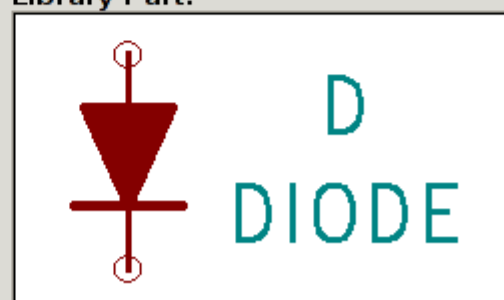
Instances of this symbol:

Reference	Value
D1	DIODE
D2	DIODE
D3	DIODE


Cached Part:



Library Part:



Never Show Again

 Cancel

 OK

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.

If you would prefer not to see this dialog, you can press "Never Show Again". The default will be to do nothing and allow the new components to be loaded. This option can be changed back in the Libraries preferences.


Esquemas Jerarquicos

Introducción

Una representación jerárquica es generalmente una buena solución para proyectos mas grandes que unas pocas hojas. Si quiere gestionar este tipo de proyectos, sera necesario:

- Usar hojas grandes, lo que resulta en problemas para imprimir y manejar las hojas.
- Usar varias hojas, lo que conduce a una estructura jerárquica.

El esquema completo consiste en una hoja principal, llamada hoja raíz, y una serie de hojas hijas que constituyen la jerarquía. Además, una subdivisión adecuada del diseño en hojas separadas, a menudo mejora su legibilidad.

From the root sheet, you must be able to find all sub-sheets. Hierarchical schematics management is very easy with KiCad, thanks to an integrated "hierarchy navigator" accessible via the icon  of the top toolbar.

There are two types of hierarchy that can exist simultaneously: the first one has just been evoked and is of general use. The second consists in creating symbols in the library that appear like traditional symbols in the schematic, but which actually correspond to a schematic which describes their internal structure.

Este segundo tipo es usado en el desarrollo de circuitos integrados, pero en este caso tiene que usar bibliotecas de funciones en el esquema que este dibujando.

KiCad currently doesn't treat this second case.

Una jerarquía puede ser:

- simple: una hoja dada es usada solo una vez
- compleja: una hoja dada es usada mas de una vez (múltiples instancias)
- plana: que es una jerarquía simple, donde las conexiones entre hojas no son dibujadas.


KiCad can deal with all these hierarchies.

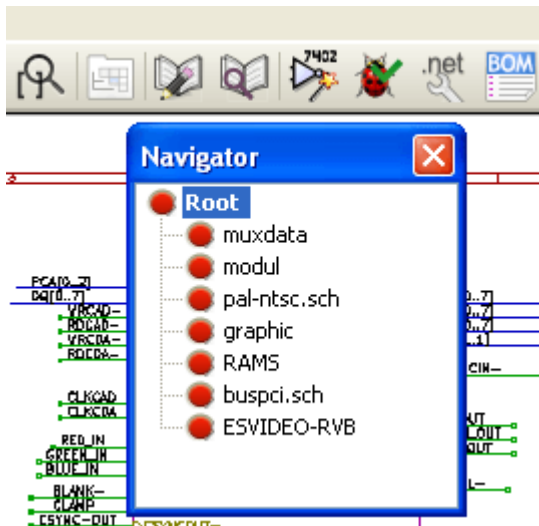
La creación de esquemas jerárquicos es fácil, la jerarquía completa es gestionada comenzando desde el esquema raíz, como si solo tuviera un esquema.

Los dos pasos importantes que debe entender son:

- Como crear una hoja hija.
- How to build electrical connections between sub-sheets.

Navegando en la Jerarquía

Navigation among sub-sheets is acheived by using the navigator tool accessible via the button  on the top toolbar.






Each sheet is reachable by clicking on its name. For quick access, right click on a sheet name, and choose to Enter Sheet or double click within the bounds of the sheet.

In order to exit the current sheet to the parent sheet, right click anywhere in the schematic where there is no object and select "Leave Sheet" in the context menu or press Alt+Backspace.

Etiquetas locales, jerárquicas y globales

Propiedades

Local labels, tool , are connecting signals only within a sheet. Hierarchical labels (tool ) are connecting signals only within a sheet and to a hierarchical pin placed in the parent sheet.

Global labels (tool ) are connecting signals across all the hierarchy. Power pins (type *power in* and *power out*) invisible are like global labels because they are seen as connected between them across all the hierarchy.

NOTE

Dentro de una jerarquía (simple o compleja) se pueden usar etiquetas jerárquicas y/o etiquetas globales.

Summary of hierarchy creation


Tiene que:

- Colocar en la hoja padre un símbolo de jerarquía llamado "hoja símbolo".
- Entre dentro del nuevo esquema (hoja hija) con el navegador y dibuje su esquema en ella, como cualquier otro esquema.
- Draw the electric connections between the two schematics by placing Global Labels (HLabels) in the new schematic (sub-sheet), and labels having the same name in the root sheet, known as SheetLabels. These SheetLabels will be connected to the sheet symbol of the root sheet to the other elements of the schematic like standard symbol pins.

Símbolo de la hoja

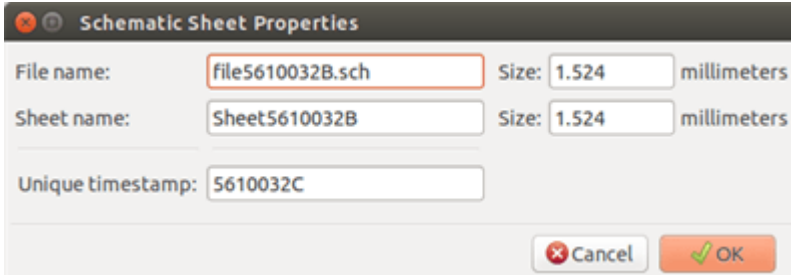
Dibuje un rectángulo, definido por dos puntos de su diagonal, que simbolizará la hoja hija.

El tamaño de este rectángulo debe permitirle colocar mas tarde etiquetas particulares, pines de jerarquía, correspondientes a las etiquetas globales (HLabels) en la hoja hija.

These labels are similar to usual symbol pins. Select the tool .

Haga clic para colocar la esquina superior izquierda del rectángulo. Haga clic de nuevo para colocar la esquina inferior derecha, dejando un rectángulo suficientemente grande.

Se le pedirá que escriba el nombre de archivo y en nombre de la hoja para esta hoja hija (con el fin de alcanzar el correspondiente esquema, usando el navegador de jerarquía)




Debe introducir al menos el nombre de archivo. Si no especifica un nombre de hoja, se usará el nombre de fichero como nombre de pagina (modo usual de hacerlo)

Conexion - Pines de Jerarquia

Debe crear aquí los puntos de conexión (pines de jerarquía) para el símbolo que acaba de crear.


These points of connection are similar to normal symbol pins, with however the possibility to connect a complete bus with only one point of connection.

Importing Hierarchical Sheet Pins

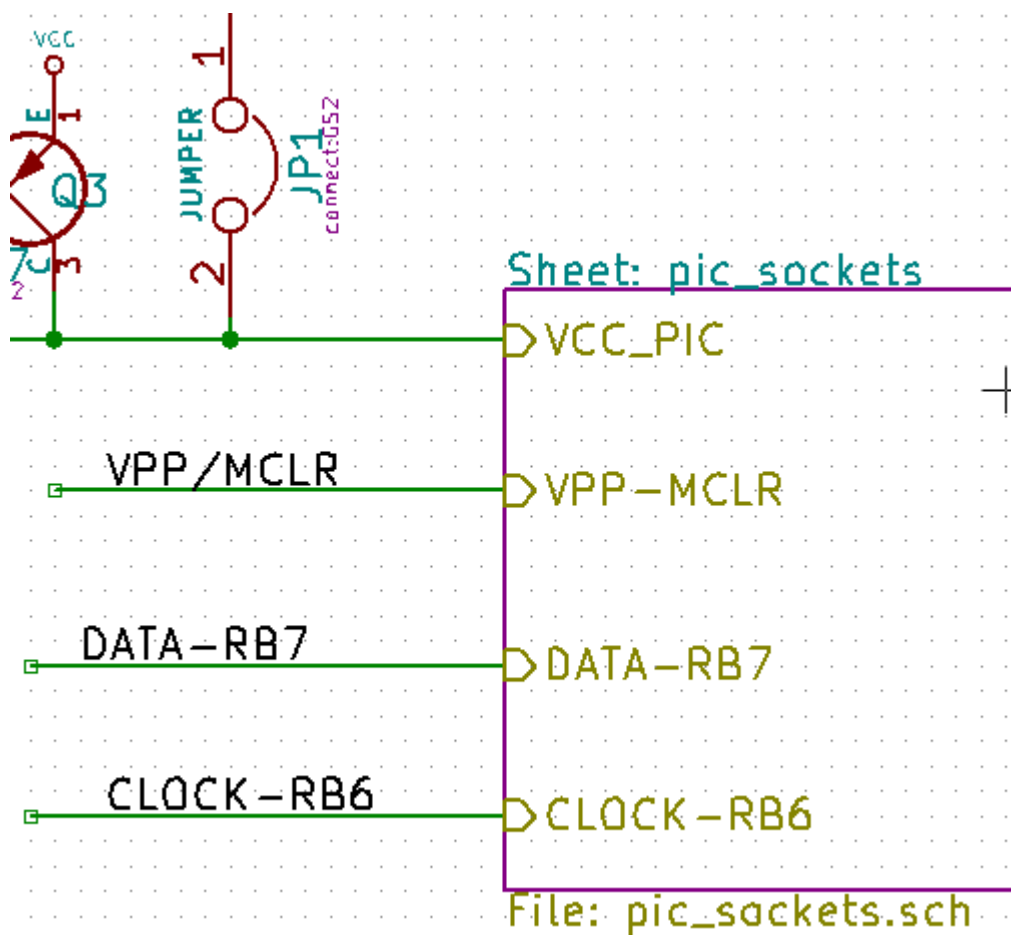
- Select the tool .
- Click on the hierarchical sheet from where you want to import the pins corresponding to hierarchical labels placed in the corresponding schematic. A hierarchical pin appears, if a new hierarchical label exists, i.e. not corresponding to an already placed pin.
- Haga clic donde quiera colocar este pin.

All necessary pins can thus be placed quickly and without error. Their aspect is in accordance with corresponding hierarchical labels.

Conexiones - Etiquetas de Jerarquía

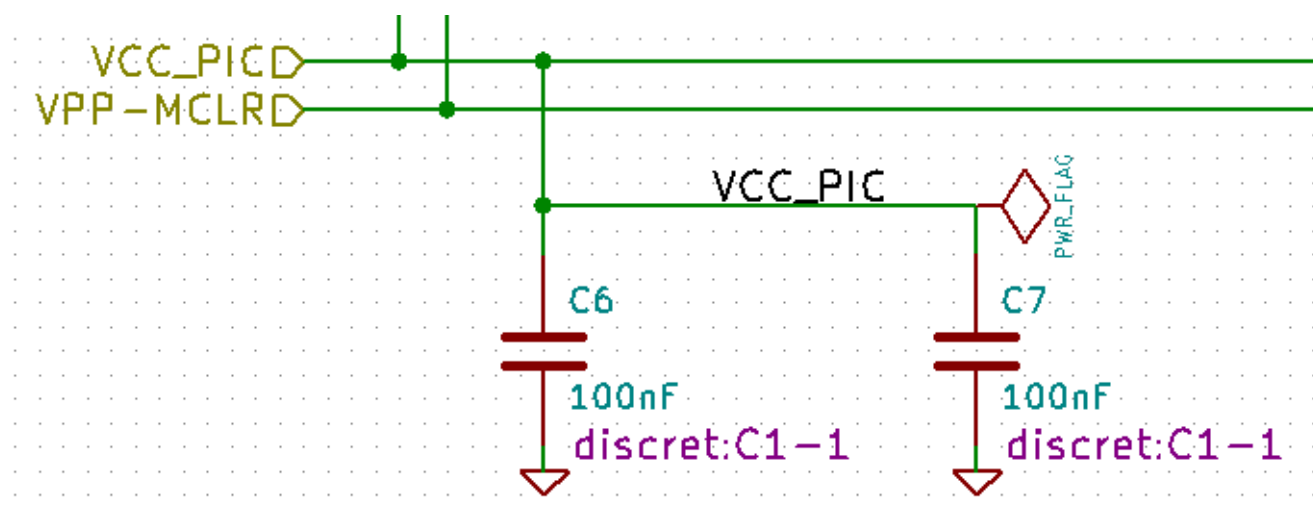
Each pin of the sheet symbol just created, must correspond to a label called hierarchical Label in the sub-sheet. Hierarchical labels are similar to labels, but they provide connections between sub-sheet and root sheet. The graphical representation of the two complementary labels (pin and hierarchical labels) is similar. Hierarchical labels are made with the tool .

Debajo se muestra un ejemplo de hoja padre:



Observe el pin VCC_PIC conectado al conector JP1

Aquí se muestra las conexiones correspondientes en la hoja hija:



Encontramos, de nuevo, las dos correspondientes etiquetas de jerarquía, que proveen la conexión entre las dos hojas jerárquicas.

NOTE

Puede usar etiquetas y pines de jerarquía para conectar dos buses, de acuerdo con la sintaxis (Bus[N..m]) anteriormente descrita.

Etiquetas, etiquetas jerárquicas, etiquetas globales y pines de potencia invisibles

A continuación se indican varios modos de proveer conexión, además de las conexiones mediante hilos.

Etiquetas simples

Las etiquetas simples tienen capacidad de conexión local, limitada a la hoja del esquema donde están ubicadas. Esto es debido a que:

- Cada hoja tiene un número de hoja.
- Este número de hoja es asociado a las etiquetas.

Entonces, si coloca la etiqueta "TOTO" en la hoja nº 3, en realidad el verdadero nombre de la etiqueta es "TOTO_3". Si también coloca la etiqueta "TOTO" en la hoja nº 1 (hoja raíz) está colocando una etiqueta llamada "TOTO_1", diferente de "TOTO_3". Esto es siempre cierto, incluso si hay una sola hoja.

Etiquetas de Jerarquía

Todo lo dicho para las etiquetas simples es también cierto para las etiquetas jerárquicas.

Thus in the same sheet, a hierarchical label "TOTO" is considered to be connected to a local label "TOTO", but not connected to a hierarchical label or label called "TOTO" in another sheet.

A hierarchical label is considered to be connected to the corresponding sheet pin symbol in the hierarchical symbol placed in the parent sheet.

Pines de potencia invisibles

It was seen that invisible power pins were connected together if they have the same name. Thus all the power pins declared "Invisible Power Pins" and named VCC are connected all symbol invisible power pins named VCC only within the sheet they are placed.

Esto significa que si coloca una etiqueta VCC en una hoja hija, ésta no estará conectada a los pines VCC ya que la etiqueta es realmente VCC_n, donde n es el número de hoja.

If you want this label VCC to be really connected to the VCC for the entire schematic, it will have to be explicitly connected to an invisible power pin via a VCC power symbol.

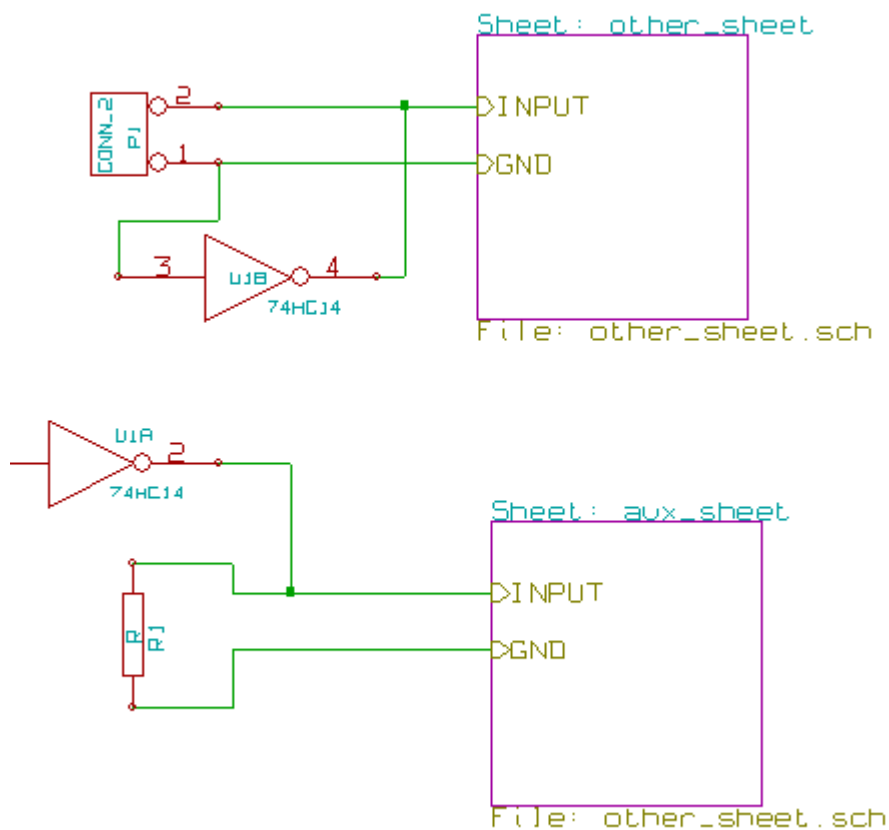
Etiquetas globales

Las etiquetas globales que tienen idéntico nombre están conectadas a lo largo de la jerarquía completa.

(Las etiquetas de potencia como vcc ... son etiquetas globales)

Jerarquías Complejas

Here is an example. The same schematic is used twice (two instances). The two sheets share the same schematic because the file name is the same for the two sheets (`other_sheet.sch`). The sheet names must be unique.

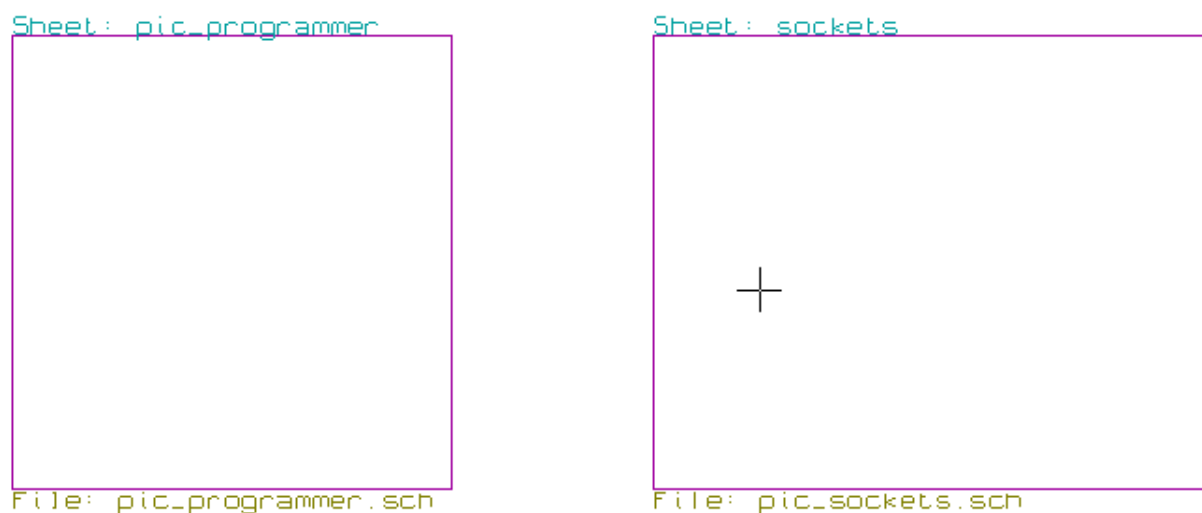


Jerarquías Planas

You can create a project using many sheets without creating connections between these sheets (flat hierarchy) if the following rules are observed:

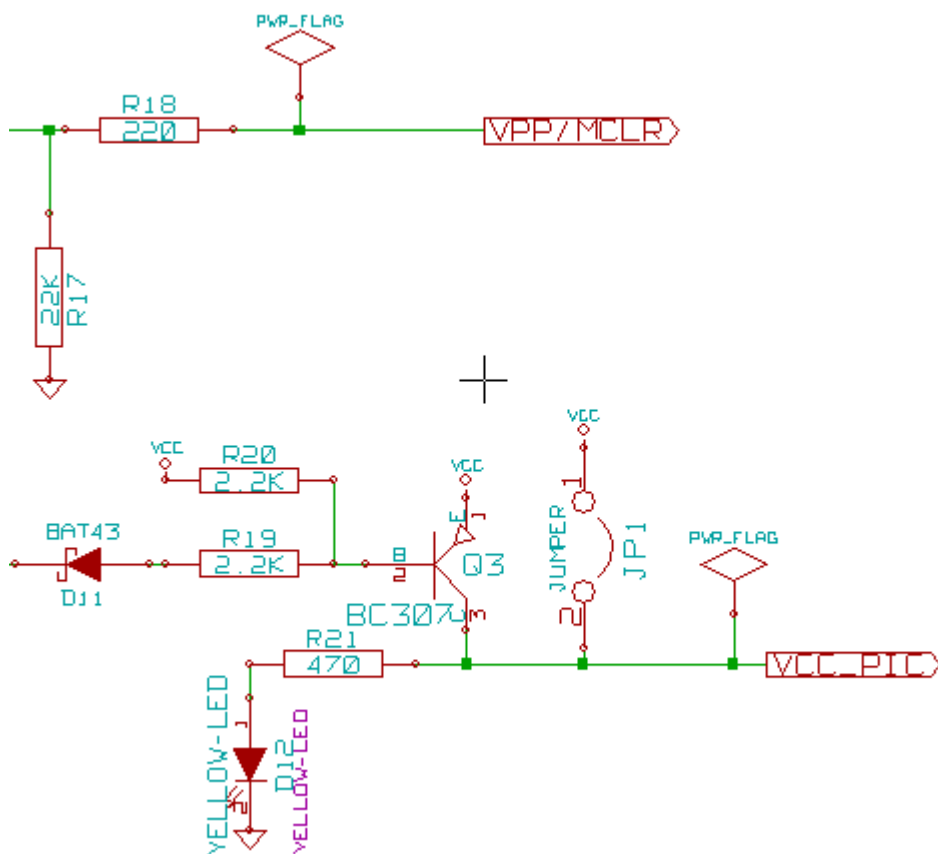
- Create a root sheet containing the other sheets which acts as a link between others sheets.
- No se necesitan conexiones explícitas.
- Use global labels instead of hierarchical labels in all sheets.

Aquí se muestra un ejemplo de la hoja padre.

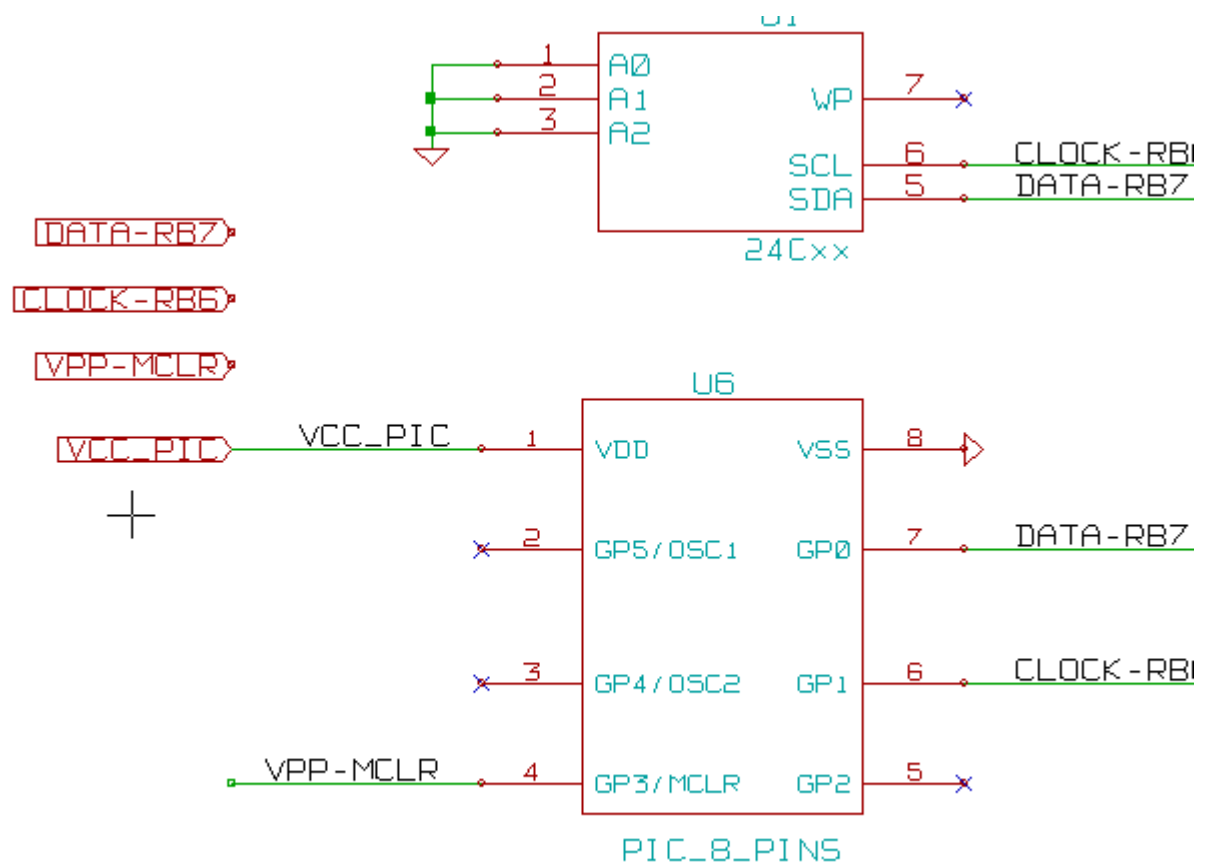


Estas son las dos paginas, conectadas por etiquetas globales.

Esta es la hoja pic_programmer.sch.



Y esta pic_sockets.sch.



Observe las etiquetas globales.


DATA-RB7

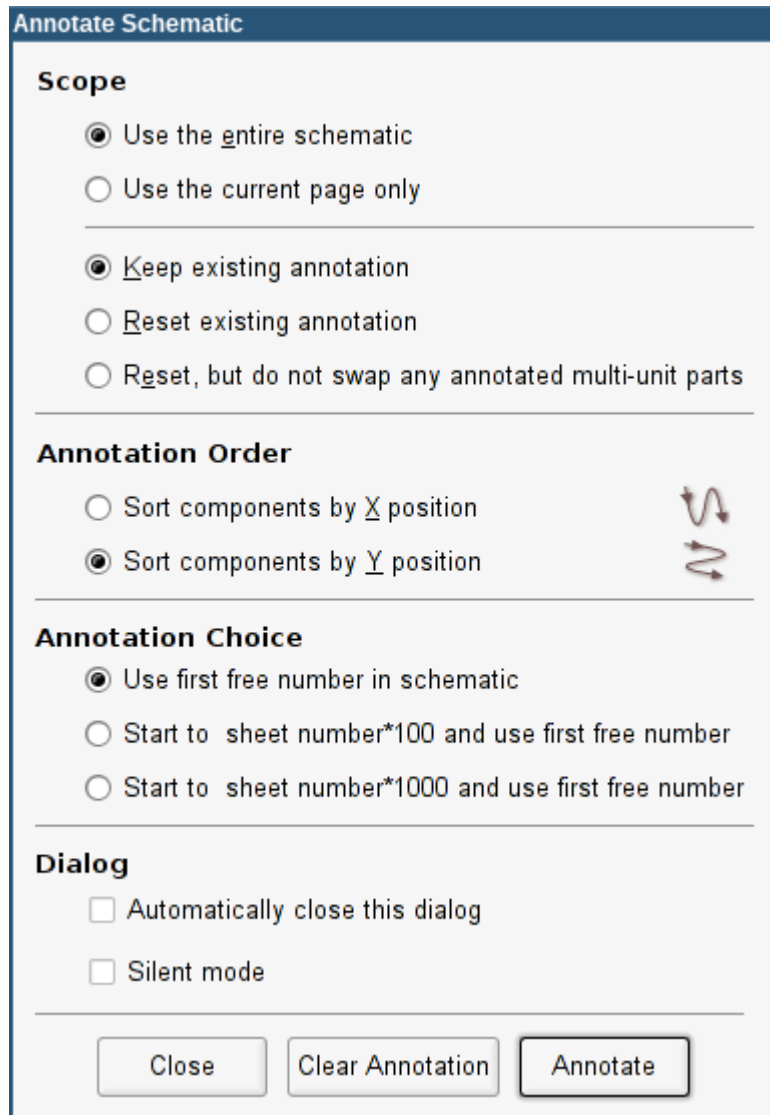
CLOCK-RB6

VPP-MCLR

Symbol Annotation Tool

Introducción

The annotation tool allows you to automatically assign a designator to symbols in your schematic. Annotation of symbols with multiple units will assign a unique suffix to minimize the number of these symbols. The annotation tool is accessible via the icon . Here you find its main window.



