

KiCad Command-Line Interface

The KiCad Team

Table of Contents

Introduction to the KiCad Command-Line Interface	2
<code>kicad-cli</code> <code>commands</code>	3
Footprint	3
PCB	4
Schematic	12
Symbol	17
Version	18

Reference manual

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 © 2023 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.

All trademarks within this guide belong to their legitimate owners.

Contributors

Graham Keeth

Feedback

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

Introduction to the KiCad Command-Line Interface

KiCad provides a command-line interface, which is available by running the `kicad-cli` binary. With the command-line interface, you can perform a number of actions on schematics, PCBs, symbols, and footprints in an automated fashion, such as plotting Gerber files from a PCB design or upgrading a symbol library from a legacy file format to a modern format.

NOTE

On macOS, the `kicad-cli` executable is located at `/Applications/KiCad/KiCad.app/Contents/MacOS/kicad-cli`.

kicad-cli commands

The `kicad-cli` command has 5 subcommands: `fp`, `pcb`, `sch`, `sym`, and `version`. Each subcommand may have its own subcommands and arguments. For example, to export Gerber files from a PCB you could run `kicad-cli pcb export gerbers example.kicad_pcb`.

You can add the `--help` or `-h` flag to see information about each subcommand. For example, running `kicad-cli pcb -h` prints usage information about the `pcb` subcommand, and `kicad-cli pcb export gerbers -h` prints usage information specifically for the `pcb export gerbers` subcommand.

Footprint

The `fp` subcommand exports footprints to another format or upgrades the footprint libraries to the current version of the KiCad footprint file format.

Footprint export

The `fp export svg` command exports one or more footprints from the specified library into SVG files.

Usage: `kicad-cli fp export svg [-h] [--output VAR] [--layers VAR] [--theme VAR] [--footprint VAR] [--black-and-white] input`

Positional arguments:

<code>input</code>	Footprint library to export (.pretty).
--------------------	--

Optional arguments:

<code>-h, --help</code>	Show help for the footprint SVG export command.
<code>-o <output dir>, --output <output dir></code>	The output directory for the exported files. When this argument is not used, the files are exported to the current directory.
<code>-l <layer list>, --layers <layer list></code>	A comma-separated list of layer names to export from the footprint, such as F.Cu,B.Cu. If no layers are given, all layers are exported.
<code>-t <theme name>, --theme <theme name></code>	The name of the theme to use for export. If no theme is given, the footprint editor's currently selected theme is used.
<code>--fp <footprint>, --footprint <footprint></code>	The specific footprint to export from the library. When this argument is not used, all footprints in the library are exported.
<code>--black-and-white</code>	Export footprints in black and white.

Footprint upgrade

The `fp upgrade` command upgrades the the specified footprint library from a legacy footprint format to the native format for the current version of KiCad. If the input library is already in the current file format, no action is taken.

Usage: `kicad-cli fp upgrade [-h] [--output VAR] [--force] input`

Positional arguments:

input	Footprint library to upgrade (.pretty).
-------	---

Optional arguments:

-h, --help	Show help for the footprint upgrade command.
-o <output dir>, --output <output dir>	The output directory for the upgraded footprints. When this argument is not used, the upgraded footprints are saved over the original footprints.
--force	Re-save the input library even if it is already in the current file format.

PCB

The `pcb export` command exports a board to various other file formats, including fabrication and 3D files. Each file format has its own options.

PCB drill file export

The `pcb export drill` command exports a drill file from a board.

Usage: `kiCad-cli pcb export drill [-h] [--output VAR] [--format VAR] [--drill-origin VAR] [--excellon-zeros-format VAR] [--excellon-units VAR] [--excellon-mirror-y] [--excellon-min-header] [--excellon-separate-th] [--generate-map] [--map-format VAR] [--gerber-precision VAR] input`

Positional arguments:

input	Board file to export.
-------	-----------------------

Optional arguments:

