



kicad



kicad

Eeschema

**9 października 2018**

---

# Spis treści

<b>1</b>	<b>Wprowadzenie do Eeschema</b>	<b>1</b>
1.1	Opis	1
1.2	Opis techniczny	1
<b>2</b>	<b>Podstawowe polecenia</b>	<b>3</b>
2.1	Polecenia związane z myszą	4
2.1.1	Podstawowe polecenia	4
2.1.2	Block operations	4
2.2	Hotkeys	5
2.3	Grid	7
2.4	Wybór powiększenia - Zoom	7
2.5	Wyświetlanie pozycji kursora	8
2.6	Pasek menu	8
2.7	Górny pasek narzędzi	8
2.8	Prawy pasek narzędzi	11
2.9	Lewy pasek narzędzi	12
2.10	Menu kontekstowe i szybka edycja komponentów	12
<b>3</b>	<b>Menu główne</b>	<b>15</b>
3.1	Menu Plik	15
3.2	Menu ustawień	16
3.2.1	Manage Symbol Library Tables	17
3.2.1.1	Add a new library	17
3.2.1.2	Remove a library	17
3.2.1.3	Library properties	18

---

3.2.2	General Options . . . . .	18
3.2.2.1	Display . . . . .	18
3.2.2.2	Editing . . . . .	19
3.2.2.3	Controls . . . . .	20
3.2.2.4	Colors . . . . .	21
3.2.2.5	Default Fields . . . . .	22
3.3	Menu Pomoc . . . . .	23
<b>4</b>	<b>Główny pasek narzędziowy</b>	<b>24</b>
4.1	Zarządzanie ustawieniami arkuszy/stron . . . . .	24
4.2	Znajdź oraz Znajdź i zamień . . . . .	24
4.3	Narzędzie Listy sieci . . . . .	25
4.4	Numeracja komponentów . . . . .	26
4.5	Kontrola reguł projektowych - ERC . . . . .	27
4.5.1	Główne okno narzędzia ERC . . . . .	28
4.5.2	Opcje sprawdzania ERC . . . . .	29
4.6	Lista materiałowa - BOM . . . . .	30
4.7	Edit Fields tool . . . . .	32
4.8	Import tool for footprint assignment . . . . .	33
4.8.1	Dostęp: . . . . .	33
<b>5</b>	<b>Manage Symbol Libraries</b>	<b>34</b>
5.1	Symbol Library Table . . . . .	34
5.1.1	Global Symbol Library Table . . . . .	35
5.1.2	Project Specific Symbol Library Table . . . . .	35
5.1.3	Initial Configuration . . . . .	36
5.1.4	Adding Table Entries . . . . .	36
5.1.5	Environment Variable Substitution . . . . .	36
5.1.6	Usage Patterns . . . . .	37
5.1.7	Legacy Project Remapping . . . . .	37

---

<b>6 Tworzenie i edycja schematu</b>	<b>39</b>
6.1 Wprowadzenie	39
6.2 Uwagi ogólne	39
6.3 Proces tworzenia	40
6.4 Symbol placement and editing	40
6.4.1 Find and place a symbol	40
6.4.2 Porty zasilania	42
6.4.3 Symbol Editing and Modification (already placed component)	42
6.4.3.1 Symbol modification	42
6.4.3.2 Modyfikacja pól tekstowych elementów	42
6.5 Połączenia, Magistrale, Etykiety i Symbole zasilania	43
6.5.1 Wprowadzenie	43
6.5.2 Połączenia (Łączy i etykiety)	44
6.5.3 Połączenia - Magistrale	45
6.5.3.1 Składniki magistral	45
6.5.3.2 Połączenia pomiędzy składnikami magistral	45
6.5.3.3 Globalne połączenia pomiędzy magistralami	46
6.5.4 Połączenia z symbolami zasilania	47
6.5.5 Flaga "Nie połączone"	48
6.6 Elementy uzupełniające	48
6.6.1 Komentarze	48
6.6.2 Tabelka	48
6.7 Rescuing cached symbols	49
<b>7 Schematy o strukturze hierarchicznej</b>	<b>51</b>
7.1 Wprowadzenie	51
7.2 Nawigacja wewnątrz hierarchii	52
7.3 Etykiety lokalne, hierarchiczne i globalne	52
7.3.1 Właściwości	52
7.4 Summary of hierarchy creation	53
7.5 Symbole arkuszy podrzędnych	53
7.6 Połączenia - Piny hierarchiczne	54
7.7 Etykiety hierarchiczne	55

---

7.7.1	Etykiety, etykiety hierarchiczne, etykiety globalne oraz piny ukryte . . . . .	56
7.7.1.1	Zwykłe etykiety . . . . .	56
7.7.1.2	Etykiety hierarchiczne . . . . .	56
7.7.1.3	Ukryte piny zasilania . . . . .	57
7.7.2	Etykiety globalne . . . . .	57
7.8	Hierarchia złożona . . . . .	57
7.9	Hierarchia płaska . . . . .	58
<b>8</b>	<b>Symbol Annotation Tool</b>	<b>61</b>
8.1	Wprowadzenie . . . . .	61
8.2	Przykłady . . . . .	62
8.2.1	Zmiany porządku numeracji . . . . .	62
8.2.2	Wybór numeracji . . . . .	63
<b>9</b>	<b>Kontrola reguł projektowych - Electrical Rules Check</b>	<b>66</b>
9.1	Wprowadzenie . . . . .	66
9.2	Używanie narzędzia testu ERC . . . . .	67
9.3	Przykład testu ERC . . . . .	68
9.4	Wyświetlanie informacji o znaczniku błędu . . . . .	68
9.5	Piny zasilania i flagi zasilania . . . . .	69
9.6	Konfiguracja . . . . .	70
9.7	Plik raportu ERC . . . . .	71
<b>10</b>	<b>Generowanie list sieci</b>	<b>73</b>
10.1	Przegląd zagadnień . . . . .	73
10.2	Format listy sieci . . . . .	74
10.3	Przykłady list sieci . . . . .	74
10.4	Uwagi odnośnie list sieci . . . . .	77
10.4.1	Zalecane środki ostrożności . . . . .	77
10.4.2	Listy sieci PSPICE . . . . .	77
10.5	Inne formaty . . . . .	79
10.5.1	Inicjowanie okna dialogowego . . . . .	79
10.5.2	Format linii poleceń . . . . .	80
10.5.3	Konwerter i arkusz stylów (wtyczka) . . . . .	80
10.5.4	Format pośredniej listy sieci . . . . .	80

---

<b>11 Drukowanie i rysowanie schematów na drukarkach lub ploterach</b>	<b>81</b>
11.1 Wprowadzenie	81
11.2 Polecenia wspólne	81
11.3 Rysuj w formacie PostScript	81
11.4 Rysuj w formacie PDF	82
11.5 Rysuj w formacie SVG	83
11.6 Rysuj w formacie DXF	83
11.7 Rysowanie w formacie HPGL	83
11.7.1 Wybór rozmiaru arkusza	84
11.7.2 Ustawienie przesunięcia strony	84
11.8 Drukuj	85
<b>12 Symbol Library Editor</b>	<b>86</b>
12.1 General Information About Symbol Libraries	86
12.2 Symbol Library Overview	86
12.3 Symbol Library Editor Overview	87
12.3.1 Główny pasek menu	87
12.3.2 Pasek narzędzi edycji symbolu	89
12.3.3 Pasek opcji	90
12.4 Wybór biblioteki	90
12.4.1 Select and Save a Symbol	91
12.4.1.1 Symbol Selection	91
12.4.1.2 Save a Symbol	91
12.4.1.3 Transfer Symbols to Another Library	92
12.4.1.4 Discarding Symbol Changes	92
12.5 Creating Library Symbols	92
12.5.1 Create a New Symbol	92
12.5.2 Create a Symbol from Another Symbol	94
12.5.3 Symbol Properties	94
12.5.4 Symbols with Alternate Symbolic Representation	95
12.6 Elementy graficzne symbolu	96
12.6.1 Przynależność elementów graficznych	96
12.6.2 Tekst jako grafika w symbolu	98

---

12.7 Multiple Units per Symbol and Alternate Body Styles . . . . .	98
12.7.1 Example of a Symbol Having Multiple Units with Different Symbols: . . . . .	99
12.7.1.1 Elementy geometryczne w symbolach . . . . .	100
12.8 Tworzenie i edycja wyprowadzeń (pinów) . . . . .	101
12.8.1 Wyprowadzenia - Informacje podstawowe . . . . .	101
12.8.2 Właściwości wyprowadzeń . . . . .	102
12.8.3 Style graficzne pinów . . . . .	102
12.8.4 Typy elektryczne . . . . .	103
12.8.5 Wyprowadzenia - Zmiany globalne . . . . .	104
12.8.6 Wyprowadzenia - Symbole wieloczęściowe i podwójna reprezentacja . . . . .	104
12.9 Symbol Fields . . . . .	105
12.9.1 Editing Symbol Fields . . . . .	105
12.10 Tworzenie symboli zasilania . . . . .	106
<b>13 LibEdit - Symbols</b>	<b>108</b>
13.1 Przegląd zagadnień . . . . .	108
13.2 Position a symbol anchor . . . . .	109
13.3 Symbol aliases . . . . .	109
13.4 Symbol fields . . . . .	110
13.5 Symbol documentation . . . . .	111
13.5.1 Symbol keywords . . . . .	111
13.5.2 Symbol documentation (Doc) . . . . .	112
13.5.3 Dołączony plik dokumentacji (_Nazwa pliku z dokumentacją_) . . . . .	112
13.5.4 Filtrowanie footprintów dla CvPcb . . . . .	112
13.6 Biblioteka wzorców . . . . .	114
13.6.1 Eksport/Tworzenie wzorca . . . . .	114
13.6.2 Importowanie wzorca . . . . .	114
<b>14 Symbol Library Browser</b>	<b>115</b>
14.1 Wprowadzenie . . . . .	115
14.2 Ekran główny . . . . .	116
14.3 Symbol Library Browser Top Toolbar . . . . .	117

---



<b>15 Tworzenie własnych list sieci i plików BOM</b>	<b>119</b>
15.1 Plik pośredniej listy sieci . . . . .	119
15.1.1 Przykładowy schemat . . . . .	119
15.1.2 Przykład pośredniej listy sieci . . . . .	120
15.2 Konwersja na nowy format listy sieci . . . . .	123
15.3 Konwerter XSLT . . . . .	124
15.3.1 Przykład tworzenia pliku z listą sieci PADS-PCB . . . . .	124
15.3.2 Przykład tworzenia listy sieci programu Cadstar . . . . .	126
15.3.3 Przykład tworzenia listy sieci programu OrcadPCB2 . . . . .	130
15.3.4 Używanie systemu wtyczek Eeschema . . . . .	135
15.3.4.1 Inicjowanie okna dialogowego . . . . .	135
15.3.4.2 Ustawienia nowych wtyczek . . . . .	136
15.3.4.3 Generowanie list sieci za pomocą linii poleceń . . . . .	136
15.3.4.4 Format linii poleceń: przykład z xsltproc . . . . .	137
15.3.5 Generowanie list materiałowych (BOM) . . . . .	137
15.4 Format polecenia: Przykład skryptu Python . . . . .	138
15.5 Plik pośredni listy sieci . . . . .	138
15.5.1 Struktura ogólna . . . . .	140
15.5.2 Sekcja nagłówka . . . . .	140
15.5.3 Sekcja komponentów . . . . .	140
15.5.3.1 Uwagi na temat odcisków czasowych dla komponentów . . . . .	141
15.5.4 Sekcja elementów bibliotecznych . . . . .	141
15.5.5 Sekcja bibliotek . . . . .	142
15.5.6 Sekcja sieci . . . . .	142
15.6 Więcej informacji na temat xsltproc . . . . .	143
15.6.1 Wprowadzenie . . . . .	143
15.6.2 Synopsis . . . . .	144
15.6.3 Opcje linii poleceń . . . . .	144
15.6.4 Zwracane wartości . . . . .	146
15.6.5 Więcej informacji na temat xsltproc . . . . .	146

---

<b>16 Simulator</b>	<b>147</b>
16.1 Assigning models . . . . .	147
16.1.1 Passive . . . . .	148
16.1.2 Model . . . . .	150
16.1.3 Source . . . . .	152
16.2 Spice directives . . . . .	154
16.3 Simulation . . . . .	154
16.3.1 Menu . . . . .	155
16.3.1.1 File . . . . .	155
16.3.1.2 Simulation . . . . .	155
16.3.1.3 View . . . . .	155
16.3.2 Toolbar . . . . .	155
16.3.3 Plot panel . . . . .	156
16.3.4 Output console . . . . .	156
16.3.5 Signals list . . . . .	156
16.3.6 Cursors list . . . . .	156
16.3.7 Tune panel . . . . .	157
16.3.8 Tuner tool . . . . .	157
16.3.9 Probe tool . . . . .	157
16.3.10 Simulation settings . . . . .	158

---

*Podręcznik użytkownika*

## **Prawa autorskie**

This document is Copyright © 2010-2018 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.

Wszystkie znaki towarowe użyte w tym dokumencie należą do ich właścicieli.

## **Współtwórcy**

Jean-Pierre Charras, Fabrizio Tappero.

## **Tłumaczenie**

Kerusey Karyu <[keruseykaryu@o2.pl](mailto:keruseykaryu@o2.pl)>, 2014-2015.

## **Kontakt**

Wszelkie zauważone błędy, sugestie lub nowe wersje dotyczące tego dokumentu prosimy kierować do:

- W sprawie dokumentacji: <https://github.com/KiCad/kicad-doc/issues>
- W sprawie oprogramowania: <https://bugs.launchpad.net/kicad>
- About KiCad translation: <https://github.com/KiCad/kicad-i18n/issues>

## **Data publikacji i wersja oprogramowania**

Published on May 30, 2015.

# Rozdział 1

## Wprowadzenie do Eeschema

### 1.1 Opis

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

- Linux
- Apple macOS
- Windows

Niezależnie jaki system został użyty, generowane przez program pliki są w pełni kompatybilne pomiędzy systemami.

Eeschema to zintegrowane oprogramowanie gdzie wszystkie funkcje jakie oferuje: rysowanie, kontrola poprawności, zarządzanie bibliotekami i dostęp do oprogramowania do tworzenia PCB są dostępne z poziomu Eeschema.

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

Eeschema 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:

- Sprawdzanie poprawności projektu (DRC) dla zautomatyzowanego procesu kontroli poprawności połączeń, czy testowania niepołączonych wejść elementów.
- Generowanie wydruków w różnych formatach (Postscript, PDF, HPGL oraz SVG).
- Bill of Materials generation (via Python or XSLT scripts, which allow many flexible formats).

### 1.2 Opis techniczny

Eeschema 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 diagrams, the representation is hierarchical.

Eeschema can use multi-sheet diagrams in a few ways:

---

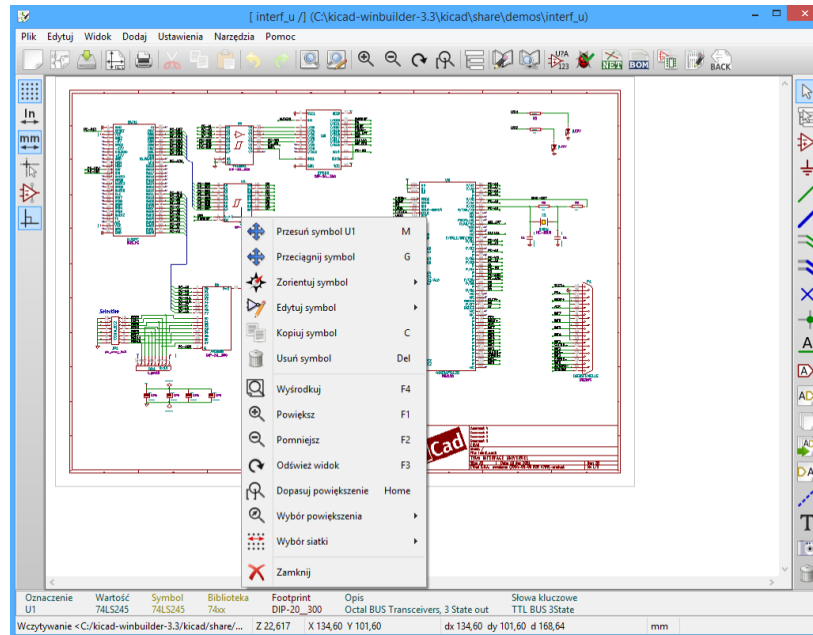
- Hierarchii prostych (każdy schemat jest użyty tylko jeden raz),
  - Hierarchii złożonych (niektóre ze schematów są użyte wielokrotnie),
  - Hierarchii płaskich (niektóre schematy nie są wyraźnie połączone na głównym schemacie).
-

## Rozdział 2

# Podstawowe polecenia

Commands can be executed by:

- Kliknięcia w pasek menu (na górze ekranu).
  - Kliknięcia w pasek ikon na górze ekranu (podstawowe polecenia).
  - Kliknięcia w pasek ikon po prawej stronie ekranu (podstawowe polecenia lub narzędzia”).
  - Kliknięcia w pasek ikon z lewej strony ekranu (opcje wyświetlania).
  - Kliknięcia za pomocą klawisza myszy (najważniejsze polecenia). W praktyce kliknięcie prawym klawiszem otwiera menu kontekstowe, zależne od tego jaki element znajduje się na pozycji kursora (powiększenie, siatka oraz edycja elementów).
  - Function keys (F1, F2, F3, F4, Insert and Space keys). Specifically: Escape key cancels the command in progress. Insert key allows the duplication of the last element created.
  - Pressing hot keys which typically perform a select tool command and begin tool action at the current cursor location. For a list of hot keys, see the "Help→List Hotkeys" menu entry or press ? key.
-



## 2.1 Polecenia związane z myszą

### 2.1.1 Podstawowe polecenia

#### Lewy klawisz

- Single click: displays the characteristics of the symbol or text under the cursor in the status bar.
- Double click: edit (if the element is editable) the symbol or text.

#### Prawy klawisz

- Otwarcie menu kontekstowego.

### 2.1.2 Block operations

Można przesuwać, przeciągać, kopiować oraz usuwać wybrane obszary w każdym z menu Eeschema.

Areas are selected by drawing a box around items using the left mouse button.

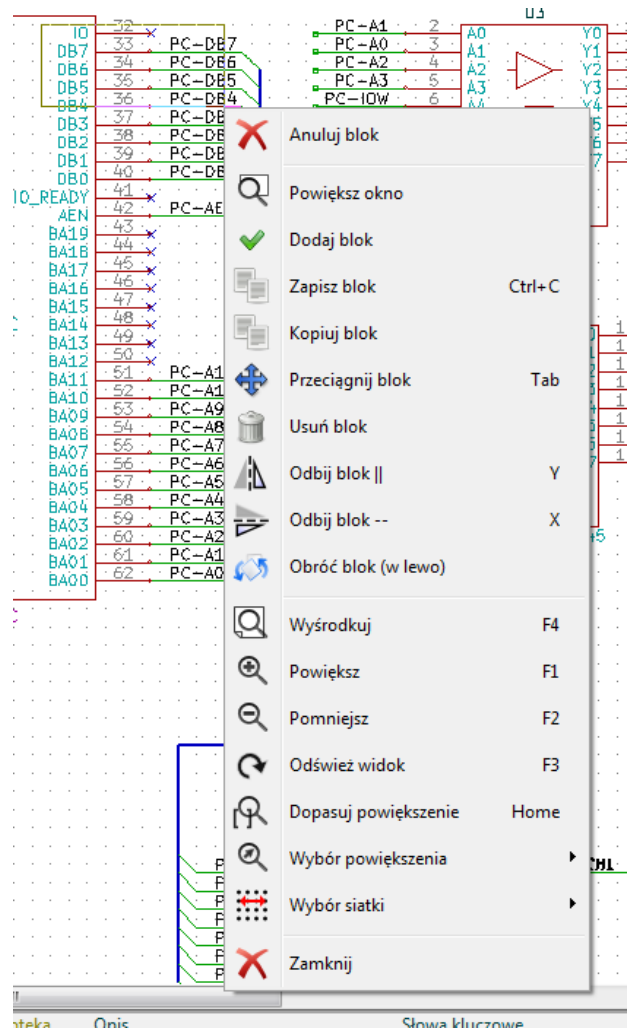
Holding "Shift", "Ctrl", or "Shift + Ctrl" during selection respectively performs copying, dragging and deletion:

Lewy przycisk myszy	Przesunięcie zaznaczenia.
Shift + Lewy przycisk myszy	Skopiowanie zaznaczenia.
Ctrl + Lewy przycisk myszy	Przeciąganie zaznaczenia.
Ctrl + Shift + Lewy przycisk myszy	Usunięcie zaznaczenia.

Podczas przeciągania lub kopiowania można:

- Kliknąć ponownie by umieścić elementy.
- Click the right button or press Escape key to cancel.

If a block move command has started, another command can be selected using the right-click pop-up menu.



## 2.2 Hotkeys

- Klawisz "?" pokazuje bieżącą listę klawiszy skrótów.
- Hotkeys might be redefined in Controls tab of Schematic Editor Options dialog (menu Preferences → General Options).

Here is the default hotkey list:

Help (this window)	?
Zoom In	F1
Zoom Out	F2
Zoom Redraw	F3



Zoom Center	F4
Fit on Screen	Home
Zoom to Selection	@
Reset Local Coordinates	Space
Edit Item	E
Delete Item	Del
Rotate Item	R
Drag Item	G
Undo	Ctrl+Z
Redo	Ctrl+Y
Mouse Left Click	Return
Mouse Left Double Click	End
Save Schematic	Ctrl+S
Load Schematic	Ctrl+O
Find Item	Ctrl+F
Find Next Item	F5
Find Next DRC Marker	Shift+F5
Find and Replace	Ctrl+Alt+F
Repeat Last Item	Ins
Move Block → Drag Block	Tab
Copy Block	Ctrl+C
Paste Block	Ctrl+V
Cut Block	Ctrl+X
Move Schematic Item	M
Duplicate Symbol or Label	C
Add Symbol	A
Add Power	P
Mirror X	X
Mirror Y	Y
Orient Normal Symbol	N
Edit Symbol Value	V
Edit Symbol Reference	U
Edit Symbol Footprint	F
Edit with Symbol Editor	Ctrl+E
Begin Wire	W
Begin Bus	B
End Line Wire Bus	K
Add Label	L
Add Hierarchical Label	H
Add Global Label	Ctrl+L
Add Junction	J
Add No Connect Flag	Q
Add Sheet	S
Add Wire Entry	Z

Add Bus Entry	/
Add Graphic PolyLine	I
Add Graphic Text	T
Update PCB from Schematic	F8
Autoplace Fields	O
Leave Sheet	Alt+BkSp
Delete Node	BkSp
Highlight Connection	Ctrl+X

All hotkeys can be redefined using the hotkey editor (menu Preferences→General Options→[Controls](#)).

It is possible to import/export hotkey settings using menu Preferences→Import and Export→Import/Export Hotkeys.

## 2.3 Grid

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

- Size might be changed using the pop-up menu or using the Preferences/Options menu.
- Color might be changed in Colors tab of the Schematic Editor Options dialog (menu Preferences → General Options).
- Visibility might be switched using the left-hand toolbar button.

Domyślnym rozmiarem siatki jest 50 milsów (0,050 cala) lub inaczej 1,27 milimetrów.

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.