Annotate Schematic

Scope

☒ Use the entire schematic

☐ Use the current page only

☒ Keeep existing annotation

☐ Reset existing annotation

☐ Reset, but do not swap any annotated multi-unit parts

Annotation Order

☐ Sort components by X position

☒ Sort components by Y position

Annotation Choice

☒ Use first free number in schematic

☐ Start to sheet number*100 and use first free number

☐ Start to sheet number*1000 and use first free number

Dialog

☐ Automatically close this dialog

☐ Silent mode

Available annotation schemes:

- Annotate all the symbols (reset existing annotation option)
- Annotate all the symbols, but do not swap any previously annotated multi-unit parts.
- Annotate only symbols that are currently not annotated. Symbols that are not annotated will have a designator which ends with a '?' character.
- Anotar toda la arquitectura (use la opción: usar esquema completo)
- Anotar solo la hoja actual (use la opción: solo página actual)

The ``Reset, but do not swap any annotated multi-unit parts" option keeps all existing associations between symbols with multiple units. For example, U2A and U2B may be reannotated to U1A and U1B respectively but they will never be reannotated to U1A and U2A, nor to U2B and U2A. This is useful if you want to ensure that pin groupings are maintained.

La elección del orden de anotación determina el método usado para fijar los números de referencias dentro de cada hoja de la arquitectura.

Exceptuando casos particulares, una anotación automática se aplica al proyecto completo (todas las hojas) y a todos los componentes nuevos, si no desea cambiar las anotaciones realizadas previamente.

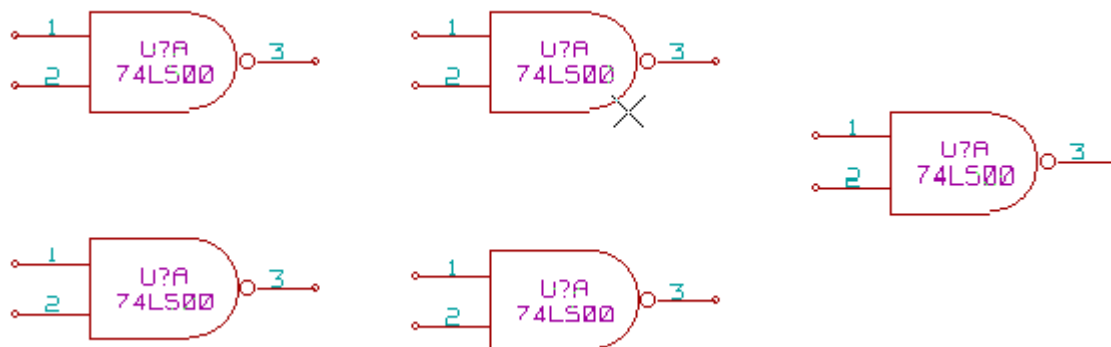
The Annotation Choice gives the method used to calculate reference:

- Use first free number in schematic: components are annotated from 1 (for each reference prefix). If a previous annotation exists, only unused numbers will be used.
- Comenzar por el número de hoja*100 y usar el primer número libre: la anotación comienza por 101 para la hoja 1, 201 para la hoja 2, etc. Si existen mas de 99 elementos con el mismo prefijo (U, R) dentro de la hoja 1, la herramienta de anotado utiliza los números 200 y sucesivos, y el referenciado de la hoja 2 comenzara por el siguiente número libre.
- Comenzar por el número de hoja*1000 y usar el primer número libre: La anotación comienza por 1001 para la hoja 1, 2001 para la hoja 2.

Algunos ejemplos

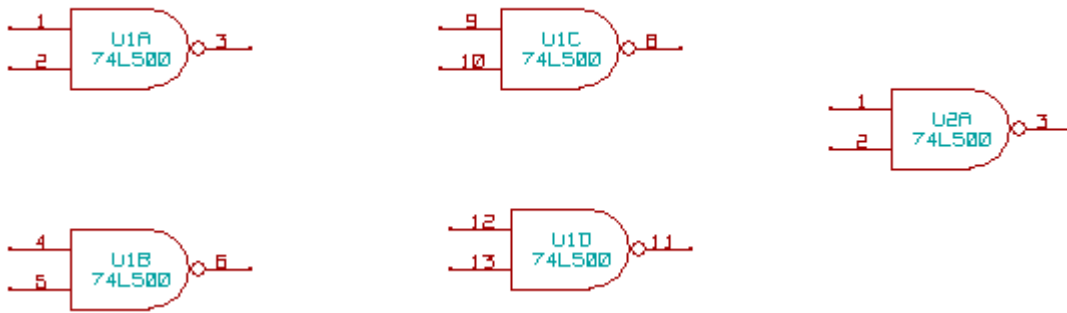
Orden de anotación

Este ejemplo muestra 5 elementos ubicados pero no anotados.

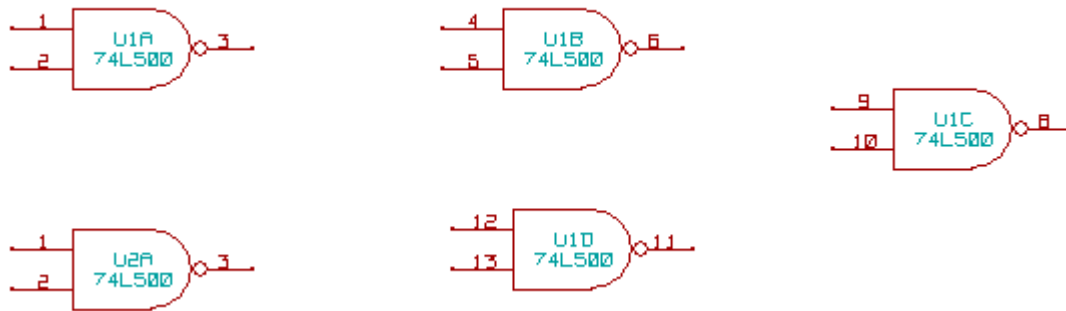


Después de ejecutar la anotación, se obtiene el siguiente resultado.

Ordenar con respecto a la posición X.



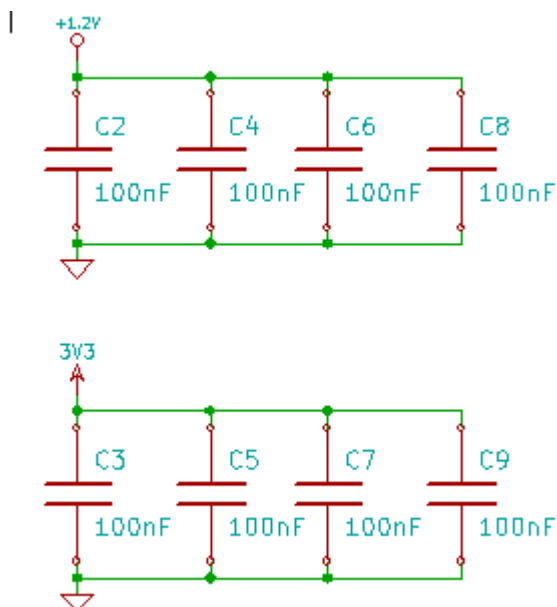
Ordenar con respecto a la posición Y.



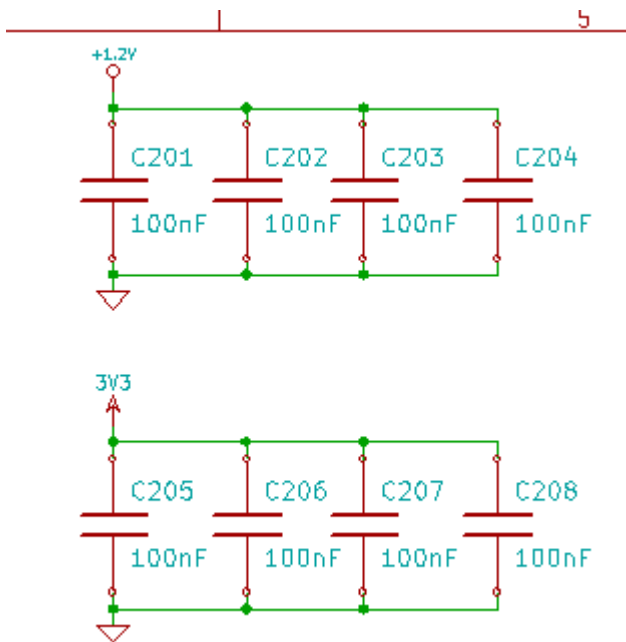
Puede observarse que cuatro puertas 74LS00 fueron distribuidas en el encapsulado U1, y que la quinta puerta 74LS00 ha sido asignada al siguiente encapsulado, U2.

Preferencias de anotado

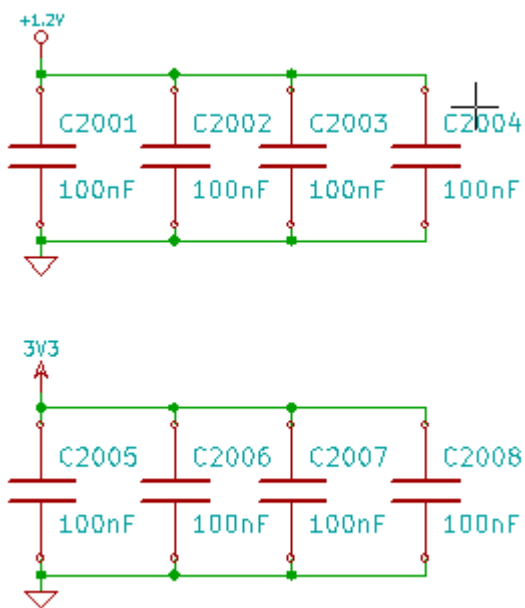
Esta es una anotación en la hoja 2 donde fue usada la opción usar primer número libre en el esquema.



La opción comenzar por el número de hoja*100 y usar el primer número libre arrojó el siguiente resultado.



La opción comenzar por el número de hoja*1000 y usar el primer número libre produjo el siguiente resultado.

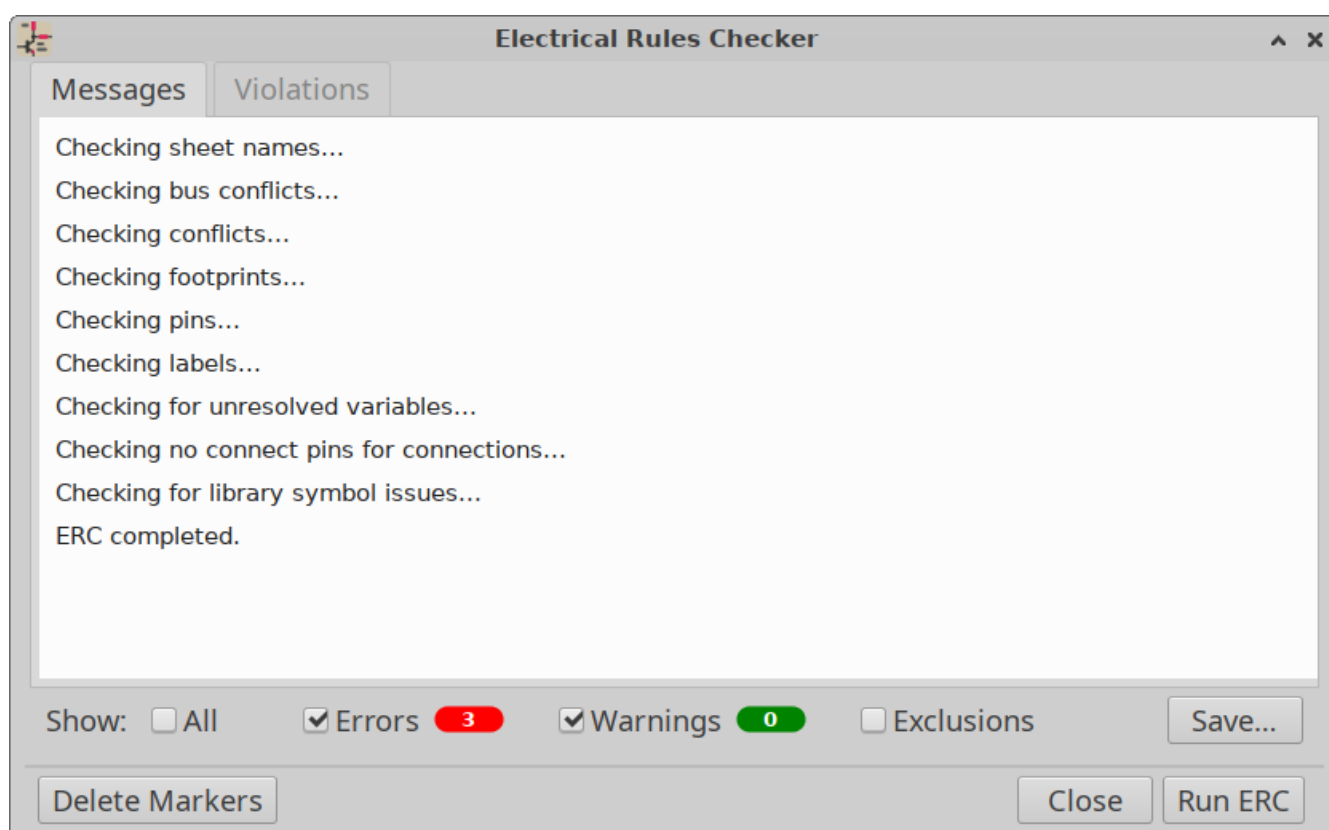


Verificación del diseño con el Comprobador de Reglas Eléctricas.

Introducción

El Comprobador de Reglas Eléctricas (ERC) realiza un test automático de su esquema. El ERC examina cualquier error en su hoja, como pines no conectados, símbolos de las jerarquías no conectados, salidas cortocircuitadas, etc. Naturalmente, un test automático no es infalible, y el software que hace posible detectar todos los errores de diseño no esta 100% completo. Este test es muy útil, ya que le permite detectar muchos descuidos y pequeños errores.

In fact all detected errors must be checked and then corrected before proceeding as normal. The quality of the ERC is directly related to the care taken in declaring electrical pin properties during symbol library creation. ERC output is reported as `errors''` or `warnings''`.



Como usar el ERC

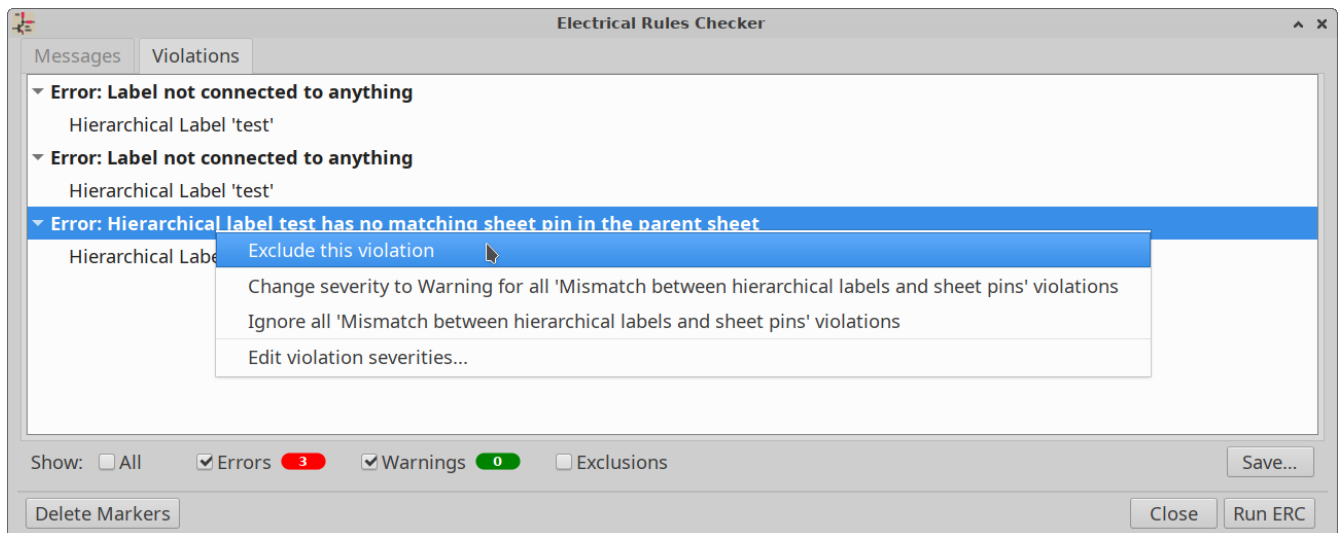
ERC can be started by clicking on the icon .

Los avisos son indicados en los elementos del esquema que generan un error ERC (pines o etiquetas)

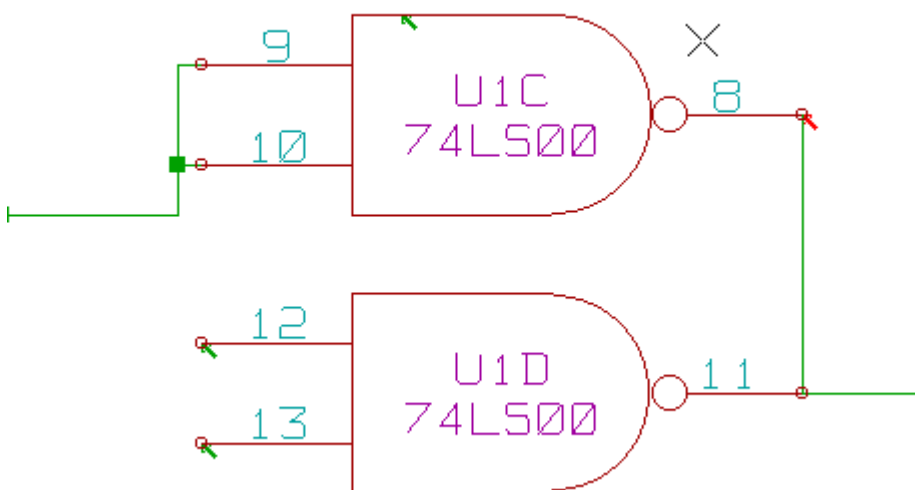
NOTE

- En esta ventana, cuando se hace clic sobre un mensaje de error puede saltar a la correspondiente marca en el esquema.
- En el esquema haga clic derecho sobre una marca para acceder al correspondiente mensaje de diagnóstico.

You can also delete error markers from the dialog and set specific ERC messages to be suppressed by using the right-click context menu.



Ejemplo de ERC

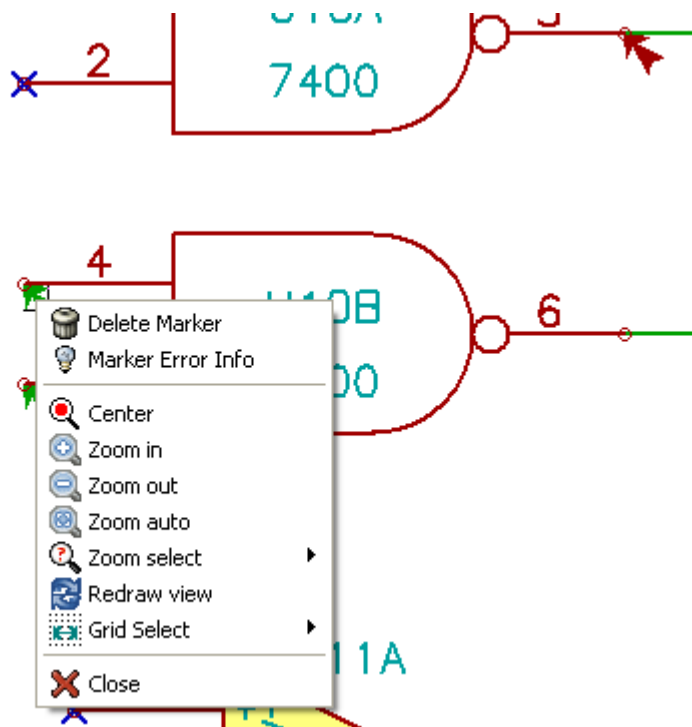


Se pueden observar cuatro errores:

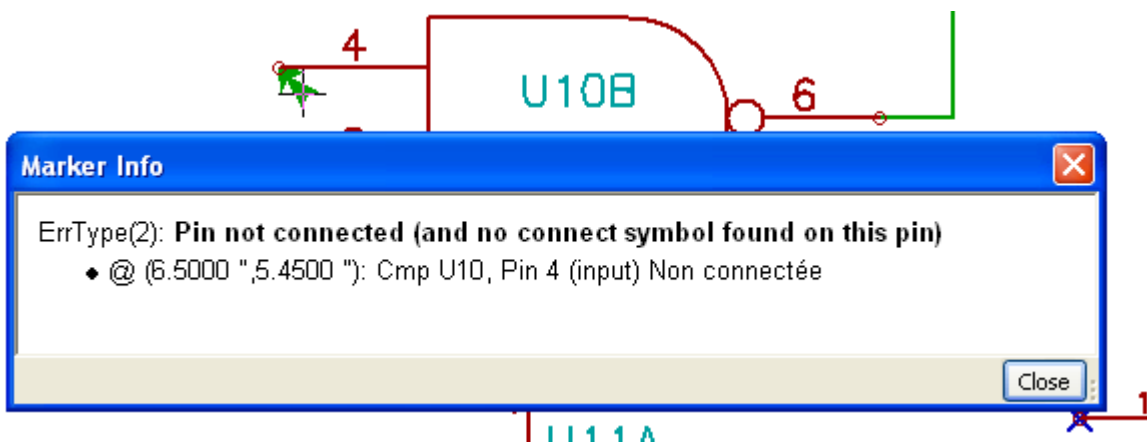
- Dos salidas han sido erróneamente conectadas juntas (flecha roja)
- Dos entradas se han dejado sin conectar (flecha verde)
- Hay un error en un puerto de alimentación invisible, no se encuentra el indicador de alimentación (flecha verde en la parte superior)

Mostrando diagnósticos

Haciendo clic con el botón derecho sobre una marca se despliega un menú que le permite acceder a la ventana de diagnóstico de marcas ERC.



y haciendo clic en "Información de la Marca de Error" puede obtener una descripción del error.

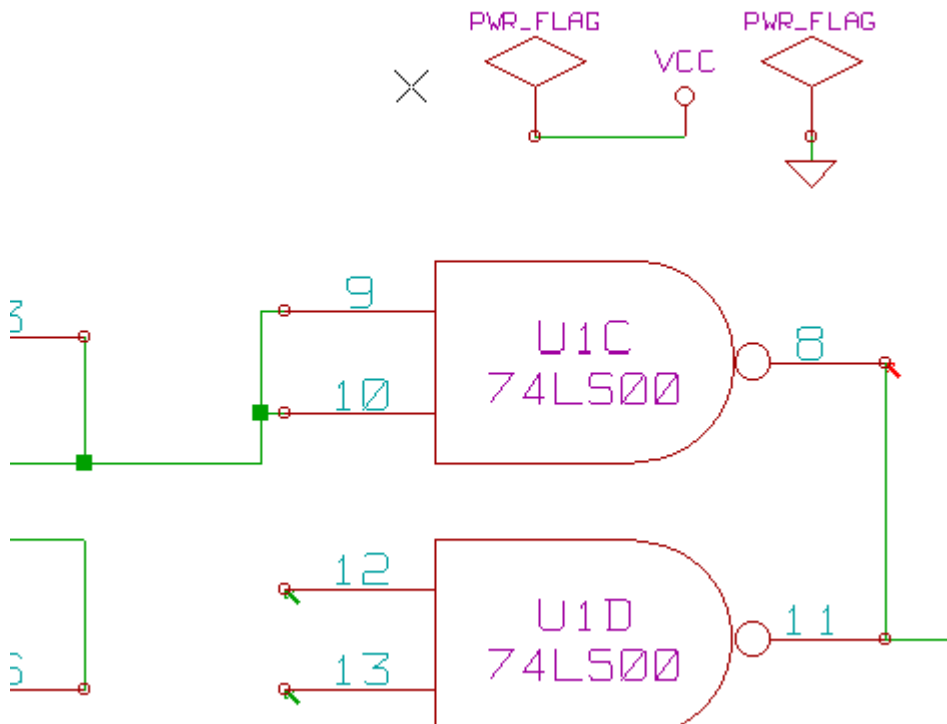


Pines de potencia e indicadores de potencia

Es usual tener un error o aviso en los pines de potencia, aunque todo parezca norma. Mire el ejemplo anterior. Esto sucede porque, en la mayoría de diseños, la alimentación es suministrada mediante conectores que no son fuentes de alimentación (como las salidas de reguladores, que están declaradas como salidas de potencia)

El ERC no detecta ningún pin de salida de potencia alimentando este hilo y lo declara como "no alimentado por una fuente de potencia"

Para evitar este aviso tiene que colocar un "PWR_FLAG" sobre estos puertos de potencia. Eche un vistazo al siguiente ejemplo:

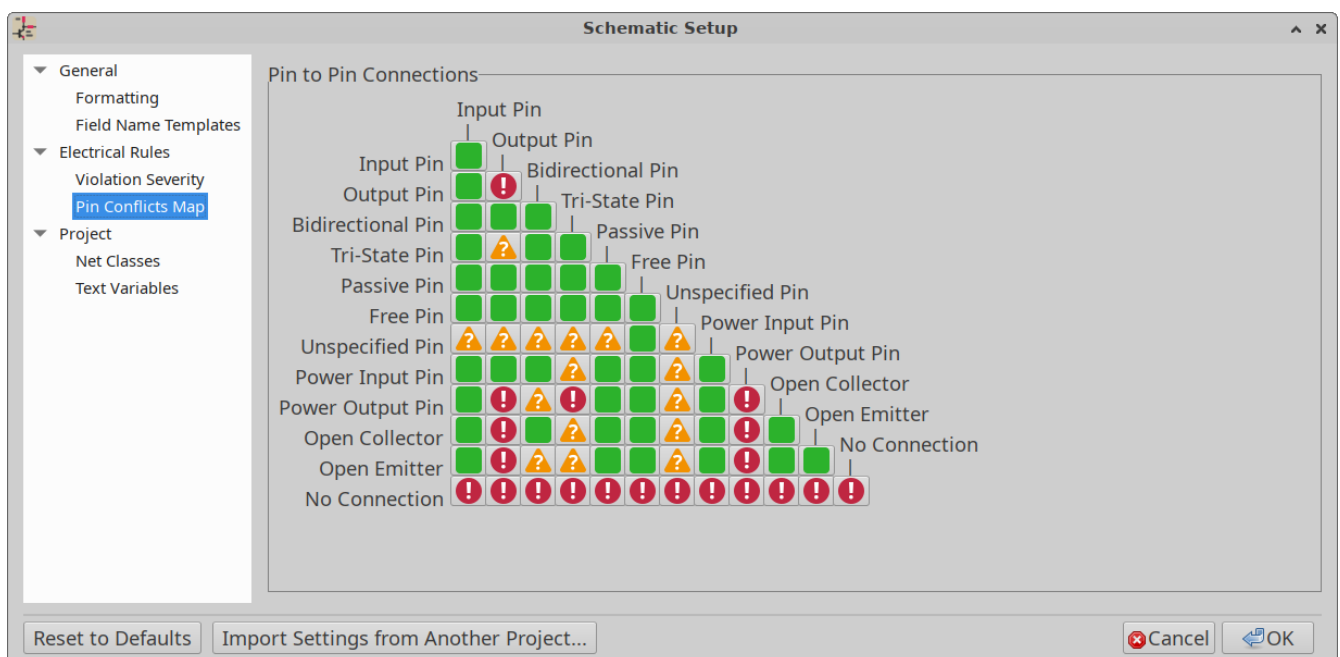


La marca de error ha desaparecido.