<code>-h, --help</code>	Show help for the drill file export command.
<code>-o <output dir>, --output <output dir></code>	The output directory for the drill file. When this argument is not used, the drill file is saved in the current directory.
<code>--format <format></code>	The drill file format. Options are <code>excellon</code> (default) or <code>gerber</code> .
<code>--drill-origin <origin></code>	The coordinate origin for the drill file. Options are <code>absolute</code> (default) to use the board's absolute origin or <code>plot</code> to use the board's drill/placement origin.
<code>--excellon-zeros-format <format></code>	The zeros format for the drill file. Options are <code>decimal</code> (default), <code>suppressleading</code> , <code>suppresstrailing</code> , or <code>keep</code> . Only applies to Excellon format drill files.
<code>-u <units>, --excellon-units <units></code>	The units for the drill file. Options are <code>mm</code> (default) or <code>in</code> . Only applies to Excellon format drill files.
<code>--excellon-mirror-y</code>	Mirror the drill file in the Y direction. Only applies to Excellon format drill files.
<code>--excellon-min-header</code>	Use a minimal header in the drill file. Only applies to Excellon format drill files.
<code>--excellon-separate-th</code>	Generate separate drill files for plated and non-plated through holes. Only applies to Excellon format drill files.
<code>--excellon-oval-format <format></code>	Control the oval holes drill mode. Options are <code>route</code> (default) and <code>alternate</code> . Only applies to Excellon format drill files.
<code>--generate-map</code>	Generate a map file in addition to the drill file.
<code>--map-format <format></code>	The map file format. Options are <code>pdf</code> (default), <code>gerberx2</code> , <code>ps</code> , <code>dxg</code> , or <code>svg</code> .
<code>--gerber-precision <precision></code>	The precision (number of digits) for the drill file. Valid options are <code>5</code> or <code>6</code> (default). Only applies to Gerber format drill files.

PCB DXF export

The `pcb export dxf` command exports a board design to a DXF file.

Usage: `kicad-cli pcb export dxf [-h] [--output VAR] [--layers VAR] [--exclude-refdes] [--exclude-value] [--use-contours] [--output-units VAR] input`

Positional arguments:

<code>input</code>	Board file to export.
--------------------	-----------------------

Optional arguments:

<code>-h, --help</code>	Show help for the DXF export command.
<code>-o <output filename>, --output <output filename></code>	The output filename. When this argument is not used, the output filename will be the same as the input file, with the <code>.dxf</code> file extension.
<code>-l <layer list>, --layers <layer list></code>	A comma-separated list of layer names to export from the footprint, such as <code>F.Cu,B.Cu</code> . At least one layer must be given.
<code>--erd, --exclude-refdes</code>	Exclude footprint reference designators from plot.
<code>--ev, --exclude-value</code>	Exclude footprint values from plot.
<code>--uc, --use-contours</code>	Plot graphic items using their contours.
<code>--ou <unit>, --output-units <unit></code>	Output units. Options are <code>mm</code> or <code>in</code> (default).

PCB Gerber export: one layer per file

The `pcb export gerbers` command exports a board design to Gerber files, with one layer per file.

Usage: `kicad-cli pcb export gerbers [-h] [--output VAR] [--layers VAR] [--exclude-refdes] [--exclude-value] [--include-border-title] [--no-x2] [--no-netlist] [--subtract-soldermask] [--disable-aperture-macros] [--use-drill-file-origin] [--precision VAR] [--no-protel-ext] [--common-layers VAR] [--board-plot-params] input`

Positional arguments:

<code>input</code>	Board file to export.
--------------------	-----------------------

Optional arguments:

<code>-h, --help</code>	Show help for the Gerber export command.
<code>-o <output dir>, --output <output dir></code>	The output folder for the exported files. When this argument is not used, the files are exported to the current directory.
<code>-l <layer list>, --layers <layer list></code>	A comma-separated list of layer names to plot from the board, such as <code>F.Cu,B.Cu</code> . If this argument is not used, all layers will be plotted.
<code>--erd, --exclude-refdes</code>	Exclude footprint reference designators from plot.
<code>--ev, --exclude-value</code>	Exclude footprint values from plot.
<code>--ibt, --include-border-title</code>	Include the sheet border and title block.
<code>--no-x2</code>	Do not use the extended X2 format.
<code>--no-netlist</code>	Do not include netlist attributes.
<code>--subtract-soldermask</code>	Remove silkscreen from areas without soldermask.
<code>--disable-aperture-macros</code>	Disable aperture macros.
<code>--use-drill-file-origin</code>	Use drill/place file origin instead of absolute origin.
<code>--precision <precision></code>	The precision (number of digits) for the Gerber files. Valid options are 5 or 6 (default).
<code>--no-protel-ext</code>	Use <code>.gbr</code> file extension instead of Protel file extensions (<code>.gbl</code> , <code>.gtl</code> , etc.).
<code>--cl <layer list>, --common-layers <layer list></code>	A comma-separated list of layer names to plot on all layers, such as <code>F.Cu,B.Cu</code> .
<code>--board-plot-params</code>	Use the Gerber plot settings already configured in the board file.