## 2.4 Wybór powiększenia - Zoom

By zmienić powiększenie (Zoom):

- Należy kliknąć prawym klawiszem by otworzyć menu kontekstowe i wybrać potrzebne powiększenie.
- Lub użyć klawiszy funkcyjnych:
  - F1: Zoom in
  - F2: Zoom out
  - F4 or simply click on the middle mouse button (without moving the mouse): Center the view around the cursor pointer position
- Powiększanie widoku:

- Mouse wheel: Zoom in/out
- Shift+Mouse wheel: Pan up/down
- Ctrl+Mouse wheel: Pan left/right

## 2.5 Wyświetlanie pozycji kursora

The display units are in inches or millimeters. However, Eeschema always uses 0.001 inch (mil/thou) as its internal unit.

Informacje wyświetlane na dole okna od prawej strony są następujące:

- Poziom powiększenia
- Pozycja absolutna kursora
- Względna pozycja kursora

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

Z 8	X 5,400 Y 6,650	dx 5,400 dy 6,650 d 8,566	cale
-----	-----------------	---------------------------	------

## 2.6 Pasek menu

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

Plik	Edytuj	Widok	Dodaj	Ustawienia	Narzędzia	Pomoc
------	--------	-------	-------	------------	-----------	-------




## 2.7 Górny pasek narzędzi

Ten pasek umożliwia dostęp do głównych funkcji programu Eeschema.

Gdy Eeschema jest uruchomiony jako samodzielna aplikacja, pasek narzędzi wygląda następująco:



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.

	Select the sheet size and edit the title block.
	Open print 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 to fit.
	Zoom in and out.
	View and navigate the hierarchy tree.
	Leave the current sheet and go up in the hierarchy.
	Call the symbol library editor to view and modify libraries and symbols.
	Browse symbol libraries.
	Annotate symbols.
	Electrical Rules Checker (ERC), automatically validate electrical connections.
	Call CvPcb to assign footprints to symbols.
	Export a netlist (Pcbnew, SPICE and other formats).
	Edit symbol fields.
	Generate the Bill of Materials (BOM).
	Call Pcbnew to perform a PCB layout.























Back-import footprint assignment (selected using CvPcb or Pcbnew) into the "footprint" fields.

## 2.8 Prawy pasek narzędzi

Ten pasek zawiera narzędzia do:







- Place symbols, wires, buses, junctions, labels, text, etc.
- Create hierarchical subsheets and connection symbols.

	Cancel the active command or tool.
	Highlight a net by marking its wires and net labels with a different color. If KiCad runs in project mode then copper corresponding to the selected net will be highlighted in Pcbnew 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.
	Draw bus-to-bus entry points.
	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 <b>in the same sheet</b> . 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.
	Place a hierarchical pin in a subsheet. This command can be executed only on hierarchical subsheets. It will create arbitrary hierarchical pins, even if they do not exist 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 selected element.

## 2.9 Lewy pasek narzędzi

Ten pasek narzędzi zarządza opcjami wyświetlania:

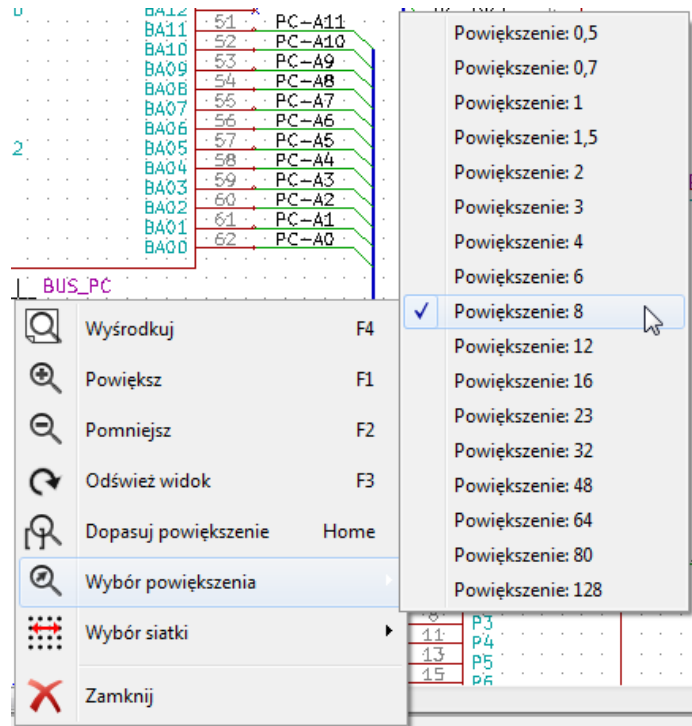
	Toggle grid visibility.
	Switch units to inches.
	Switch units to millimeters.
	Choose the cursor shape (full screen/small).
	Toggle visibility of "invisible" pins.
	Toggle free angle/90 degrees wires and buses placement.

## 2.10 Menu kontekstowe i szybka edycja komponentów

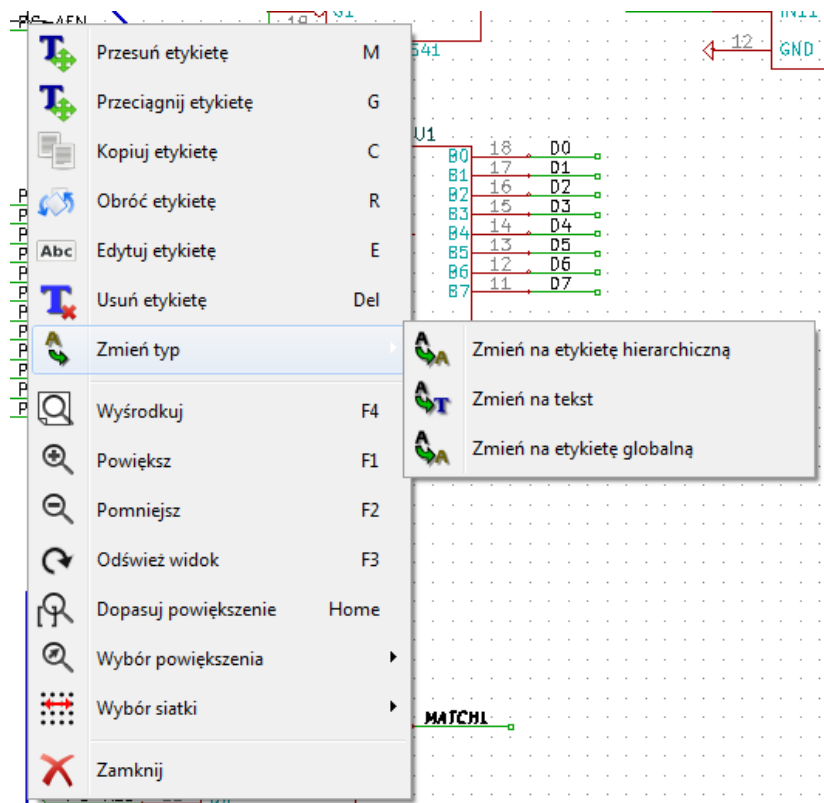
Kliknięcie prawym klawiszem otwiera menu kontekstowe dla wybranego elementu. Zawiera ono:

- Rozmiar powiększenia.
- Ustawienie siatki.
- Główne opcje edycyjne dla wybranego elementu.

Menu podręczne bez wybranego elementu.

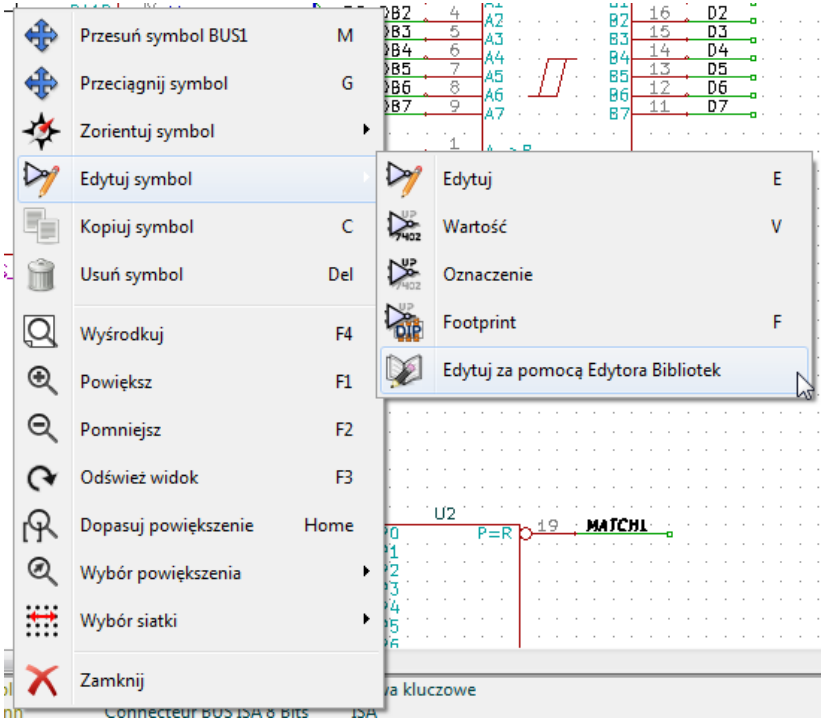


Editing a label.



Editing a symbol.

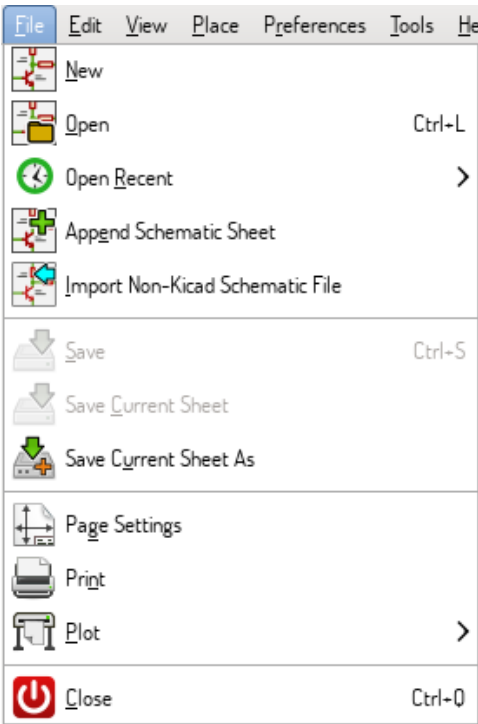




# Rozdział 3

## Menu główne

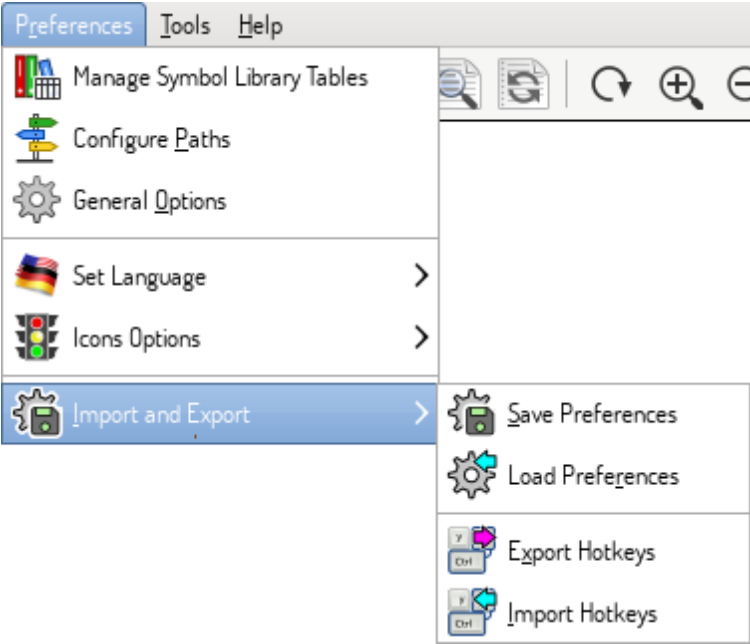
### 3.1 Menu Plik



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).
Append Schematic Sheet	Insert the contents of another sheet into the current one.
Import Non-Kicad Schematic File	Imports a schematic project saved in another file format.
Save	Save current sheet and all its subsheets.
Save Current Sheet	Save only the current sheet, but not others in the project.

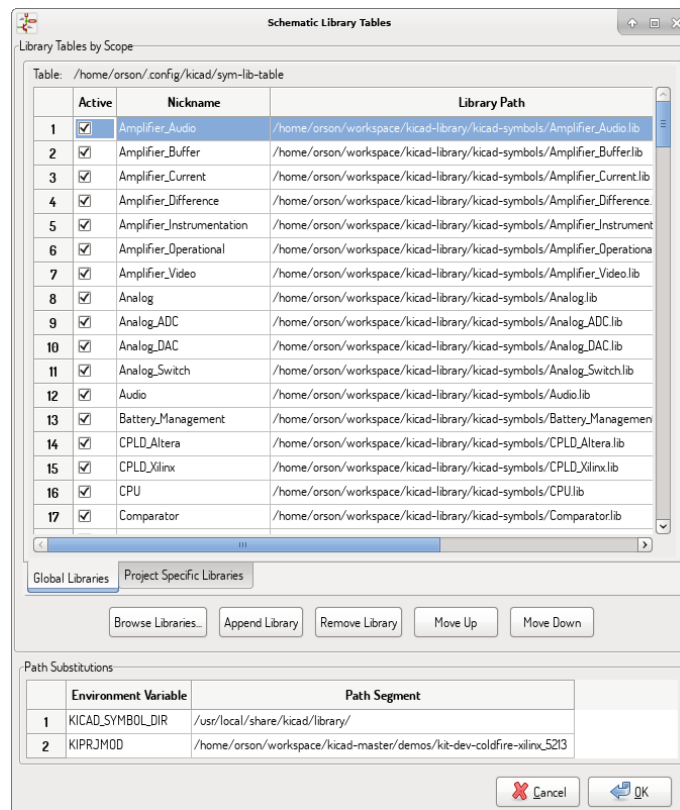
Save Current Sheet As...	Save the current sheet under a new name.
Page Settings	Configure page dimensions and title block.
Print	Print schematic project (See also chapter <a href="#">Plot and Print</a> ).
Plot	Export to PDF, PostScript, HPGL or SVG format (See chapter <a href="#">Plot and Print</a> ).
Close	Terminate the application.

3.2 Menu ustawień



Manage Symbol Library Tables	Add/remove symbol libraries.
Configure Paths	Set the default search paths.
General Options	Preferences (units, grid size, field names, etc.).
Set Language	Select interface language.
Icons Options	Icons visibility settings.
Import and Export	Transfer preferences to/from file.

### 3.2.1 Manage Symbol Library Tables



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

- Global Libraries

Libraries listed in the Global Libraries table are available to every project. They are saved in **sym-lib-table** in your home directory (exact path is dependent on the operating system; check the path above the table).

- Project Specific Libraries

Libraries listed in Project Specific Libraries table are available to the currently opened project. They are saved in **sym-lib-table** file in the project directory (check the path above the table).

You can view either list by clicking on "Global Libraries" or "Project Specific Libraries" tab below the library table.

#### 3.2.1.1 Add a new library

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

#### 3.2.1.2 Remove a library

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

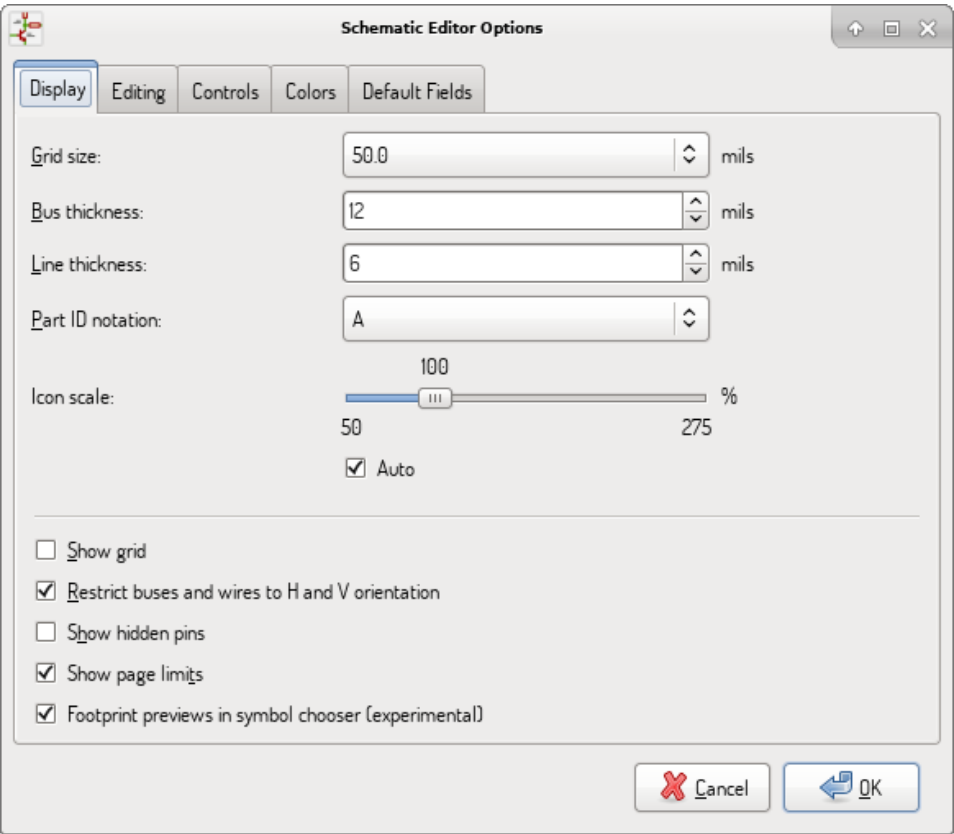
3.2.1.3 Library properties

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 <i>&lt;Library Nickname&gt;:&lt;Symbol Name&gt;</i> strings.
Library Path	Path points to the library location.
Plugin Type	Determines the library file format.
Options	Stores library specific options, if used by plugin.
Description	Briefly characterizes the library contents.

3.2.2 General Options

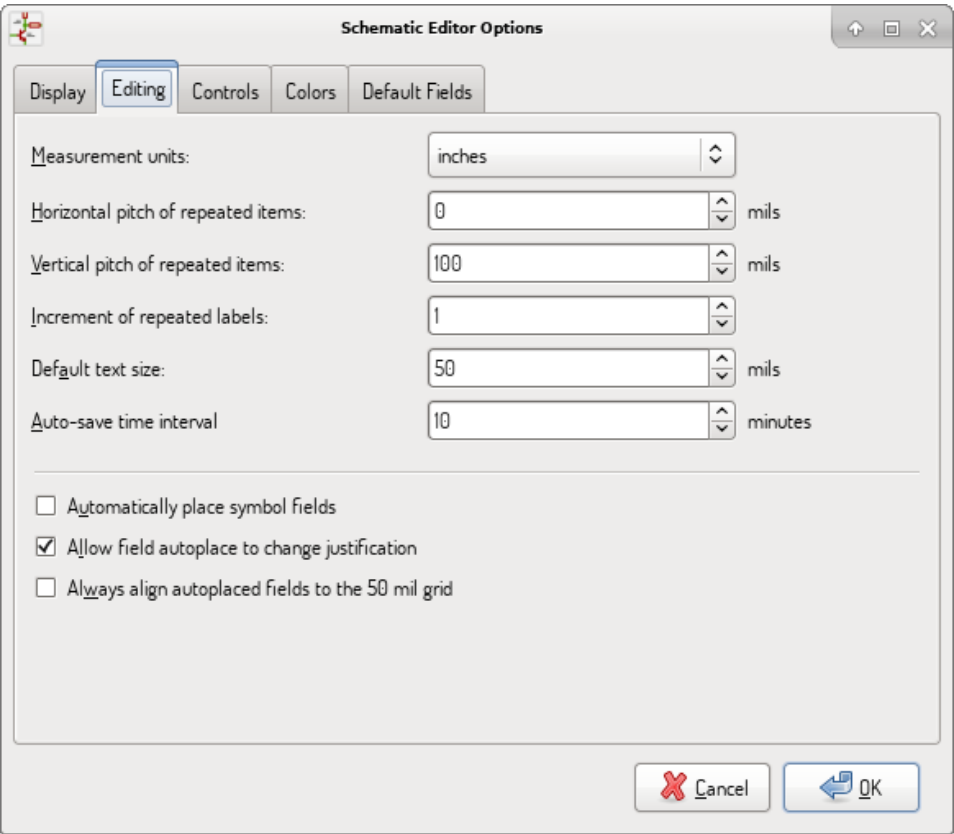
3.2.2.1 Display



Grid Size	Grid size selection. It is <b>recommended</b> to work with normal grid (0.050 inches or 1,27 mm). Smaller grids are used for component building.
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	Displays a footprint preview frame and footprint selector when placing a new symbol. <b>Note:</b> it may cause problems or delays, use at your own risk.

3.2.2.2 Editing

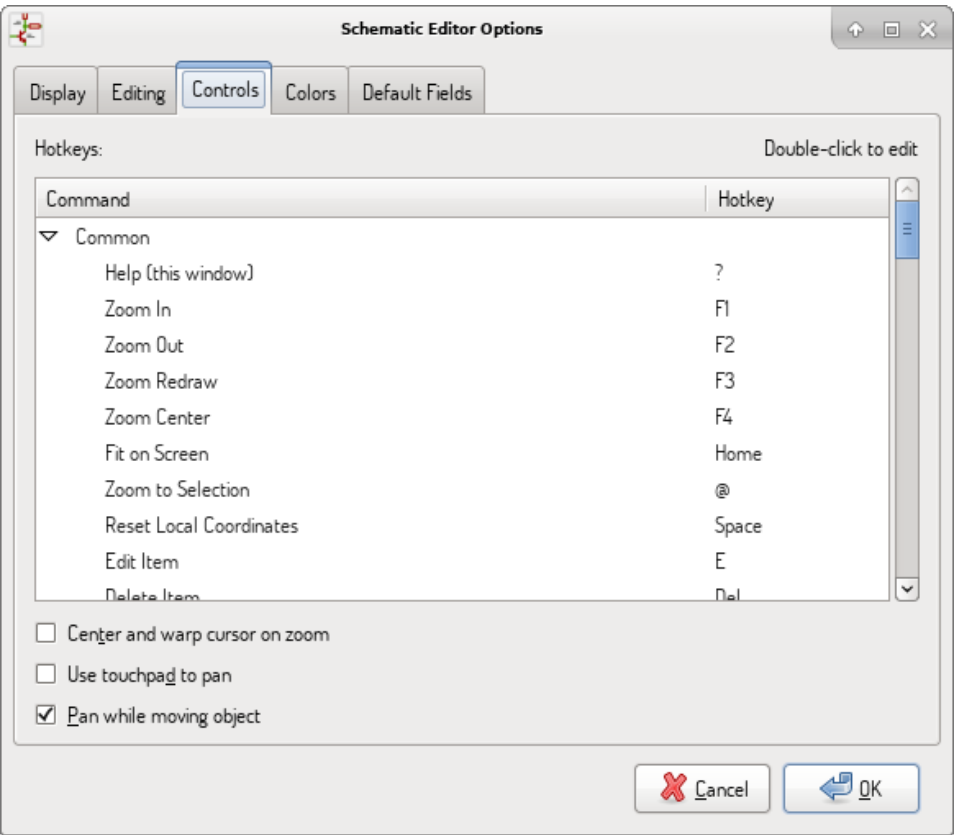


Measurement units	Select the display and the cursor coordinate units (inches or millimeters).
Horizontal pitch of repeated items	Increment on X axis during element duplication (default: 0) (after placing an item like a symbol, label or wire, a duplication is made by the <i>Insert</i> key)

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 <i>Automatically place symbol fields</i> option. Enable text justification adjustment for symbol fields when placing a new part.
Always align autoplaced fields to the 50 mil grid	Extension of <i>Automatically place symbol fields</i> option. If checked, fields are autoplaced using 50 mils grid, otherwise they are placed freely.

3.2.2.3 Controls

Redefine hotkeys and set up the user interface behavior.