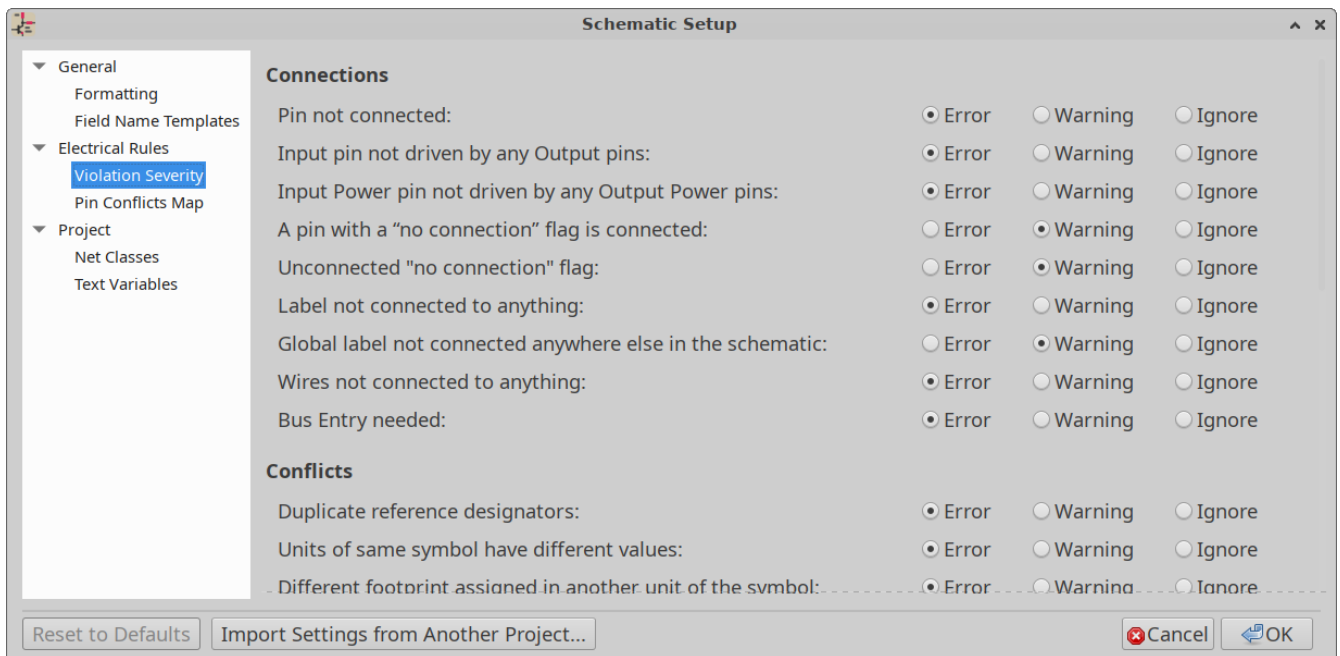
Most of the time, a PWR_FLAG must be connected to GND, because regulators have outputs declared as power out, but ground pins are never power out (the normal attribute is power in), so grounds never appear connected to a power source without a power flag symbol.

Configuración

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



Las reglas se pueden cambiar haciendo clic sobre el cuadrado oportuno de la matriz, esto alterna su estado cíclicamente entre: normal, aviso, error.



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

Archivo de reporte ERC

An ERC report file can be generated and saved by checking the option Write ERC report. The file extension for ERC report files is .erc. Here is an example ERC report file.

```
ERC control (4/1/1997-14:16:4)

***** Sheet 1 (INTERFACE UNIVERSAL)
ERC: Warning Pin input Unconnected @ 8.450, 2.350
ERC: Warning passive Pin Unconnected @ 8.450, 1.950
ERC: Warning: BiDir Pin connected to power Pin (Net 6) @ 10.100, 3.300
ERC: Warning: Power Pin connected to BiDir Pin (Net 6) @ 4.950, 1.400

>> Errors ERC: 4
```

Creación del Netlist

Resumen

A netlist is a file which describes electrical connections between symbols. These connections are referred to as nets. In the netlist file you can find:

- The list of the symbols
- The list of connections (nets) between symbols.

Many different netlist formats exist. Sometimes the symbols list and the list of nets are two separate files. This netlist is fundamental in the use of schematic capture software, because the netlist is the link with other electronic CAD software such as:

- PCB layout software.
- Schematic and electrical signal simulators.
- Compiladores de CPLD (y otros Circuitos Integrados programables).

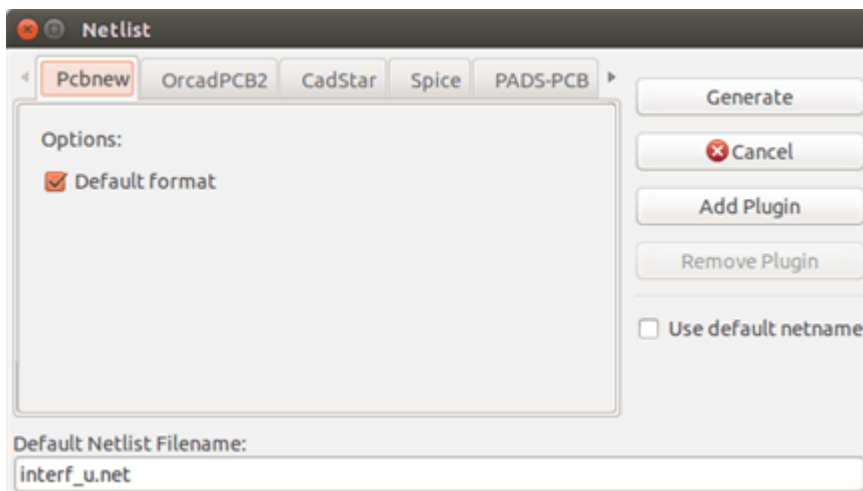
The KiCad Schematic Editor supports several netlist formats.

- Formato PCBNEW (circuito impreso).
- Formato ORCAD PCB2 (circuito impreso)
- Formato CADSTAR (circuito impreso)
- Formato Spice, para simuladores Spice (el formato Spice es usado también por otros simuladores).

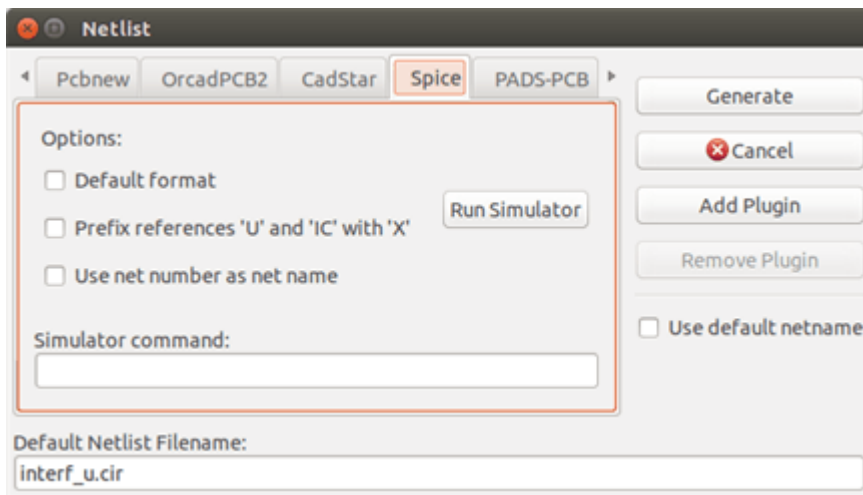
Formatos de Netlist

Select the tool  Netlist icon to open the netlist creation dialog.

Seleccione Pcbnew



Seleccione Spice

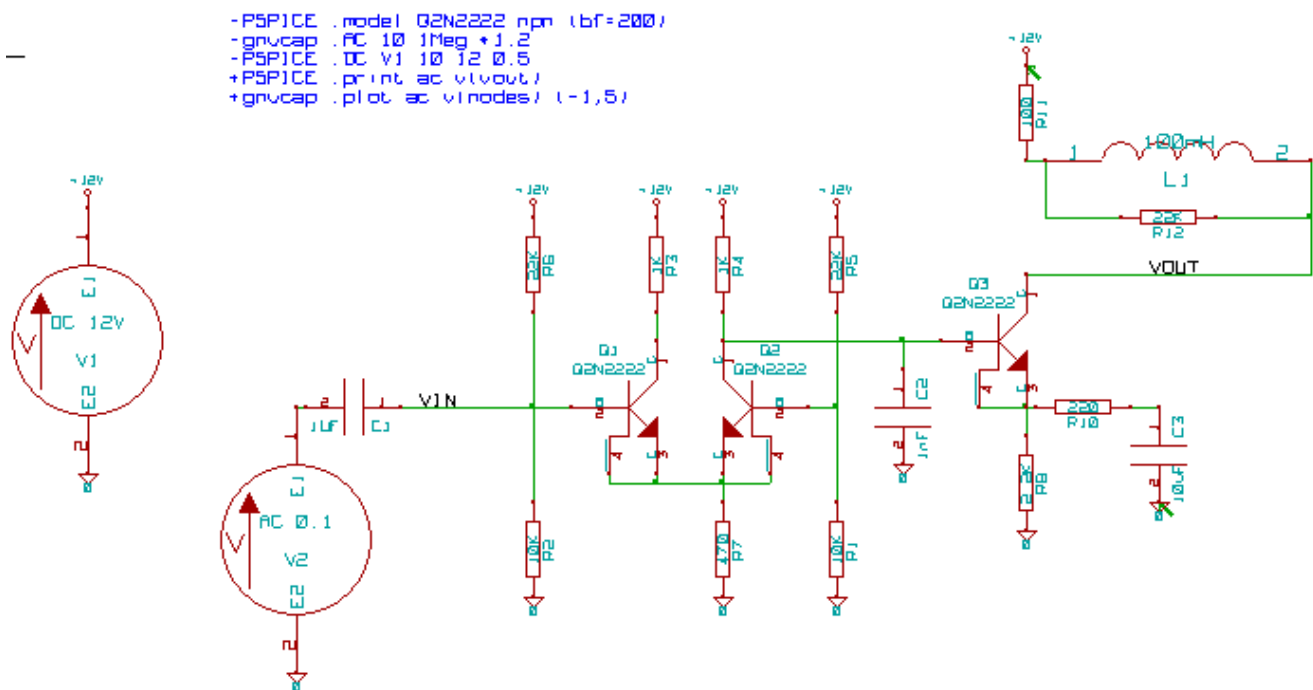


Using the different tabs you can select the desired format. In Spice format you can generate netlists with either net names which makes the SPICE file more human readable or net numbers which are used by older Spice. By clicking the Netlist button, you will be asked for a netlist file name.

NOTE | The netlist generation can take up to several minutes for large schematics.

Ejemplos de Netlist

Puede ver debajo un esquema diseñado usando la biblioteca PSPICE:



Ejemplo de un fichero de netlist tipo PCBNEW:

```

# Eeschema Netlist Version 1.0 genereee le 21/1/1997-16:51:15
(
(32E35B76 $noname C2 1NF {Lib=C}
(1 0)
(2 VOUT_1)
)
(32CFC454 $noname V2 AC_0.1 {Lib=VSOURCE}
(1 N-000003)
(2 0)
)
(32CFC413 $noname C1 1UF {Lib=C}
(1 INPUT_1)
(2 N-000003)
)
(32CFC337 $noname V1 DC_12V {Lib=VSOURCE}
(1 +12V)
(2 0)
)
(32CFC293 $noname R2 10K {Lib=R}
(1 INPUT_1)
(2 0)
)
(32CFC288 $noname R6 22K {Lib=R}
(1 +12V)
(2 INPUT_1)
)
(32CFC27F $noname R5 22K {Lib=R}
(1 +12V)
(2 N-000008)
)
(32CFC277 $noname R1 10K {Lib=R}
(1 N-000008)
(2 0)
)
(32CFC25A $noname R7 470 {Lib=R}
(1 EMET_1)
(2 0)
)
(32CFC254 $noname R4 1K {Lib=R}
(1 +12V)
(2 VOUT_1)
)
(32CFC24C $noname R3 1K {Lib=R}
(1 +12V)
(2 N-000006)
)
(32CFC230 $noname Q2 Q2N2222 {Lib=NPN}
(1 VOUT_1)
(2 N-000008)
(3 EMET_1)
)
(32CFC227 $noname Q1 Q2N2222 {Lib=NPN}
(1 N-000006)
(2 INPUT_1)
(3 EMET_1)
)
)
# End

```

En formato PSPICE, el netlist es como sigue:

```
* Eeschema Netlist Version 1.1 (Spice format) creation date: 18/6/2008-08:38:03

.model Q2N2222 npn (bf=200)
.AC 10 1Meg \*1.2
.DC V1 10 12 0.5


R12  /VOUT N-000003 22K
R11  +12V N-000003 100
L1   N-000003 /VOUT 100mH
R10  N-000005 N-000004 220
C3   N-000005 0 10uF
C2   N-000009 0 1nF
R8   N-000004 0 2.2K
Q3   /VOUT N-000009 N-000004 N-000004 Q2N2222
V2   N-000008 0 AC 0.1
C1   /VIN N-000008 1uF
V1   +12V 0 DC 12V
R2   /VIN 0 10K
R6   +12V /VIN 22K
R5   +12V N-000012 22K
R1   N-000012 0 10K
R7   N-000007 0 470
R4   +12V N-000009 1K
R3   +12V N-000010 1K
Q2   N-000009 N-000012 N-000007 N-000007 Q2N2222
Q1   N-000010 /VIN N-000007 N-000007 Q2N2222


.print ac v(vout)
.plot ac v(nodes) (-1,5)

.end
```

Notas en los Netlists

Precauciones en los nombres de los netlist

Many software tools that use netlists do not accept spaces in the component names, pins, nets or other informations. Avoid using spaces in labels, or names and value fields of components or their pins to ensure maximum compatibility.

In the same way, special characters other than letters and numbers can cause problems. Note that this limitation is not related to KiCad, but to the netlist formats that can then become untranslatable to software that uses netlist files.

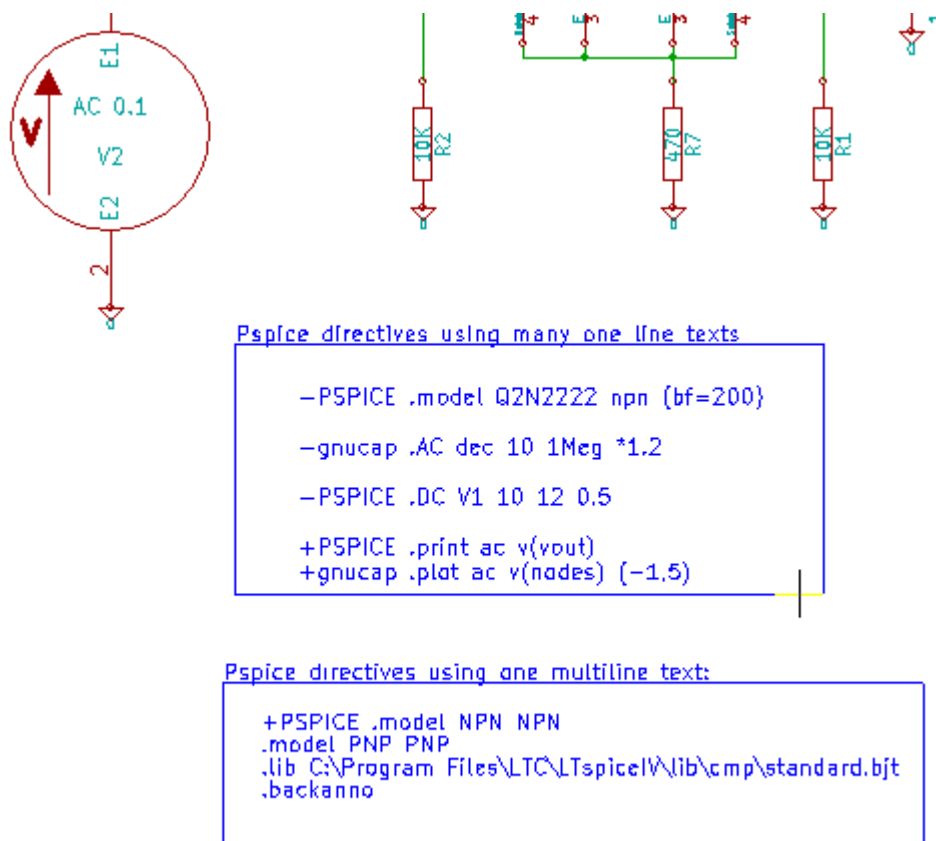
Netlist PSPICE

Para el simulador Pspice, tiene que incluir algunas lineas de comando en el propio netlist (.PROBE, .AC, etc.).

Cualquier linea de texto incluida en el diagrama de esquema comenzando con la clave **-pspice** o **-gnucap** será insertada (sin la clave) al inicio del netlist.

Cualquier línea de texto incluida en el diagrama de esquema comenzando con la clave **+pspice** o **+gnuicap** será insertada (sin la clave) al inicio del netlist.

Aquí se muestra un ejemplo usando varias textos de una línea y textos multi-línea.



Por ejemplo, si escribes el siguiente texto (¡No use una etiqueta!):

```
-PSPICE .PROBE
```

una línea **.PROBE** se insertará en el netlist.

En el ejemplo anterior tres líneas fueron insertadas al inicio del netlist y dos al final con esta técnica.

Si está usando textos multilinea, las claves **+pspice** or **+gnuicap** serán necesarias solo una vez:

```
+PSPICE .model NPN NPN
.model PNP PNP
.lib C:\Program Files\LTC\LTspiceIV\lib\cmp\standard.bjt
.backanno
```

Crea las siguientes cuatro líneas:

```
.model NPN NPN
.model PNP PNP
.lib C:\Program Files\LTC\LTspiceIV\lib\cmp\standard.bjt
.backanno
```

Also note that the GND net must be named 0 (zero) for Pspice.

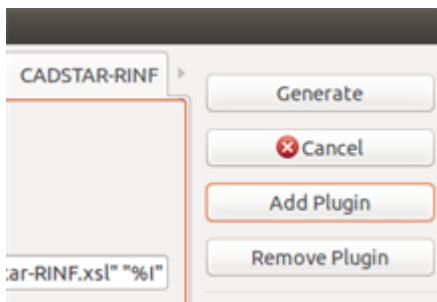
Otros Formatos

For other netlist formats you can add netlist converters in the form of plugins. These converters are automatically launched by KiCad. Chapter 14 gives some explanations and examples of converters.

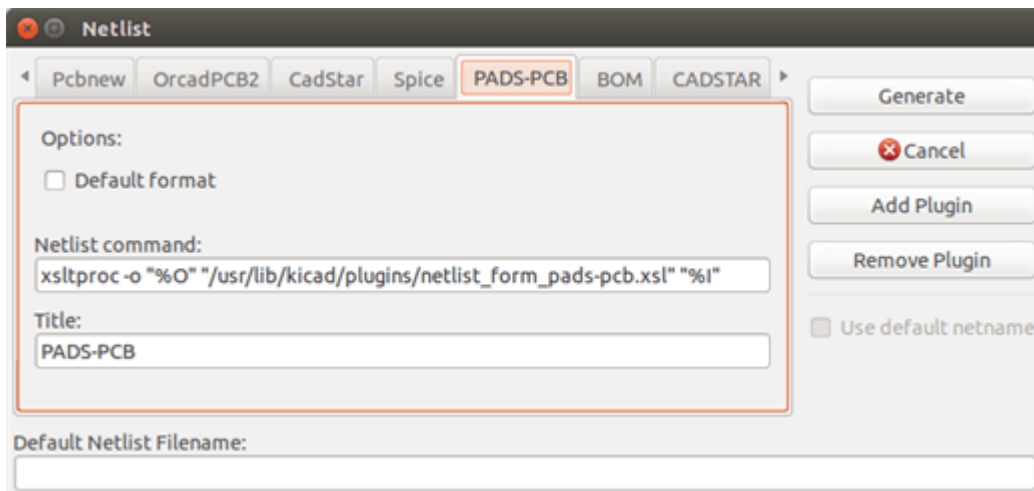
A converter is a text file (xsl format) but one can use other languages like Python. When using the xsl format, a tool (xsltproc.exe or xsltproc) read the intermediate file created by KiCad, and the converter file to create the output file. In this case, the converter file (a sheet style) is very small and very easy to write.

Inicie la ventana de diálogo

Puede añadir un nuevo plugin de netlist mediante el botón Añadir Plugin.



Aquí se muestra la ventana de ajuste del plugin PadsPcb:



La configuración requerirá:

- Un título (por ejemplo, el nombre del formato de netlist)
- El plug-in a lanzar.

Cuando el netlist es generado:

1. KiCad creates an intermediate file *.tmp, for example test.tmp.
2. KiCad runs the plug-in, which reads test.tmp and creates test.net.

Formato de Línea de Comando

Aquí se muestra un ejemplo usando xsltproc.exe como herramienta para convertir archivos .xsl, y un archivo metlist_from_pads-pcb.xsl como hoja de estilo conversora:

f:/kicad/bin/xsltproc.exe -o %O.net f:/kicad/bin/plugins/netlist_form_pads-pcb.xsl %I

Con:

f:/kicad/bin/xsltproc.exe	A tool to read and convert xsl file
-o %O.net	Output file: %O will define the output file.
f:/kicad/bin/plugins/netlist_form_pads-pcb.xsl	File name converter (a sheet style, xsl format).
%I	Will be replaced by the intermediate file created by KiCad (*.tmp).

Para un fichero de esquema llamado test.sch, la línea de comando actual es:

f:/kicad/bin/xsltproc.exe -o test.net f:/kicad/bin/plugins/netlist_form_pads-pcb.xsl test.tmp.

Conversor y Hoja de Estilo (plug-in)

Este es un software muy simple porque su propósito es simplemente convertir un fichero de entrada (el fichero de texto intermedio) a otro fichero de texto. Además, desde el fichero de texto intermedio, se puede crear una lista de materiales (BOM).

Cuando se usa xsltproc como herramienta de conversión solo será generado el fichero tipo hoja.

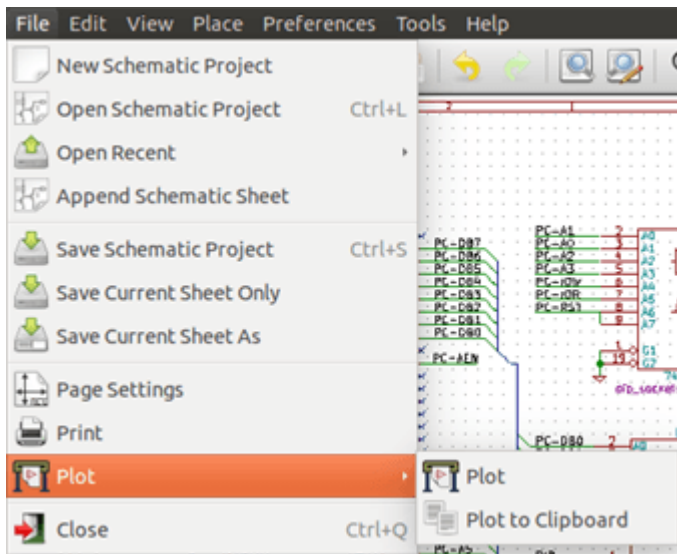
Formatos de fichero netlist intermedios

Vea el capítulo 14 para más explicaciones sobre xsltproc, descripciones del formato de archivo intermedio, y algunos ejemplos de conversores de estilo hoja.

Plot e Imprimir

Introducción

Puede acceder a ambos comandos mediante del menú archivo.



Los formatos de salida soportados son Postscript, PDF, SVG, DXF and HPGL. Puede además imprimir directamente a su impresora.

Comandos comunes de impresión

Imprimir página actual

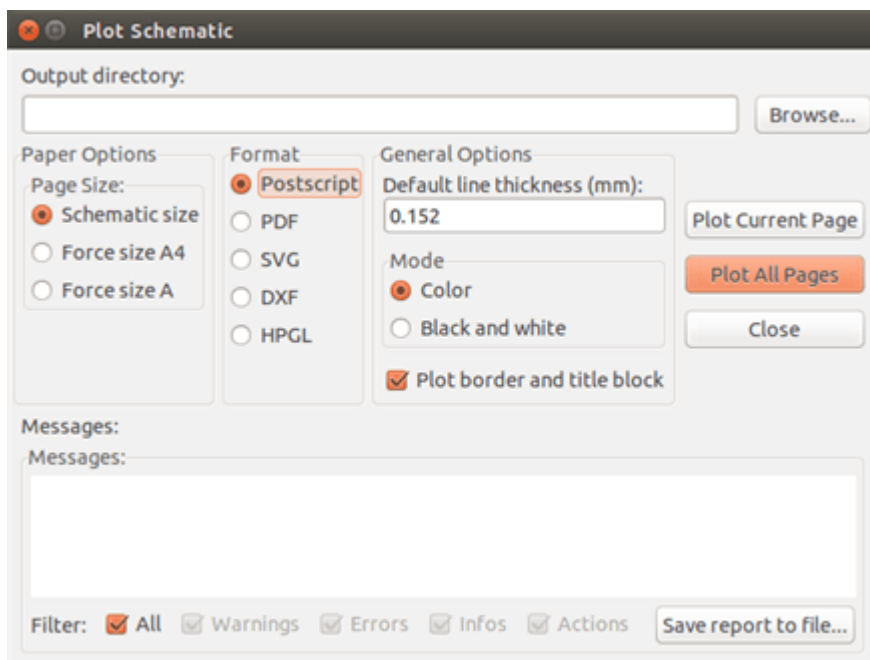
Imprime un fichero solo para la página actual.

Imprimir todas las páginas

Le permite imprimir la jerarquía completa (se genera un fichero para cada hoja)

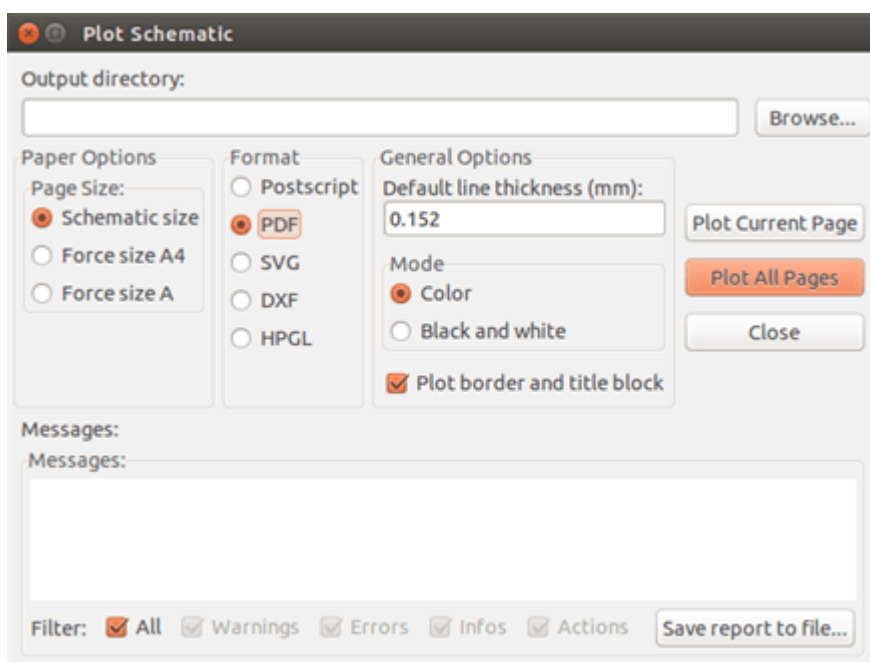
Imprimir en Postscript

Este comando le permite crear archivos PostScript.



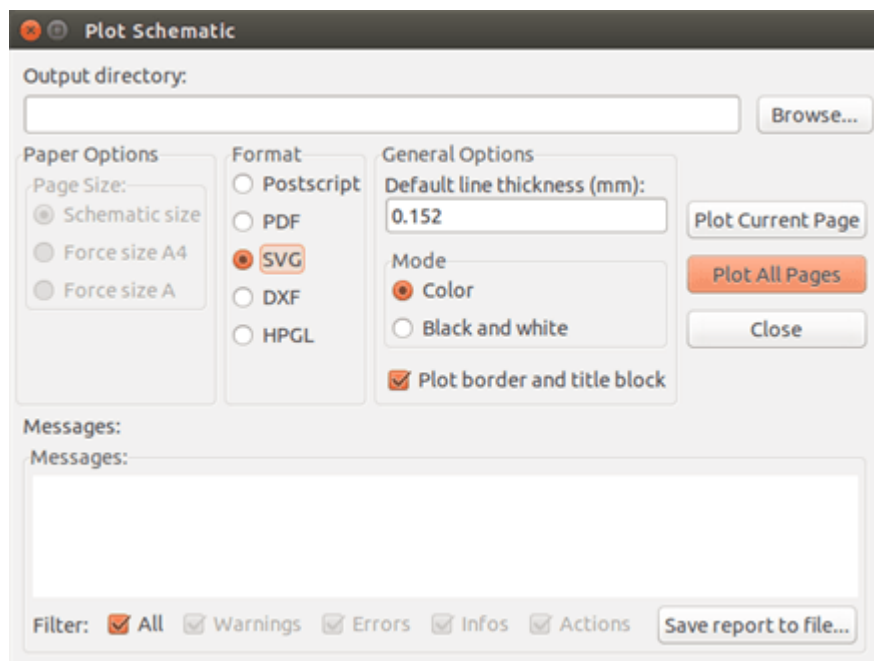
El nombre del archivo es el nombre de la hoja con la extensión .ps. Puede deshabilitar la opción "Plot border and title block". Esto es útil si quiere crear un fichero postscript para encapsular (formato .eps) un diagrama en un software de procesamiento de texto. La ventana de mensaje muestra el nombre de archivo creado.