PCB Gerber export: multiple layers per file

The `pcb export gerber` command exports one or more board layers to a single Gerber file.

Usage: `kicad-cli pcb export gerber [-h] [--output VAR] [--layers VAR] [--exclude-refdes] [--exclude-value] [--include-border-title] [--no-x2] [--no-netlist] [--subtract-soldermask] [--disable-aperture-macros] [--use-drill-file-origin] [--precision VAR] [--no-protel-ext] input`

Positional arguments:

input	Board file to export.
-------	-----------------------

Optional arguments:

<code>-h, --help</code>	Show help for the Gerber export command.
<code>-o <output filename>, --output <output filename></code>	The output filename. When this argument is not used, the output filename will be the same as the input file, with the <code>.gbr</code> file extension.
<code>-l <layer list>, --layers <layer list></code>	A comma-separated list of layer names to plot from the board, such as <code>F.Cu,B.Cu</code> . All layers will be plotted in the output file. At least one layer must be given.
<code>--erd, --exclude-refdes</code>	Exclude footprint reference designators from plot.
<code>--ev, --exclude-value</code>	Exclude footprint values from plot.
<code>--ibt, --include-border-title</code>	Include the sheet border and title block.
<code>--no-x2</code>	Do not use the extended X2 format.
<code>--no-netlist</code>	Do not include netlist attributes.
<code>--subtract-soldermask</code>	Remove silkscreen from areas without soldermask.
<code>--disable-aperture-macros</code>	Disable aperture macros.
<code>--use-drill-file-origin</code>	Use drill/place file origin instead of absolute origin.
<code>--precision <precision></code>	The precision (number of digits) for the Gerber files. Valid options are 5 or 6 (default).
<code>--no-protel-ext</code>	Use <code>.gbr</code> file extension instead of Protel file extensions (<code>.gbl</code> , <code>.gtl</code> , etc.).

PCB PDF export

The `pcb export pdf` command exports a board design to a PDF file.

Usage: `kicad-cli pcb export pdf [-h] [--output VAR] [--layers VAR] [--mirror] [--exclude-refdes] [--exclude-value] [--include-border-title] [--negative] [--black-and-white] [--theme VAR] input`

Positional arguments:

input	Board file to export.
-------	-----------------------

Optional arguments:

<code>-h, --help</code>	Show help for the PDF export command.
<code>-o <output filename>, --output <output filename></code>	The output filename. When this argument is not used, the output filename will be the same as the input file, with the <code>.pdf</code> file extension.
<code>-l <layer list>, --layers <layer list></code>	A comma-separated list of layer names to export from the board, such as <code>F.Cu,B.Cu</code> . At least one layer must be given.
<code>-m, --mirror</code>	Mirror the board. This can be useful for showing bottom layers.
<code>--erd, --exclude-refdes</code>	Exclude footprint reference designators from plot.
<code>--ev, --exclude-value</code>	Exclude footprint values from plot.
<code>--ibt, --include-border-title</code>	Include the sheet border and title block.
<code>-n, --negative</code>	Plot in negative.
<code>--black-and-white</code>	Plot in black and white.
<code>-t <theme name>, --theme <theme name></code>	The name of the theme to use for export. If no theme is given, the board editor's currently selected theme is used.

PCB position file export

The `pcb export pos` command exports a position file from a board design.

Usage: `kicad-cli pcb export pos [-h] [--output VAR] [--side VAR] [--format VAR] [--units VAR] [--bottom-negate-x] [--use-drill-file-origin] [--smd-only] [--exclude-fp-th] [--gerber-board-edge] input`

Positional arguments:

<code>input</code>	Board file to export.
--------------------	-----------------------

Optional arguments:

<code>-h, --help</code>	Show help for the position file export command.
<code>-o <output filename>, --output <output filename></code>	The output filename. When this argument is not used, the output filename will be the same as the input file, with the <code>.pos</code> file extension.
<code>--side <side></code>	The side of the board to export. Options are <code>front</code> , <code>back</code> , or <code>both</code> (default). Gerber format does not support <code>both</code> .
<code>--format <format></code>	The position file format. Options are <code>ascii</code> (default), <code>csv</code> , or <code>gerber</code> .
<code>--units <unit></code>	Units for position file. Options are <code>in</code> (default) or <code>mm</code> . This argument has no effect for Gerber format.
<code>--bottom-negate-x</code>	Use negative X coordinates for footprints on the bottom layer. This argument has no effect for Gerber format.
<code>--use-drill-file-origin</code>	Use drill/place file origin instead of absolute origin. This option has no effect for Gerber format.
<code>--smd-only</code>	Include only surface-mount components. This option has no effect for Gerber format.
<code>--exclude-fp-th</code>	Exclude all footprints with through-hole pads. This option has no effect for Gerber format.
<code>--gerber-board-edge</code>	Include board edge layer in export (Gerber format only).