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.
Restore Default	Sets the action hotkey to its default value.
Undo All Changes	Reverts all recent hotkey changes for the action.
Restore All to Default	Sets all action hotkeys to their default values.

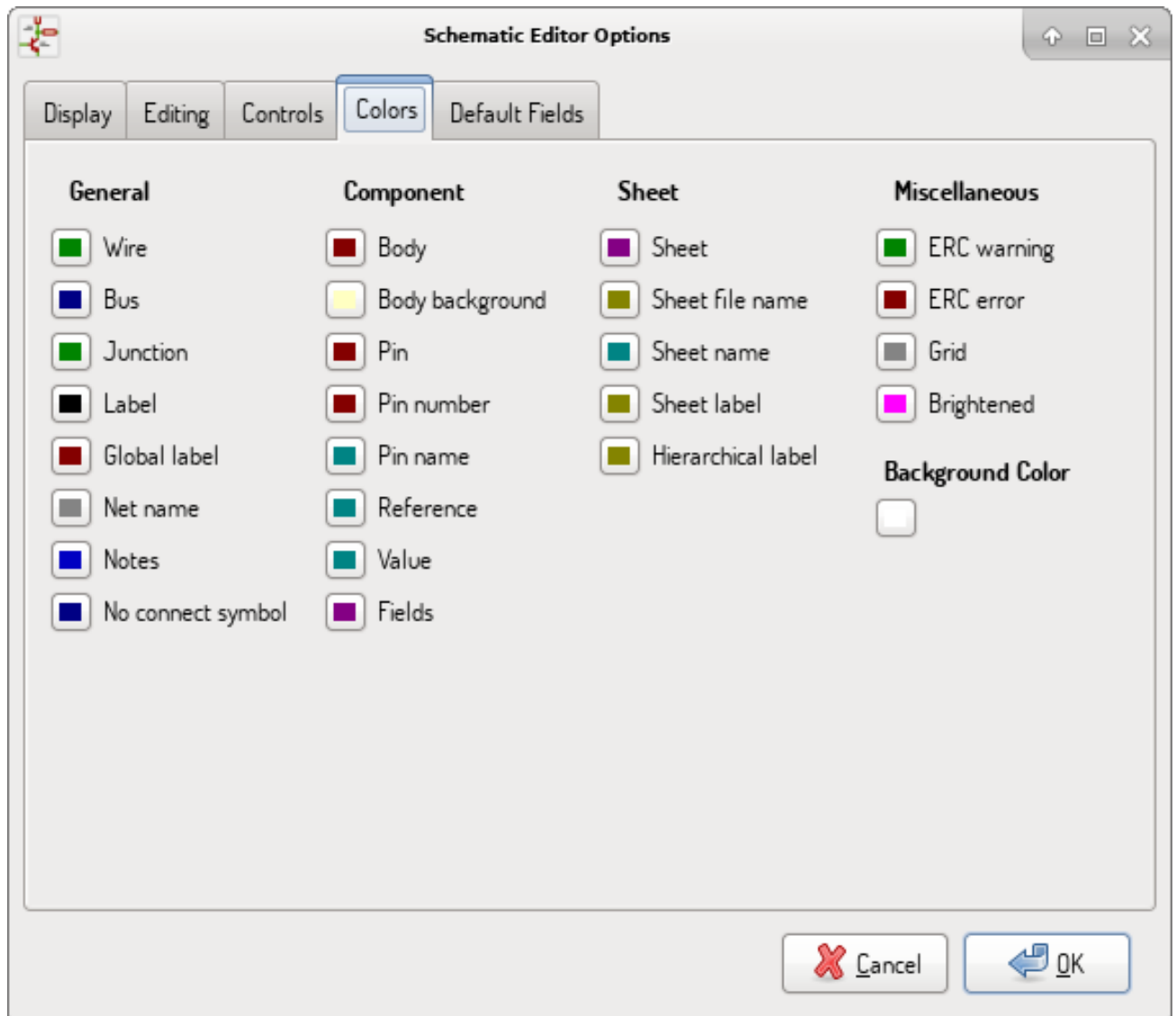
Options description:

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.

#### 3.2.2.4 Colors

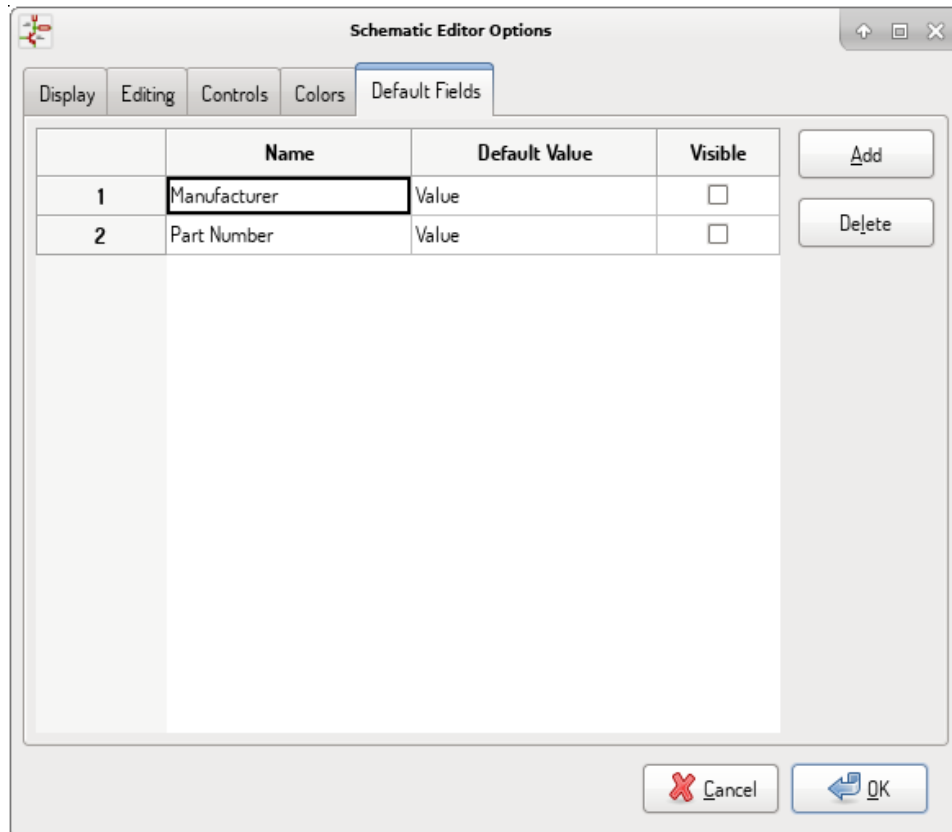
Color scheme for various graphic elements. Click on any of the color swatches to select a new color for a particular element.





### 3.2.2.5 Default Fields

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



### 3.3 Menu Pomoc

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

Use “Copy Version Information” when submitting bug reports to identify your build and system.

## Rozdział 4

# Główny pasek narzędziowy

### 4.1 Zarządzanie ustawieniami arkuszy/stron




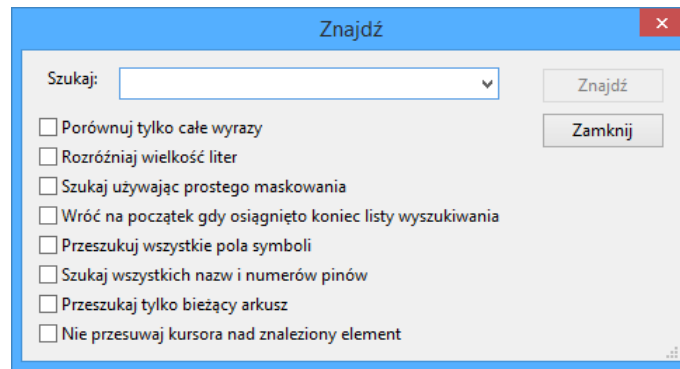
The Sheet Settings icon (  ) allows you to define the sheet size and the contents of the title block.

Numeracja arkuszy jest odświeżana automatycznie. Można ustawić datę na datę bieżącą klikając na przycisk ze strzałką, lecz data ta nie będzie się zmieniała automatycznie.

### 4.2 Znajdź oraz Znajdź i zamień



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.

## 4.3 Narzędzie Listy sieci



The Netlist icon (  ) opens the netlist generation tool.

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

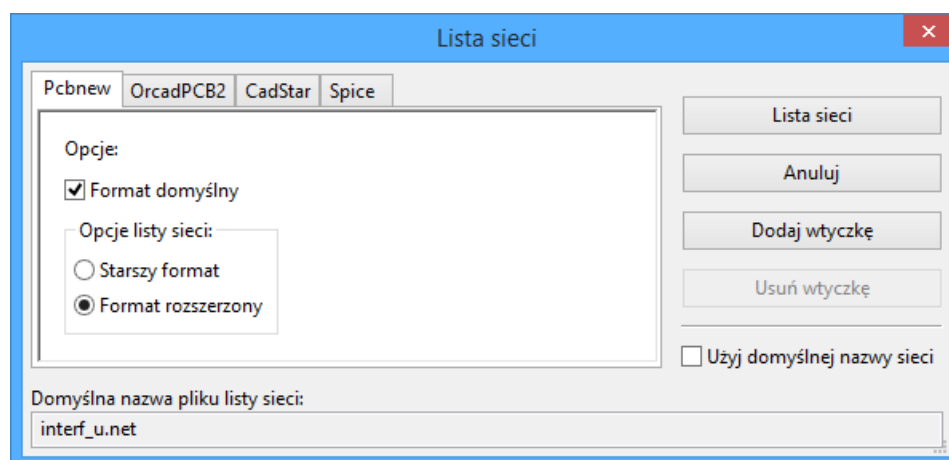
In a multisheet hierarchy, any local label is visible only inside the sheet to which it belongs. For example: the label LABEL1 of sheet 3 is different from the label LABEL1 of sheet 5 (if no connection has been intentionally introduced to connect them). This is due to the fact that the sheet name path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

### Notatka

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

### Notatka

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



Opcje:

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

Można też wygenerować netlistę w innych formatach:

- 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.

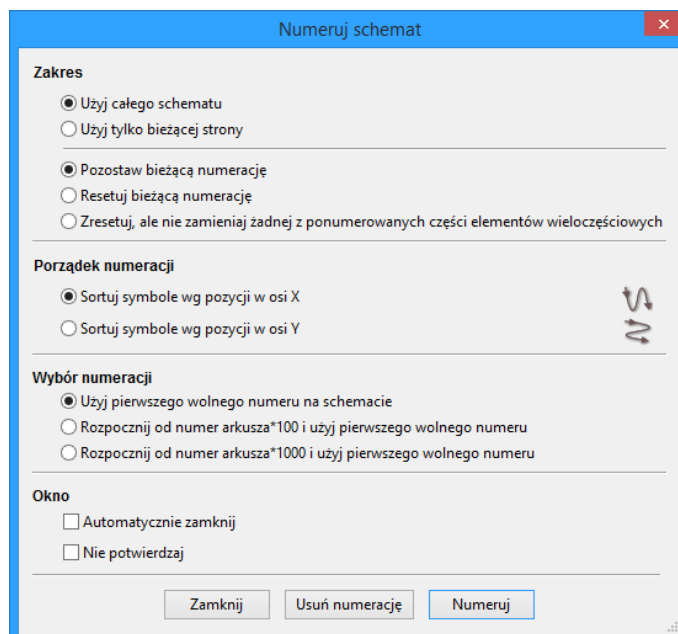
## 4.4 Numeracja komponentów



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

Dla komponentów, które składają się z kilku części (jak na przykład 7400 TTL który posiada 4 takie same 4 bramki), przyrostek oznaczający poszczególne części także będzie zachowany (w przypadku 7400 TTL przypisana nazwa U3 będzie podzielona na U3A, U3B, U3C oraz U3D).

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



### Zakres

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.

### Porządek numeracji

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

### Wybór numeracji

Selects the assigned reference format.

## 4.5 Kontrola reguł projektowych - ERC

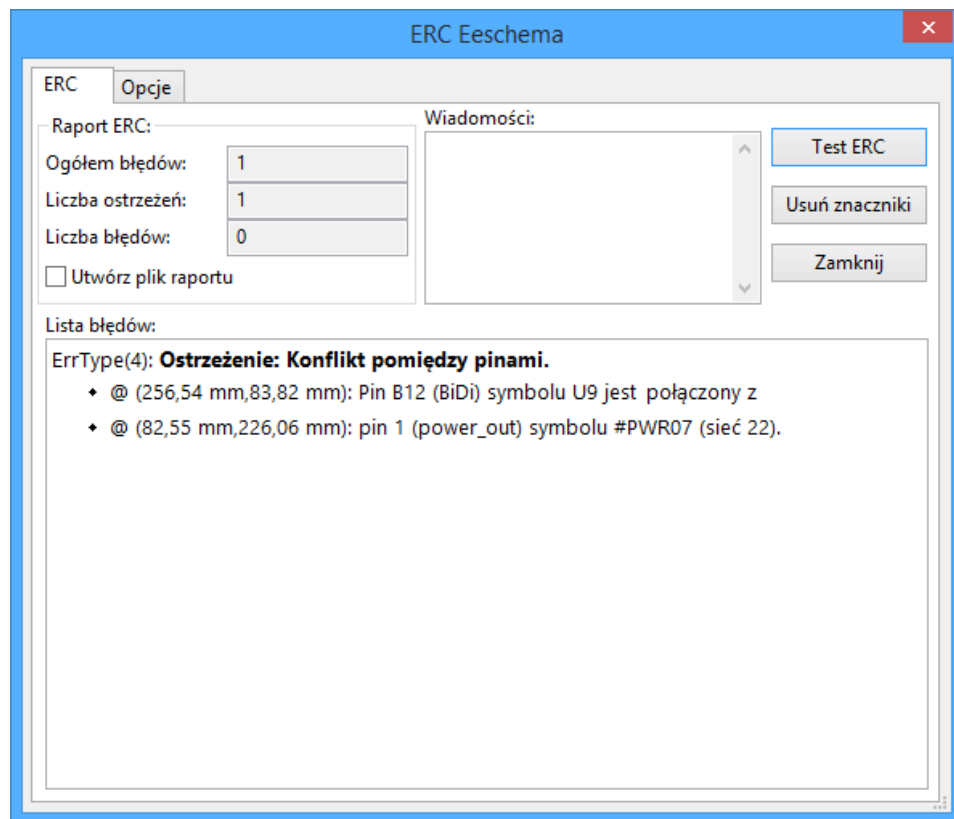


The icon launches the electrical rules check (ERC) tool.

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

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

### 4.5.1 Główne okno narzędzia ERC



Errors are displayed in the Electrical Rules Checker dialog:

- Ogółem błędów - to całkowita liczba znalezionych nieprawidłowości.
- Liczba błędów - to liczba wykrytych błędów.
- Ogółem ostrzeżeń - to liczba wykrytych ostrzeżeń.

Opcje:

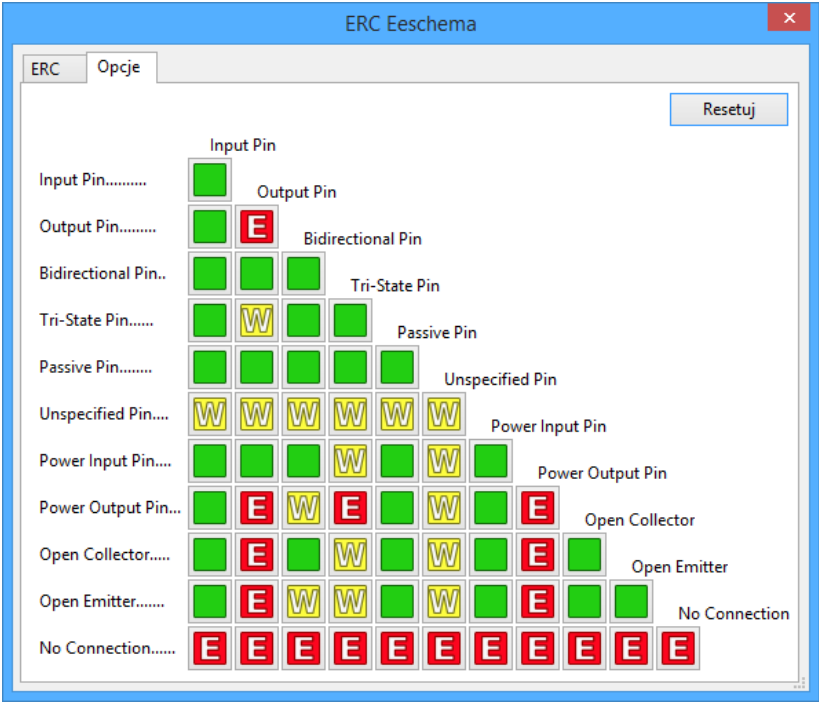
Create ERC file report	Check this option to generate an ERC report file.
------------------------	---

Polecenia:

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

- Gdy kliknie się w komunikat o błędzie na liście, kursor przeskakuje do miejsca gdzie znajduje się znacznik tego błędu.

4.5.2 Opcje sprawdzania ERC



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

- Brak błędu
- Ostrzeżenie
- Błąd

Każde pole na matrycy błędów i ostrzeżeń może być zmienione klikając w nie. Zmiany są wykonywane cykliczne. W każdej chwili można przywrócić ustawienia domyślne za pomocą przycisku Resetuj.

Opcje:

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 labels that occur only once for a particular net. Normally it is required to have at least two make a connection.

Polecenia:

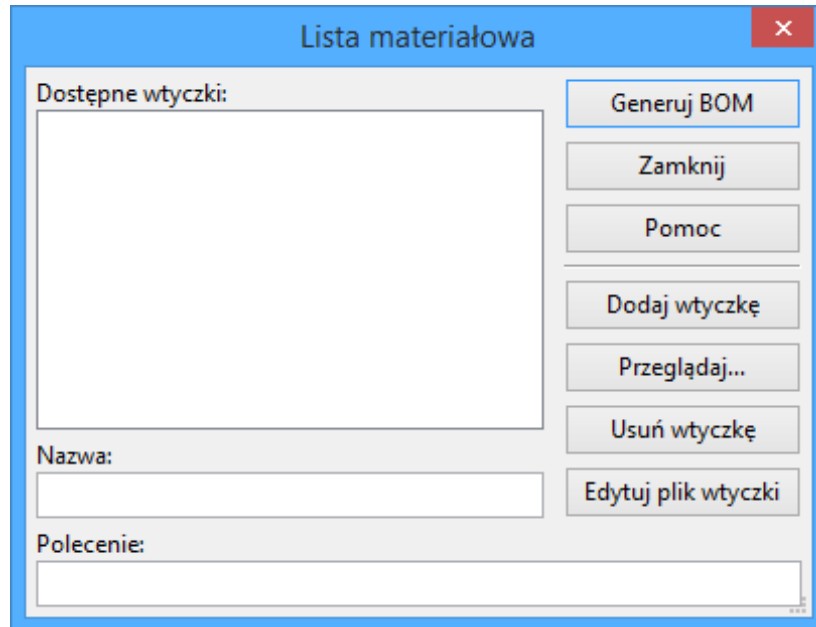
Initialize to Default	Restores the original settings.
-----------------------	---------------------------------



## 4.6 Lista materiałowa - BOM



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

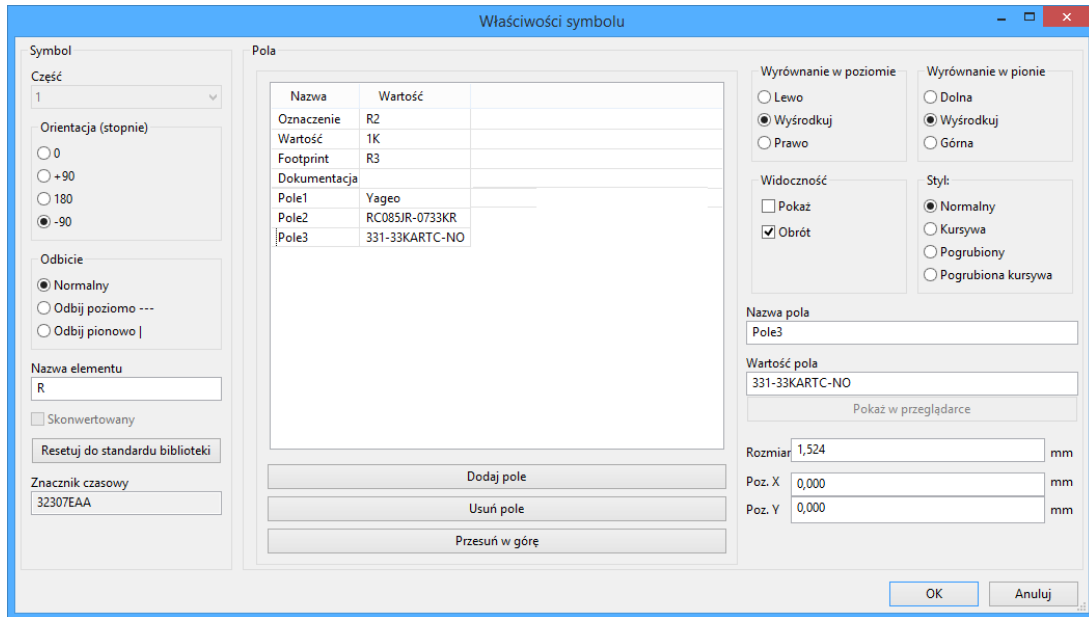


Eeschema's BOM generator makes use of external plugins, either as XSLT or Python scripts. There are a few examples installed inside the KiCad program files directory.

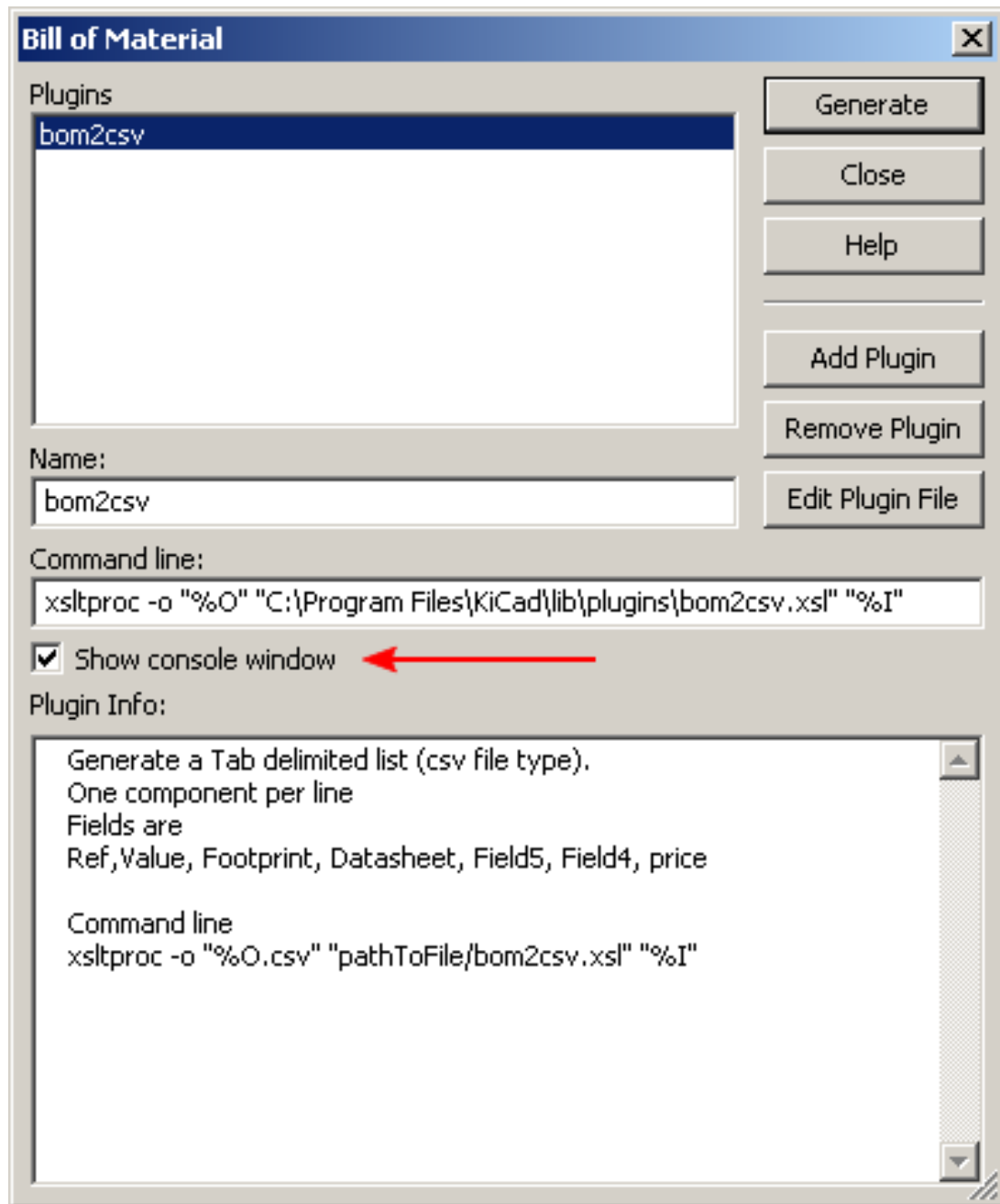
Przydatny zestaw właściwości elementów używany zwykle do tworzenia list materiałowych to:

- Wartość - unikalna nazwa dla każdego z użytych elementów,
- Obudowa - zarówno wpisany ręcznie lub za pomocą numeracji wstecznej (zobacz następny punkt),
- Pole1 - nazwa producenta,
- Pole2 - nazwa elementu według producenta,
- Pole3 - nazwa elementu według jego dystrybutora.