Imprimir en PDF



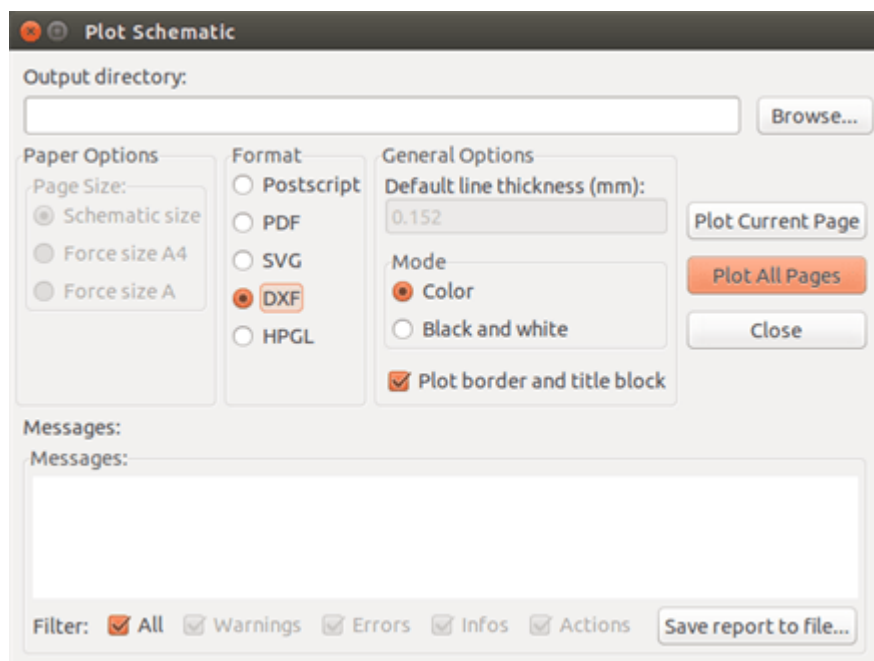
Le permite crear fichero de impresión usando el formato PDF. El nombre del archivo es el nombre de la hoja con la extensión .pdf.

Imprimir en SVG



Le permite crear fichero de impresión usando el formato vectorial SVG. El nombre del fichero es el nombre de la hoja con la extensión .svg.

Imprimir en DXF



Le permite crear un fichero de impresión usando el formato DXF. El nombre del fichero es el nombre de la hoja con extensión .dxf.

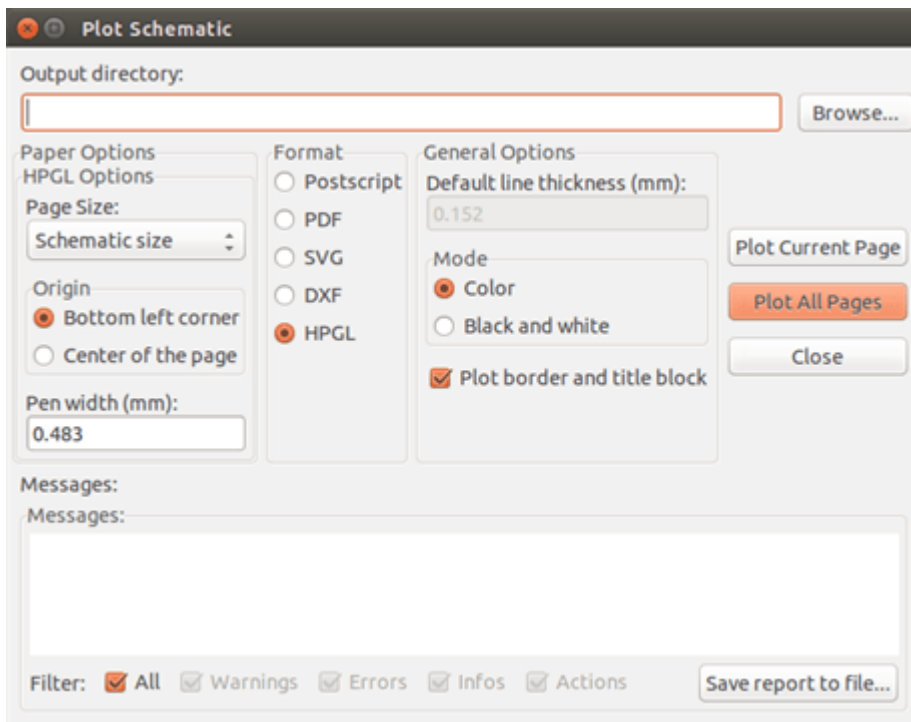
Imprimir en HPGL

Este comando le permite crear un fichero HPGL. En este formato puede definir:

- Tamaño de la página.
- Origen.

- Ancho de las líneas (en mm).

La ventana de ajustes de ploteado es la siguiente:



El nombre del fichero de salida será el nombre de la hoja con extensión .plt.

Selección del tamaño de la hoja

Normalmente el tamaño de la hoja es comprobado. En este caso, se usará el tamaño de hoja definido en el menú del bloque y la escala elegida será 1. Si se selecciona un tamaño de hoja diferente (A4 con A0, o A con E), la escala se ajusta automáticamente al tamaño de la hoja.

Ajuste de offset

Para todas las dimensiones estándar, puede ajustar el offset para centrar el dibujo tan precisamente como sea posible. Dado que los plotters tienen el punto de origen en el centro o en la esquina inferior izquierda, es necesario que sea posible introducir un offset para imprimir adecuadamente.


Generalidades:

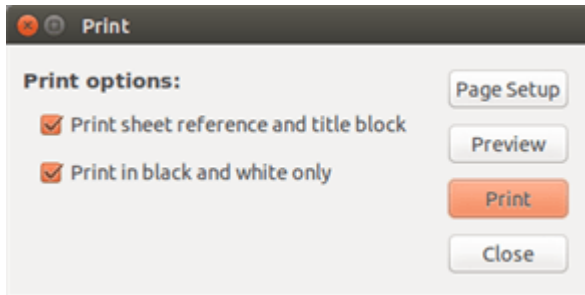
- Para plotters que tengan su punto de origen en el centro de la hoja el offset debe ser negativo y ajustarse a la mitad de las dimensiones de la hoja.
- Para plotters que tengan su punto de origen en la esquina inferior izquierda el offset debe ser ajustado a 0.

Ajustar el offset:

- Seleccionar el tamaño de la hoja.
- Ajustar el offset en X y el offset en Y.
- Pulsar en aceptar offset.

Imprimir en papel

This command, available via the icon , allows you to visualize and generate design files for the standard printer.



La opción "Imprimir referencia de hoja y bloque de título" habilita o deshabilita las referencias de la hoja y el bloque de título.

La opción "Imprimir en blanco y negro" ajusta la impresión monocromo. Esta opción normalmente es necesaria si se usa una impresora láser en blanco y negro, ya que los colores son impresos en tonos medios que normalmente no son tan legibles.

Symbol Editor

General Information About Symbol Libraries

A symbol is a schematic element which contains a graphical representation, electrical connections, and text fields describing the symbol. Symbols used in a schematic are stored in symbol libraries. KiCad provides a symbol editing tool that allows you to create libraries, add, delete or transfer symbols between libraries, export symbols to files, and import symbols from files. The symbol editing tool provides a simple way to manage symbols and symbol libraries.

Symbol Library Overview

A symbol library is composed of one or more symbols. Generally the symbols are logically grouped by function, type, and/or manufacturer.

A symbol is composed of:

- Graphical items (lines, circles, arcs, text, etc.) that determine how symbol looks in a schematic.
- Pins which have both graphic properties (line, clock, inverted, low level active, etc.) and electrical properties (input, output, bidirectional, etc.) used by the Electrical Rules Check (ERC) tool.
- Campos como referencias, valores, huella correspondiente para el diseño del PCB, etc.

Symbols can be derived from another symbol in the same library. Derived symbols share the base symbol's graphical shape and pin definitions, but can override the base symbol's property fields (value, footprint, footprint filters, datasheet, description, etc.). Derived symbols can be used to define symbols that are similar to a base part. For example, 74LS00, 74HC00, and 7437 symbols could all be derived from a 7400 symbol. In previous versions of KiCad, derived symbols were referred to as aliases.

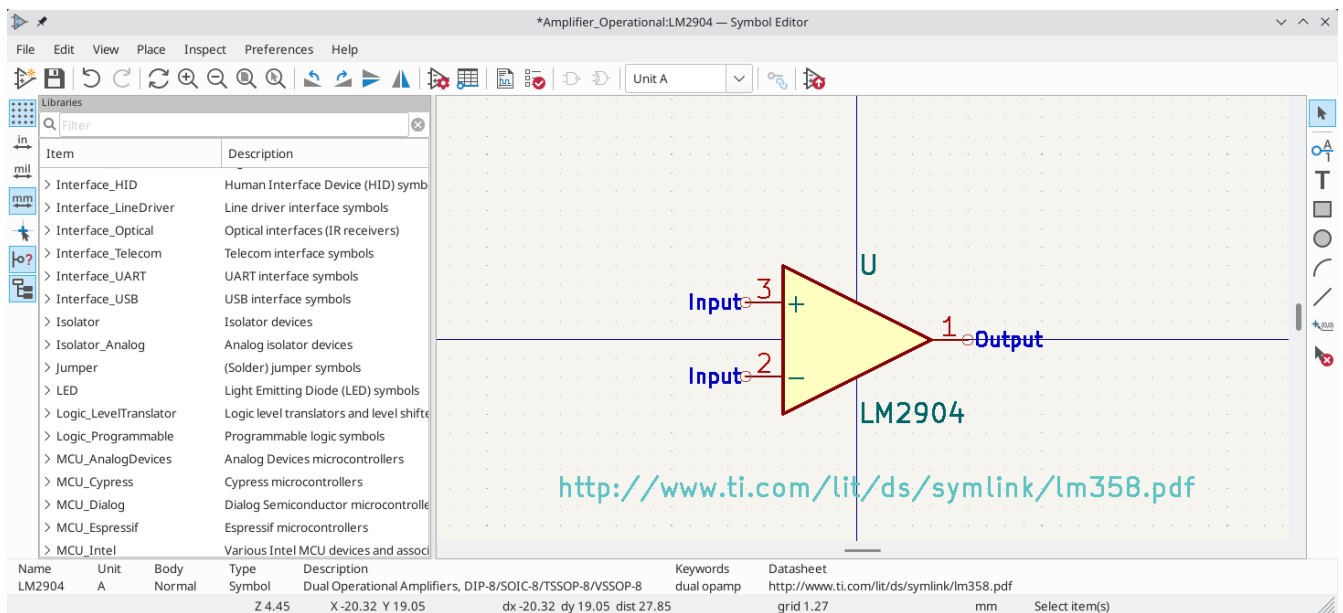
Proper symbol designing requires:

- Defining if the symbol is made up of one or more units.
- Defining if the symbol has an alternate body style (also known as a De Morgan representation).
- Diseñar su representación simbólica usando líneas, rectángulos, círculos, polígonos y texto.
- Añadir terminales definiendo cuidadosamente cada representación gráfica, su nombre, número y propiedades eléctricas (entrada, salida, tri-estado, alimentación, etc).
- Determining if the symbol should be derived from another symbol with the same graphical design and pin definition.
- Añadir campos opcionales como nombre de la huella usada por el programa de diseño de PCB y/o definiendo su visibilidad.
- Documenting the symbol by adding a description string and links to data sheets, etc.
- Salvar el componente en la biblioteca deseada.

Symbol Library Editor Overview

The symbol library editor main window is shown below. It consists of three tool bars for quick access to common features and a symbol viewing/editing area. Not all commands are available on the tool bars but






















can be accessed using the menus.



Barra de herramientas principal










The main tool bar is located at the top of the main window. It consists of the undo/redo commands, zoom commands, symbol properties dialogs, and unit/representation management controls.



	Create a new symbol in the selected library.
	Save the currently selected library. All modified symbols in the library will be saved.
	Undo last edit.
	Redo last undo.
	Refresh display.
	Zoom in.
	Zoom out.
	Zoom to fit symbol in display.
	Zoom to fit selection.
	Rotate counter-clockwise.
	Rotate clockwise.
	Mirror horizontally.
	Mirror vertically.
	Edit the current symbol properties.
	Edit the symbol's pins in a tabular interface.
	Open the symbol's datasheet. The button will be disabled if no datasheet is defined for the current symbol.
	Test the current symbol for design errors.
	Select the normal body style. The button is disabled if the current symbol does not have an alternate body style.
	Select the alternate body style. The button is disabled if the current symbol does not have an alternate body style.
	Select the unit to display. The drop down control will be disabled if the current symbol is not derived from a symbol with multiple units.
	Enable synchronized pins edit mode. When this mode is enabled, any pin modifications are propagated to all other symbol units. Pin number changes are not propagated. This mode is automatically enabled for symbols with multiple interchangeable units and cannot be enabled for symbols with only one unit.








Barra de herramientas de elementos

The vertical toolbar located on the right hand side of the main window allows you to place all of the elements required to design a symbol.


	Select tool. Right-clicking with the select tool opens the context menu for the object under the cursor. Left-clicking with the select tool displays the attributes of the object under the cursor in the message panel at the bottom of the main window. Double-left-clicking with the select tool will open the properties dialog for the object under the cursor.
	Pin tool. Left-click to add a new pin.
	Graphical text tool. Left-click to add a new graphical text item.
	Rectangle tool. Left-click to begin drawing the first corner of a graphical rectangle. Left-click again to place the opposite corner of the rectangle.
	Circle tool. Left-click to begin drawing a new graphical circle from the center. Left-click again to define the radius of the circle.
	Arc tool. Left-click to begin drawing a new graphical arc item from the first arc end point. Left-click again to define the second arc end point. Adjust the radius by dragging the arc center point.
	Connected line tool. Left-click to begin drawing a new graphical line item in the current symbol. Left-click for each additional connected line. Double-left-click to complete the line.
	Anchor tool. Left-click to set the anchor position of the symbol.
	Delete tool. Left-click to delete an object from the current symbol.

Barra de herramientas Opciones

The vertical tool bar located on the left hand side of the main window allows you to set some of the editor drawing options.


	Toggle grid visibility on and off.
	Set units to inches.
	Set units to mils (0.001 inch).
	Set units to millimeters.
	Toggle full screen cursor on and off.
	Toggle display of pin electrical types.
	Toggle display of libraries and symbols.

Selección y mantenimiento de bibliotecas

The selection of the current library is possible via the  icon which shows you all available libraries and allows you to select one. When a symbol is loaded or saved, it will be put in this library. The library name of a symbol is the contents of its `Value` field.

Select and Save a Symbol

Symbol Selection

Clicking the  icon on the left tool bar toggles the treeview of libraries and symbols. Clicking on a symbol opens that symbol.

NOTE

Some symbols are derived from other symbols. Derived symbol names are displayed in *italics* in the treeview. If a derived symbol is opened, its symbol graphics will not be editable. Its symbol fields will be editable as normal. To edit the graphics of a base symbol and all of its derived symbols, open the base symbol.

Save a Symbol

After modification, a symbol can be saved in the current library or a different library.

To save the modified symbol in the current library, click the  icon. The modifications will be written to the existing symbol.

NOTE


Saving a modified symbol also saves all other modified symbols in the same library.

To save the symbol changes to a new symbol, click **File** → **Save As....** The symbol can be saved in the current library or a different library. A new name can be set for the symbol.

To create a new file containing only the current symbol, click **File** → **Export** → **Symbol....** This file will be a standard library file which will contain only one symbol.

Creating Library Symbols

Create a New Symbol

A new symbol can be created by clicking the  icon. You will be asked for a number of symbol properties.

- A symbol name (this name is used as the default value for the `Value` field in the schematic editor)
- An optional base symbol to derive the new symbol from. The new symbol will use the base symbol's graphical shape and pin configuration, but other symbol information can be modified in the derived symbol. The base symbol must be in the same library as the new derived symbol.
- The reference designator prefix (U , C , R ...).
- The number of units per package, and whether those units are interchangeable (for example a 7400 is made of 4 units per package).
- If an alternate body style (sometimes referred to as a "De Morgan equivalent") is desired.
- Whether the symbol is a power symbol. Power symbols appear in the "Add Power Port" dialog in the Schematic editor, their `Value` fields are not editable in the schematic, they cannot be assigned a

footprint and they are not added to the PCB, and they are not included in the bill of materials.

- Whether the symbol should be excluded from the bill of materials.
- Whether the symbol should be excluded from the PCB.

There are also several graphical options.

- The offset between the end of each pin and its pin name.
- Whether the pin number and pin name should be displayed.
- Whether the pin names should be displayed alongside the pins or at the ends of the pins inside the symbol body.

These properties can also be changed later in the [Symbol Properties window](#).

New Symbol

Symbol name:

Derive from existing symbol: ▼

Default reference designator:

Number of units per package: - +

☐ Units are not interchangeable

☐ Create symbol with alternate body style (De Morgan)

☐ Create symbol as power symbol

☐ Exclude from schematic bill of materials

☐ Exclude from board

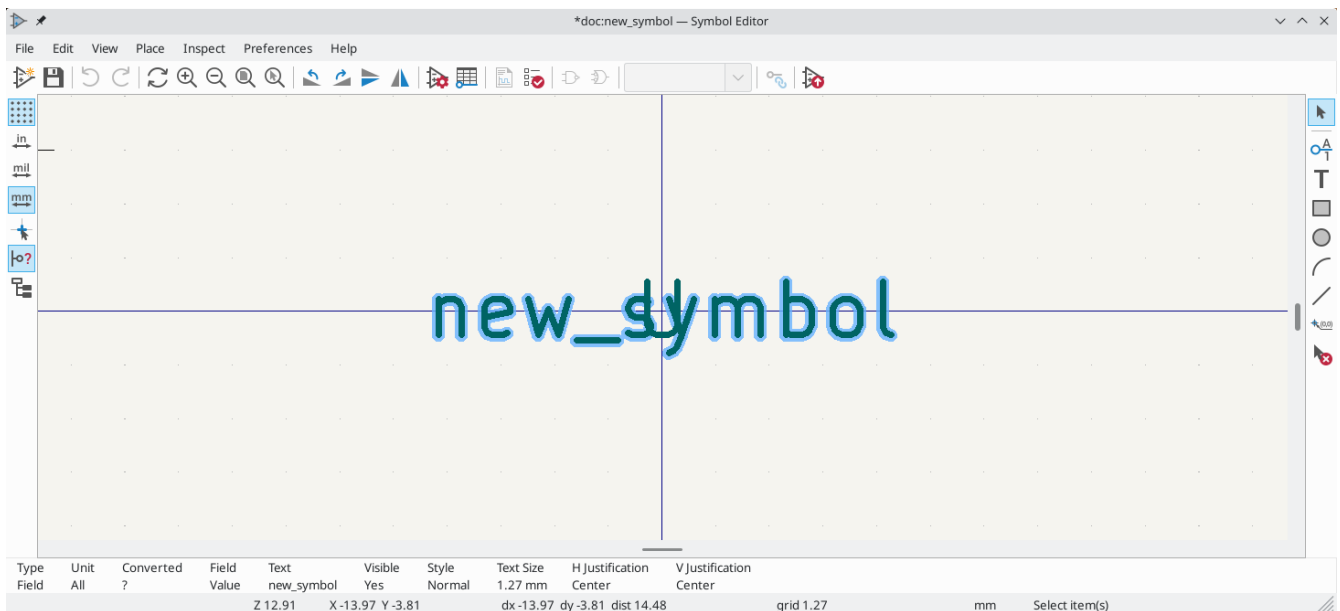
Pin name position offset: mm


☒ Show pin number text

☒ Show pin name text

☒ Pin name inside

A new symbol will be created using the properties above and will appear in the editor as shown below.




The blue cross in the center is the symbol anchor, which specifies the symbol origin i.e. the coordinates (0, 0). The anchor can be repositioned by selecting the  icon and clicking on the new desired anchor position.

Create a Symbol from Another Symbol

Often, the symbol that you want to make is similar to one already in a symbol library. In this case it is easy to load and modify an existing symbol.

- Load the symbol which will be used as a starting point.
- Save a new copy of the symbol using **File** → **Save As....** The Save As dialog will prompt for a name for the new symbol and the library to save it in.
- Edit the new symbol as required.
- Save the modified symbol.

Symbol Properties

Symbol properties are set when the symbol is created but they can be modified at any point. To change the symbol properties, click on the  icon to show the dialog below.

Library Symbol Properties

General **Footprint Filters**

Fields

Name	Value	Show	H Align	V Align	Italic	Bold	Text Size
Reference	U	<input type="checkbox"/>	Left	Center	<input type="checkbox"/>	<input type="checkbox"/>	1.27 mm
Value	LM2904	<input checked="" type="checkbox"/>	Left	Center	<input type="checkbox"/>	<input type="checkbox"/>	1.27 mm
Footprint		<input type="checkbox"/>	Center	Center	<input type="checkbox"/>	<input type="checkbox"/>	1.27 mm
Datasheet	http://www.ti.com/lit/ds/symlink/lm358.pdf	<input type="checkbox"/>	Center	Center	<input type="checkbox"/>	<input type="checkbox"/>	1.27 mm

+ ↑ ↓

Symbol name: LM2904

Description: Dual Operational Amplifiers, DIP-8/SOIC-8/TSSOP-8/VSSOP-8

Keywords: dual opamp

Derive from symbol:

Symbol

☐ Has alternate body style (De Morgan)

☐ Define as power symbol

☐ Exclude from schematic bill of materials

☐ Exclude from board

Number of Units: 3 — +

☐ All units are interchangeable

Pin Text Options

☒ Show pin number

☒ Show pin name

☒ Place pin names inside

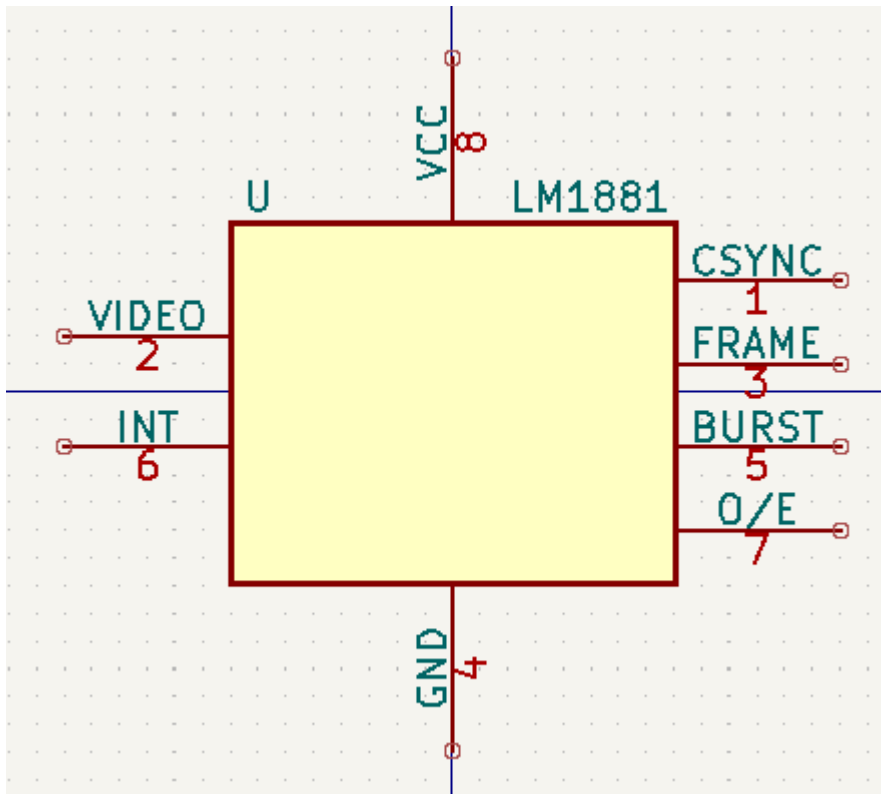
Position offset: 0.127 mm

Edit Spice Model... Cancel OK

It is important to correctly set the number of units per package and the alternate symbolic representation, if enabled, because when pins are edited or created the corresponding pins for each unit will be affected. If you change the number of units per package after pin creation and editing, there will be additional work to specify the pins and graphics for the new unit. Nevertheless, it is possible to modify these properties at any time.

The graphic options "Show pin number" and "Show pin name" define the visibility of the pin number and pin name text. The option "Place pin names inside" defines the pin name position relative to the pin body. The pin names will be displayed inside the symbol outline if the option is checked. In this case the "Pin Name Position Offset" property defines the shift of the text away from the body end of the pin. A value from 0.02 to 0.05 inches is usually reasonable.

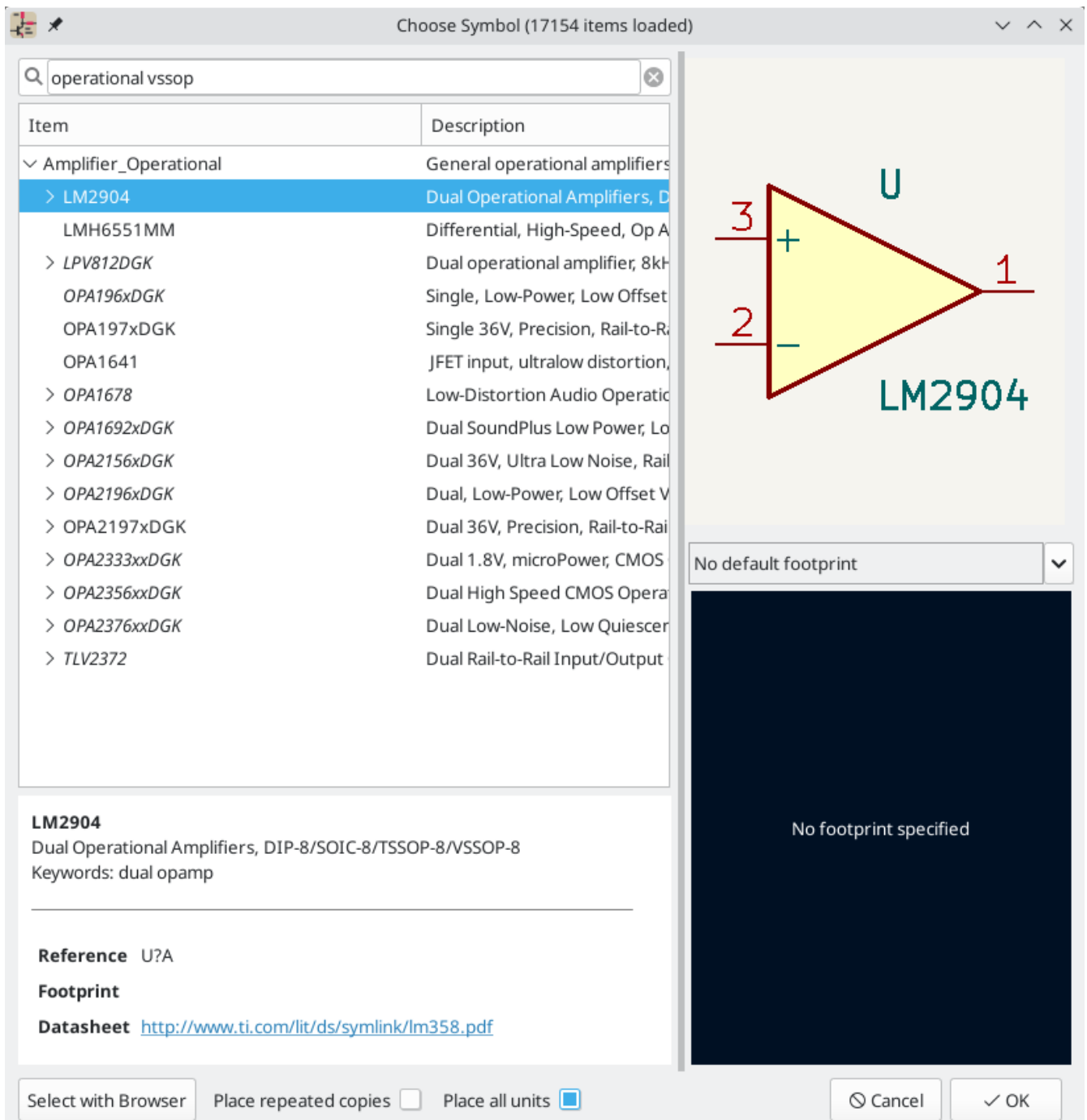
The example below shows a symbol with the "Place pin name inside" option unchecked. Notice the position of the names and pin numbers.