PCB STEP export

The `pcb export step` command exports a board design to a STEP file.

Usage: `kicad-cli pcb export step [-h] [--drill-origin] [--grid-origin] [--no-virtual] [--subst-models] [--force] [--board-only] [--min-distance VAR] [--user-origin VAR] [--output VAR] input`

Positional arguments:

<code>input</code>	Board file to export.
--------------------	-----------------------

Optional arguments:

<code>-h, --help</code>	Show help for the STEP file export command.
<code>--drill-origin</code>	Use drill origin as origin of output file.
<code>--grid-origin</code>	Use grid origin as origin of output file.
<code>--no-virtual</code>	Exclude 3D models of components with 'virtual' attribute.
<code>--subst-models</code>	Replace VRML models in footprints with STEP models of the same name, if they exist.
<code>-f, --force</code>	Overwrite output file.
<code>--board-only</code>	Only include the board itself in the generated model; exclude all component models.
<code>--min-distance <min distance></code>	Tolerance for considering two points to be in the same location. Default: 0.01mm.
<code>--user-origin <output origin></code>	Specify a custom origin for the output file, with X and Y coordinates. For example, <code>1x1in</code> , <code>1x1inch</code> , or <code>25.4x25.4mm</code> . The default unit is millimeters.
<code>-o <output filename>, --output <output filename></code>	The output filename. When this argument is not used, the output filename will be the same as the input file, with the <code>.step</code> file extension.

PCB SVG export

The `pcb export svg` command exports a board design to an SVG file.

Usage: `kicad-cli pcb export svg [-h] [--output VAR] [--layers VAR] [--mirror] [--theme VAR] [--negative] [--black-and-white] [--page-size-mode VAR] [--exclude-drawing-sheet] input`

Positional arguments:

<code>input</code>	Board file to export.
--------------------	-----------------------

Optional arguments:

<code>-h, --help</code>	Show help for the SVG file export command.
<code>-o <output filename>, --output <output filename></code>	The output filename. When this argument is not used, the output filename will be the same as the input file, with the <code>.svg</code> file extension.
<code>-l <layer list>, --layers <layer list></code>	A comma-separated list of layer names to export from the board, such as <code>F.Cu,B.Cu</code> . At least one layer must be given.
<code>-m, --mirror</code>	Mirror the board. This can be useful for showing bottom layers.
<code>-t <theme name>, --theme <theme name></code>	The name of the theme to use for export. If no theme is given, the board editor's currently selected theme is used.
<code>-n, --negative</code>	Plot in negative.
<code>--black-and-white</code>	Plot in black and white.
<code>--page-size-mode <mode></code>	Set page sizing mode. Options are 0 (default), 1, or 2. 0 sets the output page size to fit the entire sheet, including drawing sheet frame and title block. 1 sets the output page size to match the current page size. 2 sets the output page size to the size of the board itself.
<code>--exclude-drawing-sheet</code>	Plot SVG without a drawing sheet.

Schematic

The `sch export` command exports a schematic to various other file formats, or exports a bill of materials or netlist. Each file format has its own options.

Schematic DXF export

The `sch export dxf` command exports a schematic to a DXF file. Each sheet in the design is exported to its own file.

Usage: `kicad-cli sch export dxf [-h] [--output VAR] [--theme VAR] [--black-and-white] [--exclude-drawing-sheet] [--no-background-color] [--plot-one] input`

Positional arguments:

<code>input</code>	Schematic file to export.
--------------------	---------------------------

Optional arguments:

<code>-h, --help</code>	Show help for the DXF file export command.
<code>-o <output dir>, --output <output dir></code>	The output folder for the exported files. When this argument is not used, the files are exported to the current directory.
<code>-t <theme name>, --theme <theme name></code>	The name of the theme to use for export. If no theme is given, the schematic editor's currently selected theme is used.
<code>-b, --black-and-white</code>	Export schematic in black and white.
<code>-e, --exclude-drawing-sheet</code>	Plot DXF without a drawing sheet.
<code>-n, --no-background-color</code>	Export schematic without a background color, regardless of theme.
<code>-o, --plot-one</code>	Only export the input sheet; do not export any subsheets.