Przykładowo:



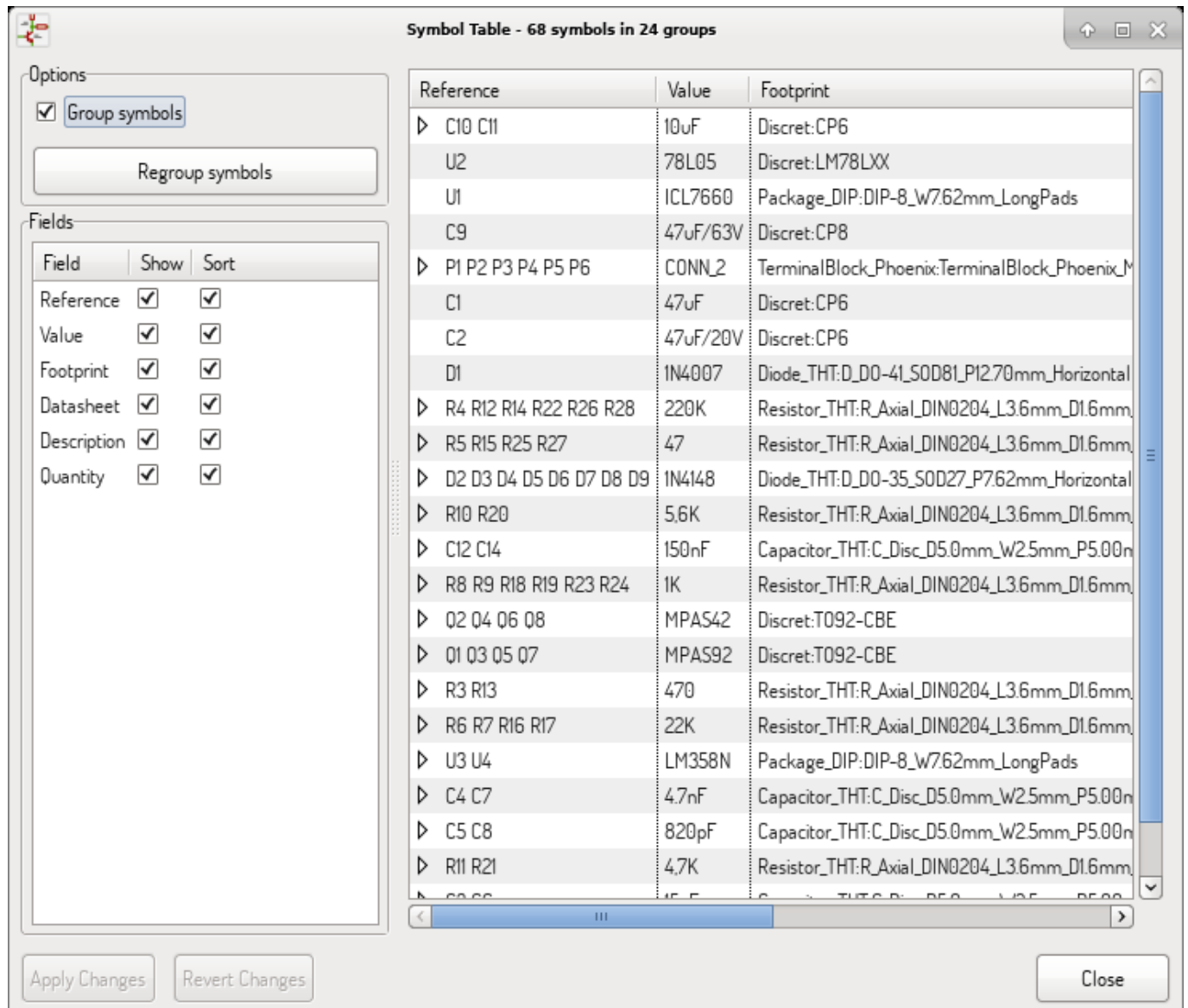
On **MS Windows**, BOM generator dialog has a special option (pointed by red arrow) that controls visibility of external plugin window. + By default, BOM generator command is executed console window hidden and output is redirected to *Plugin info* field. Set this option to show the window of the running command. It may be necessary if plugin has provides a graphical user interface.



## 4.7 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.

## 4.8 Import tool for footprint assignment

### 4.8.1 Dostęp:



The icon **BACK** launches the back-annotate tool.

Narzędzie to pozwala na uaktualnienie danych o footprintach, które zostały zmienione w programie Pcbnew, z powrotem do schematu, tak by nie było rozbieżności pomiędzy oboma dokumentami.

## Rozdział 5

# Manage Symbol Libraries

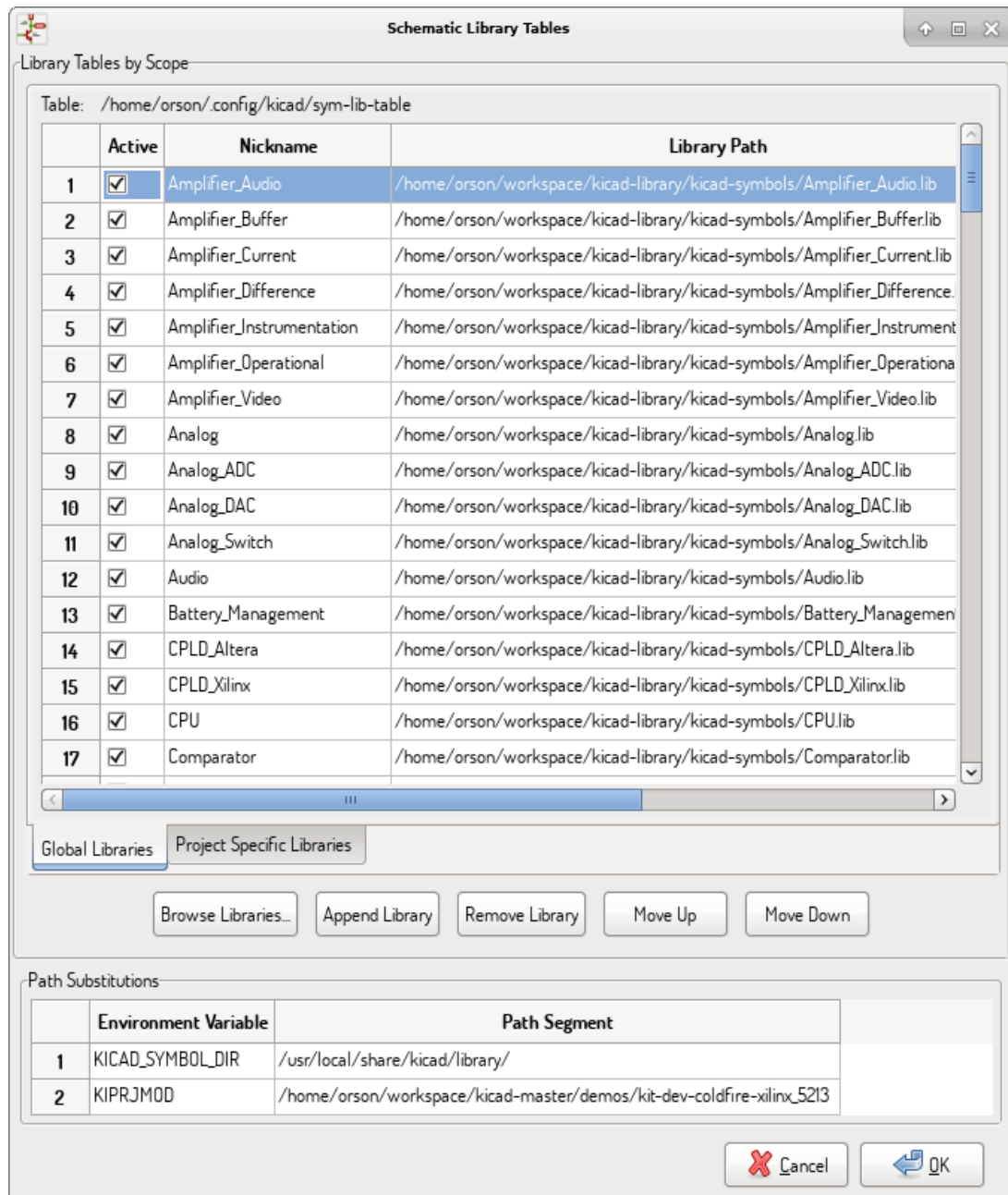
Symbol libraries hold collections of symbols used when creating schematics. Each symbol in a schematic is uniquely identified by a full name that is composed of a library nickname and a symbol name. An example is **Audio:AD1853**.

### 5.1 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, Eeschema 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 Library Tables” entry in the “Preferences” menu.



### 5.1.1 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 home folder. The location of this folder is dependent upon the operating system being used.

### 5.1.2 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.

### 5.1.3 Initial Configuration

The first time Eeschema is run and the global symbol table file **sym-lib-table** is not found in the user's home folder, Eeschema will attempt to copy the default symbol table file `sym-lib-table` stored in the system's KiCad template folder to the file `sym-lib-table` in the user's home folder. If the default template `sym-lib-table` file cannot be found, a dialog will prompt for an alternate location for the `sym-lib-table` file. If no `sym-lib-table` is found or the dialog is dismissed, an empty symbol library table will be created in the user's home folder. If this happens, the user can either copy `sym-lib-table` manually or configure the table by hand.

---

#### Notatka

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.

---

### 5.1.4 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.

**Each library entry must have a unique nickname.**

This 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 plug in type must be selected in order for the library to be properly read. KiCad currently supports only legacy symbol library files plug-in.

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.

### 5.1.5 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**:

---

- the **KIPRJMOD** environment variable that always points to the currently open project directory. **KIPRJMOD** cannot be modified.
- the **KICAD\_SYMBOL\_DIR** environment variable. This points to the path where the default symbol libraries that were installed with KiCad.

You can override **KICAD\_SYMBOL\_DIR** by defining it yourself in preferences/ Configure Path which allows you to substitute your own libraries in place of the default KiCad symbol libraries.

**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.

### 5.1.6 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 home 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.

### 5.1.7 Legacy Project Remapping

When loading a schematic created prior to the symbol library table implementation, Eeschema 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.



#### Ostrzeżenie

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.

---



**Ostrzeżenie**

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.

---

**Notatka**

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 Eeschema prior to the symbol library table implementation.

---

## Rozdział 6

# Tworzenie i edycja schematu

### 6.1 Wprowadzenie

Schemat może zostać przedstawiony na jednym arkuszu, ale głównie będzie to jednak wymagać kilku arkuszy.

Schemat prezentowany na kilku arkuszach jest zwany schematem hierarchicznym, a wszystkie te arkusze (każdy reprezentowany przez własny plik) stanowi dla Eeschema projekt. Operacje na arkuszach hierarchicznych będą opisane w rozdziale [Schematy hierarchiczne](#).

### 6.2 Uwagi ogólne

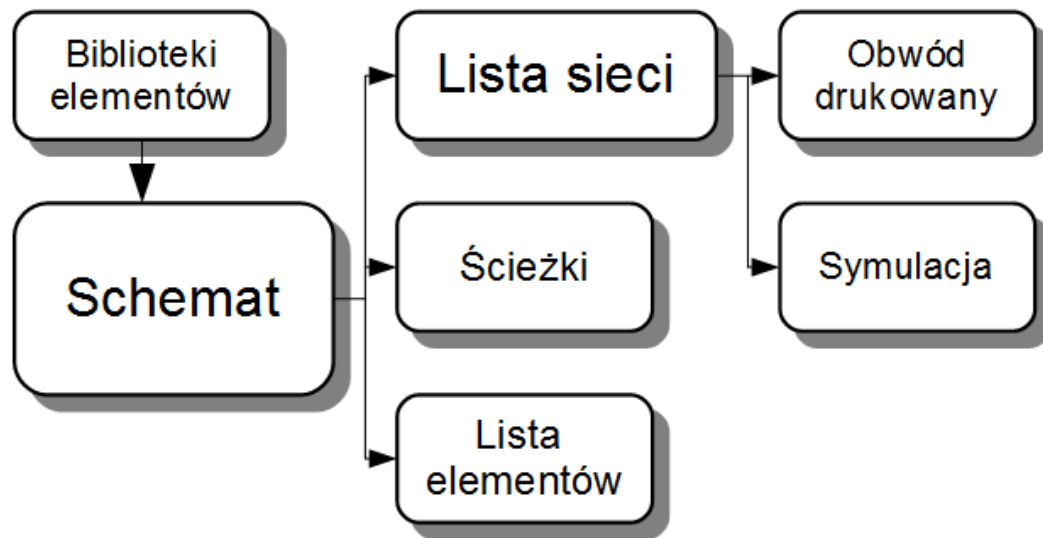
Schemat zaprojektowany za pomocą Eeschema jest czymś więcej niż tylko prostą reprezentacją graficzną urządzenia elektronicznego. Zwykle jest punktem wyjścia dla łańcucha rozwoju, który umożliwia:

- Walidację za pomocą zbioru reguł ([ERC](#)) by wykryć proste błędy oraz braki na schemacie.
- Automatyczne wygenerowanie listy materiałowej ([BOM](#)).
- [Generowanie listy sieci](#) dla oprogramowania do symulacji, takiego jak PSpice.
- [Generowanie listy sieci](#) dla oprogramowania do projektowania obwodów drukowanych PCB.

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.

---

## 6.3 Proces tworzenia

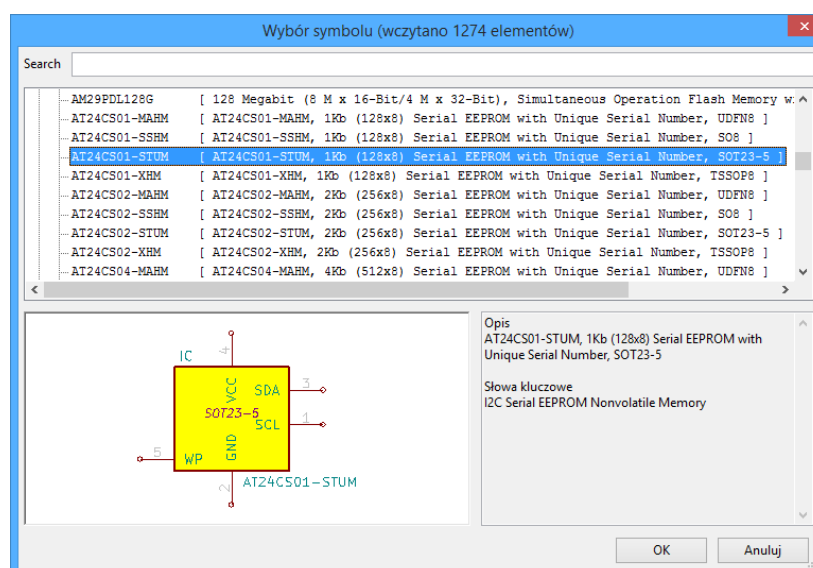


Symbols are added to the schematic from symbol libraries. After the schematic is made, a netlist is generated, which is later used to import the set of connections and footprints into PcbNew.

## 6.4 Symbol placement and editing

### 6.4.1 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:

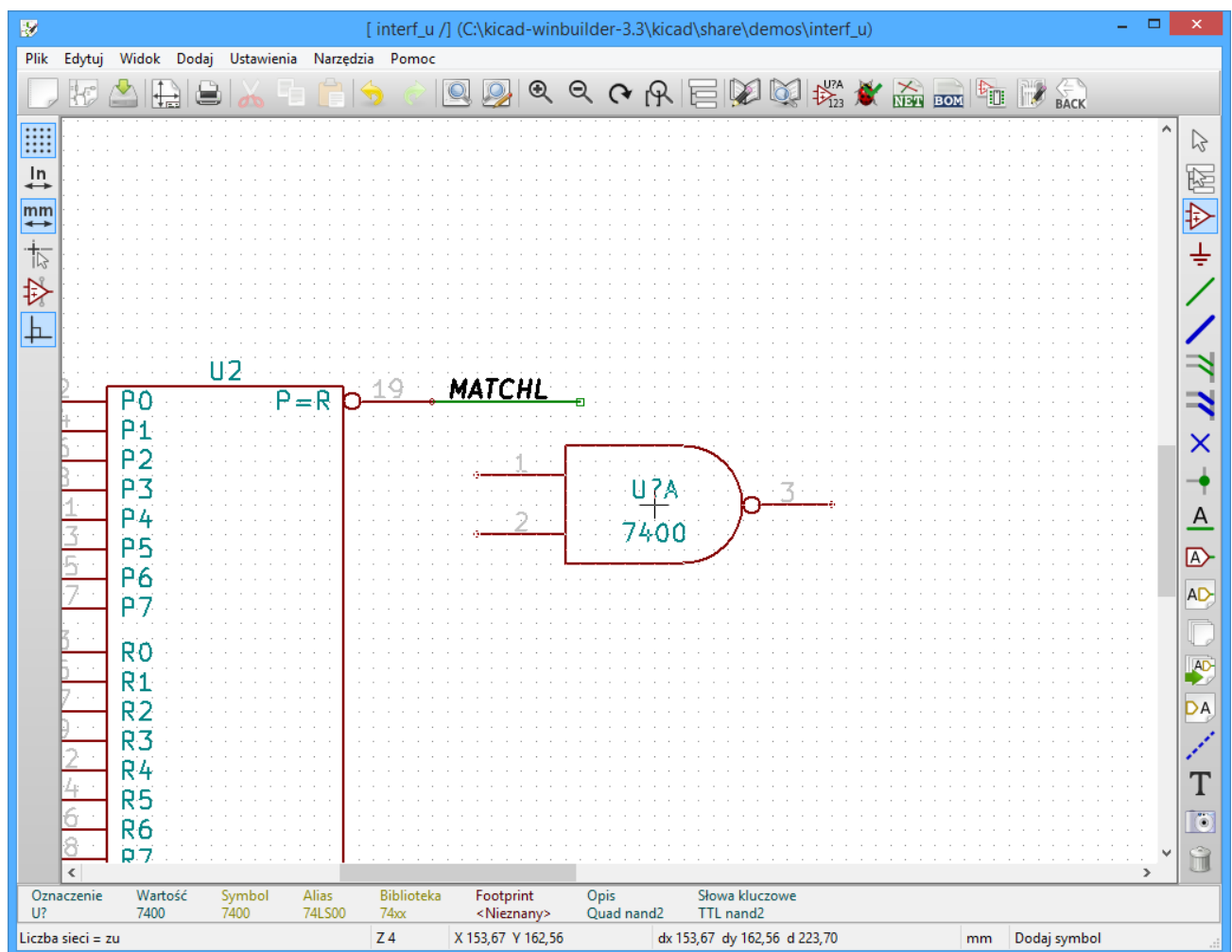
p	n	u	m	k	meg	g	t
$10^{-12}$	$10^{-9}$	$10^{-6}$	$10^{-3}$	$10^3$	$10^6$	$10^9$	$10^{12}$

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.

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. This can be done the same way after placement.

Here is a symbol during placement:



## 6.4.2 Porty zasilania

A power port symbol is a symbol (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.

## 6.4.3 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.

### 6.4.3.1 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.
- Kliknąć prawym przyciskiem myszy, aby otworzyć menu podręczne, i użyć jednego z wyświetlonych poleceń: Przesuń, Zorientuj, Edytuj, Usuń, itp.

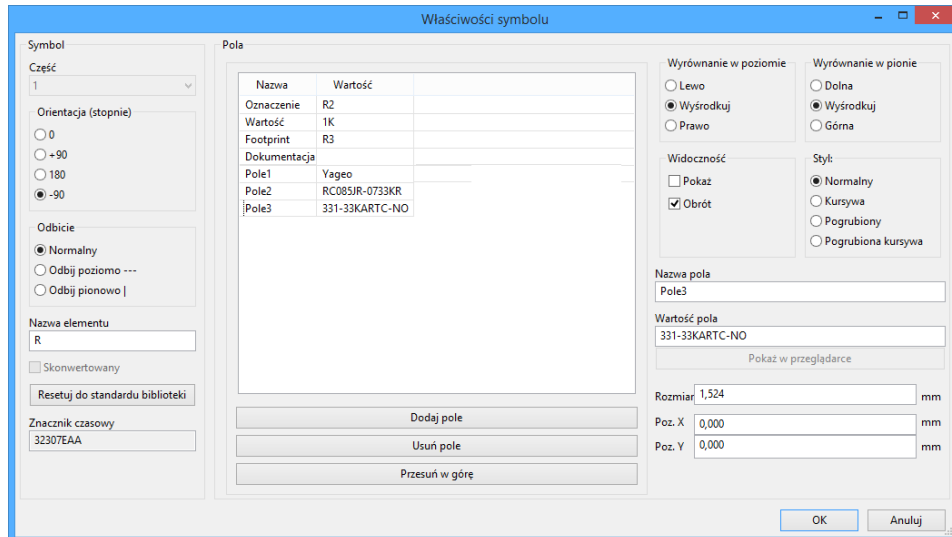
### 6.4.3.2 Modyfikacja pól tekstowych elementów

Można dokonać modyfikacji oznaczeń, wartości, pozycji, orientacji, rozmiaru i widoczności pól:

- Kliknąć dwukrotnie na pole tekstowe aby go zmienić.
- Kliknąć prawym przyciskiem myszy, aby otworzyć menu podręczne, i użyć jednego z wyświetlonych poleceń: Przesuń, Zorientuj, Edytuj, Usuń, itp.

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

---



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 option “Reset to Library Defaults” sets the symbol to the original orientation, and resets the options, size and position of each field. However, texts fields are not modified because this could break the schematic.

## 6.5 Połączenia, Magistrale, Etykiety i Symbole zasilania

### 6.5.1 Wprowadzenie

Wszystkie te elementy rysunkowe mogą zostać umieszczone na schemacie za pomocą narzędzi z prawego pionowego paska narzędziowego.