Symbol Name, Description, and Keywords

The symbol's name is the same as the `Value` field. When the symbol name is changed the value also changes, and vice versa. The symbol's name in the library also changes accordingly.

The symbol description should contain a brief description of the component, such as the component function, distinguishing features, and package options. The keywords should contain additional terms related to the component. Keywords are used primarily to assist in searching for the symbol.



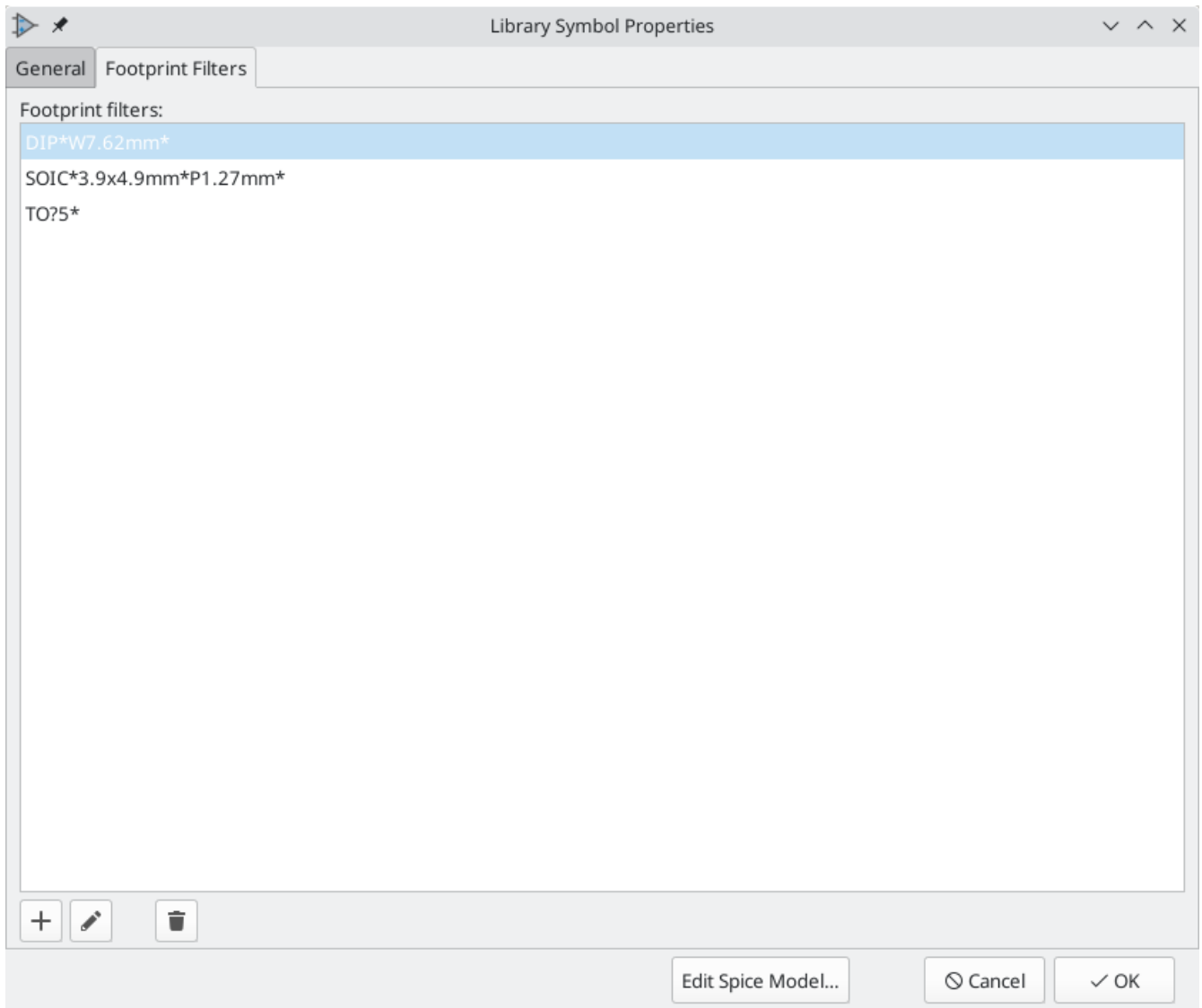
A symbol's name, description, and keywords are all used when searching for symbols in the Symbol Editor and Add a Symbol dialog. The description and keywords are displayed in the Symbol Library Browser and Add a Symbol dialog.

Footprint Filters


The footprint filters tab is used to define which footprints are appropriate to use with the symbol. The filters can be applied in the Footprint Assignment tool so that only appropriate footprints are displayed for each symbol.

Multiple footprint filters can be defined. Footprints that match any of the filters will be displayed; if no filters are defined, then all footprints will be displayed.

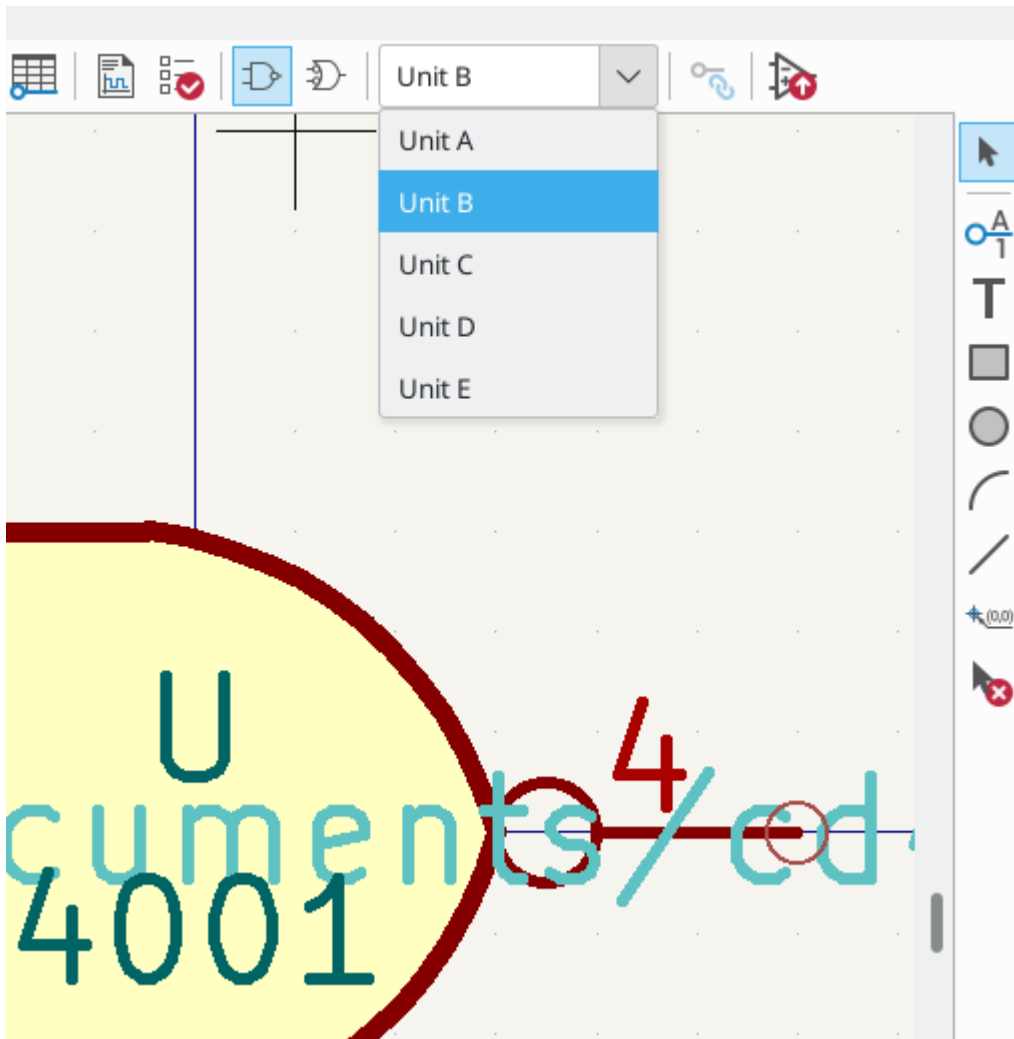
Filters can use wildcards: `*` matches any number of characters, including zero, and `?` matches zero or one characters. For example, `SOIC-*` would match the `SOIC-8_3.9x4.9mm_P1.27mm` footprint as well as any other footprint beginning with `SOIC-`. The filter `SOT?23` matches `SOT23` as well as `SOT-23`.



Symbols with Alternate Symbolic Representation

If the symbol has an alternate body style defined, one body style must be selected for editing at a time. To edit the normal representation, click the  icon.

To edit the alternate representation, click on the  icon. Use the  dropdown shown below to select the unit you wish to edit.



Elementos gráficos

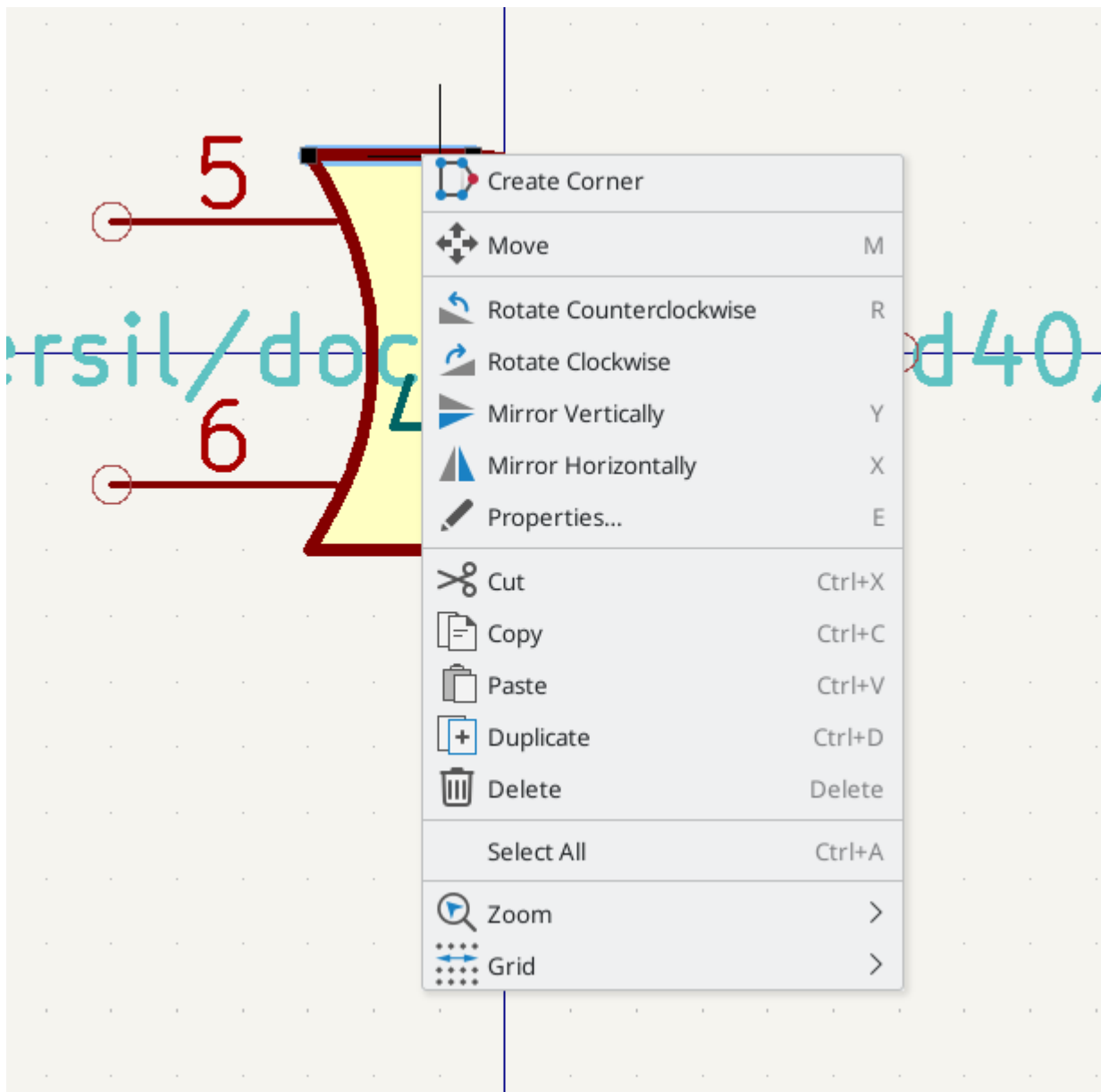
Graphical elements create the visual representation of a symbol and contain no electrical connection information. Graphical elements are created with the following tools:

- Líneas y polígonos definidos mediante sus puntos iniciales y finales.
- Rectángulos definidos mediante dos esquinas opuestas.
- Círculos definidos mediante su centro y radio.
- Arcos definidos mediante sus puntos inicial y final así como el centro de la circunferencia que lo contiene. Un arco va desde 0° a 180°.

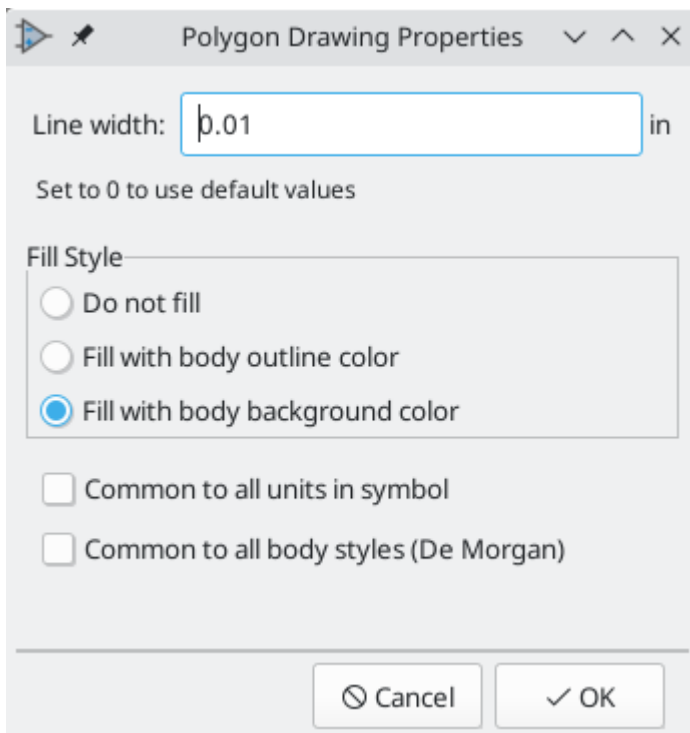
The vertical toolbar on the right hand side of the main window allows you to place all of the graphical elements required to design the representation of a symbol.

Características de los elementos gráficos

Cada elemento gráfico (línea, arco, círculo, etc.) puede definirse como común a todas las unidades y/o estilos de símbolo, o específicos a una unidad o estilo de símbolo concreto. Puede accederse rápidamente a las opciones de los elementos mediante un menú contextual haciendo clic derecho en el elemento. Debajo se muestra el menú contextual para un elemento línea.



Además puede realizar doble clic sobre un elemento para modificar sus propiedades. Debajo se muestra el formulario de propiedades para un elemento polígono.



Las propiedades de un elemento gráfico son:

- "Line width" defines the width of the element's line in the current drawing units.
- "Fill Style" determines if the shape defined by the graphical element is to be drawn unfilled, background filled, or foreground filled.
- "Common to all units in symbol" determines if the graphical element is drawn for each unit in symbol with more than one unit per package or if the graphical element is only drawn for the current unit.
- "Common to all body styles (De Morgan)" determines if the graphical element is drawn for each symbolic representation in symbols with an alternate body style or if the graphical element is only drawn for the current body style.

Elementos de texto

The **T** icon allows for the creation of graphical text. Graphical text is automatically oriented to be readable, even when the symbol is mirrored. Please note that graphical text items are not the same as symbol fields.

Multiple Units per Symbol and Alternate Body Styles

Symbols can have up to two body styles (a standard symbol and an alternate symbol often referred to as a "De Morgan equivalent") and/or have more than one unit per package (logic gates for example). Some symbols can have more than one unit per package each with different symbols and pin configurations.

Consider for instance a relay with two switches, which can be designed as a symbol with three different units: a coil, switch 1, and switch 2. Designing a symbol with multiple units per package and/or alternate body styles is very flexible. A pin or a body symbol item can be common to all units or specific to a given unit or they can be common to both symbolic representation so are specific to a given symbol representation.

By default, pins are specific to a unit and body style. When a pin is common to all units or all body styles, it only needs to be created once. This is also the case for the body style graphic shapes and text, which may be

common to each unit, but typically are specific to each body style).

Example of a Symbol With Multiple Noninterchangeable Units

For an example of a symbol with multiple units that are not interchangeable, consider a relay with 3 units per package: a coil, switch 1, and switch 2.

The three units are not all the same, so "All units are interchangeable" should be deselected in the Symbol Properties dialog. Alternatively, this option could have been specified when the symbol was initially created.

The dialog box is titled "Library Symbol Properties" and has two tabs: "General" and "Footprint Filters". The "General" tab is active.

Fields

Name	Value	Show	H Align	V Align	Italic	Bold	Text Size
Reference	K	<input type="checkbox"/>	Left	Center	<input type="checkbox"/>	<input type="checkbox"/>	0.05 in
Value	G5V-2_Split	<input checked="" type="checkbox"/>	Left	Center	<input type="checkbox"/>	<input type="checkbox"/>	0.05 in
Footprint	Relay_THT:Relay_DPDT_Omron_G5V-2	<input type="checkbox"/>	Left	Center	<input type="checkbox"/>	<input type="checkbox"/>	0.05 in
Datasheet	http://omronfs.omron.com/en_US/ecb/products/pdf/en-g5v	<input type="checkbox"/>	Center	Center	<input type="checkbox"/>	<input type="checkbox"/>	0.05 in

Below the table are icons for adding (+), moving up (↑), moving down (↓), and deleting (trash).

Symbol name: G5V-2_Split

Description: Relay Miniature Omron DPDT

Keywords: Miniature Relay Dual Pole DPDT Omron

Derive from symbol: (dropdown menu)

Symbol

- ☐ Has alternate body style (De Morgan)
- ☐ Define as power symbol
- ☐ Exclude from schematic bill of materials
- ☐ Exclude from board

Number of Units: 3 (with minus and plus buttons)

☐ All units are interchangeable

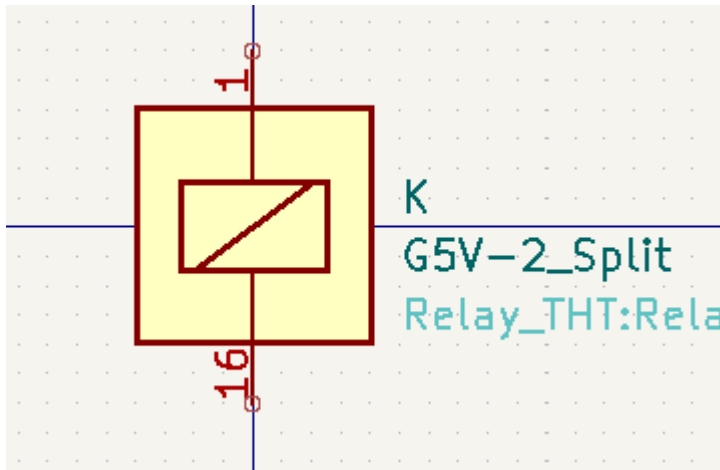
Pin Text Options

- ☒ Show pin number
- ☒ Show pin name
- ☒ Place pin names inside

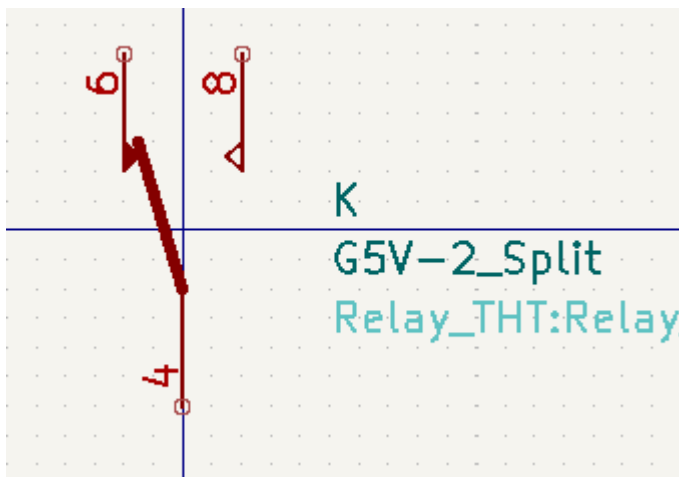
Position offset: 0.02 in

Buttons at the bottom: Edit Spice Model..., Cancel, OK.

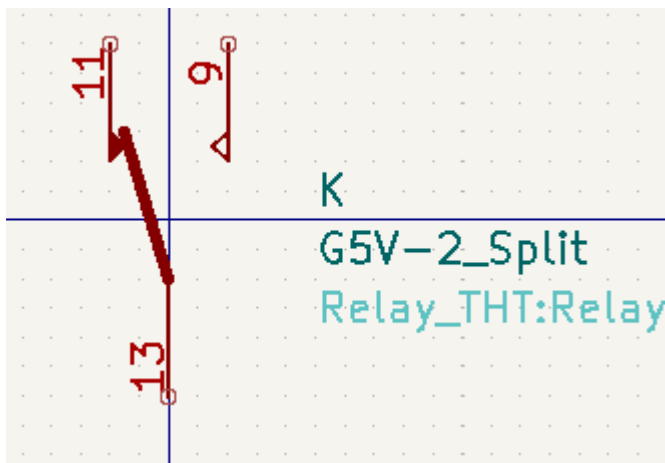
Unit A



Unit B




Unit C



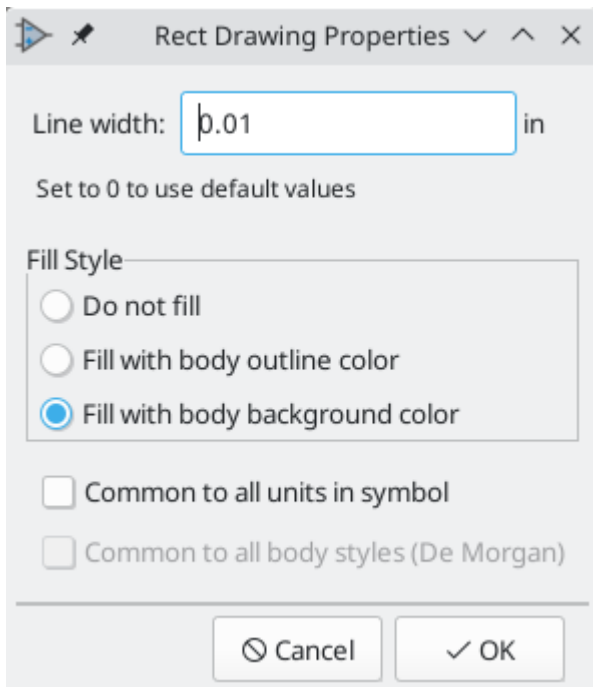
Unit A does not have the same symbol and pin layout as Units B and C, so the units are not interchangeable.

NOTE

"Synchronized Pins Edit Mode" can be enabled by clicking the  icon. In this mode, pin modifications are propagated between symbol units; changes made in one unit will be reflected in the other units as well. When this mode is disabled, pin changes made in one unit do not affect other units. This mode is enabled automatically when "All units are interchangeable" is checked, but it can be disabled. The mode cannot be enabled when "All units are interchangeable" is unchecked or when the symbol only has one unit.

Elementos Gráficos Simbólicos

Shown below are properties for a graphic body element. In the relay example above, the three units have different symbolic representations. Therefore, each unit was created separately and the graphical body elements have the "Common to all units in symbol" setting disabled.



Edición y Creación de Pines

You can click on the  icon to create and insert a pin. The editing of all pin properties is done by double-clicking on the pin or right-clicking on the pin to open the pin context menu. Pins must be created carefully, because any error will have consequences on the PCB design. Any pin already placed can be edited, deleted, and/or moved.

Resumen de Pines

A pin is defined by its graphical representation, its name and its number. The pin's name and number can contain letters, numbers, and symbols, but not spaces. For the Electrical Rules Check (ERC) tool to be useful, the pin's electrical type (input, output, tri-state...) must also be defined correctly. If this type is not defined properly, the schematic ERC check results may be invalid.

Notas importantes:

- Symbol pins are matched to footprint pads by number. The pin number in the symbol must match the corresponding pad number in the footprint.
- Do not use spaces in pin names and numbers. Spaces will be automatically replaced with underscores (`_`).
- To define a pin name with an inverted signal (overline) use the `~` (tilde) character followed by the text to invert in braces. For example `~{FO}O` would display \overline{FO} O.
- If the pin name is empty, the pin is considered unnamed.
- Pin names can be repeated in a symbol.

Pin numbers must be unique in a symbol.

Propiedades de Pin

Pin name:

Pin number:

Electrical type:

Graphic style:

X position: in

Y position: in

Orientation:

Pin length: in

Name text size: in

Number text size: in

☐ Common to all units in symbol

☐ Common to all body styles (De Morgan)

☒ Visible

Preview:

> Alternate pin definitions

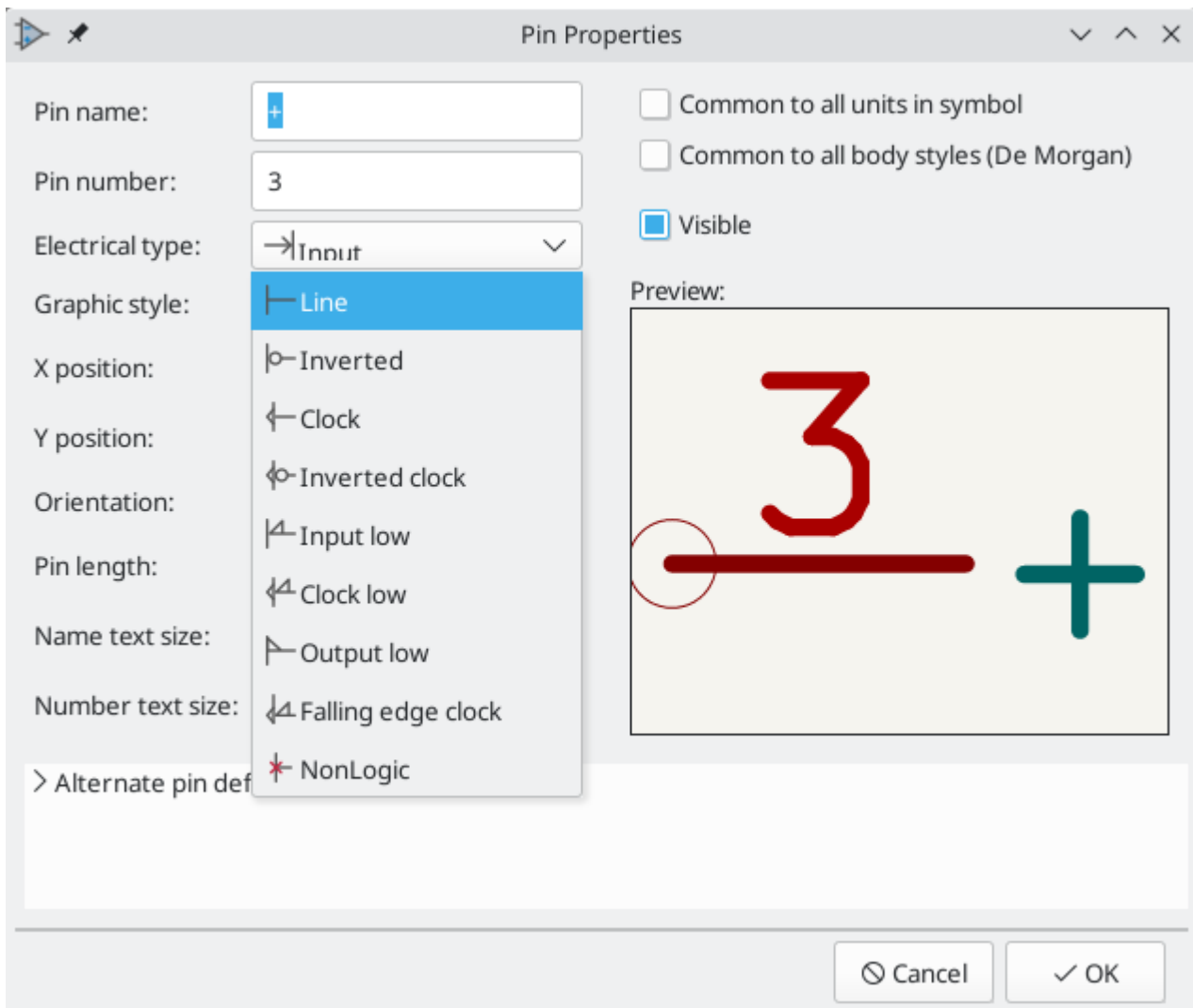
Cancel OK

El formulario de propiedades del pin le permite editar todas las características de un pin. Este formulario aparece automáticamente cuando crea un pin o cuando hace doble clic en un pin que ya exista. Desde este formulario podrá modificar:

- The pin name and text size.
- The pin number and text size.
- The pin length.
- The pin electrical type and graphical style.
- Si el pin está definido para todos las unidades y símbolos alternativos.
- Pin visibility.
- [Alternate pin definitions](#).