Schematic HPGL export

The `sch export hpgl` command exports a schematic to an HPGL file for a pen plotter. Each sheet in the design is exported to its own file.

Usage: `kicad-cli sch export hpgl [-h] [--output VAR] [--theme VAR] [--black-and-white] [--exclude-drawing-sheet] [--no-background-color] [--plot-one] [--pen-size VAR] [--origin VAR] input`

Positional arguments:

<code>input</code>	Schematic file to export.
--------------------	---------------------------

Optional arguments:

<code>-h, --help</code>	Show help for the HPGL file export command.
<code>-o <output dir>, --output <output dir></code>	The output folder for the exported files. When this argument is not used, the files are exported to the current directory.
<code>-t <theme name>, --theme <theme name></code>	The name of the theme to use for export. If no theme is given, the schematic editor's currently selected theme is used.
<code>-b, --black-and-white</code>	Export schematic in black and white.
<code>-e, --exclude-drawing-sheet</code>	Plot HPGL without a drawing sheet.
<code>-n, --no-background-color</code>	Export schematic without a background color, regardless of theme.
<code>-o, --plot-one</code>	Only export the input sheet; do not export any subsheets.
<code>-p <pen size>, --pen-size <pen size></code>	Set the pen width. The default pen size is 0.5 mm.
<code>-r <origin>, --origin <origin></code>	Set plotter origin and scale. Options are 0, 1 (default), 2, or 3. 0 sets the origin to the bottom left and uses plotter units. 1 sets the origin to the center and uses plotter units. 2 scales to the page, and 3 scales to the content within the page.

Schematic netlist export

The `sch export netlist` command exports a netlist in [various formats](#) from a schematic.

Usage: `kicad-cli sch export netlist [-h] [--output VAR] [--format VAR] input`

Positional arguments:

<code>input</code>	Schematic file to export.
--------------------	---------------------------

Optional arguments:

<code>-h, --help</code>	Show help for the netlist export command.
<code>-o <output filename>, --output <output filename></code>	The output filename. When this argument is not used, the output filename will be the same as the input file, with a <code>.net</code> file extension.
<code>-f <format>, --format <format></code>	The netlist output format. Options are <code>kicadsexpr</code> (default), <code>kicadxml</code> , <code>cadstar</code> , <code>orcadpcb2</code> , <code>spice</code> , or <code>spicemodel</code> .

Schematic PDF export

The `sch export pdf` command exports a schematic to a PDF file. Each sheet in the design is exported to its own page in the PDF file.

Usage: `kicad-cli sch export pdf [-h] [--output VAR] [--theme VAR] [--black-and-white] [--exclude-drawing-sheet] [--no-background-color] [--plot-one] input`

Positional arguments:

input	Schematic file to export.
-------	---------------------------

Optional arguments:

-h, --help	Show help for the PDF file export command.
-o <output filename>, --output <output filename>	The output filename. When this argument is not used, the output filename will be the same as the input file, with a .pdf file extension.
-t <theme name>, --theme <theme name>	The name of the theme to use for export. If no theme is given, the schematic editor's currently selected theme is used.
-b, --black-and-white	Export schematic in black and white.
-e, --exclude-drawing-sheet	Plot PDF without a drawing sheet.
-n, --no-background-color	Export schematic without a background color, regardless of theme.
-o, --plot-one	Only export the input sheet; do not export any subsheets.

Schematic PostScript export

The `sch export ps` command exports a schematic to a PostScript file. Each sheet in the design is exported to its own file.

Usage: `kicad-cli sch export ps [-h] [--output VAR] [--theme VAR] [--black-and-white] [--exclude-drawing-sheet] [--no-background-color] [--plot-one] input`

Positional arguments:

input	Schematic file to export.
-------	---------------------------

Optional arguments:

<code>-h, --help</code>	Show help for the PS file export command.
<code>-o <output dir>, --output <output dir></code>	The output folder for the exported files. When this argument is not used, the files are exported to the current directory.
<code>-t <theme name>, --theme <theme name></code>	The name of the theme to use for export. If no theme is given, the schematic editor's currently selected theme is used.
<code>-b, --black-and-white</code>	Export schematic in black and white.
<code>-e, --exclude-drawing-sheet</code>	Plot PS without a drawing sheet.
<code>-n, --no-background-color</code>	Export schematic without a background color, regardless of theme.
<code>-o, --plot-one</code>	Only export the input sheet; do not export any subsheets.