Te elementy to:

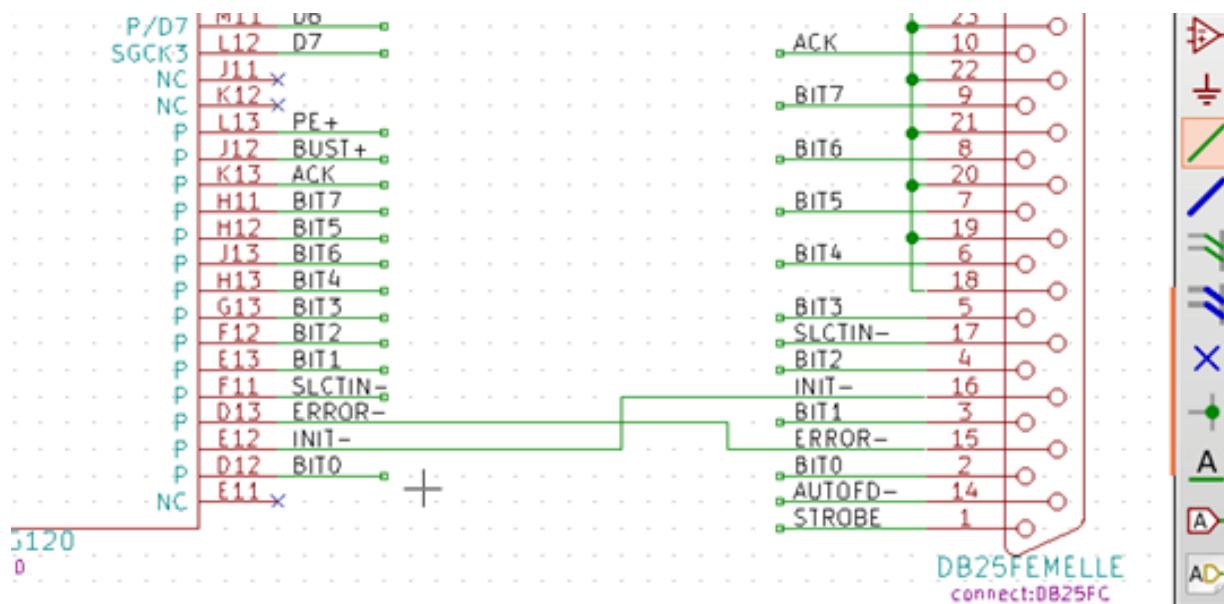
- **Wires:** most connections between symbols.
- **Magistrale:** do łączenia sygnałów w magistrale.
- **Linie łamane:** dla graficznych prezentacji.
- **Węzły:** by wymusić połączenia pomiędzy przecinającymi się połączeniami lub magistralami.
- **Wejścia magistral:** by pokazać, że dane połączenie wchodzi w skład magistrali. Jest to tylko element graficzny!
- **Etykiety:** do nadawania nazw połączeniom w aktywnym arkuszu.
- **Etykiety globalne:** do połączeń pomiędzy arkuszami.
- **Teksty** do umieszczania komentarzy lub adnotacji.
- Symbol **Ńie połączone** by wskazać pin, który nie wymaga żadnego połączenia.
- **Arkusze hierarchiczne**, oraz ich wyprowadzenia.

### 6.5.2 Połączenia (łącza i etykiety)

Są dwie możliwości tworzenia połączeń:

- Połączenia bezpośrednie pomiędzy wyprowadzeniami.
- Połączenia z pomocą etykiet.

Poniższy obrazek pokazuje obie te metody:



Uwaga 1:

Punktem "kontaktu"(lub podpięcia) etykiet to dolny lewy narożnik pierwszej litery z lewej strony. Punkt ten jest wyróżniony małym prostokątem gdy nie ma połączenia.

Aby etykieta była brana pod uwagę, punkt ten musi być w kontakcie z połączeniem lub nakładać się na styku połączenia z wyprowadzeniem.

**Uwaga 2:**

By nawiązać połączenie, jeden z segmentów połączenia musi być dołączony swoim końcem do innego zakończenia segmentu lub do punktu aktywnego u wyprowadzenia elementu.

Jeśli połączenie się nakłada na wyprowadzenie (gdy połączenie przechodzi przez wyprowadzenie ale nie trafia w jego punkt aktywny), wtedy takie połączenie nie jest prawidłowym połączeniem.

**Uwaga 3:**

Połączenia przecinające się nie są domyślnie łączone. W takich przypadkach konieczne jest manualne połączenie ich za pomocą węzła.

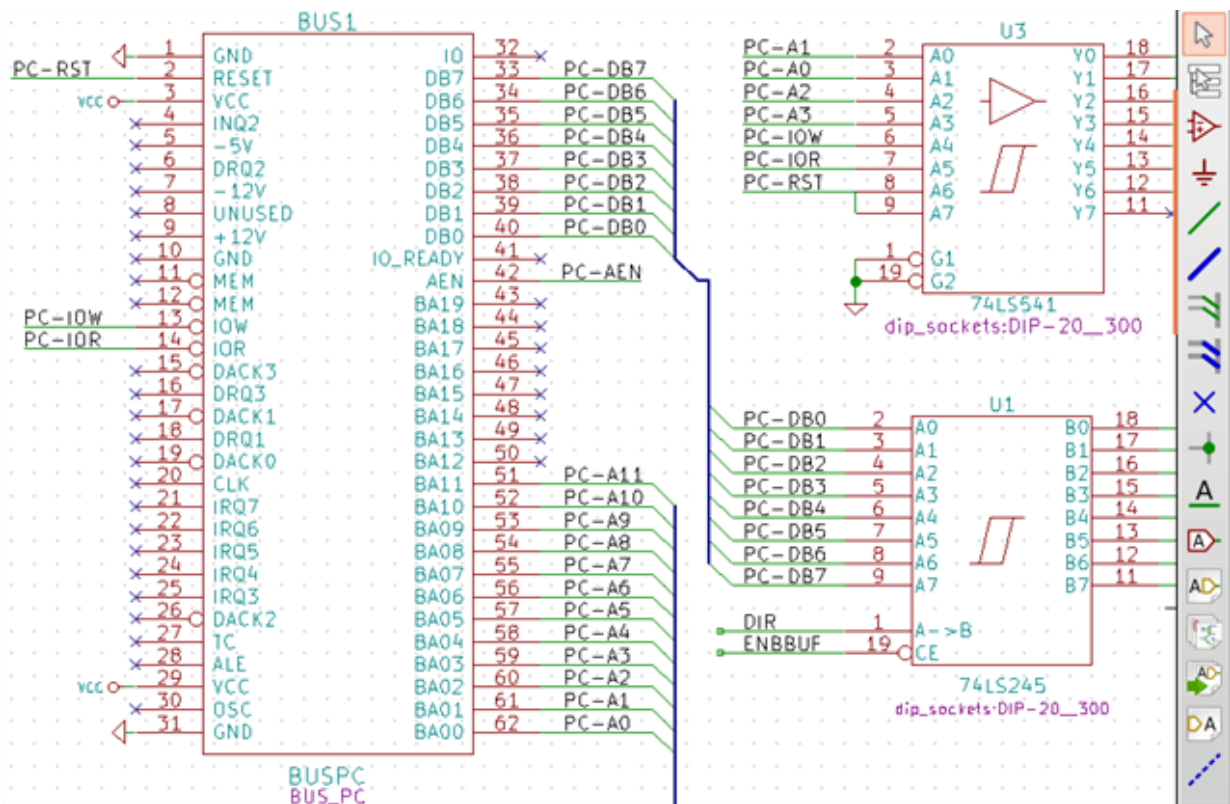
Poprzedni rysunek (połączenia doprowadzone do szpilek 22, 21, 20, 19 złącza DB25FEMALE) ukazuje taki przypadek połączeń za pomocą węzła.

**Uwaga 4:**

Jeśli dwie różne etykiety są umieszczone na tym samym połączeniu, zostają one połączone ze sobą i stają się równoważne: wszystkie inne elementy związane z jedną lub drugą etykietą zostają połączone razem.

### 6.5.3 Połączenia - Magistrale

Na poniższym schemacie, wiele pinów jest połączonych z magistralami.



#### 6.5.3.1 Składniki magistral

Z punktu widzenia schematu, magistrala jest zbiorem sygnałów, począwszy od wspólnego prefiksu, a skończywszy na liczbie. Przykładowo, PCA0, PCA1, PCA2 są członkami magistrali PCA.

Cała magistrala jest zwana PCA[N...m], gdzie N i m to pierwszy i ostatni numer połączenia tej magistrali. Zatem, jeśli PCA posiada 20-tu członków od 0 do 19, kompletna magistrala to PCA [0...19]. Ale zbiór sygnałów, takich jak PCA0, PCA1, PCA2, WRITE, READ nie może być zawarty w magistrali.

#### 6.5.3.2 Połączenia pomiędzy składnikami magistral

Wyprowadzenia między tymi samymi członkami magistrali muszą być połączone poprzez etykiety. Nie jest możliwe bezpośrednie połączenie pinu z magistralą; ponieważ takie połączenia będą ignorowane przez Eeschema.

W powyższym przykładzie, połączenia wykonane są za pomocą etykiet umieszczonych na połączeniach podłączonych do wyprowadzeń. Połączenia poprzez wejścia do magistral (odcinki połączeń pod kątem 45 stopni) mają wyłącznie wartość estetyczną, i nie są konieczne na poziomie czystego schematu.

W rzeczywistości, ze względu na polecenie powtórzenia (klawisz *Insert*), połączenia mogą być bardzo szybko wykonane tym sposobem, jeśli wyprowadzenia elementu są ustawione w porządku rosnącym (częsty przypadek w praktyce, na przykład w pamięciach, procesorach...):



- Najpierw należy umieścić pierwszą etykietę (np. PCA0)
- Użyć polecenia powtórzenia tyle razy ile potrzeba, aby umieścić kolejne etykiety. Eeschema automatycznie utworzy następne etykiety (PCA1, PCA2 ...) pionowo, teoretycznie w miejscu innych wyprowadzeń.
- Narysować połączenie pod pierwszą z etykiet. Następnie użyć polecenia powtarzania umieszczając dalsze połączenia pod etykietami.
- W razie potrzeby umieścić wejścia do magistrali w ten sam sposób (Umieścić pierwsze wejście, a następnie użyć polecenia powtarzania).

---

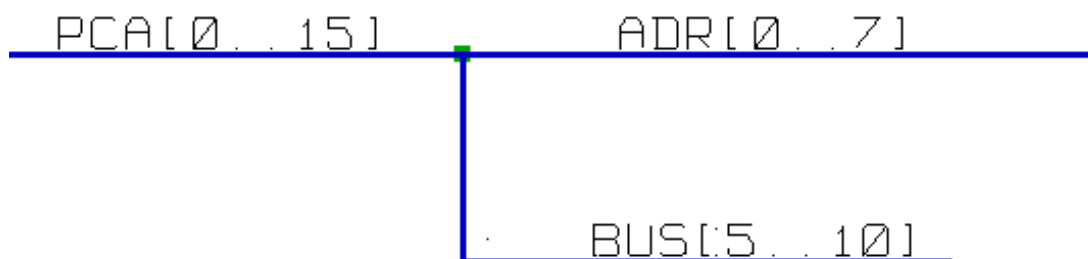
**Notatka**

W menu **Ustawienia/Opcje edytora schematów**, można ustalić parametry powtarzania:

- Przeskok w pionie.
  - Przeskok w poziomie.
  - Przyrost etykiety (który może być dodatni np. 2, 3. lub ujemny -2, -3).
- 

### 6.5.3.3 Globalne połączenia pomiędzy magistralami

Konieczne może być też połączenie pomiędzy magistralami, w celu połączenia dwóch magistral o różnych nazwach, lub w przypadku hierarchii, do tworzenia połączeń między różnymi arkuszami. Można dokonać tych połączeń w następujący sposób:



Magistrale PCA [0..15], ADR [0..7] oraz BUS [5..10] są ze sobą połączone (Należy zwrócić uwagę na węzeł, ponieważ segment pionowej magistrali łączy się w środku poziomego segmenty magistrali).

Dokładniej, odpowiedni członkowie każdej z magistral są ze sobą złączeni: PCA0, ADR0 są połączone (tak samo jak PCA1 i ADR1 ...PCA7 i ADR7).

Ponadto PCA5, BUS5 i ADR5 są podłączone (tak jak PCA6, BUS6 i ADR6 jak PCA7, BUS7 i ADR7).

A także PCA8 i BUS8 są podłączone (podobnie jak PCA9 i BUS9, PCA10 i BUS10).

---

### 6.5.4 Połączenia z symbolami zasilania

When the power pins of the symbols are visible, they must be connected, as for any other signal.


Symbols such as gates and flip-flops may have invisible power pins. Care must be taken with these because:

- Nie można ich połączyć, ze względu na ich niewidzialność.
- Nie znamy ich nazw.

Poza tym, złym pomysłem będzie ich uwidocznienie i łączenie tak jak inne wyprowadzenia, bo schemat stanie się nieczytelny, i nie będzie zgody z przyjętą konwencją.

#### Notatka

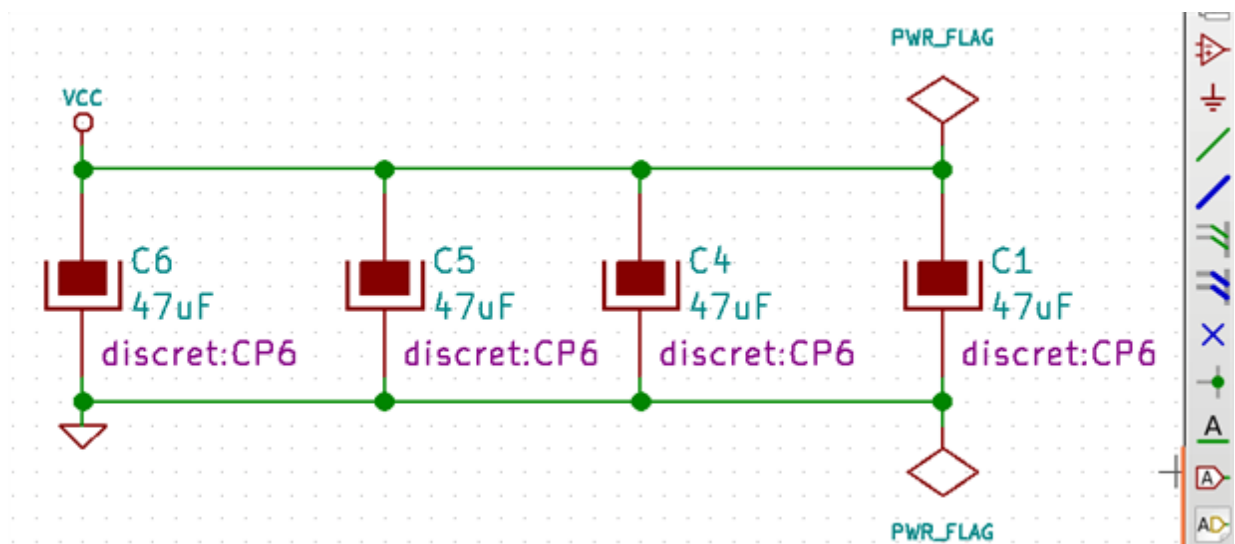
Jeśli chcemy aby były one widoczne, należy zaznaczyć opcję "Pokaż ukryte piny" w oknie dialogowym **Ustawienia/Opcje**

**edytora schematów** dostępnym z głównego menu aplikacji, lub za pomocą ikony  znajdującej się na lewym pasku narzędzi (pasek opcji).

Eeschema automatycznie łączy niewidoczne piny zasilania o tej samej nazwie do sieci o tej samej nazwie. Może być zatem potrzeba jawnego połączenia pinów zasilania o różnych nazwach (np. "GND" w symbolach TTL i "VSS" w symbolach CMOS); za pomocą dwóch portów zasilania.

Nie należy w tym celu używać etykiet, które mają tylko *lokalne* możliwości łączeniowe i nie mogą łączyć niewidocznych wyprowadzeń zasilania.

Poniższy rysunek przedstawia przykład połączenia portów zasilania:




W tym wypadku, masa (GND) jest połączona z portem zasilania o nazwie VSS, a port zasilania VCC jest połączony do VDD.

Widoczne są też dwa symbole PWR\_FLAG. Informują one, że dwa porty zasilania VCC i GND są rzeczywiście podłączone do źródła napięcia zasilania. Bez tych dwóch flag, narzędzie ERC wykaże ostrzeżenie: *Ostrzeżenie: pin podłączony do innych pinów ale brak pinu sterowania.*

All these symbols can be found in the "power" symbol library.



### 6.5.5 Flaga “Nie połączone”

Te symbole są bardzo użyteczne podczas sprawdzania poprawności schematu (ERC) by narzędzie to, nie zwracało ostrzeżeń o niepodłączonych wyprowadzeniach.

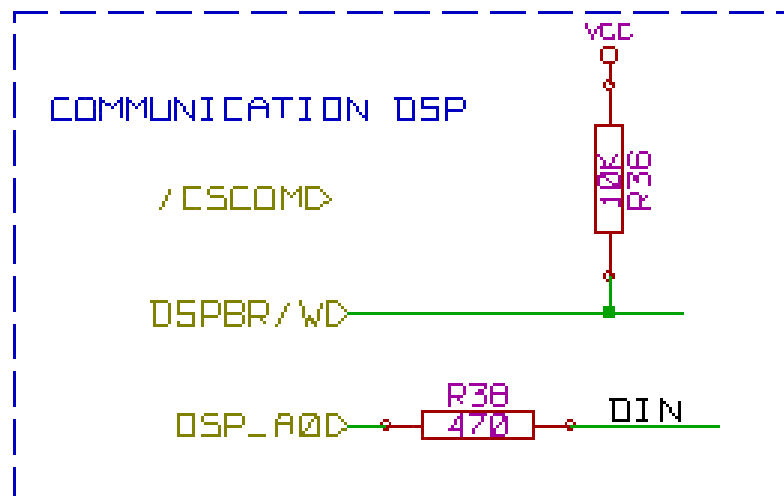
Jeśli wyprowadzenia muszą pozostać niepodłączone, wymagane jest aby dołączyć do tych wyprowadzeń specjalny symbol *Nie połączone* (narzędziem ). Symbole te nie mają żadnego wpływu na tworzone listy połączeń.

## 6.6 Elementy uzupełniające

### 6.6.1 Komentarze

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

Przykład ramki z zawartością w postaci tekstu.



### 6.6.2 Tabelka

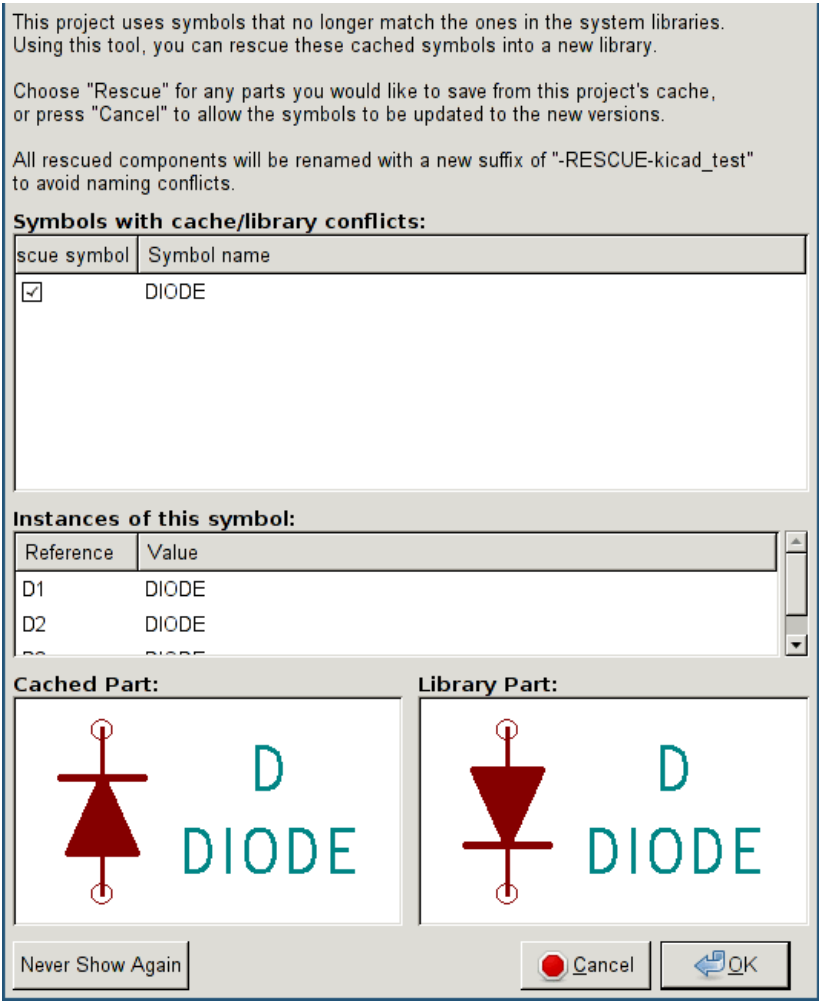
Ramka tytułowa jest modyfikowana za pomocą narzędzia .

Numer arkusza (*Sheet X/Y*) jest automatycznie aktualizowany.

## 6.7 Rescuing cached symbols

By default, Eeschema 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, Eeschema will scan the libraries for conflicts. Any conflicts found will be listed in the following dialog:



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 Eeschema 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.

## Rozdział 7

# Schematy o strukturze hierarchicznej


### 7.1 Wprowadzenie

Hierarchiczna reprezentacja jest szczególnie dobrym rozwiązaniem dla projektów większych niż kilka arkuszy. Jeśli chcielibyśmy zarządzać tego rodzaju projektem, niezbędne będzie:

- Użycie dużych arkuszy, co spowoduje problem z drukowaniem i obsługą.
- Wykorzystanie kilku arkuszy, która tym samym doprowadzi do stworzenia hierarchii.

Cały schemat następnie składa się w głównym arkuszu schematu zwanym arkuszem nadrzędnym (głównym) i arkuszy podrzędnych stanowiących hierarchię. Co więcej, umiejętny podział projektu na oddzielne arkusze często poprawia jego czytelność.

Z arkusza głównego, musi być możliwość odnalezienia dowolnego arkusza podrzędnego. Zarządzanie schematem hierarchicznym jest bardzo proste w Eeschema dzięki zintegrowanemu narzędziu jakim jest Źawigator”dostęny za pomocą

ikony  na górnym i prawym pasku narzędzi.

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.

Ten drugi typ jest raczej wykorzystany do opracowania układów scalonych, ponieważ w tym przypadku należy skorzystać z bibliotek funkcji w schemacie który rysujemy.

Eeschema obecnie nie obsługuje tego drugiego przypadku.

Hierarchia może być:

- prosta: dany arkusz jest używany tylko raz,
  - złożona: dany arkusz jest używany więcej niż raz (przypadek zwielokrotnienia),
  - płaska, która jest prostą hierarchią, ale połączenia między arkuszami nie są rysowane.
-

Eeschema uznaje wszystkie te hierarchie.

Stworzenie struktury hierarchicznej schematu jest łatwe, gdyż całość hierarchii jest obsługiwana z poziomu schematu głównego, tak jak gdyby był to tylko jeden schemat.