Pin Graphic Styles

Shown in the figure below are the different pin graphic styles. The choice of graphic style does not have any influence on the pin's electrical type.



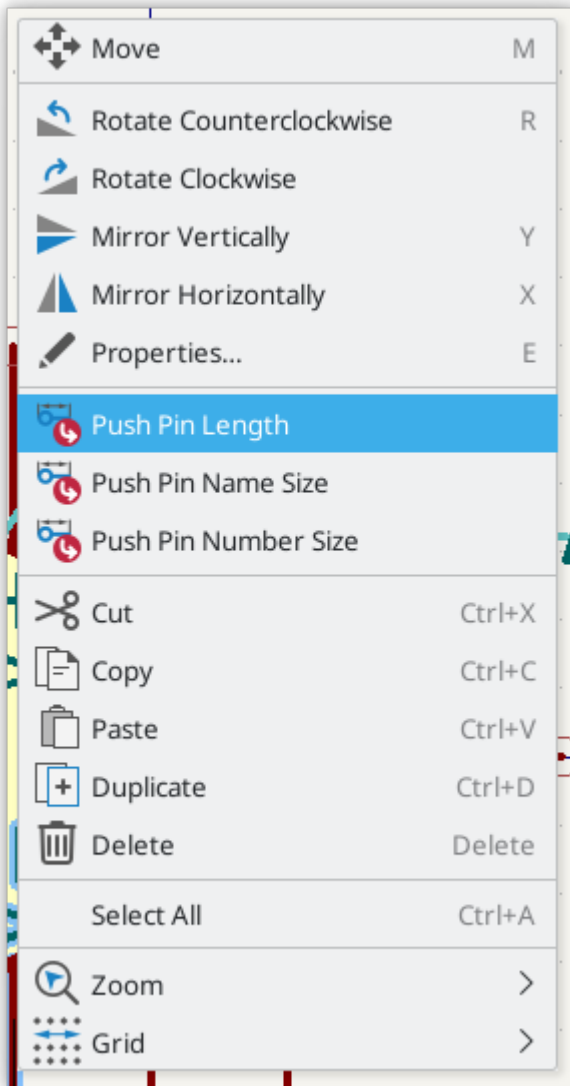
Tipos Eléctricos de Pin

Choosing the correct electrical type is important for the schematic ERC tool. ERC will check that pins are connected appropriately, for example ensuring that input pins are driven and power inputs receive power from an appropriate source.

Pin Type	Description
Input	A pin which is exclusively an input.
Output	A pin which is exclusively an output.
Bidirectional	A pin that can be either an input or an output, such as a microcontroller data bus pin.
Tri-state	A three state output pin (high, low, or high impedance)
Passive	A passive symbol pin: resistors, connectors, etc.
Free	A pin that can be freely connected to any other pin without electrical concerns.
Unspecified	A pin for which the ERC check does not matter.
Power input	A symbol's power pin. As a special case, power input pins that are marked invisible are automatically connected to the net with the same name. See the Power Ports section for more information.
Power output	A pin that provides power to other pins, such as a regulator output.
Open collector	An open collector logic output.
Open emitter	An open emitter logic output.
Unconnected	A pin that should not be connected to anything.


Pushing Pin Properties to Other Pins

You can apply the length, name size, or number size of a pin to the other pins in the symbol by right clicking the pin and selecting **Push Pin Length**, **Push Pin Name Size**, or **Push Pin Number Size**, respectively.





Definición de pines para múltiples unidades y representaciones de símbolo alternativas

Symbols with multiple units and/or graphical representations are particularly problematic when creating and editing pins. The majority of pins are specific to each symbol unit (because each unit has a different set of pins) and to each body style (because the form and position is different between the normal body style and the alternate form).


The symbol library editor allows the simultaneous creation of pins. By default, changes made to a pin are made for all units of a multiple unit symbol and to both representations for symbols with an alternate symbolic representation. The only exception to this is the pin's graphical type and name, which remain unlinked between symbol units and body styles. This dependency was established to allow for easier pin creation and editing in most cases. This dependency can be disabled by toggling the  icon on the main tool bar. This will allow you to create pins for each unit and representation completely independently.

Pins can be common or specific to different units. Pins can also be common to both symbolic representations or specific to each symbolic representation. When a pin is common to all units, it only has to be drawn once. Pins are set as common or specific in the pin properties dialog.

An example is the output pin in the 7400 quad dual input NAND gate. Since there are four units and two symbolic representations, there are eight separate output pins defined in the symbol definition. When

creating a new 7400 symbol, unit A of the normal symbolic representation will be shown in the library editor. To edit the pin style in the alternate symbolic representation, it must first be enabled by clicking the  button on the tool bar. To edit the pin number for each unit, select the appropriate unit using the  drop down control.

Pin Table

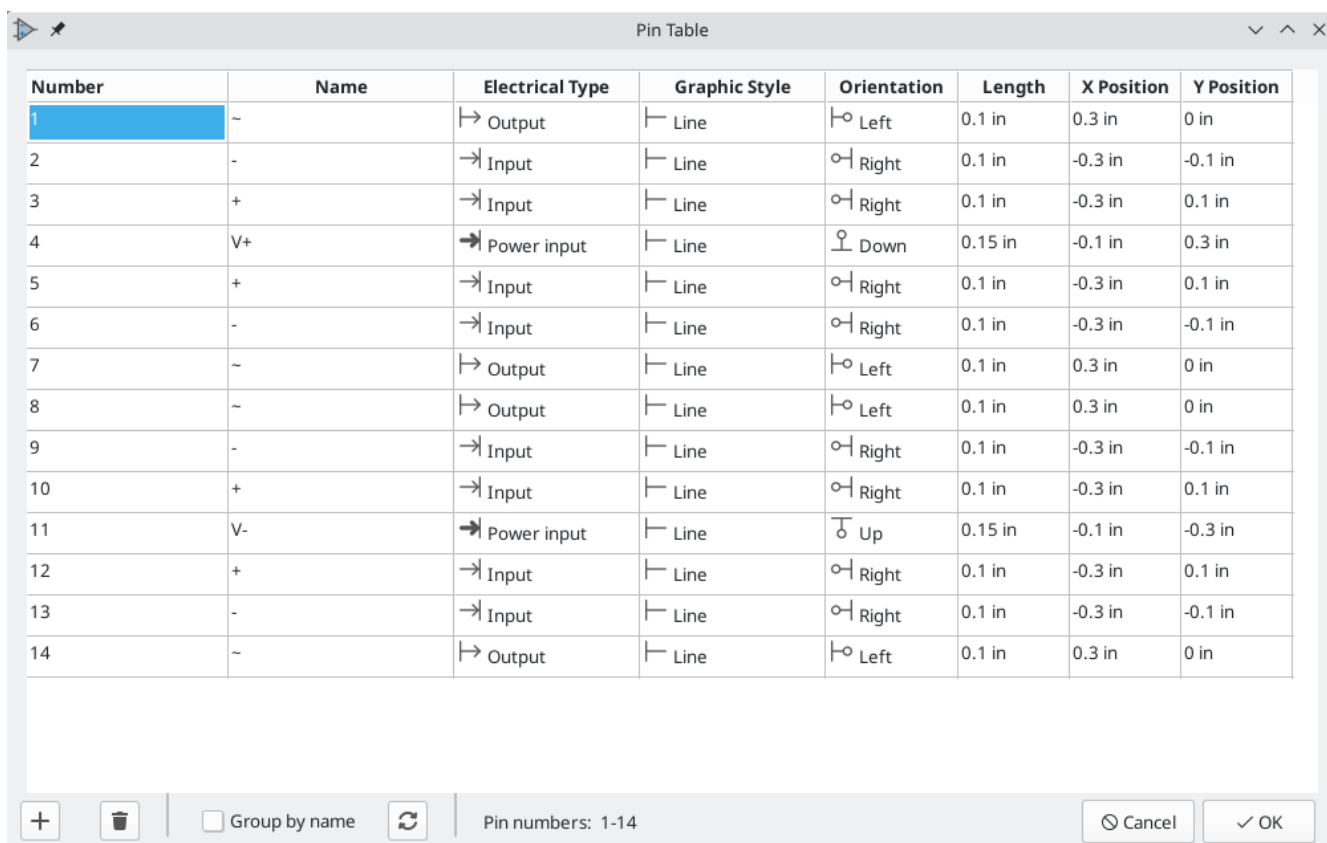
Another way to edit pins is to use the Pin Table, which is accessible via the  icon. The Pin Table displays all of the pins in the symbol and their properties in a table view, so it is useful for making bulk pin changes.

Any pin property can be edited by clicking on the appropriate cell. Pins can be added and removed with the  and  icons, respectively.

NOTE

Columns of the pin table can be shown or hidden by right-clicking on the header row and checking or unchecking additional columns. Some columns are hidden by default.

The screenshot below shows the pin table for a quad opamp.

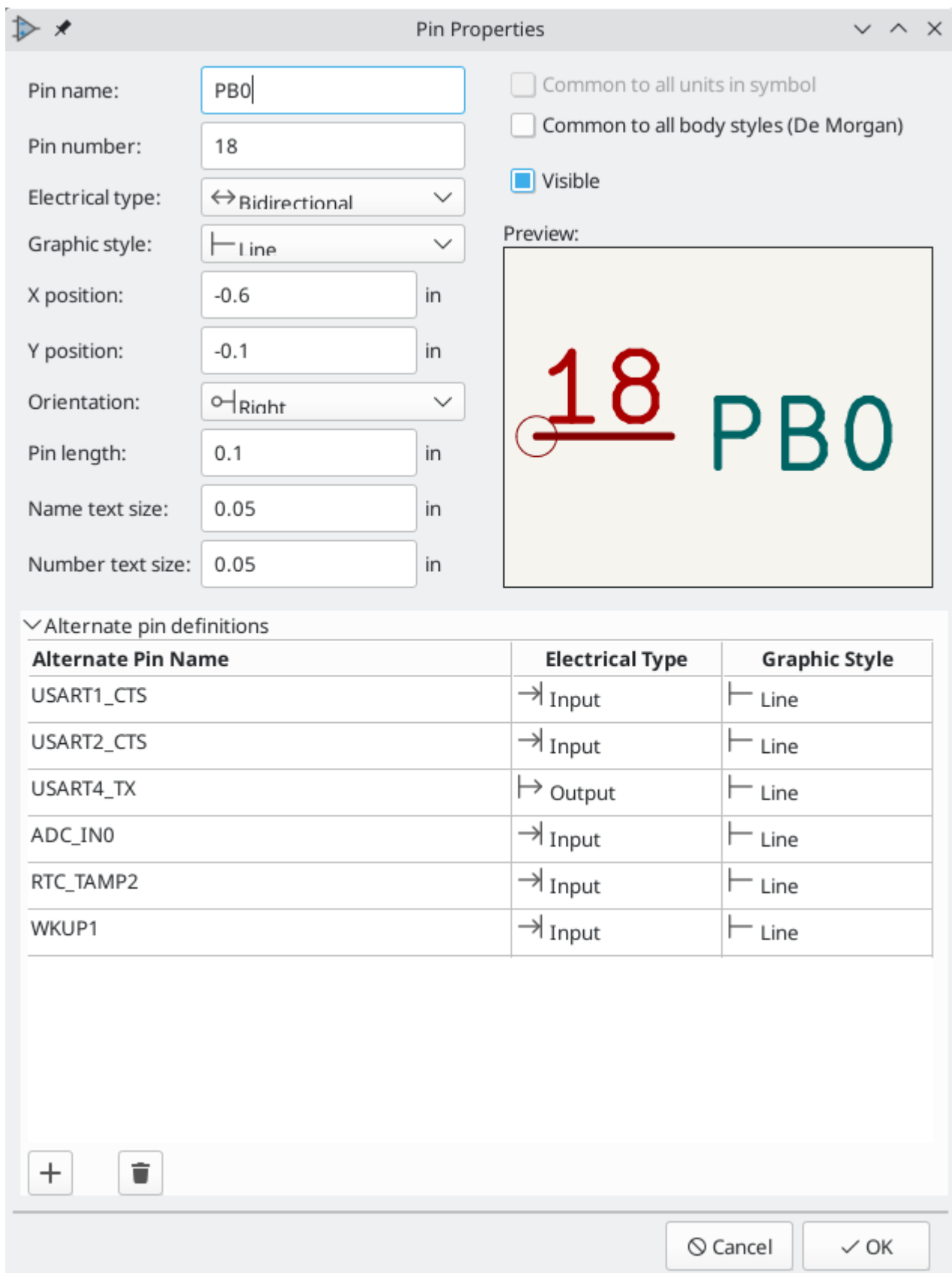


Number	Name	Electrical Type	Graphic Style	Orientation	Length	X Position	Y Position
1	~	Output	Line	Left	0.1 in	0.3 in	0 in
2	-	Input	Line	Right	0.1 in	-0.3 in	-0.1 in
3	+	Input	Line	Right	0.1 in	-0.3 in	0.1 in
4	V+	Power input	Line	Down	0.15 in	-0.1 in	0.3 in
5	+	Input	Line	Right	0.1 in	-0.3 in	0.1 in
6	-	Input	Line	Right	0.1 in	-0.3 in	-0.1 in
7	~	Output	Line	Left	0.1 in	0.3 in	0 in
8	~	Output	Line	Left	0.1 in	0.3 in	0 in
9	-	Input	Line	Right	0.1 in	-0.3 in	-0.1 in
10	+	Input	Line	Right	0.1 in	-0.3 in	0.1 in
11	V-	Power input	Line	Up	0.15 in	-0.1 in	-0.3 in
12	+	Input	Line	Right	0.1 in	-0.3 in	0.1 in
13	-	Input	Line	Right	0.1 in	-0.3 in	-0.1 in
14	~	Output	Line	Left	0.1 in	0.3 in	0 in

Alternate Pin Definitions

Pins can have alternate pin definitions added to them. Alternate pin definitions allow a user to select a different name, electrical type, and graphical style for a pin when the symbol has been placed in the schematic. This can be used for pins that have multiple functions, such as microcontroller pins.

Alternate pin definitions are added in the Pin Properties dialog as shown below. Each alternate definition contains a pin name, electrical type, and graphic style. This microcontroller pin has all of its peripheral functions defined in the symbol as alternate pin names.



The Pin Properties dialog box is used to configure the properties of a pin symbol. It includes fields for Pin name, Pin number, Electrical type, Graphic style, X position, Y position, Orientation, Pin length, Name text size, and Number text size. It also has checkboxes for 'Common to all units in symbol', 'Common to all body styles (De Morgan)', and 'Visible'. A Preview window shows the resulting symbol. Below the main settings is a section for 'Alternate pin definitions' containing a table of alternate pin names, electrical types, and graphic styles. At the bottom are buttons for '+', a trash icon, 'Cancel', and 'OK'.

Pin name:
☐ Common to all units in symbol
☐ Common to all body styles (De Morgan)
☒ Visible

Pin number:

Electrical type:

Graphic style:

X position: in


Y position: in

Orientation:

Pin length: in

Name text size: in

Number text size: in

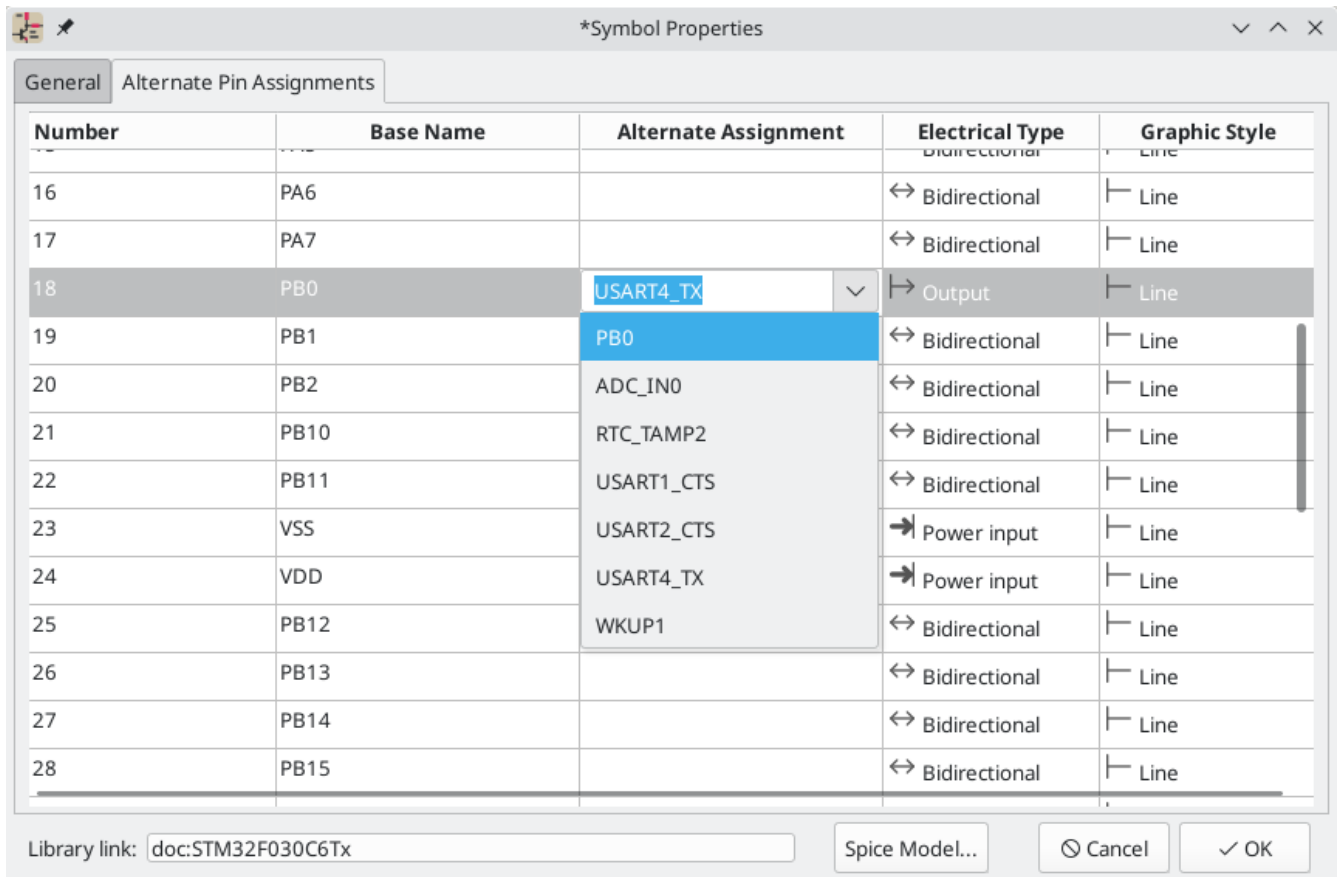
Preview:


The preview shows a pin symbol with the number '18' in red and the text 'PB0' in teal. The number '18' is underlined with a red line, and a small red circle is at the end of the line.

▼ Alternate pin definitions

Alternate Pin Name	Electrical Type	Graphic Style
USART1_CTS	→ Input	└─ Line
USART2_CTS	→ Input	└─ Line
USART4_TX	└─ Output	└─ Line
ADC_IN0	→ Input	└─ Line
RTC_TAMP2	→ Input	└─ Line
WKUP1	→ Input	└─ Line

Alternate pin definitions are selected in the Schematic Editor once the symbol has been placed in the schematic. The alternate pin is assigned in the Alternate Pin Assignments tab of the Symbol Properties dialog. Alternate definitions are selectable in the dropdown in the Alternate Assignment column.



Symbol Fields

All library symbols are defined with four default fields. The reference designator, value, footprint assignment, and datasheet link fields are created whenever a symbol is created or copied. Only the reference designator and value fields are required.

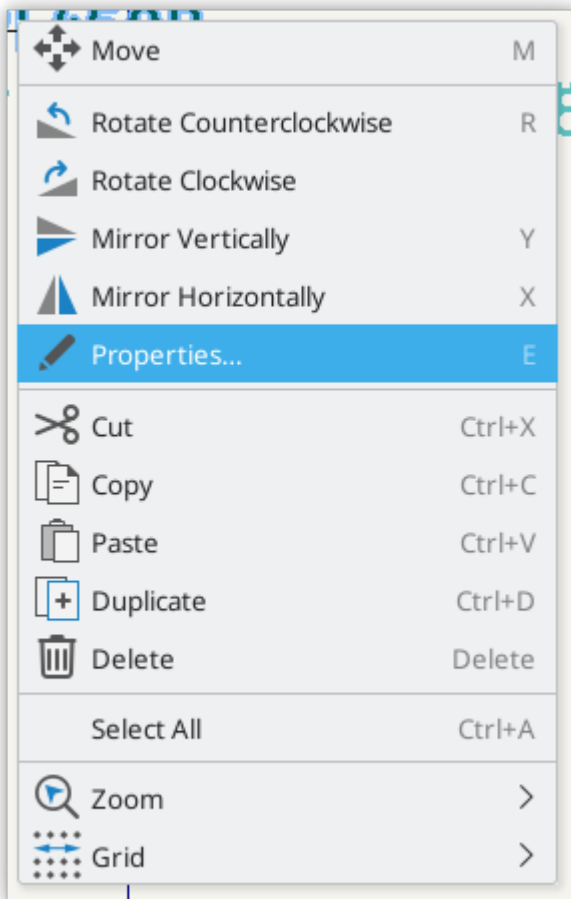
Symbols defined in libraries are typically defined with only these four default fields. Additional fields such as vendor, part number, unit cost, etc. can be added to library symbols but generally this is done in the schematic editor so the additional fields can be applied to all of the symbols in the schematic.


NOTE

A convenient way to create additional empty symbol fields is to use define field name templates. Field name templates define empty fields that are added to each symbol when it is inserted into the schematic. Field name templates can be defined globally (for all schematics) in the Schematic Editor Preferences, or they can be defined locally (specific to each project) in the Schematic Setup dialog.

Editing Symbol Fields

To edit an existing symbol field, right-click on the field text to show the field context menu shown below.



To add new fields, delete optional fields, or edit existing fields, use the  icon on the main tool bar to open the [Symbol Properties dialog](#).

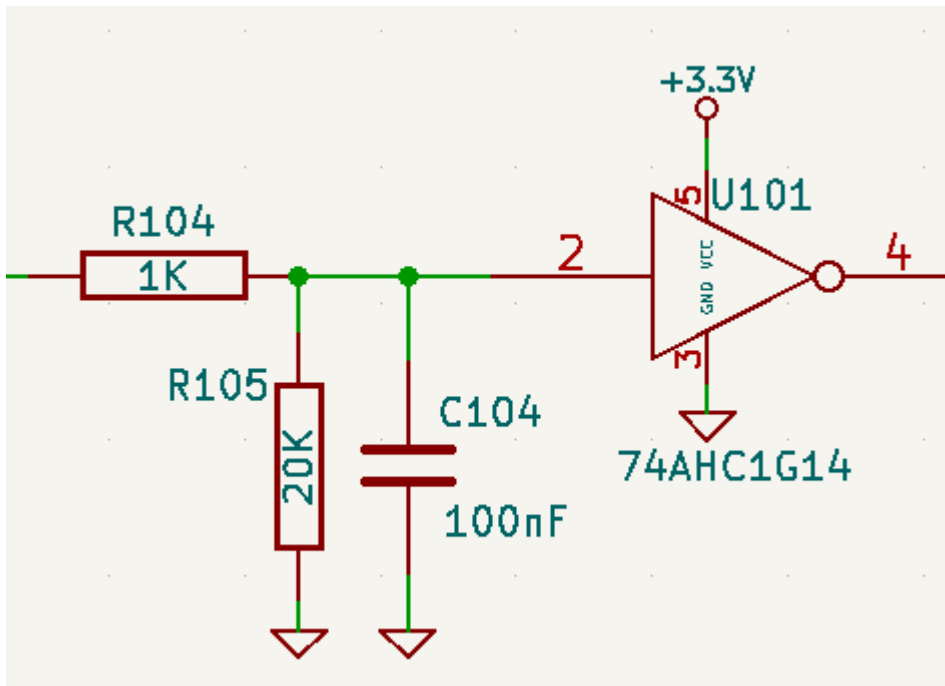
Fields are text information associated a the symbol. Do not confuse them with text in the graphic representation of a symbol.

Notas importantes:

- Modifying the `Value` field changes the name of the symbol. The symbol's name in the library will change when the symbol is saved.
- The Symbol Properties dialog must be used to edit a field that is empty or has the invisible attribute enabled because such fields cannot be clicked on.
- The footprint is defined as an absolute footprint using the `LIBNAME:FOOTPRINTNAME` format where `LIBNAME` is the name of the footprint library defined in the footprint library table (see the "Footprint Library Table" section in the PCB Editor manual) and `FOOTPRINTNAME` is the name of the footprint in the library `LIBNAME`.

Power Ports

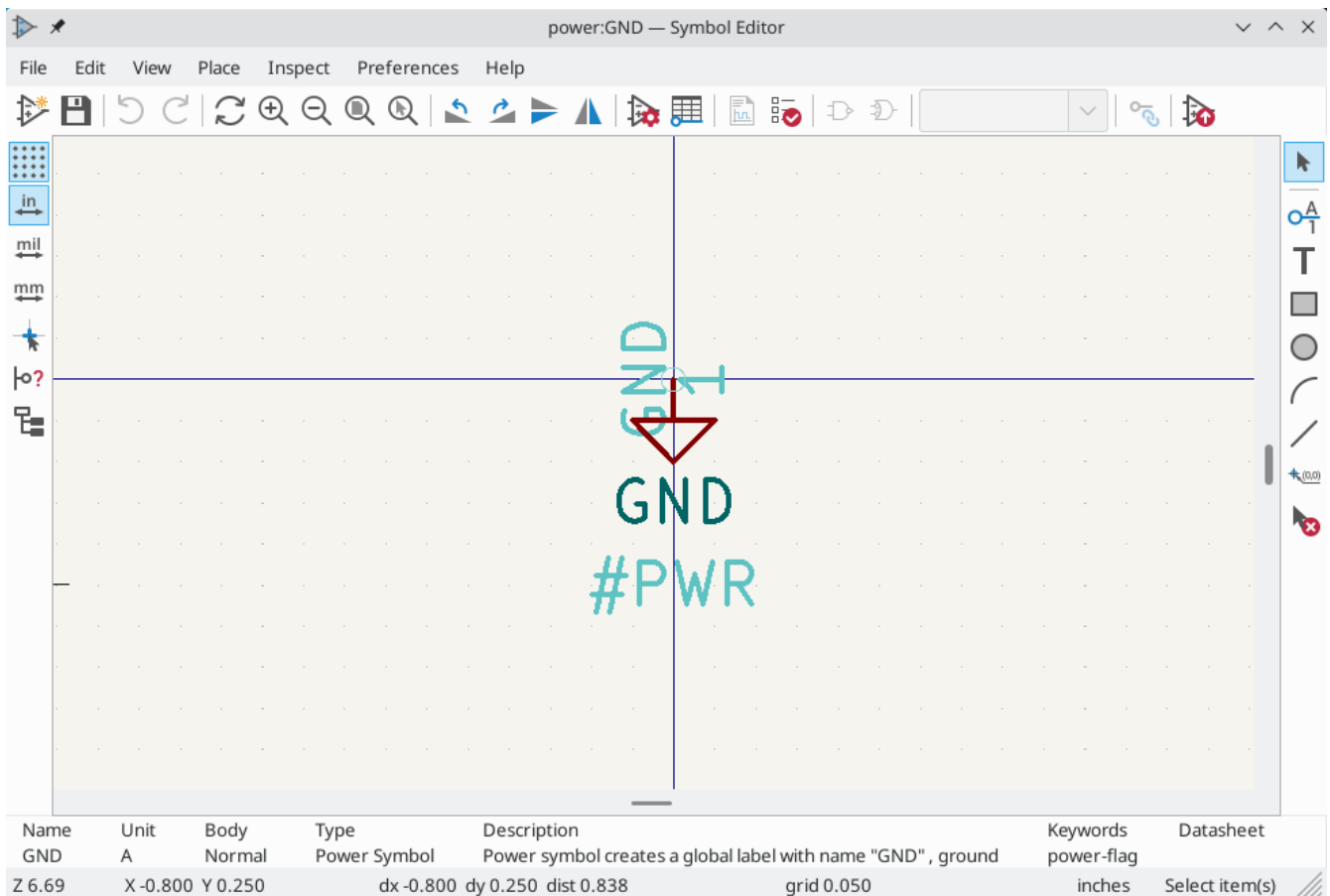
Power ports, or power symbols, are conventionally used to label a wire as part of a power net, like `VCC`, `+5V`, or `GND`. In the schematic below, the `+3.3V` and `GND` symbols are power ports. In addition to acting as a visual indicator that a net is a power rail, a power port will determine the name of the net it is attached to. This is true even if there is another net label attached to the net; the net name determined by the power symbol overrides any other net names.