Schematic bill of materials export

The `sch export python-bom` command exports an XML BOM file from a schematic. The XML BOM file can then be processed into your desired BOM format using a custom script or one of the scripts described in the [schematic BOM export documentation](#).

Usage: `kicad-cli sch export python-bom [-h] [--output VAR] input`

Positional arguments:

input	Schematic file to export.
-------	---------------------------

Optional arguments:

<code>-h, --help</code>	Show help for the BOM export command.
<code>-o <output filename>, --output <output filename></code>	The output filename. When this argument is not used, the output filename will be the same as the input file, with a <code>-bom.xml</code> suffix and file extension.

Schematic SVG export

The `sch export svg` command export a schematic to an SVG file. Each sheet in the design is exported to its own file.

Usage: `kicad-cli sch export svg [-h] [--output VAR] [--theme VAR] [--black-and-white] [--exclude-drawing-sheet] [--no-background-color] [--plot-one] input`

Positional arguments:

input	Schematic file to export.
-------	---------------------------

Optional arguments:

<code>-h, --help</code>	Show help for the SVG file export command.
<code>-o <output dir>, --output <output dir></code>	The output folder for the exported files. When this argument is not used, the files are exported to the current directory.
<code>-t <theme name>, --theme <theme name></code>	The name of the theme to use for export. If no theme is given, the schematic editor's currently selected theme is used.
<code>-b, --black-and-white</code>	Export schematic in black and white.
<code>-e, --exclude-drawing-sheet</code>	Plot SVG without a drawing sheet.
<code>-n, --no-background-color</code>	Export schematic without a background color, regardless of theme.
<code>-o, --plot-one</code>	Only export the input sheet; do not export any subsheets.

Symbol

The `sym` subcommand exports symbols to another format or upgrades symbol libraries to the current version of the KiCad symbol file format.

Symbol export

The `sym export svg` command exports one or more symbols from the specified library into SVG files.

Usage: `kiCad-cli sym export svg [-h] [--output VAR] [--theme VAR] [--symbol VAR] [--black-and-white] [--include-hidden-pins] [--include-hidden-fields] input`

Positional arguments:

<code>input</code>	Symbol library file to use for export.
--------------------	--

Optional arguments:

<code>-h, --help</code>	Show help for the symbol SVG export command.
<code>-o <output dir>, --output <output dir></code>	The output folder for the exported files. When this argument is not used, the files are exported to the current directory.
<code>-t <theme name>, --theme <theme name></code>	The name of the theme to use for export. If no theme is given, the symbol editor's currently selected theme is used.
<code>-s <symbol name>, --symbol <symbol name></code>	The specific symbol to export from the library. When this argument is not used, all symbols in the library are exported.
<code>--black-and-white</code>	Export symbols in black and white.
<code>--include-hidden-pins</code>	Export hidden pins in the exported SVG.
<code>--include-hidden-fields</code>	Export hidden symbol fields in the exported SVG.

Symbol upgrade

The `sym upgrade` command upgrades the the specified symbol library from a legacy symbol format to the native format for the current version of KiCad. If the input library is already in the current file format, no action is taken.

Usage: `kicad-cli sym upgrade [-h] [--output VAR] [--force] input`

Positional arguments:

<code>input</code>	Symbol library to upgrade.
--------------------	----------------------------

Optional arguments:

<code>-h, --help</code>	Show help for the symbol upgrade command.
<code>-o <output filename>, --output <output filename></code>	The output filename for the upgraded symbol library. When this argument is not used, the upgraded symbol library is saved over the original library.
<code>--force</code>	Re-save the input library even if it is already in the current file format.

Version

The `version` subcommand prints the KiCad version. Without any arguments, it simply prints the version number, for example `7.0.7`. You can print the version in several other formats using the `--format` argument.

NOTE

Use `kicad-cli version --format about` for version information to include when submitting bug reports or feature requests on Gitlab.

Usage: `kicad-cli version [-h] [--format VAR]`

Optional arguments:

<code>--format <format></code>	Format of the version number. Options are <code>plain</code> (default), <code>commit</code> , or <code>about</code> . <code>plain</code> prints the version number (e.g. <code>7.0.7</code>), which is the default if the <code>--format</code> argument is not used. <code>commit</code> prints the hash of the git commit for the build of KiCad you are using. <code>about</code> prints the full version information, including library versions and basic system information. You can use the <code>about</code> version information in bug reports.
--------------------------------------	--