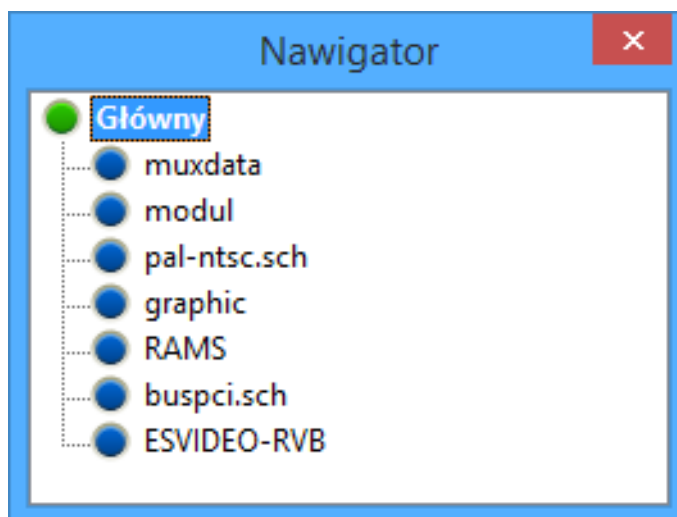
By opanować tworzenie hierarchii należy poznać dwie rzeczy:

- Jak stworzyć arkusz podrzędny.
- How to build electrical connections between sub-sheets.

## 7.2 Nawigacja wewnątrz hierarchii



Navigation among sub-sheets is achieved 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.


## 7.3 Etykiety lokalne, hierarchiczne i globalne

### 7.3.1 Właściwości



Lokalne etykiety (narzędzie ) łączą sygnały tylko w bieżącym arkuszu. Hierarchiczne etykiety (narzędzie ) łączą sygnały tylko wewnątrz arkusza z hierarchicznym pinem umieszczonym w arkuszu nadrzędnym.



Etykiety globalne (narzędzie ) łączą sygnały we wszystkich elementach hierarchii. Niewidoczne wyprowadzenia zasilania (typu *wejście zasilania* i *wyjście zasilania*) są traktowane jak globalne etykiety, ponieważ są one postrzegane jako związane między sobą we wszystkich elementach hierarchii.

---

**Notatka**

Wewnątrz hierarchii (prostej lub złożonej) można wykorzystać obie: hierarchiczne jak i globalne etykiety.

---

## 7.4 Summary of hierarchy creation

Aby stworzyć hierarchię prostą należy:

- Umieścić w arkuszu głównym symbol zwany symbolem arkusza hierarchicznego”.
- Wejść do nowego schematu (arkusza podrzędnego) za pomocą narzędzi nawigacji i narysować schemat tak jak zwykle.
- 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.

## 7.5 Symbole arkuszy podrzędnych

Aby narysować arkusz podrzędny, należy narysować prostokąt symbolizujący arkusz podrzędny.

Rozmiar tego prostokąta musi pozwolić na umieszczenie później etykiet czy pinów hierarchicznych odpowiadających etykiетom hierarchicznym wewnątrz arkusza podrzędnego.



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

Kliknąć w miejscu górnego lewego narożnika prostokąta. Kliknąć ponownie w miejscu dolnego prawego narożnika, pozostawiając prostokąt o dostatecznym rozmiarze.

Następnie program poprosi o wpisanie nazwy pliku oraz nazwy arkusza dla tego arkusza podrzędnego (w celu dotarcia do odpowiedniego schematu, z pomocą nawigatora).

Schematic Sheet Properties			
File name:	file5610032B.sch	Size:	1.524 millimeters
Sheet name:	Sheet5610032B	Size:	1.524 millimeters
Unique timestamp:	5610032C		
		<input type="button" value="Cancel"/> <input type="button" value="OK"/>	

Ostatecznie można podać tylko samą nazwę pliku. Jeśli nie ma nazwy arkusza, nazwa pliku będzie pełniła rolę nazwy arkusza (zwykle tak się robi).

---



## 7.6 Połączenia - Piny hierarchiczne


W tym punkcie stworzone zostaną punkty połączeń (piny hierarchiczne) dla symbolu, który został właśnie utworzony. 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.

Są dwie możliwości by wykonać takie połączenia:

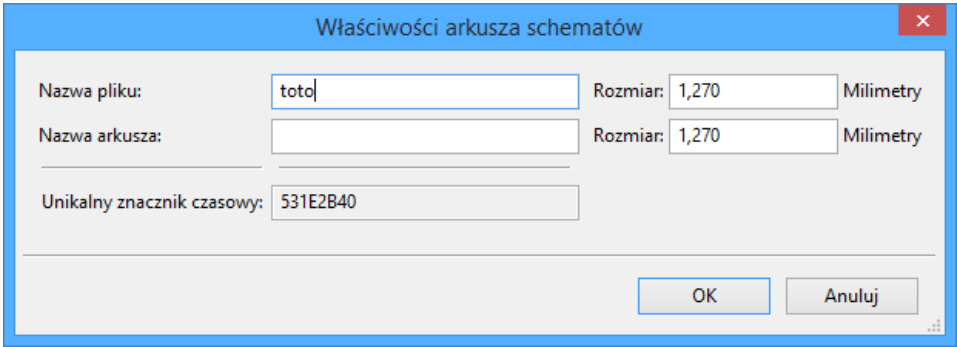
- Umieścić różne piny hierarchiczne w symbolu arkusza przed narysowaniem zawartości arkusza podrzędnego (wstawianie ręczne).
- Umieścić różne piny hierarchiczne po narysowaniu zawartości arkuszy podrzędnych oraz etykiet hierarchicznych (wstawianie półautomatyczne).

Drugie rozwiązanie wydaje się bardziej korzystne.

### Wstawianie ręczne pinów hierarchicznych

- ERC można uruchomić klikając w ikonę .
- Kliknąć na symbolu hierarchicznym gdzie chcielibyśmy utworzyć taki pin.


Poniżej znajduje się przykład tworzenia pinu hierarchicznego nazwanego "CONNECTION".



Można później zdefiniować dodatkowe atrybuty graficzne oraz rozmiar opisu edytując pin hierarchiczny, klikając prawym klawiszem, wybierając polecenie Edycja w podręcznym menu:

Wewnątrz arkusza muszą znaleźć się te same nazwy etykiet hierarchicznych co pinów hierarchicznych. Zapewnienie zgodności pomiędzy nimi musi być przeprowadzone manualnie, dlatego preferowana jest druga metoda opisana niżej.

### Wstawianie półautomatyczne pinów hierarchicznych

- Wybierz narzędzie .
- Kliknąć na symbol hierarchiczny, z którego chcemy zaimportować piny odpowiadające etykietom hierarchicznym umieszczonym w odpowiednim schemacie. Piny hierarchiczne pojawiają się, jeśli istnieją nowe etykiety, tzn. pominięte zostaną już dodane wcześniej piny.
- Kliknąć w miejscu gdzie taki pin ma się pojawić.

Wszystkie niezbędne piny mogą być więc umieszczone szybko i bez błędów. Przewagą w stosunku do wstawiania ręcznego jest to, że są one zgodne z odpowiednimi etykietami hierarchicznymi jakie istnieją na schemacie.

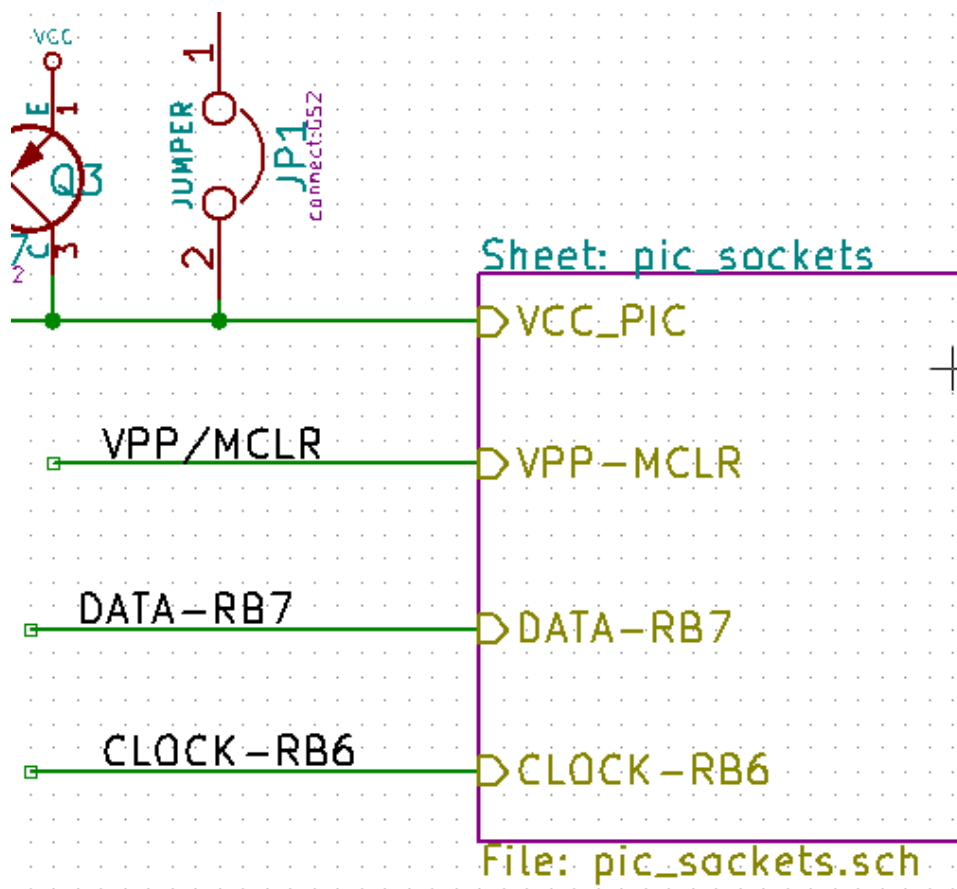
## 7.7 Etykiety hierarchiczne

Każdy stworzony pin w symbolu arkusza musi odpowiadać etykiecie hierarchicznej w arkuszu podrzędnym. Etykiety hierarchiczne są podobne do zwykłych etykiet, ale w przeciwieństwie do nich dostarczają one połączeń pomiędzy arkuszami podrzędnymi a arkuszem głównym (nadrzędnym). Graficzna reprezentacja dwóch etykiet uzupełniających (pinu i etykiety hierarchicznej) jest podobna. Tworzeniem etykiet hierarchicznych (HLabel) zajmuje się narzędzie



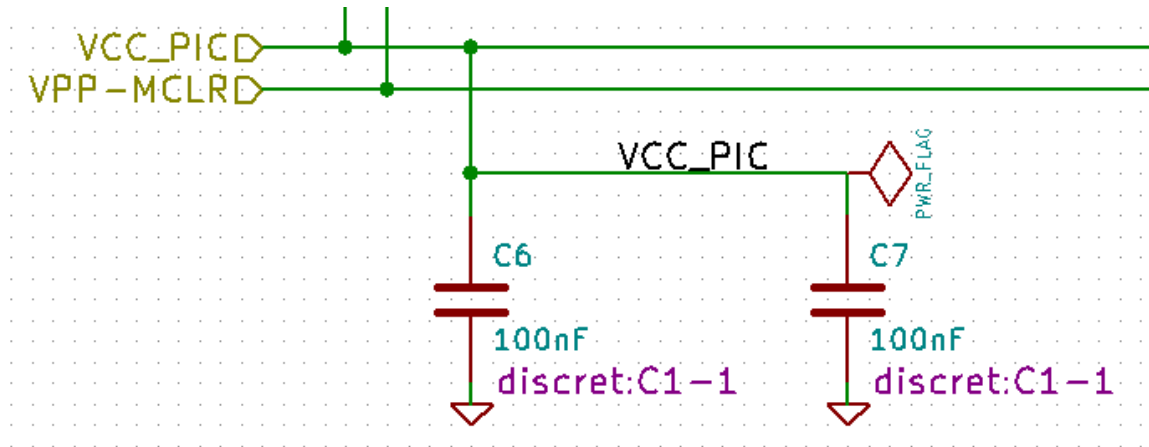
ukryte pod ikoną

Poniżej znajduje się przykład arkusza głównego:



Należy zwrócić uwagę na pin VCC\_PIC, połączony ze złączem JP1.

Na następnym rysunku znajdują się odpowiednie połączenia w arkuszu podrzędnym:



Należy zwrócić też uwagę na dwie odpowiadające pinom hierarchicznym etykiety hierarchiczne, pozwalające na zbudowanie połączenia pomiędzy arkuszami hierarchicznymi.

#### Notatka

Za pomocą etykiet hierarchicznych oraz pinów hierarchicznych, istnieje możliwość łączenia dwóch magistral, zgodnie z tym co opisano wcześniej przy opisie ich składników (Magistrala [N. .m]).

### 7.7.1 Etykiety, etykiety hierarchiczne, etykiety globalne oraz piny ukryte

Oto kilka uwag na temat różnych sposobów zapewnienia połączeń innych niż połączenia bezpośrednie.

#### 7.7.1.1 Zwykłe etykiety

Etykiety (zwykłe) mają lokalne możliwości łączeniowe, czyli ograniczone do arkusza schematu w którym się znajdują. Wynika to z faktu, że:

- Każdy arkusz posiada numer arkusza.
- Numer arkusza jest związany z nazwą etykiety.

Dlatego, jeśli umieścimy etykietę "TOTO" na arkuszu nr 3, w rzeczywistości prawdziwa jej nazwa będzie brzmieć "TOTO\_3". Jeśli w arkuszu numer 1 (arkusz główny) również umieścimy etykietę "TOTO" to w rzeczywistości zostanie tam umieszczona etykieta "TOTO\_1", różna od "TOTO\_3". Dzieje się tak zawsze, nawet jeśli istnieje tylko jeden arkusz.

#### 7.7.1.2 Etykiety hierarchiczne

To, co zostało napisane w związku z etykietami prostymi również jest prawdą w stosunku do etykiet hierarchicznych.

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.

### 7.7.1.3 Ukryte piny zasilania

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.

Gdyby etykieta o nazwie VCC została umieszczona na arkuszu podrzędnym, to nie byłaby ona połączona z wyprowadzeniem VCC, ponieważ etykieta byłaby faktycznie VCC\_n, gdzie n to numer arkusza.

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.

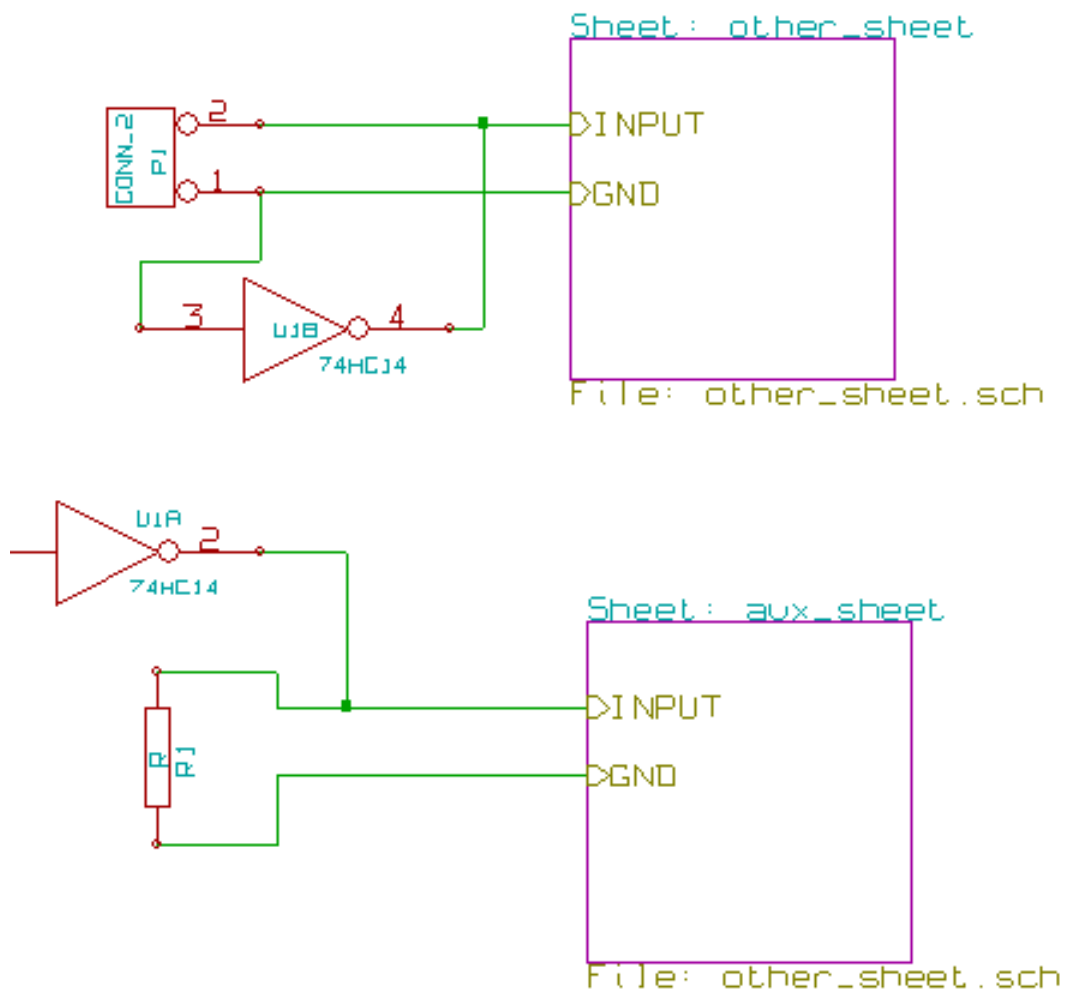
### 7.7.2 Etykiety globalne

Globalne etykiety, które mają identyczne nazwy połączone są w całej hierarchii.

(Porty zasilania jak np. VCC...są właśnie globalnymi etykietami).

## 7.8 Hierarchia złożona

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.

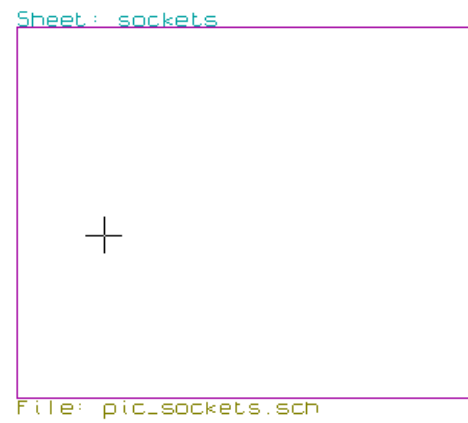
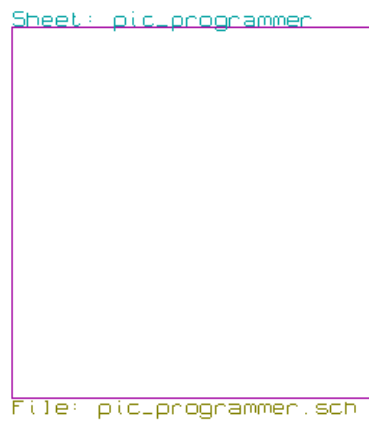


## 7.9 Hierarchia płaska

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

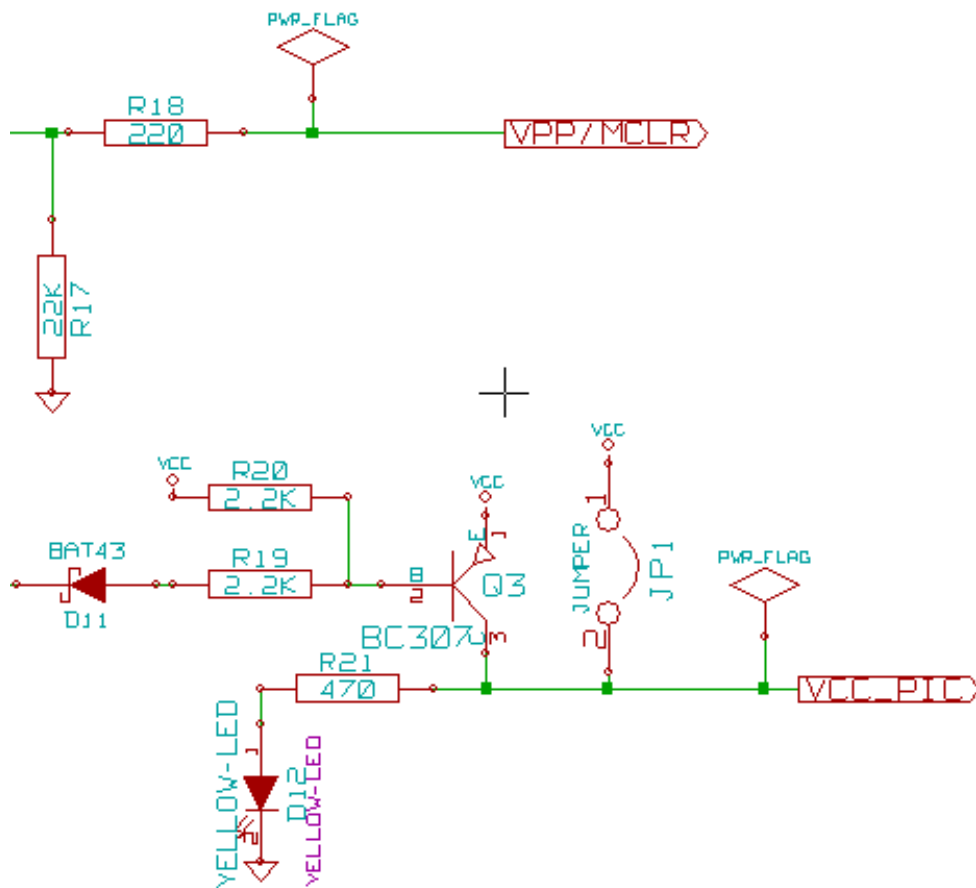
- Create a root sheet containing the other sheets which acts as a link between others sheets.
- Nie są potrzebne wyraźne połączenia między nimi.
- Use global labels instead of hierarchical labels in all sheets.

Poniżej znajduje się przykład głównego schematu:

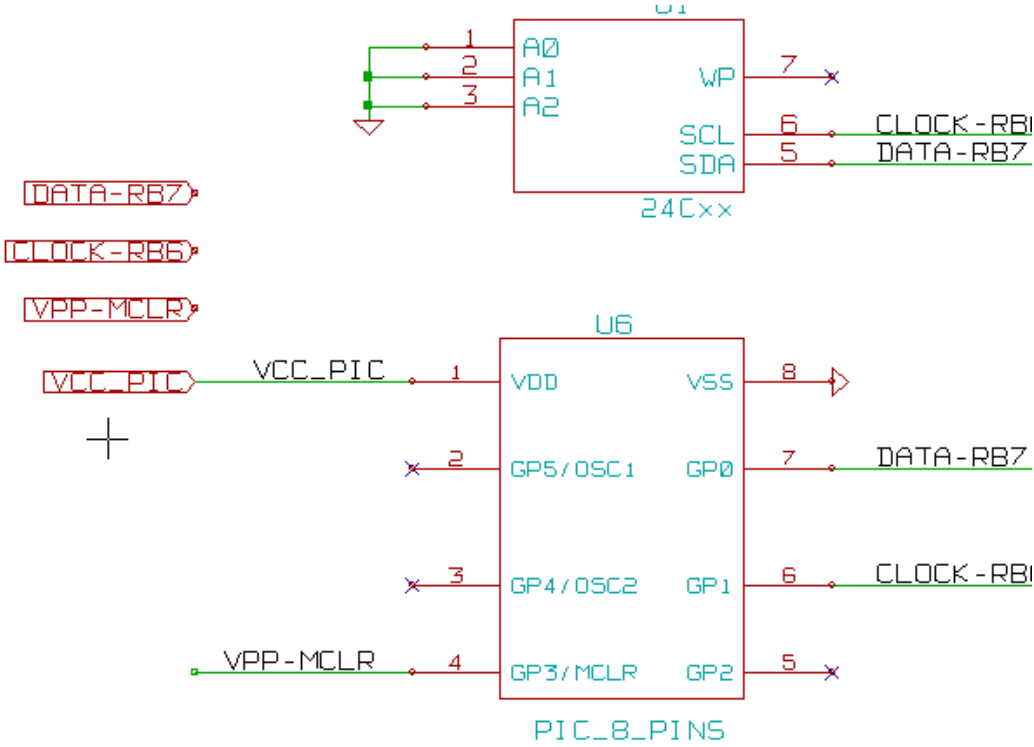


Poniżej znajdują się dwa arkusze, połączone za pomocą etykiet globalnych.