It may be useful to place power symbols in a dedicated library. KiCad's symbol library places power symbols in the `power` library, and users may create libraries to store their own power symbols. If the "Define as power symbol" box is checked in a symbol's properties, that symbol will appear in the Schematic Editor's "Add Power Port" dialog for convenient access.

Power symbols are handled and created the same way as normal symbols, but there are several additional considerations described below. They consist of a graphical symbol and a pin of the type "Power input" that is marked hidden.

Below is an example of a `GND` power symbol.



Creating a Power Port Symbol

Power Port symbols consist of a pin of type "Power input" that is marked invisible. Invisible power input pins have a special property of automatically connecting to a net with the same name as the pin name. A net that is wired to an invisible power input pin will therefore be named after the pin, even if there are other net labels on the net. This connection is global.

NOTE

If the power symbol has the "Define as power symbol" property checked, the power input pin does not need to be marked invisible. However, the convention is to make these pins invisible anyway.

Pin Properties

Pin name:

Pin number:

Electrical type:

Graphic style:

X position: in

Y position: in

Orientation:

Pin length: in

Name text size: in

Number text size: in

☐ Common to all units in symbol

☐ Common to all body styles (De Morgan)

☐ Visible

Preview:

> Alternate pin definitions

Cancel OK

Para crear un símbolo de potencia siga los siguientes pasos:

- Add a pin of type "Power input", with "Visible" unchecked, and the pin named according to the desired net. Make the pin number 1, the length 0, and set the graphic style to "Line". The pin name establishes the connection to the net; in this case the pin will automatically connect to the net GND. The pin number, length, and line style do not matter electrically.
- Place the pin on the symbol anchor.
- Use the shape tools to draw the symbol graphics.
- Set the symbol value. The symbol value does not matter electrically, but it is displayed in the schematic. To eliminate confusion, it should match the pin name (which determines the connected net name).
- Check the "Define as power symbol" box in Symbol Properties window. This makes the symbol appear in the "Add Power Port" dialog, makes the Value field read-only in the schematic, prevents the symbol from being assigned a footprint, and excludes the symbol from the board, BOM, and netlists.
- Set the symbol reference and uncheck the "Show" box. The reference text is not important except for the first character, which should be #. For the power port shown above, the reference could be #GND. Symbols with references that begin with # are not added to the PCB, are not included in Bill of Materials exports or netlists, and they cannot be assigned a footprint in the footprint assignment tool. If a power port's reference does not begin with #, the character will be inserted automatically when the annotation or footprint assignment tools are run.


An easier method to create a new power port symbol is to use another symbol as a starting point, [as described earlier](#).

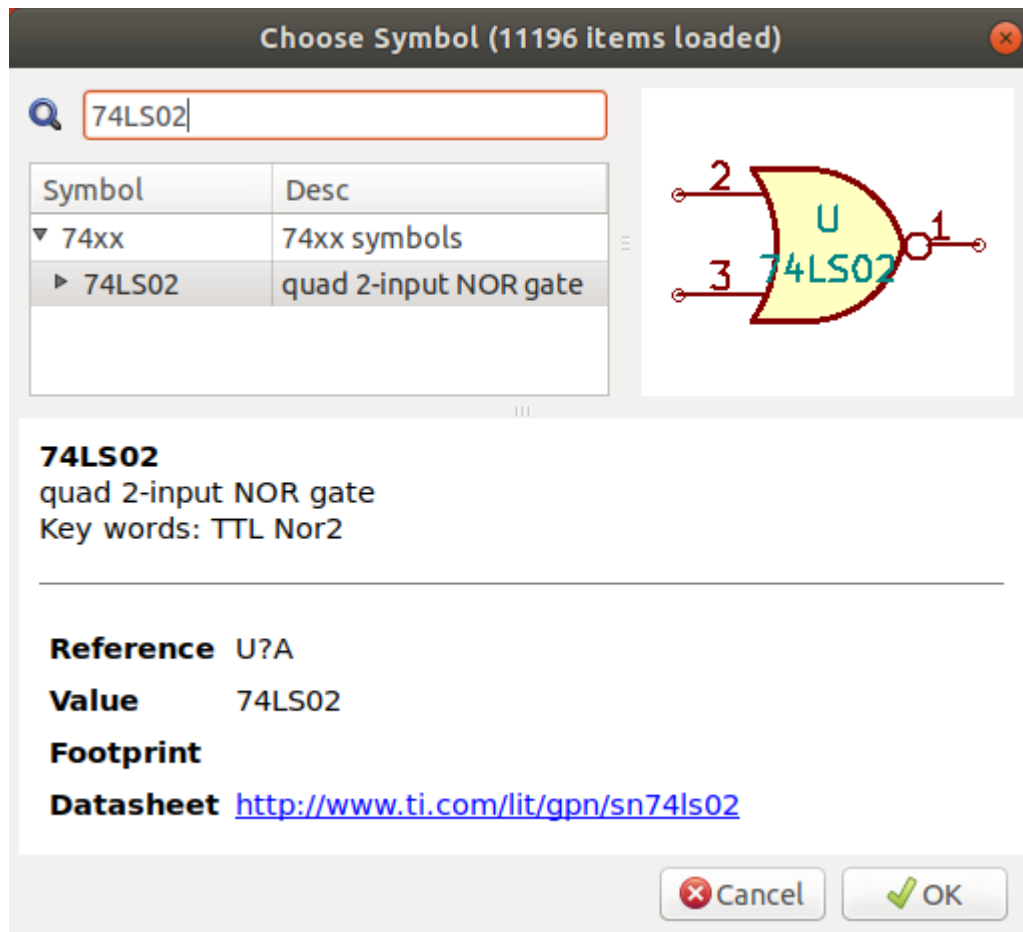
NOTE

When modifying an existing power port symbol, make sure to rename the pin name so that the new symbol connects to the appropriate power net.

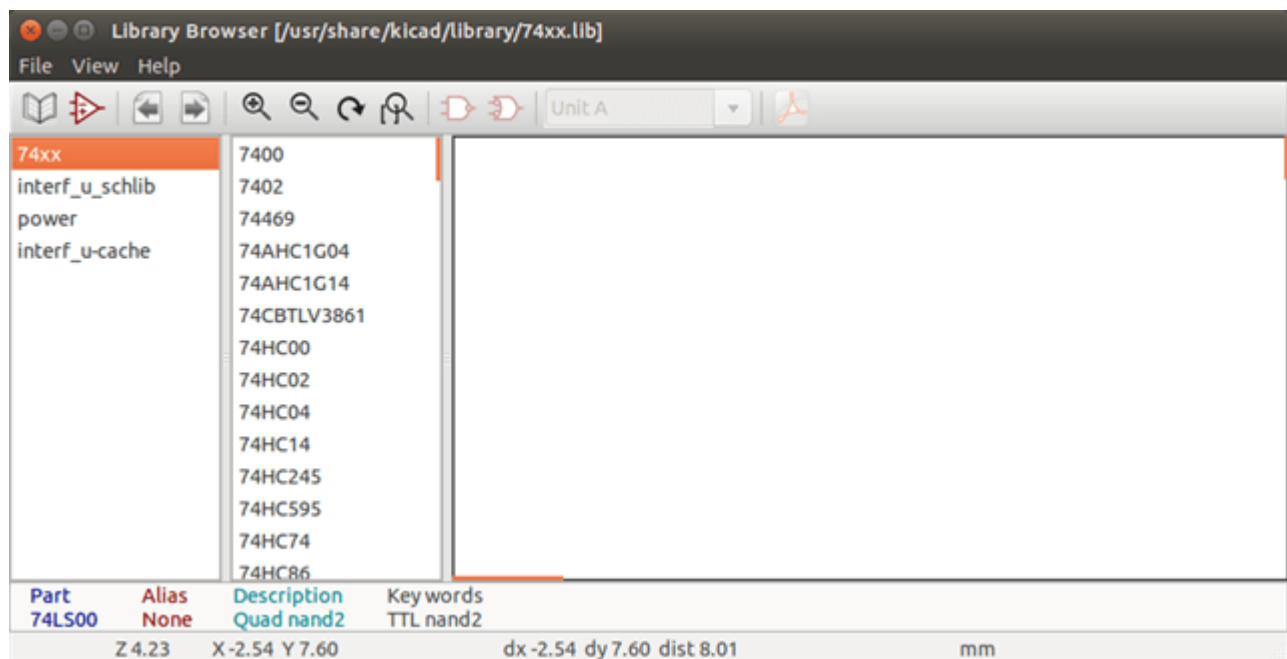
Symbol Library Browser

Introducción

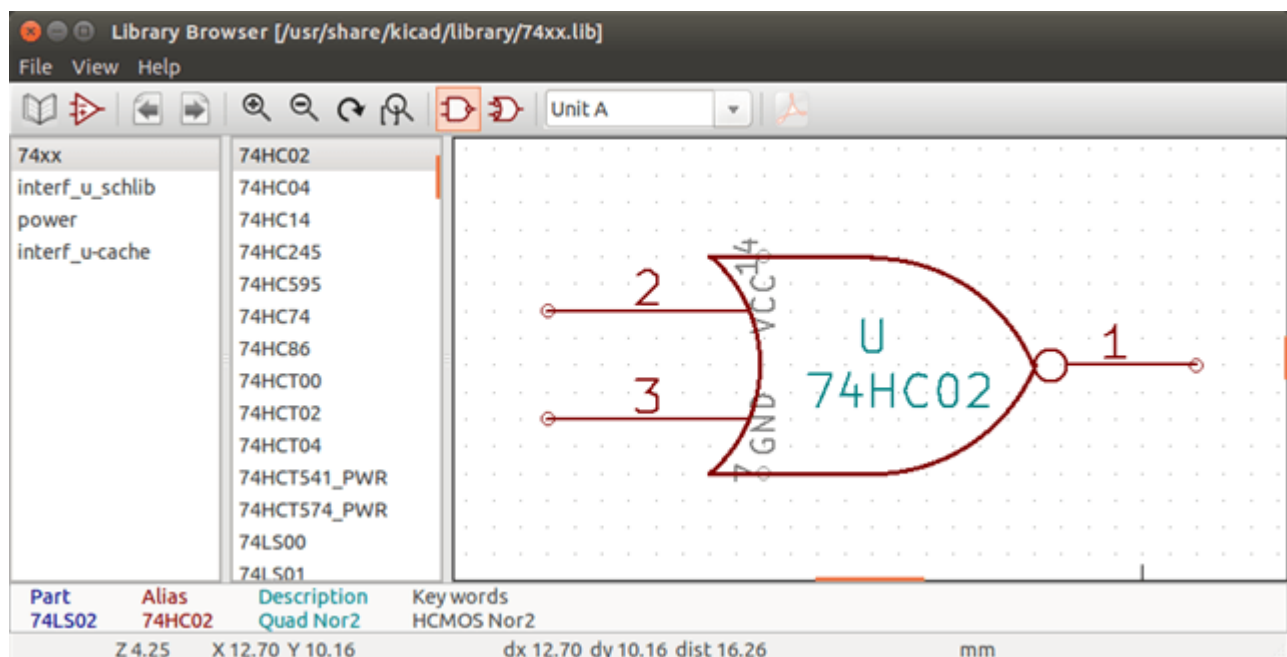
The Symbol Library Browser allows you to quickly examine the content of symbol libraries. The Symbol Library Viewer can be accessed by clicking  icon on the main toolbar, **View** → **Symbol Library Browser**..., or clicking **Select With Browser** in the "Choose Symbol" window.



Viewlib - pantalla principal



To examine the contents of a library, select a library from the list on the left hand pane. All symbols in the selected library will appear in the second pane. Select a symbol name to view the symbol.





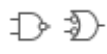
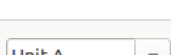




Symbol Library Browser Top Toolbar

The top tool bar in Symbol Library Browser is shown below.



The available commands are:

	Selection of the symbol which can be also selected in the displayed list.
	Display previous symbol.
	Display next symbol.
	Zoom tools.
	Selection of the representation (normal or alternate) if an alternate representation exists.
	Selection of the unit for symbols that contain multiple units.
	If they exist, display the associated documents.
	Close the browser and place the selected symbol in the schematic.

Creando Netlists y Ficheros BOM Personalizados

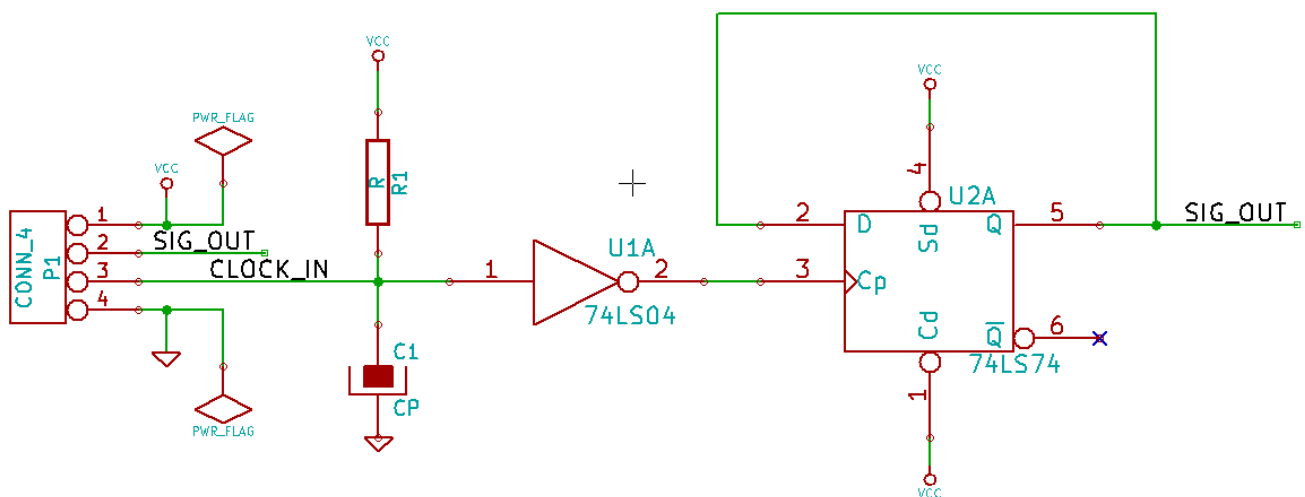
Ficheros Netlist Intermedios

BOM files and netlist files can be converted from an Intermediate netlist file created by KiCad.

Este fichero usa sintaxis XML y es llamado el netlist intermedio. Este netlist intermedio incluye gran cantidad de datos sobre su placa y por ello puede ser usado mediante post-procesamiento para crear una lista BOM u otros informes.

Dependiendo de la salida (BOM o netlist), diferentes subconjuntos del fichero netlist intermedio completo serán usados en el post-procesado.

Ejemplo de Esquema



Ejemplo del fichero Netlist Intermedio

El netlist intermedio correspondiente (usando sintaxis XML) al circuito anterior es mostrado a continuación.

```

<?xml version="1.0" encoding="utf-8"?>
<export version="D">
  <design>
    <source>F:\kicad_aux\netlist_test\netlist_test.sch</source>
    <date>29/08/2010 20:35:21</date>
    <tool>eeschema (2010-08-28 BZR 2458)-unstable</tool>
  </design>
  <components>
    <comp ref="P1">
      <value>CONN_4</value>
      <libsource lib="conn" part="CONN_4"/>
      <sheetpath names="/" tstamps="/" />
      <tstamp>4C6E2141</tstamp>
    </comp>
    <comp ref="U2">
      <value>74LS74</value>
      <libsource lib="74xx" part="74LS74"/>
      <sheetpath names="/" tstamps="/" />
      <tstamp>4C6E20BA</tstamp>
    </comp>
    <comp ref="U1">
      <value>74LS04</value>
      <libsource lib="74xx" part="74LS04"/>
      <sheetpath names="/" tstamps="/" />
      <tstamp>4C6E20A6</tstamp>
    </comp>
    <comp ref="C1">
      <value>CP</value>
      <libsource lib="device" part="CP"/>
      <sheetpath names="/" tstamps="/" />
      <tstamp>4C6E2094</tstamp>
    </comp>
    <comp ref="R1">
      <value>R</value>
      <libsource lib="device" part="R"/>
      <sheetpath names="/" tstamps="/" />
      <tstamp>4C6E208A</tstamp>
    </comp>
  </components>
  <libparts>
    <libpart lib="device" part="C">
      <description>Condensateur non polarise</description>
      <footprints>
        <fp>SM*</fp>
        <fp>C?</fp>
        <fp>C1-1</fp>
      </footprints>
      <fields>
        <field name="Reference">C</field>
        <field name="Value">C</field>
      </fields>
      <pins>
        <pin num="1" name="~" type="passive"/>
        <pin num="2" name="~" type="passive"/>
      </pins>
    </libpart>
    <libpart lib="device" part="R">
      <description>Resistance</description>
      <footprints>
        <fp>R?</fp>
        <fp>SM0603</fp>
        <fp>SM0805</fp>
      </footprints>
    </libpart>
  </libparts>
</export>

```

Conversión a un nuevo formato de netlist

Aplicando un filtro de post-procesado al fichero netlist intermedio puede generar ficheros netlist en otros formatos, así como ficheros de listas BOM. Como esta conversión es una transformación tipo texto a texto, este filtro de post-procesado puede ser escrito usando Python, XSLT, u otra herramienta capaz de aceptar XML como entrada

XSLT itself is an XML language very suitable for XML transformations. There is a free program called *xsltproc* that you can download and install. 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.

Introduccion de XSLT

El documento que describe las transformaciones XSL (XSLT) está disponible aqui:

<http://www.w3.org/TR/xslt>

Crear un fichero netlist Pads-Pcb

El formato pads-pcb está compuesto por dos secciones.

- La lista de huellas.
- La lista de nodos: Agrupando las referencias de los pads por nodos.

Justo debajo se muestra una hoja de estilo que convierte el fichero netlist intermedio al formato netlist pads-pcb:

```

<?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.

How to use:
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 -->
<xsl:template match="node">
  <xsl:text> </xsl:text>

```

Y aquí se muestra el fichero de salida tipo netlist pads-pcb tras ejecutar xsltproc:

```
*PADS-PCB*
*PART*
P1 unknown
U2 unknown
U1 unknown
C1 unknown
R1 unknown
*NET*
*SIGNAL* GND
U1.7
C1.2
U2.7
P1.4
*SIGNAL* VCC
R1.1
U1.14
U2.4
U2.1
U2.14
P1.1
*SIGNAL* N-4
U1.2
U2.3
*SIGNAL* /SIG_OUT
P1.2
U2.5
U2.2
*SIGNAL* /CLOCK_IN
R1.2
C1.1
U1.1
P1.3

*END*
```

La línea de comando para realizar esta conversión es:

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

Crear un fichero netlist tipo Cadstar

El formato Cadstar está compuesto por dos secciones.

- La lista de huellas.
- La lista de nodos: Agrupando las referencias de los pads por nodos.

Aquí se muestra un fichero de hoja de estilo que realiza esta conversión:

```

<?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 -->
      <xsl:if test="count(node)>1">
            <xsl:variable name="netname">

```

A continuación se muestra el fichero de salida formato Caadstar.

```
.HEA
.TIM 21/08/2010 08:12:08
.APP "eeschema (2010-08-09 BZR 2439)-unstable"
.ADD_COM P1 "CONN_4"
.ADD_COM U2 "74LS74"
.ADD_COM U1 "74LS04"
.ADD_COM C1 "CP"
.ADD_COM R1 "R"

.ADD_TER U1.7 "GND"
. 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
```

Create an OrcadPCB2 netlist file

Este formato tiene solo una sección que es la lista de huellas. Cada huella incluye su lista de pads con referencias a nodos.

Aquí se muestra la hoja de estilo para esta conversión:

```

<?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.

How to use:
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>

<!--
  This template read each component
  (path = /export/components/comp)
  creates lines:
  ( 3EBF7DBD $noname U1 74LS125
    ... pin list ...
  )
  and calls "create_pin_list" template to build the pin list
-->
<xsl:template match="comp">
  <xsl:text> ( </xsl:text>

```


Este es el fichero de salida en formato OrcadPCB2

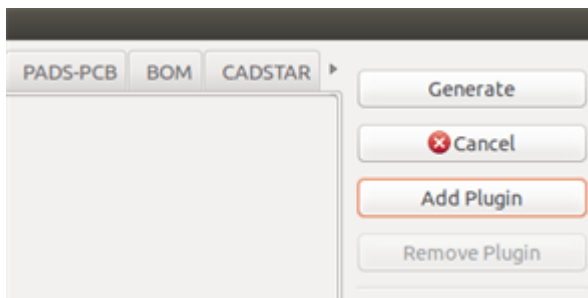
```
( { Eeschema Netlist Version 1.1 29/08/2010 21:07:51
eeschema (2010-08-28 BZR 2458)-unstable}
( 4C6E2141 $noname P1 CONN_4
( 1 VCC )
( 2 /SIG_OUT )
( 3 /CLOCK_IN )
( 4 GND )
)
( 4C6E20BA $noname U2 74LS74
( 1 VCC )
( 2 /SIG_OUT )
( 3 N-04 )
( 4 VCC )
( 5 /SIG_OUT )
( 6 ? )
( 7 GND )
( 14 VCC )
)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

Netlist plugins interface

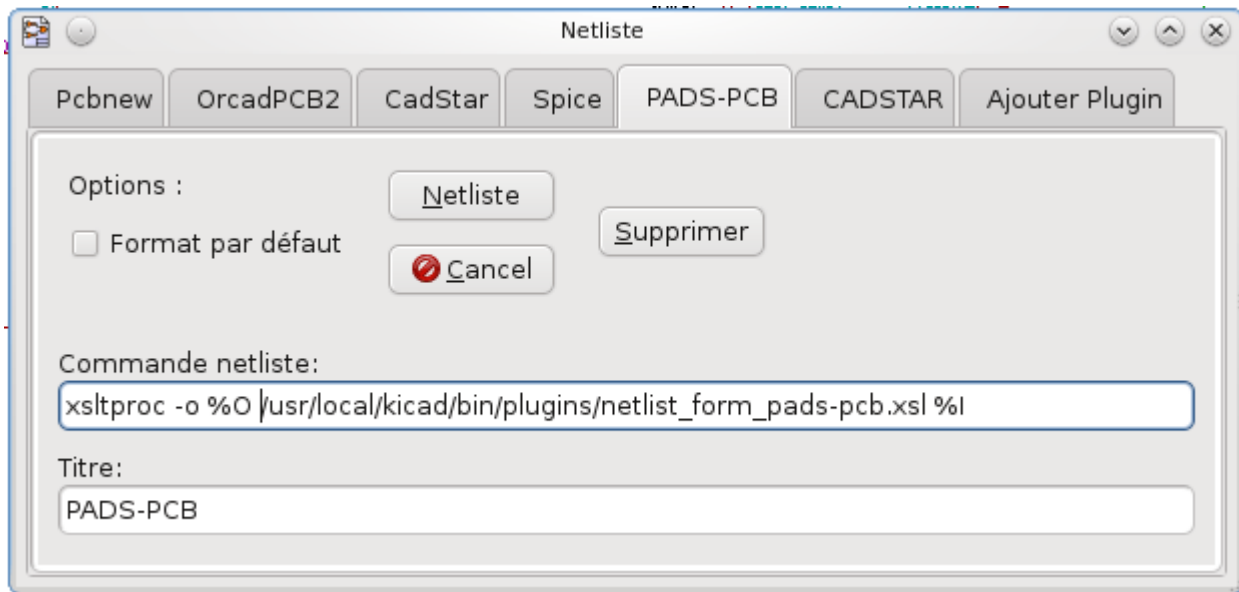
Intermediate Netlist converters can be automatically launched within the Schematic Editor.

Inicie la ventana de dialogo

Ahora puede añadir una nueva interfaz de usuario para el netlist plug-in haciendo clic sobre el botón "Añadir Plugin".



A continuación se muestra el aspecto de los datos de configuración para el formato PadsPcb.



Parámetros de configuración de Plugins

The netlist plug-in configuration dialog requires the following information:

- El título: por ejemplo, el nombre del formato de netlist.
- La comando para lanzar el conversor.

Una vez que hace clic en el boto netlist sucede lo siguiente:

1. KiCad creates an intermediate netlist file *.xml, for instance test.xml.
2. KiCad runs the plug-in by reading test.xml and creates test.net.

Generar ficheros netlist bajo linea de comando.

Suponiendo que estamos usando el programa *xsltproc.exe* para aplicar la hoja de estilo al fichero intermedio, *xsltproc.exe* se ejecutará con el siguiente comando:

```
xsltproc.exe -o <output filename> < style-sheet filename> <input XML file to convert>
```

En KiCad bajo Windows la linea de comando es la siguiente:

```
f:/kicad/bin/xsltproc.exe -o "%O" f:/kicad/bin/plugins/netlist_form_pads-pcb.xml "%I"
```

Bajo Linux la linea de comando es como sigue:

```
xsltproc -o "%O" /usr/local/kicad/bin/plugins/netlist_form_pads-pcb.xml "%I"
```

Where *netlist_form_pads-pcb.xml* is the style-sheet that you are applying. Do not forget the double quotes around the file names, this allows them to have spaces after the substitution by KiCad.

El formato de los comandos acepta parámetros como nombres de archivo.

El formato de los parámetros admitidos es:

- %B = nombre de archivo base y ruta del fichero de salida seleccionado, menos la ruta y extensión.

%I ⇒ nombre de archivo completo y ruta del fichero temporal de entrada (el fichero netlist intermedio)

- %O ⇒ nombre de archivo completo y ruta del fichero de salida elegido.

%I será reemplazado por el nombre de archivo del fichero intermedio actual.

%O será reemplazado por el nombre de archivo del fichero de salida actual.

Formato del comando: ejemplo para xsltproc

El formato del comando para xsltproc es el siguiente:

<path of xsltproc> xsltproc <xsltproc parameters>

bajo Windows:

f:/kicad/bin/xsltproc.exe -o "%O" f:/kicad/bin/plugins/netlist_form_pads-pcb.xml "%I"

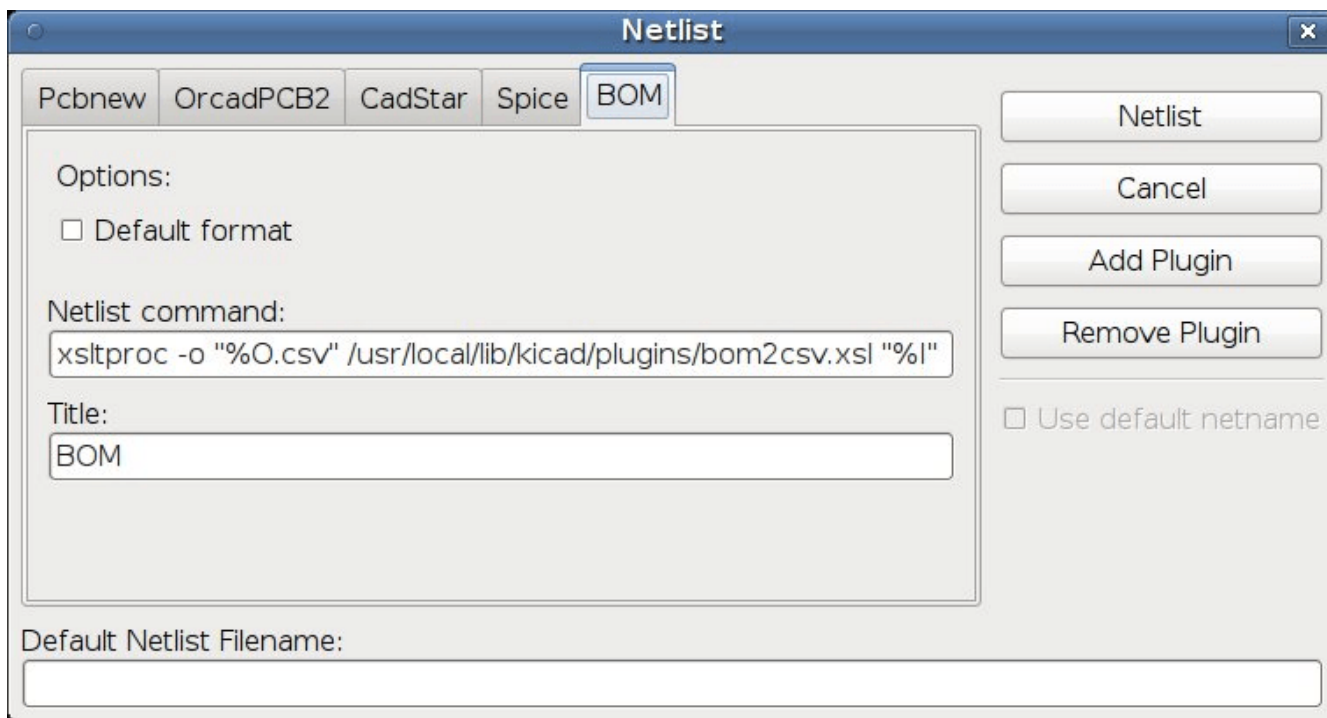
bajo Linux:

xsltproc -o "%O" /usr/local/kicad/bin/plugins/netlist_form_pads-pcb.xml "%I"

Los ejemplos anteriores suponen que xsltproc está instalado en su PC bajo Windows y todos los ficheros están ubicados en kicad/bin.

Generación de Listas de Materiales

Dado que el netlist intermedio contiene toda la información sobre los componentes usados, una lista BOM puede ser extraída de él. Aquí se muestra la ventana de ajuste del plug-in (en Linux) para crear un fichero de lista de materiales (BOM) personalizado:



La ruta al archivo de hoja de estilo bom2csv.xml depende del sistema empleado. La mejor hoja de estilo XSLT actualmente para la generación de listas de materiales es llamada *bom2csv.xml*. Siéntase libre de modificarla de acuerdo a sus necesidades, y si desarrolla algo útil en general, pregunte para que forme parte del proyecto kiCad.

Formato de Comandos: ejemplos para scripts de python

El formato de los comandos para python es algo parecido a:

python <nombre del script> <archivo de entrada> <archivo de salida>

bajo Windows:

python *.exe f:/kicad/python/my_python_script.py "%I" "%O"

bajo Linux:

python /usr/local/kicad/python/my_python_script.py "%I" "%O"

Suponiendo que python esta instalado en su PC.

Estructura del Netlist Intermedio

Este ejemplo muestra una idea del formato del fichero de netlist

```

<?xml version="1.0" encoding="utf-8"?>
<export version="D">
  <design>
    <source>F:\kicad_aux\netlist_test\netlist_test.sch</source>
    <date>29/08/2010 21:07:51</date>
    <tool>eeschema (2010-08-28 BZR 2458)-unstable</tool>
  </design>
  <components>
    <comp ref="P1">
      <value>CONN_4</value>
      <libsource lib="conn" part="CONN_4"/>
      <sheetpath names="/" tstamps="/" />
      <tstamp>4C6E2141</tstamp>
    </comp>
    <comp ref="U2">
      <value>74LS74</value>
      <libsource lib="74xx" part="74LS74"/>
      <sheetpath names="/" tstamps="/" />
      <tstamp>4C6E20BA</tstamp>
    </comp>
    <comp ref="U1">
      <value>74LS04</value>
      <libsource lib="74xx" part="74LS04"/>
      <sheetpath names="/" tstamps="/" />
      <tstamp>4C6E20A6</tstamp>
    </comp>
    <comp ref="C1">
      <value>CP</value>
      <libsource lib="device" part="CP"/>
      <sheetpath names="/" tstamps="/" />
      <tstamp>4C6E2094</tstamp>
    <comp ref="R1">
      <value>R</value>
      <libsource lib="device" part="R"/>
      <sheetpath names="/" tstamps="/" />
      <tstamp>4C6E208A</tstamp>
    </comp>
  </components>
  <libparts/>
  <libraries/>
  <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>
    <net code="3" name="">
      <node ref="U2" pin="6"/>
    </net>
    <net code="4" name="">
      <node ref="U1" pin="2"/>
      <node ref="U2" pin="3"/>
    </net>
  </nets>

```

Estructura del fichero netlist generado

El fichero netlist intermedio cuenta con cinco secciones.

- La cabecera.
- La sección de componentes.
- La sección de bloques de bibliotecas.
- La sección de bibliotecas.
- La sección de nodos.

El contenido del archivo tiene el delimitador <export>

```
<export version="D">
...
</export>
```

La Cabecera

La cabecera tiene el delimitador <design>

```
<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>
```

Esta sección puede ser considerada como una sección para comentarios.

La sección de componentes

La sección de componentes tiene el delimitador <components>

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

Esta sección contiene la lista de componentes en el esquema eléctrico. Cada componente es descrito como sigue:

```

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

```

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 (time stamps)	time stamp of the schematic file.
tstamp (time stamp)	time stamp of the component.

Nota sobre los registros temporales para los componentes

Para identificar un componente en un netlist y por lo tanto en una placa, el registro temporal es usado como único para cada componente. Sin embargo KiCad dispone de una forma auxiliar para identificar un componente que consiste en la correspondiente huella sobre la placa. Esto permite la re-anotación de componentes en el esquema del proyecto sin la pérdida del enlace entre el componente y su huella.

Una marca temporal es un identificador único para cada componente o hoja en el esquema. De todas formas, en jerarquías complejas, la misma hoja es usada mas de una vez, por eso esta hoja contiene componentes con la misma marcha temporal

Una hoja concreta dentro de una jerarquía compleja tiene un identificador único: su ruta. Un componente concreto (dentro de una jerarquía compleja) tiene un identificador único: la ruta de su hoja + su marca temporal.

La sección Libparts

La sección libparts tiene el delimitador <libparts>, y su contenido es definido en las bibliotecas del esquema. La sección libparts contiene

- The allowed footprints names (names use wildcards) delimiter <fp>.
- Los campos definidos en la biblioteca con delimitador <fields>.
- La lista de pines con delimitador <pins>.

```

<libparts>
<libpart lib="device" part="CP">
  <description>Condensateur polarise</description>
  <footprints>
    <fp>CP*</fp>
    <fp>SM*</fp>
  </footprints>
  <fields>
    <field name="Reference">C</field>
    <field name="Valeur">CP</field>
  </fields>
  <pins>
    <pin num="1" name="1" type="passive"/>
    <pin num="2" name="2" type="passive"/>
  </pins>
</libpart>
</libparts>

```

Las líneas como `<pin num="1" type="passive"/>` indican, además, las especificaciones eléctricas del pin. Los posibles tipos eléctricos para los pines son

Input	Pin de entrada
Output	Pin de salida
Bidirectional	Entrada o salida
Tri-state	Entrada/salida de bus
Passive	Finales de componentes pasivos
Unspecified	Tipo eléctrico desconocido
Power input	Pin de entrada de potencia de un componente
Power output	Pin de salida de potencia como las salidas de reguladores
Open collector	Salidas a colector abierto a menudo encontradas en comparadores analógicos
Open emitter	Salidas a emisor abierto aveces encontradas en lógica.
Not connected	Debe ser dejado sin conectar en el esquema

La sección bibliotecas

La sección bibliotecas tiene el delimitador `<libraries>`. Esta sección contiene la lista de bibliotecas usada en el proyecto.


```

<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>

```

La sección nodos

La sección nodos usa el delimitador <nets>. Esta sección especifica la "conectividad" del esquema.

```

<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>

```

Esta sección contiene todos los nodos del esquema.

Un posible nodo contiene lo siguiente.

```

<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	es un identificador interno para este nodo
name	es el nombre del nodo
node	indica la referencia de un pin conectado a este nodo

Mas sobre xsltproc

Consultar la página: <http://xmlsoft.org/XSLT/xsltproc.html>

Introducción

xsltproc es una herramienta bajo línea de comando para aplicar hojas de estilo XSLT a documentos XML. Aunque fue desarrollada como parte del proyecto GNOME, puede funcionar independientemente del escritorio GNOME.

xsltproc es invocado desde la consola con el nombre de la hoja de estilo a ser usada seguida del nombre del fichero o ficheros a los que ha de aplicarse la hoja de estilo. Usará la entrada estándar si el nombre de fichero indicado es -.

Si una hoja de estilo es incluida en un documento XML con instrucciones de procesamiento, no se necesita llamar a ninguna hoja de estilo en el comando. xsltproc detectará automáticamente la hoja de estilo incluida y la usará. Por defecto, la salida la redirige a *stdout*. Puede especificar un fichero como salida usando la opción -o.

Resumen

```
xsltproc [[-V] | [-v] | [-o *file* ] | [--timing] | [--repeat] |
[--debug] | [--novalid] | [--noout] | [--maxdepth *val* ] | [--html] |
[--param *name* *value* ] | [--stringparam *name* *value* ] | [--nonet] |
[--path *paths* ] | [--load-trace] | [--catalogs] | [--xinclude] |
[--profile] | [--dumpextensions] | [--nowrite] | [--nomkdir] |
[--writesubtree] | [--nodtdattr]] [ *stylesheet* ] [ *file1* ] [ *file2* ]
[ *....* ]
```

Opciones del comando

-V o --version

Muestra las versiones de libxml y libxslt usadas.

-v o --verbose

Muestra cada paso tomado por xsltproc durante el procesado de la hoja de estilo y el documento.

-o o --output file

Redirige la salida al fichero llamado *file*. Para múltiples salidas, también conocido como ``chunking'', -o directorio/ redirige los ficheros de salida al directorio especificado. El directorio de existir previamente.

--timing

Muestra el tiempo usado para traducir la hoja de estilo, traducir el documento, aplicar la hoja de estilo y guardar el resultado. Se muestra en milisegundos.

--repeat

Usa la transformación 20 veces. Usado para test de tiempos.

--debug

Muestra un árbol XML del documento transformado con propósitos de depuración.

--novalid

Omite la carga del DTD del documento.

--noout

No muestra el resultado.

--maxdepth value

Ajusta la profundidad máxima de la pila de la plantilla para evitar que libxslt entre en un bucle infinito. Por defecto es 500.

--html

El fichero de entrada es un fichero HTML

--param name value

Pasa un parámetro de nombre *name* y valor *value* a la hoja de estilo. Puede pasar múltiples parejas nombre/valor hasta un máximo de 32. Si el valor que se pasa es una cadena en vez de un identificador de nodo, use *--stringparam* en su lugar.

--stringparam name value

Pasa un parámetro de nombre *name* y valor *value* donde *value* es una cadena en vez de un identificador de nodo. (Nota: la cadena debe estar en formato utf-8)

--nonet

No utiliza Internet para buscar los DTD, entidades o documentos.

--path paths

Use the list (separated by space or column) of filesystem paths specified by *paths* to load DTDs, entities or documents.

--load-trace

Muestra en stderr todos los documentos cargados durante el proceso.

--catalogs

Usa el catálogo especificado en SGML_CATALOG_FILES para resolver la ubicación de entidades externas. Por defecto, xsltproc busca el catálogo especificado en XML_CATALOG_FILES. Si este no se especifica, usa /etc/xml/catalog.

--xinclude

Procesa el documento de entrada usando las especificaciones Xinclude. Mas detalles sobre esto pueden obtenerse en las especificaciones Xinclude: <http://www.w3.org/TR/xinclude/>

--profile --norman

Muestra información de evaluación indicando la cantidad de tiempo consumida en cada parte de la hoja de estilo. Esto es útil para la optimización del rendimiento de las hojas de estilo.

--dumpextensions

Vuelca la lista de todas la extensiones registradas por stdout.

--nowrite

Rechaza la escritura a cualquier archivo o recurso.

--nomkdir

Rechaza la creación de directorios.

--writesubtree path

Permite la escritura de ficheros solo en la ruta especificada en *path*.

--nodtdattr

No aplica los atributos por defecto de los DTD de los documentos.

Valores de retorno para Xsltproc

Xsltproc devuelve un numero de estado que puede ser bastante útil cuando se llama desde un script.

0: normal

1: sin argumento

2: demasiados parámetros

3: opción desconocida

4: fallo al traducir la hoja de estilo

5: error en la hoja de estilo

6: error en uno de los documentos.

7: método de salida:xsl no soportado

8: parámetro cadena contiene comilla simples y dobles

9: error de procesamiento interno

10: El proceso fue interrumpido por un mensaje de finalización

11: no se puede escribir el resultado al fichero de salida

Mas información sobre xsltproc

página web de libxml: <http://www.xmlsoft.org/>

Página W3C XSLT: <http://www.w3.org/TR/xslt>

Simulator

KiCad provides an embedded electrical circuit simulator using [ngspice](#) as the simulation engine.

When working with the simulator, you may find the official *pspice* library useful. It contains common symbols used for simulation like voltage/current sources or transistors with pins numbered to match the ngspice node order specification.

There are also a few demo projects to illustrate the simulator capabilities. You will find them in *demos/simulation* directory.

Assigning models

Before a simulation is launched, components need to have Spice model assigned.

Each component can have only one model assigned, even if component consists of multiple units. In such case, the first unit should have the model specified.

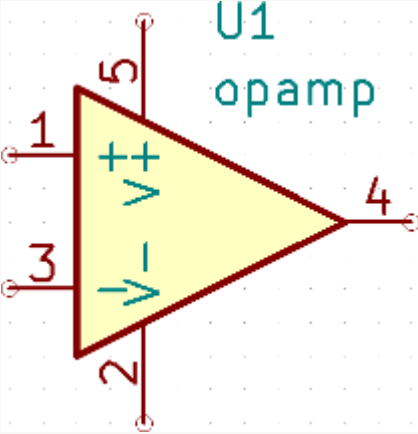
Passive components with reference matching a device type in Spice notation (*R** for resistors, *C** for capacitors, *L** for inductors) will have models assigned implicitly and use the value field to determine their properties.

NOTE

Keep in mind that in Spice notation 'M' stands for milli and 'Meg' corresponds to mega. If you prefer to use 'M' to indicate mega prefix, you may request doing so in the [simulation settings dialog](#).

Spice model information is stored as text in symbol fields, therefore you may either define it in symbol editor or schematics editor. Open symbol properties dialog and click on *Edit Spice Model* button to open Spice Model Editor dialog.

Spice Model Editor dialog has three tabs corresponding to different model types. There are two options common to all model types:

Disable symbol for simulation	When checked the component is excluded from simulation.
Alternate node sequence	<p>Allows one to override symbol pin to model node mapping. To define a different mapping, specify pin numbers in order expected by the model.</p> <p>'Example:'</p> <p>“ * connections:</p> <ul style="list-style-type: none"> * 1: non-inverting input * 2: inverting input * 3: positive power supply * 4: negative power supply * 5: output <p>.subckt tl071 1 2 3 4 5</p>  <p>To match the symbol pins to the Spice model nodes shown above, one needs to use an alternate node sequence option with value: "1 3 5 2 4". It is a list of pin numbers corresponding to the Spice model nodes order.</p>

Pasivo

Passive tab allows the user to assign a passive device model (resistor, capacitor or inductor) to a component. It is a rarely used option, as normally passive components have models assigned [implicitly](#), unless component reference does not match the actual device type.

NOTE

Explicitly defined passive device models have priority over the ones assigned implicitly. It means that once a passive device model is assigned, the reference and value fields are not taken into account during simulation. It may lead to a confusing situation when assigned model value does not match the one displayed on a schematic sheet.

Spice Model Editor

Passive

Model

Source

Type: Resistor

Passive type

Value: 1k

Spice value in simulation

In Spice values,the decimal separator is the point.
Values can use Spice unit symbols.

Spice unit symbols in values (case insensitive):

f	femto	1e-15
p	pico	1e-12
n	nano	1e-9
u	micro	1e-6
m	milli	1e-3
k	kilo	1e3
meg	mega	1e6
g	giga	1e9
t	tera	1e12

☐ Disable symbol for simulation

☐ Alternate node sequence:

Cancel

OK

Type	Selects the device type (resistor, capacitor or inductor).
Value	Defines the device property (resistance, capacitance or inductance). The value may use common Spice unit prefixes (as listed below the text input field) and should use point as the decimal separator. Note that Spice does not correctly interpret prefixes intertwined in the value (e.g. 1k5).

Model

Model tab is used to assign a semiconductor or a complex model defined in an external library file. Spice model libraries are often offered by device manufacturers.

The main text widget displays the selected library file contents. It is a common practice to put models description inside library files, including the node order.

The image shows a 'Spice Model Editor' window with three tabs: 'Passive', 'Model', and 'Source'. The 'Model' tab is active. It contains the following fields and controls:

- Library:** A text field containing 'ad8051.lib' and a 'Select file...' button.
- Model:** A dropdown menu showing 'AD8051'.
- Type:** A dropdown menu showing 'Subcircuit'.
- Model Definition Text Area:** A large text area containing the following SPICE code:

```
.SUBCKT AD8051 1      2      99      50      45
*
* INPUT STAGE
*
Q1  4  3  5  QPI
Q2  6  2  7  QPI
RC1  50 4  20.5k
RC2  50 6  20.5k
RE1  5  8  5k
RE2  7  8  5k
EOS  3  1  POLY(1) 53 98 1.7E-3 1
IOS  1  2  0.1u
FNOI1 1  0  VMEAS2 1E-4
FNOI2 2  0  VMEAS2 1E-4

CPAR1 3 50 1.7p
CPAR2 2 50 1.7p
VCMH1 99 9 1
VCMH2 99 10 1
D1     5  9  DX
D2     7 10  DX
IBIAS 99  8 73u
*
* INTERNAL VOLTAGE REFERENCE
*
EREF1 98  0  POLY(2) 99 0 50 0 0 0.5 0.5
```
- Options:** Two checkboxes at the bottom: 'Disable symbol for simulation' and 'Alternate node sequence:'. The 'Alternate node sequence:' checkbox is followed by an empty text field.
- Buttons:** 'Cancel' and 'OK' buttons at the bottom right.

File	Path to a Spice library file. This file is going to be used by the simulator, as it is added using <i>.include</i> directive.
Model	Selected device model. When a file is selected, the list is filled with available models to choose from.
Type	Selects model type (subcircuit, BJT, MOSFET or diode). Normally it is set automatically when a model is selected.

Source

Source tab is used to assign a power or signal source model. There are two sections: *DC/AC analysis* and *Transient analysis*. Each defines source parameters for the corresponding simulation type.

Source type option applies to all simulation types.

The image shows the 'Spice Model Editor' dialog box with three tabs: 'Passive', 'Model', and 'Source'. The 'Source' tab is selected. It contains two main sections: 'DC/AC analysis' and 'Transient analysis'.

DC/AC analysis:

- DC: [] Volts/Amps
- AC magnitude: 1 [] Volts/Amps
- AC phase: [] radians

Transient analysis:

Four sub-tabs are available: 'Pulse' (selected), 'Sinusoidal', 'Exponential', and 'Piece-wise Linear'.

Pulse settings:

- Initial value: [] Volts/Amps
- Pulsed value: [] Volts/Amps
- Delay time: [] seconds
- Rise time: [] seconds
- Fall time: [] seconds
- Pulse width: [] seconds
- Period: [] seconds

Source type:

- ☒ Voltage ☐ Current

Additional options:

- ☐ Disable symbol for simulation
- ☐ Alternate node sequence: []

At the bottom right are 'Cancel' and 'OK' buttons.

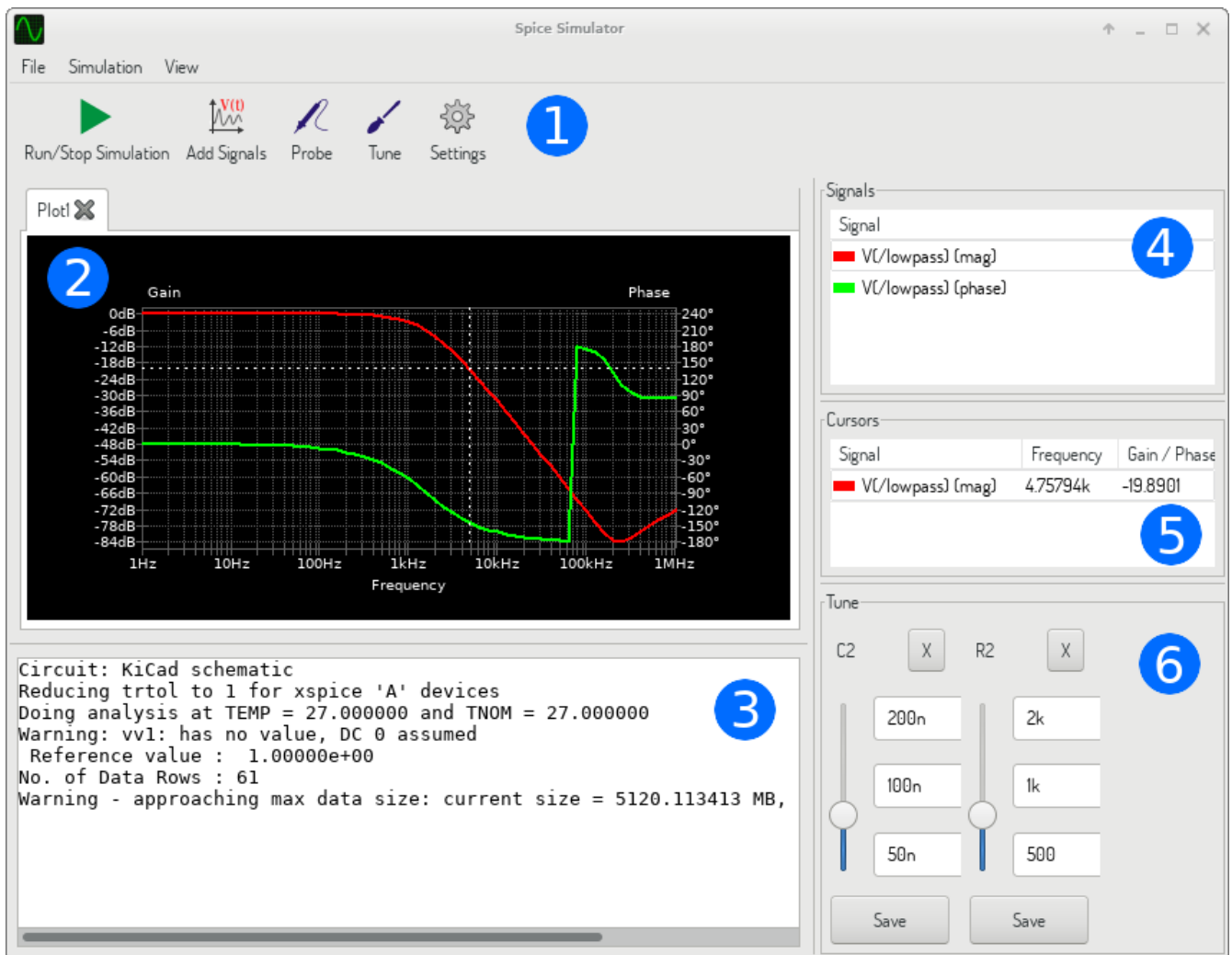
Refer to the [ngspice documentation](#), chapter 4 (Voltage and Current Sources) for more details about sources.

Spice directives

It is possible to add Spice directives by placing them in text fields on a schematic sheet. This approach is convenient for defining the default simulation type. This functionality is limited to Spice directives starting with a dot (e.g. ".tran 10n 1m"), it is not possible to place additional components using text fields.

Simulation

To launch a simulation, open *Spice Simulator* dialog by selecting menu *Tools* → *Simulator* in the schematics editor window.



The dialog is divided into several sections:

- [Toolbar](#)
- [Plot panel](#)
- [Output console](#)
- [Signals list](#)
- [Cursors list](#)
- [Tune panel](#)

Menu

File

New Plot	Create a new tab in the plot panel.
Open Workbook	Open a list of plotted signals.
Save Workbook	Save a list of plotted signals.
Save as image	Export the active plot to a .png file.
Save as .csv file	Export the active plot raw data points to a .csv file.
Exit Simulation	Close the dialog.

Simulation

Run Simulation	Perform a simulation using the current settings.
Add signals...	Open a dialog to select signals to be plotted.
Probe from schematics	Start the schematics Probe tool.
Tune component value	Start the Tuner tool.
Show SPICE Netlist...	Open a dialog showing the generated netlist for the simulated circuit.
Settings...	Open the simulation settings dialog .

View

Zoom In	Zoom in the active plot.
Zoom Out	Zoom out the active plot.
Fit on Screen	Adjust the zoom setting to display all plots.
Show grid	Toggle grid visibility.
Show legend	Toggle plot legend visibility.

Toolbar



The top toolbar provides access to the most frequently performed actions.

Run/Stop Simulation	Start or stop the simulation.
Add Signals	Open a dialog to select signals to be plotted.
Probe	Start the schematics Probe tool.
Tune	Start the Tuner tool.
Settings	Open the simulation settings dialog .

Plot panel

Visualizes the simulation results as plots. One can have multiple plots opened in separate tabs, but only the active one is updated when a simulation is executed. This way it is possible to compare simulation results for different runs.

Plots might be customized by toggling grid and legend visibility using [View](#) menu. When a legend is visible, it can be dragged to change its position.

Plot panel interaction:

- scroll mouse wheel to zoom in/out
- right click to open a context menu to adjust the view
- draw a selection rectangle to zoom in the selected area
- drag a cursor to change its coordinates

Output console

Output console displays messages from the simulator. It is advised to check the console output to verify there are no errors or warnings.

Signals list

Shows the list of signals displayed in the active plot.

Signals list interaction:

- right click to open a context menu to hide signal or toggle cursor
- double click to hide signal

Cursors list

Shows the list of cursors and their coordinates. Each signal may have one cursor displayed. Cursors visibility is set using the [Signals](#) list.

Tune panel

Displays components picked with the [Tuner](#) tool. Tune panel allows the user to quickly modify component values and observe their influence on the simulation results - every time a component value is changed, the simulation is rerun and plots are updated.

For each component there a few controls associated:

The top text field sets the maximum component value.

- The middle text field sets the actual component value.
- The bottom text field sets the minimum component value.
- Slider allows the user to modify the component value in a smooth way.
- *Save* button modifies component value on the schematics to the one selected with the slider.
- *X* button removes component from the Tune panel and restores its original value.

The three text fields recognize Spice unit prefixes.

Tuner tool

Tuner tool lets the user pick components for tuning.

To select a component for tuning, click on one in the schematics editor when the tool is active. Selected components will appear in the [Tune](#) panel. Only passive components might be tuned.

Probe tool

Probe tool provides an user-friendly way of selecting signals for plotting.

To add a signal to plot, click on a corresponding wire in the schematics editor when the tool is active.

Simulation settings

Simulation settings

AC DC Transfer Transient Custom

Number of points:

Start frequency: Hertz

Stop frequency: Hertz

☐ Adjust passive symbol values (e.g. M -> Meg; 100 nF -> 100n)

☒ Add full path for .include library directives

Cancel OK

Simulation settings dialog lets the user set the simulation type and parameters. There are four tabs:

- AC
- DC Transfer
- Transient
- Custom

The first three tabs provide forms where simulation parameters might be specified. The last tab allows the user to type in custom Spice directives to set up a simulation. You can find more information about simulation types and parameters in the [ngspice documentation](#), chapter 1.2.

An alternative way to configure a simulation is to type [Spice directives](#) into text fields on schematics. Any text field directives related to simulation type are overridden by the settings selected in the dialog. It means

that once you start using the simulation dialog, the dialog overrides the schematics directives until the simulator is reopened.

There are two options common to all simulation types:

Adjust passive symbol values	Replace passive symbol values to convert common component values notation to Spice notation.
Add full path for .include library directives	Prepend Spice model library file names with full path. Normally full path is required by ngspice to access a library file.