Tutaj znajduje się `pic_programmer.sch`.



Tutaj znajduje się `pic-sockets.sch`.



Widok na etykiety globalne.



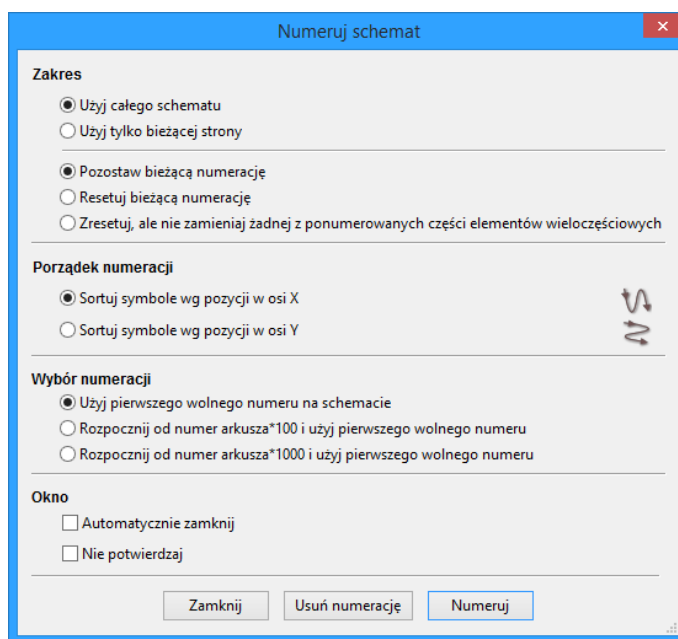
## Rozdział 8

# Symbol Annotation Tool

### 8.1 Wprowadzenie

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.



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.



- Opisywanie całej hierarchii (opcja Użyj całego schematu).
- Opisywanie tylko bieżącego arkusza (opcja Użyj tylko bieżącej strony).

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.

Opcje zawarte w grupie Wybór numeracji pozwalają wybrać metodę jaka zostanie wykorzystana podczas przypisywania numerów referencyjnych wewnątrz każdego arkusza w hierarchii.

Za wyjątkiem szczególnych przypadków, automatyczna numeracja ma zastosowanie do całego projektu (wszystkie arkusze) oraz tylko do nowych elementów, jeśli nie chcemy modyfikować poprzedniej numeracji.

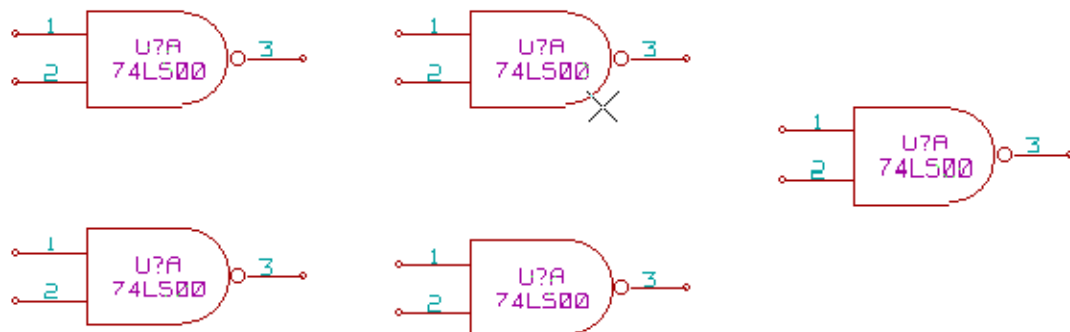
The Annotation Choice gives the method used to calculate reference:

- 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.
- Rozpocznij od numer arkusza\*100 i użyj pierwszego wolnego numeru: Numeracja zostanie rozpoczęta od liczby 101 dla arkusza 1, 201 dla arkusza 2, itd... Jeśli istnieje więcej niż 99 pozycji z tym samym prefiksem w nazwie odniesienia (np. U czy R) wewnątrz arkusza 1, numeracja będzie kontynuowana od liczby 200 i dalej, a numeracja w arkuszu 2 rozpocznie się od następnego wolnej liczby.
- Rozpocznij od numer arkusza\*1000 i użyj pierwszego wolnego numeru: Numeracja rozpocznie się od liczby 1001 dla arkusza 1, 2001 dla arkusza 2, itd...

## 8.2 Przykłady

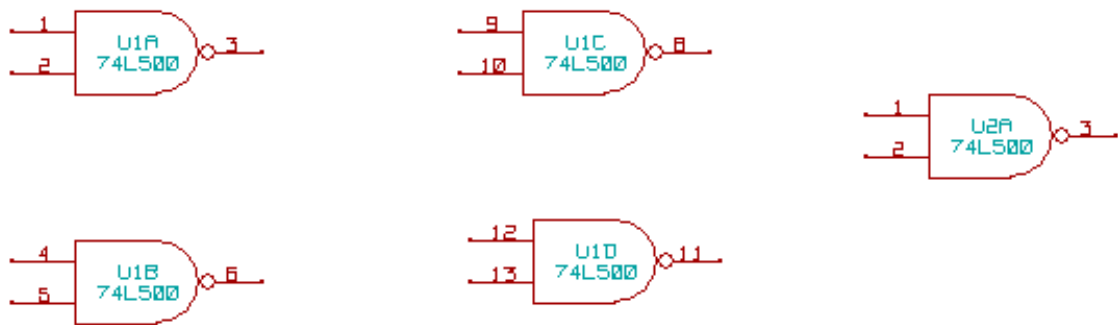
### 8.2.1 Zmiany porządku numeracji

Poniższy obrazek ukazuje 5 umieszczonych elementów, lecz jeszcze nie ponumerowanych.

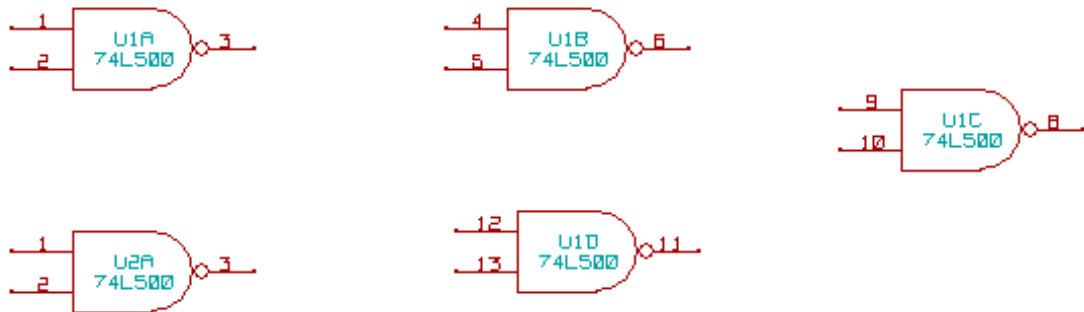


Po przeprowadzeniu automatycznej numeracji:

Z sortowaniem elementów według pozycji w osi X:



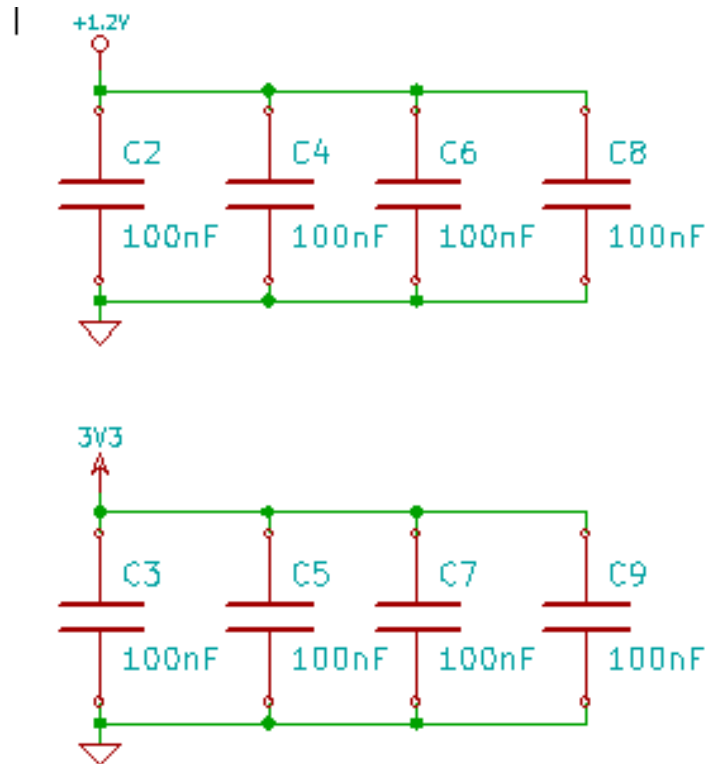
Z sortowaniem elementów według pozycji w osi Y:



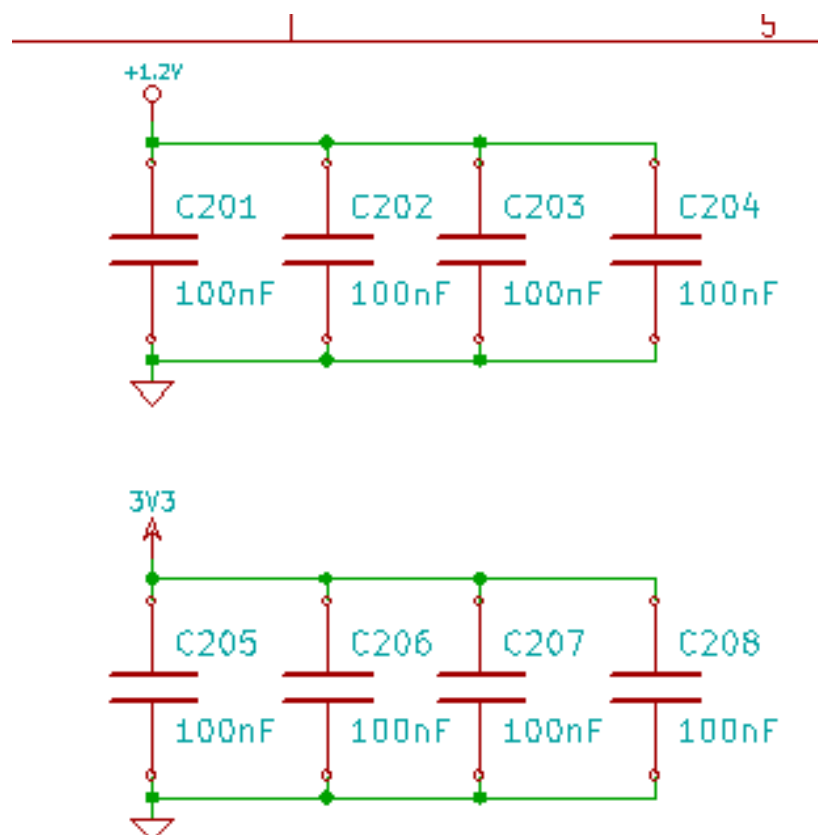
Można zauważyć, że cztery bramki układu 74LS00 zostały zawarte w układzie U1, a piąta bramka została przypisana do następnego układu U2.

### 8.2.2 Wybór numeracji

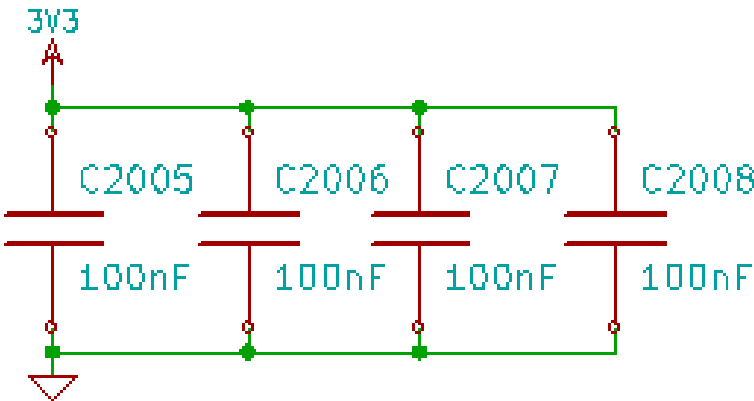
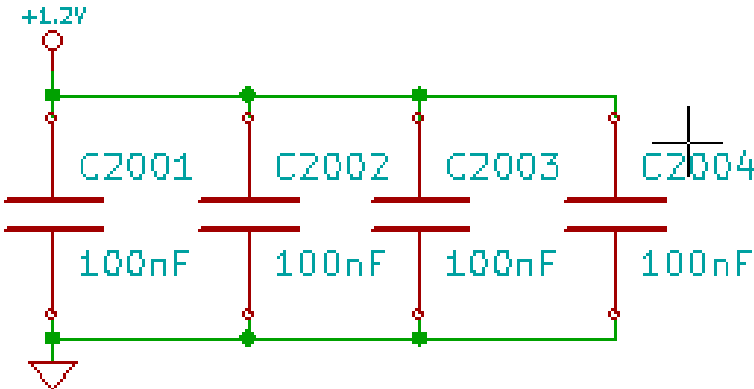
Poniżej znajdują się wyniki procesu numeracji dla arkusza numer 2, w zależności od wybranej opcji:



Opcja Rozpocznij od numer arkusza\*100 i użyj pierwszego wolnego numeru daje następujący efekt.



Opcja Rozpocznij od numer arkusza\*1000 i użyj pierwszego wolnego numeru daje następujący efekt.



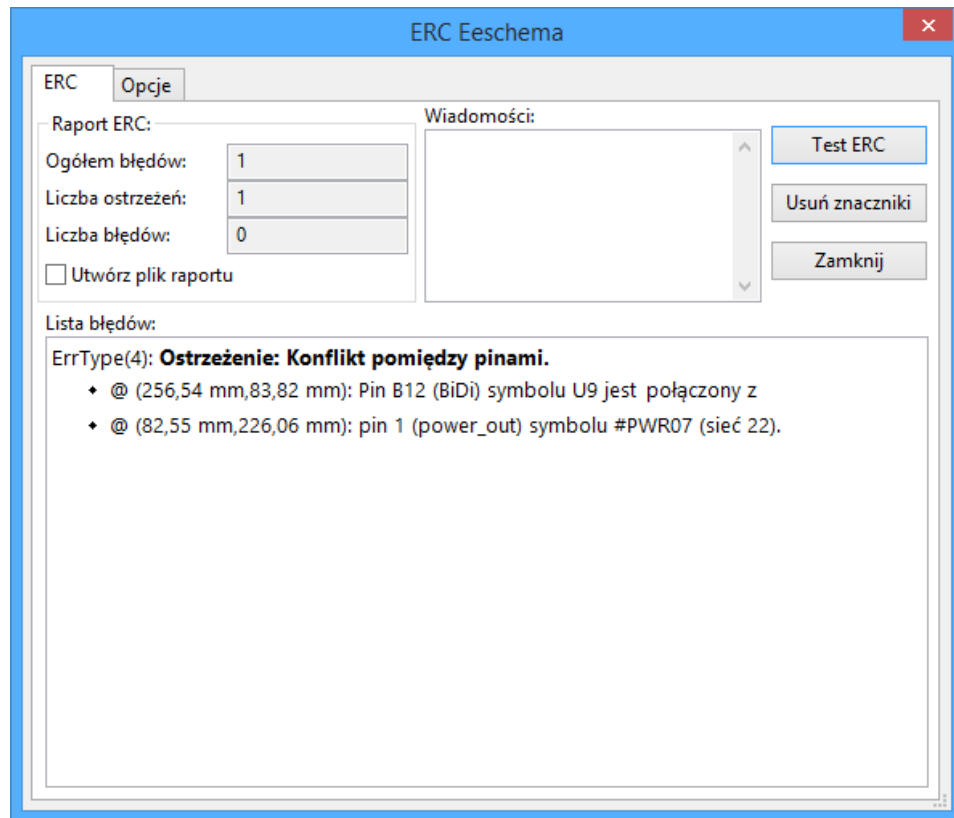
## Rozdział 9

# Kontrola reguł projektowych - Electrical Rules Check

### 9.1 Wprowadzenie

Funkcja Kontrola reguł projektowych ERC przeprowadza automatyczne sprawdzenie poprawności elektrycznej schematu. Wskazuje ona błędy na arkuszu schematu, takie jak: niepodłączone wyprowadzenia, niepodłączone wyprowadzenia w symbolach hierarchicznych, zwarcia pomiędzy wyjściami, itp. Naturalnie, sprawdzanie automatyczne nie jest bezbłędne, a oprogramowanie które mogłoby wykryć wszelkie błędy nie zostało jeszcze napisane. Aczkolwiek zwykła kontrola jest bardzo użyteczna, ponieważ pozwala na wykrycie wielu niedopatrzeń oraz małych błędów.

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".



## 9.2 Używanie narzędzia testu ERC

ERC można uruchomić klikając w ikonę .

Ostrzeżenia są umieszczane na elementach schematu, które spowodowały błąd ERC (piny lub etykiety).

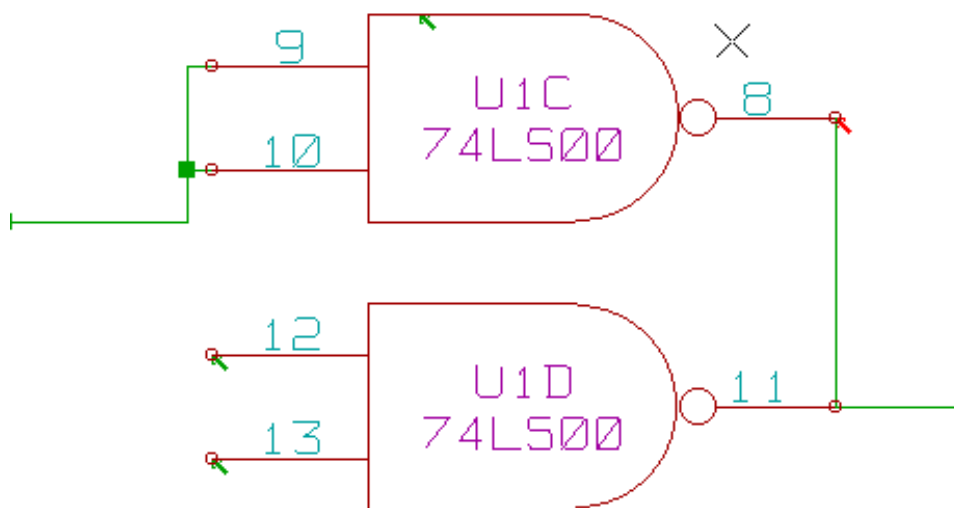
---

### Notatka

- Kliknięcie w informację o błędzie w tym oknie dialogowym spowoduje, że kursor zostanie przeniesiony do miejsca w którym znajduje się znacznik tego błędu na schemacie.
  - Na schemacie, kliknięcie prawym klawiszem na znacznik błędu umożliwia dostęp do informacji o powodzie błędu.
- 

Można także kasować znaczniki ERC z okna dialogowego.

### 9.3 Przykład testu ERC

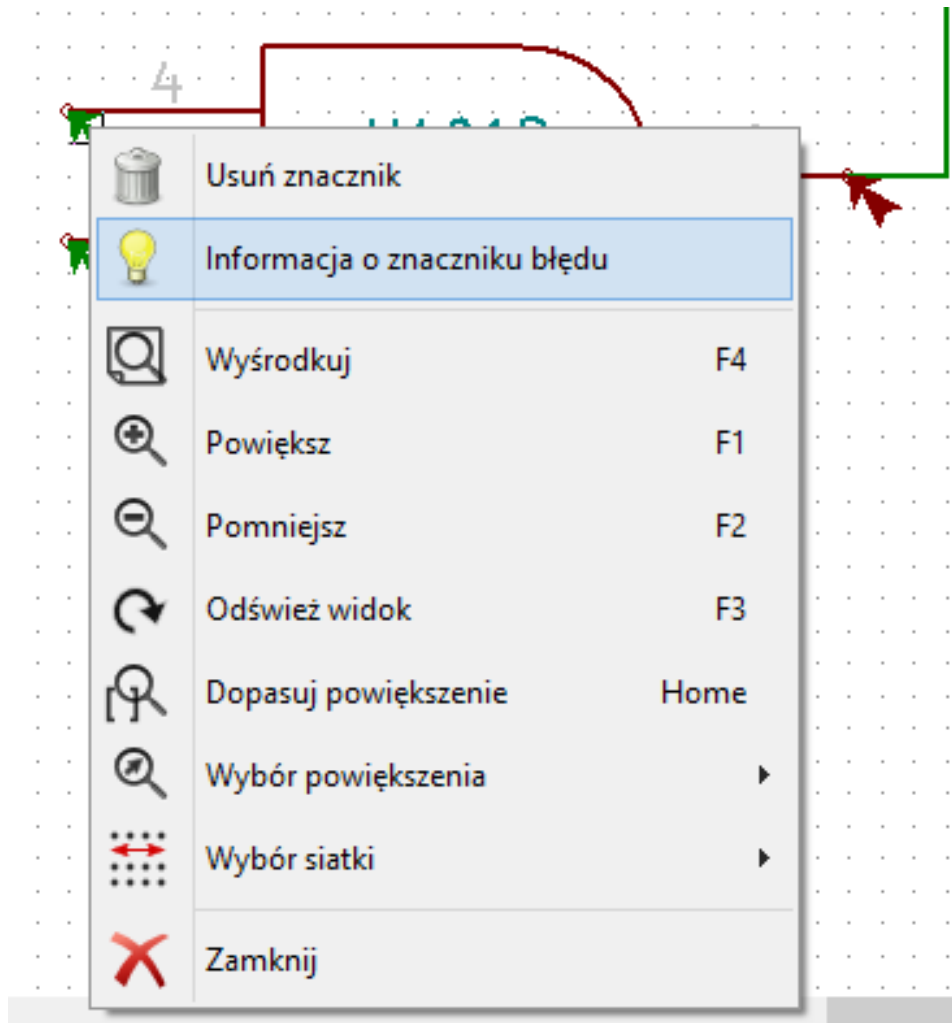


Na powyższym obrazku można zobaczyć cztery błędy:

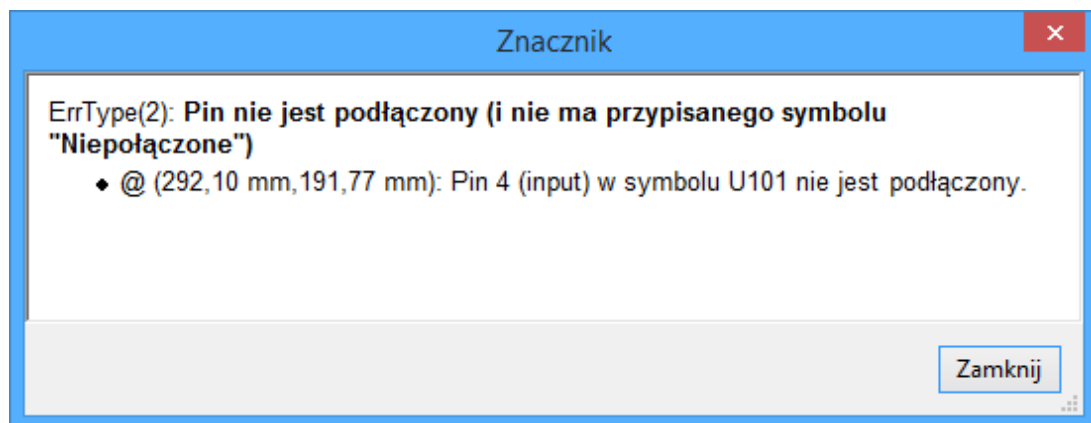
- Dwa wyjścia zostały błędnie połączone razem (czerwona strzałka).
- Dwa wejścia zostały niepodłączone (zielone strzałki).
- Jest też błąd na niewidocznym pinie zasilania, wskazujący na brak flagi zasilania (zielona strzałka na górze).

### 9.4 Wyświetlanie informacji o znaczniku błędu

Klikając prawym klawiszem na znaczniku błędu można z menu podręcznego wywołać okienko z informacją o tym błędzie ERC.



Wybierając polecenie Informacja o znaczniku błędu można zobaczyć dokładniejszy jego opis.



## 9.5 Piny zasilania i flagi zasilania

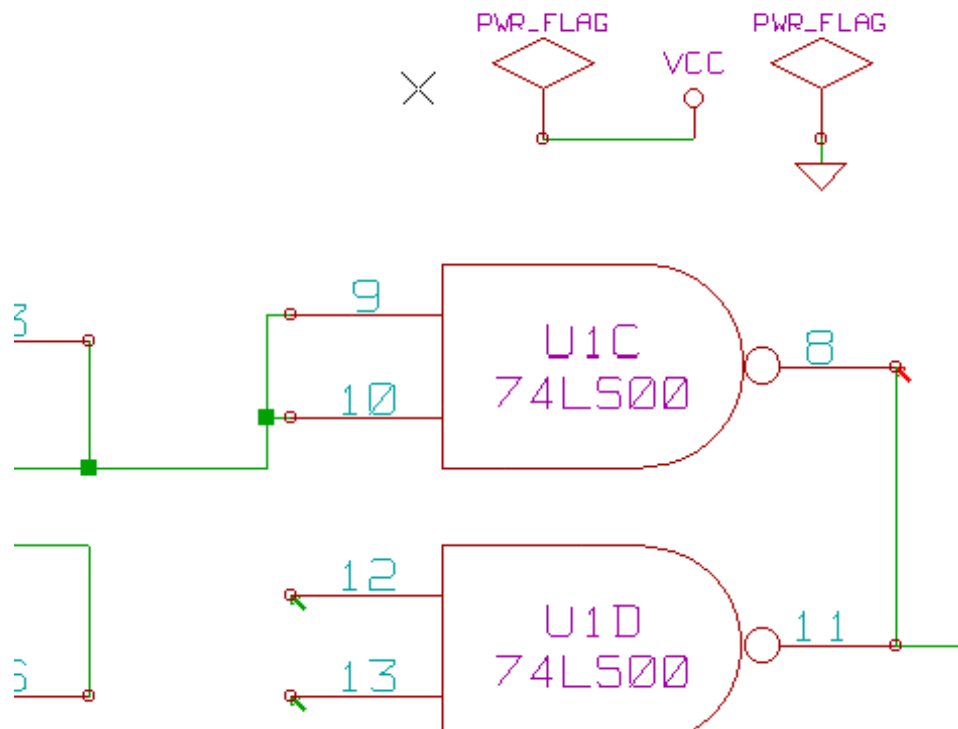
Dosyć często występuje błąd (ostrzeżenie) na wyprowadzeniach zasilania, podczas gdy na pierwszy rzut oka wszystko wydaje się poprawne (patrz przykład powyżej). To dlatego, że w większości projektów zasilanie jest dostarczane przez



złączki, które nie są źródłami zasilania gdyż ich wyprowadzenia mają funkcje pasywną (nie tak jak na przykład wyjście regulatora napięcia, którego piny są zwykle zadeklarowane jako źródło zasilania).

ERC wobec tego nie znajduje żadnego źródła zasilania do wysterowania takiej sieci i uznaje ją za nie wysterowaną (nie połączoną ze źródłem zasilania).

W takich przypadkach należy do takiej sieci przypiąć specjalny element: flagę "PWR\_FLAG" biblioteki `power.lib`, która sygnalizuje, że ta sieć jest w istocie źródłem zasilania.

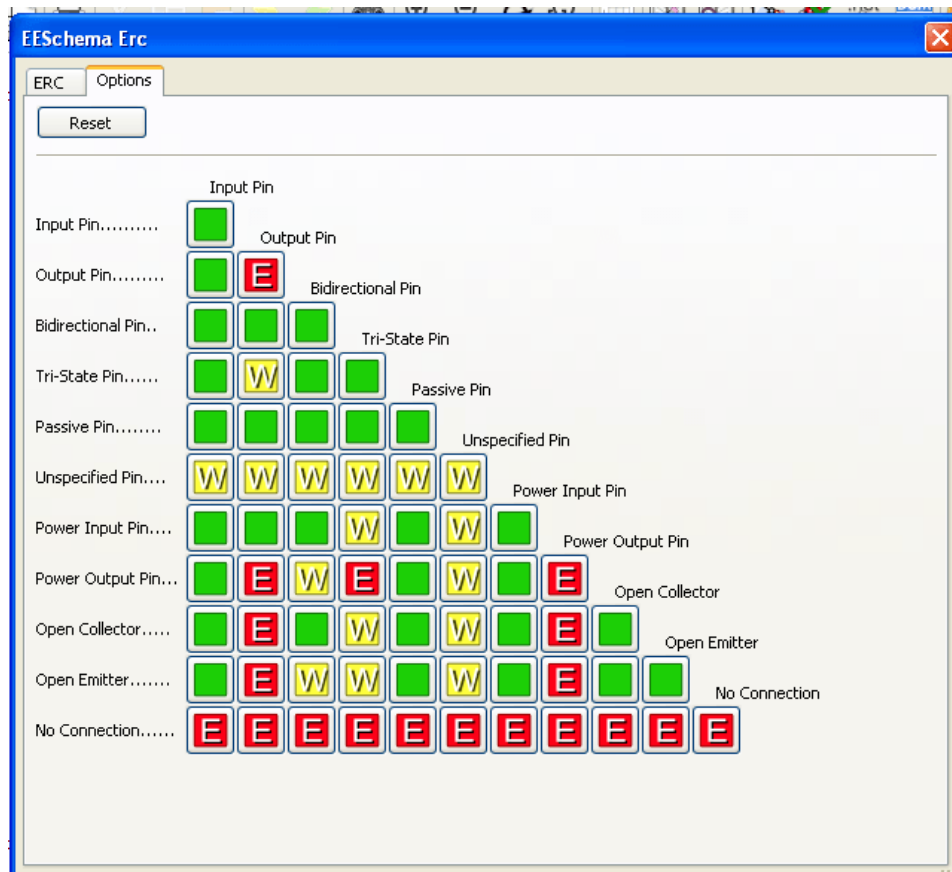


Po umieszczeniu na szynach zasilania tej flagi, błędy związane z brakiem sterowania powinny zniknąć podczas ponownej kontroli ERC.

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.

## 9.6 Konfiguracja

*Panel opcji* pozwala na skonfigurowanie reguł jakimi się ma kierować ERC w określonych przypadkach zestawienia połączenia, i czy w danym przypadku ma zostać wygenerowany błąd czy tylko ostrzeżenie, albo też takie zestawienie nie powinno generować błędu.



Poszczególne reguły mogą zostać zmienione poprzez kliknięcie na wybranym polu na powyższej matrycy. Kolejne kliknięcia pozwalają na wybranie: brak błędu (zielony), ostrzeżenie (żółty, W) i błąd (czerwony, E). Zmiany odbywają się w zamkniętym cyklu, zatem aby wrócić do poprzedniego stanu należy ponownie kliknąć (jedno- lub dwukrotnie).

## 9.7 Plik raportu ERC

Plik raportu ERC może zostać wygenerowany i zapisany poprzez zaznaczenie opcji "Utwórz plik raportu". Pliki takie są zapisywane z rozszerzeniem `.erc`. Poniżej znajduje się przykład zawartości takiego pliku:

```
Raport ERC (2011-09-24 08:46:02)
```

```
***** Arkusz / (Root)
```

```
ErrType(2): Pin nie jest podłączony (i nie ma przypisanego symbolu "Niepołączone")
```

```
@ (3,2500 ",3,0500 "): Element U10, Pin 5 (input) nie jest połączony
```

```
ErrType(3): Pin podłączony do innych pinów ale nie ma pinu sterującego
```

```
@ (3,6500 ",2,4000 "): Element U10, pin 7 (power_in) nie jest sterowany (sieć 6)
```

```
ErrType(3): Pin podłączony do innych pinów ale nie ma pinu sterującego
```

```
@ (3,6500 ",2,7500 "): Element U10, pin 14 (power_in) nie jest sterowany (sieć 5)
```

```
ErrType(5): BŁĄD: Konflikt pomiędzy pinami.
```

```
@ (4,4500 ",2,2000 "): Element U10: Pin 3 (output) połączony z
```

```
@ (4,4500 ",2,9500 "): Element U10: Pin 6 (output) (sieć 2)
```

```
ErrType(2): Pin nie jest podłączony (i nie ma przypisanego symbolu "Niepołączone")
```

```
@ (3,2500 ",2,8500 "): Element U10, Pin 4 (input) nie jest połączony
```

>> Błędy ERC: 5

## Rozdział 10

# Generowanie list sieci

### 10.1 Przegląd zagadnień

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.
- Syntetyzery układów PAL/PLA (oraz innych układów programowalnych).

Eeschema wspiera kilka formatów list sieci:

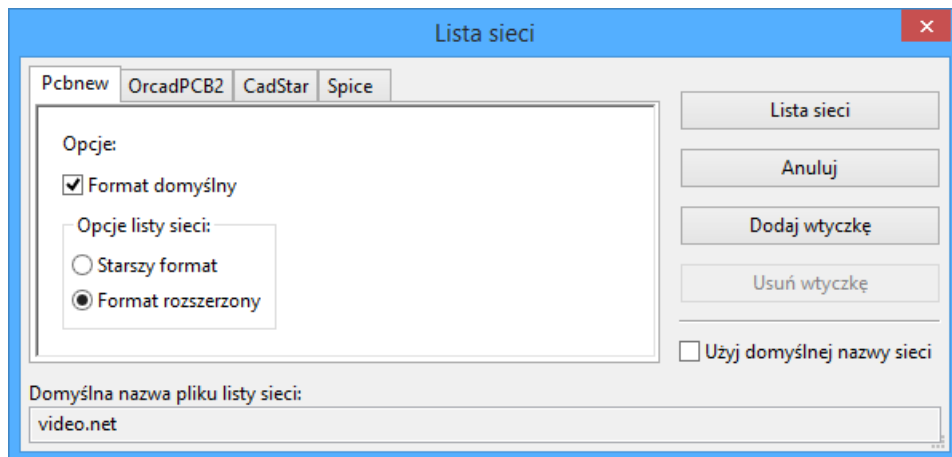
- Format Pcbnew (obwody drukowane).
  - Format ORCAD PCB2 (obwody drukowane).
  - Format CADSTAR (obwody drukowane).
  - Format Spice, używany przez sporą grupę symulatorów nie tylko przez PSpice.
-

## 10.2 Format listy sieci

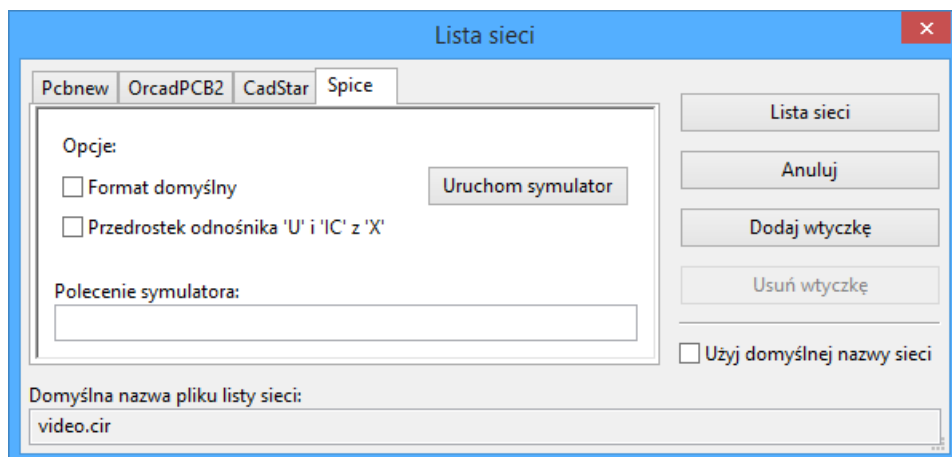


Select the tool  to open the netlist creation dialog.

Wybrany format Pcbnew



Wybrany format 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.

---

### Notatka

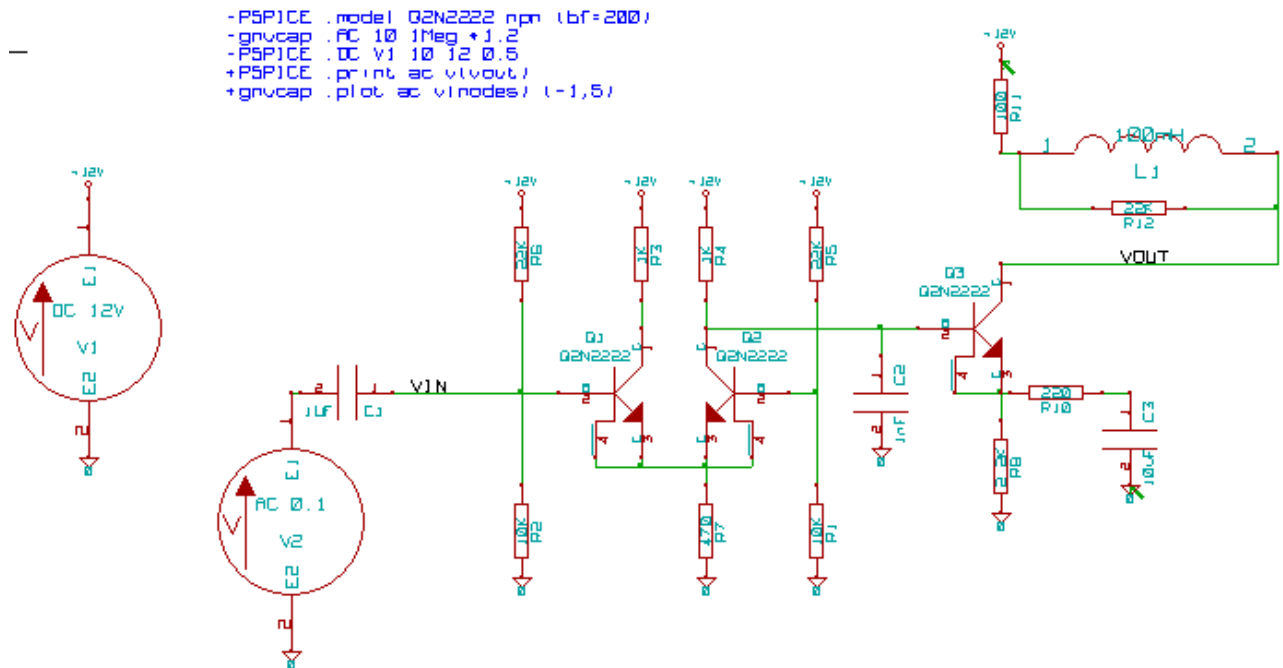
The netlist generation can take up to several minutes for large schematics.

---

## 10.3 Przykłady listy sieci

Na poniższym rysunku znajduje się schemat używający biblioteki PSPICE:

---



Struktura listy sieci programu Pcbnew:

```

# Eeschema Netlist Version 1.0 generated 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
```

W formacie PSPICE, lista sieci byłaby następująca:

```
* 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
```

## 10.4 Uwagi odnośnie list sieci

### 10.4.1 Zalecane środki ostrożności

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.

W ten sam sposób, niektóre znaki inne niż litery i cyfry mogą również powodować problemy. Należy pamiętać, że ograniczenie to nie jest związane z Eeschema, ale z formatami list sieci, które mogą następnie stać się nieprzetłumaczalne dla oprogramowania, które korzysta z tych list sieci.

### 10.4.2 Listy sieci PSPICE

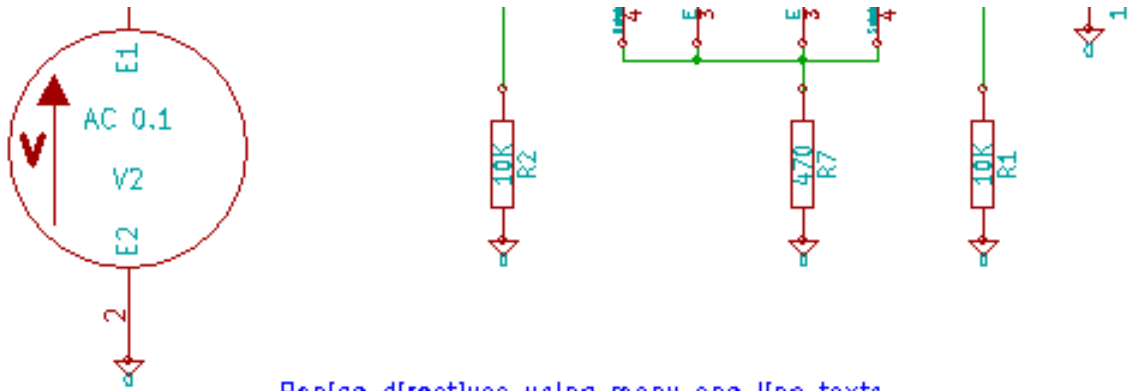
Dla symulatora PSpice trzeba do listy sieci dodać kilka linii z poleceniami dla symulatora (.PROBE, .AC ...). Można je umieścić bezpośrednio na schemacie.

Każdy wiersz tekstu umieszczonego na schemacie, rozpoczynający się od słów kluczowych: **-pspice** lub **-gnucap** zostanie wstawiony (bez słów kluczowych) na początku listy sieci.

Każdy wiersz tekstu umieszczonego na schemacie rozpoczynający się od słów kluczowych: **+gnucap** lub **+pspice** zostanie dopisany (bez słów kluczowych) na koniec listy sieci.

Poniżej znajduje się przykład, na którym użyto wielu jednoliniowych tekstów poleceń, a także jeden wieloliniowy tekst polecenia:





Pspice directives using many one line texts

```
-PSPICE .model Q2N2222 npn (bf=200)
-gnucap .AC dec 10 1Meg *1.2
-PSPICE .DC V1 10 12 0.5
+PSPICE .print ac v(vout)
+gnucap .plot ac v(nodes) (-1.5)
```

Pspice directives using one multiline text:

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

Przykładowo, jeśli zostanie wpisany następujący tekst (nie może to być etykieta!):

```
-PSPICE .PROBE
```

linia `.PROBE` zostanie wstawiona do listy sieci.

W poprzednim przykładzie dzięki tej technice, trzy linie poleceń zostaną wstawione na początek listy sieci, oraz dwie linie poleceń na końcu.

Jeśli użyty został format wieloliniowy poleceń, **+pspice** lub **+gnucap** są wymagane tylko na początku:

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

Taki zapis utworzy następujący tekst:

```
.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.

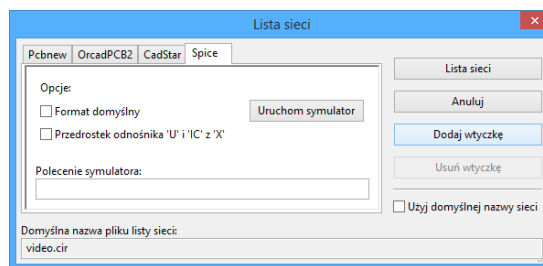
## 10.5 Inne formaty

Dla innych formatów list sieci można dodać odpowiednie konwertery w formie wtyczek. Konwertery te są automatycznie uruchamiane przez Eeschema. W rozdziale 14 znajdują się wyjaśnienia i przykłady takich konwerterów.

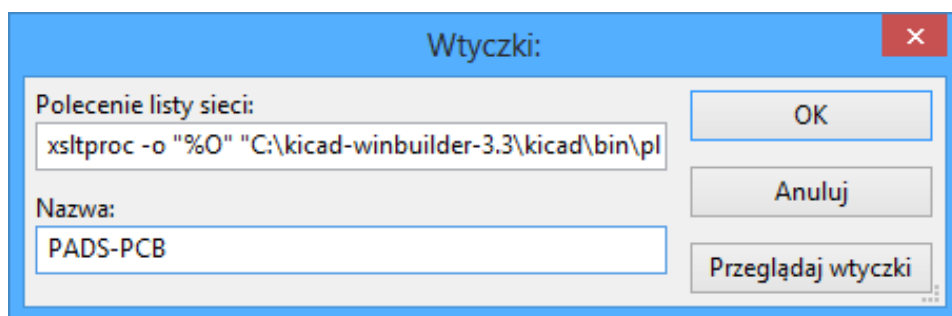
Konwerter to plik tekstowy (w formacie xsl), ale można korzystać z innych języków takich jak Python. W przypadku użycia formatu xsl, odpowiednie narzędzia (xsltproc.exe lub xsltproc) odczytuje plik pośredni stworzony przez Eeschema i plik konwertera, w celu stworzenia pliku wyjściowego. W tym przypadku plik konwertera (arkusz stylów) jest bardzo krótki i łatwy do napisania.

### 10.5.1 Inicjowanie okna dialogowego

Można dodać nową wtyczkę do list sieci używając przycisku "Dodaj wtyczkę".



Poniżej znajduje się okno ustawień nowej wtyczki PADS-PCB:



By skonfigurować wtyczkę będzie potrzebny:

- Tytułu zakładki (określająca również nazwę formatu wyjściowego listy sieci).
- Wtyczka którą należy uruchomić.

Gdy lista sieci jest generowana:

1. Eeschema tworzy pośredni plik z listą sieci `*.tmp`, na przykład `test.tmp`.
2. Eeschema uruchamia wtyczkę, która czyta plik `test.tmp` i tworzy plik `test.net`.

### 10.5.2 Format linii poleceń

Poniżej znajduje się przykład użycia `xsltproc.exe` jako konwertera plików `.xsl`, oraz pliku `netlist_form_pads-pcb.xsl` jako arkusza stylów:

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

Gdzie:

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 Eeschema (*.tmp).

Dla przykładowego schematu nazwanego `test.sch`, właściwa linia poleceń ma postać:

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

### 10.5.3 Konwerter i arkusz stylów (wtyczka)

Jest to bardzo proste oprogramowanie, ponieważ jego celem jest tylko konwersja tekstowego pliku wejściowego (pośredni plik tekstowy) do innego pliku tekstowego. Co więcej, z pośredniego pliku tekstowego można również utworzyć listę BOM.

Podczas korzystania `xsltproc` jako konwertera zostanie wygenerowany tylko arkusz stylów.

### 10.5.4 Format pośredniej listy sieci

Rozdział 14 zawiera więcej wyjaśnień na temat `xsltproc`, opisów formatu pliku pośredniego oraz niektóre przykłady arkuszy stylów dla konwerterów.

## Rozdział 11

# Drukowanie i rysowanie schematów na drukarkach lub ploterach

### 11.1 Wprowadzenie

Obie możliwości przenoszenia schematów na papier (bądź inny materiał drukarski) są dostępne z menu Plik:



Formatami wyjściowymi mogą być Postscript, HPGL, SVG, DXF lub PDF. Można także drukować bezpośrednio na zwykłej drukarce.

### 11.2 Polecenia wspólne

#### Rysuj bieżącą stronę

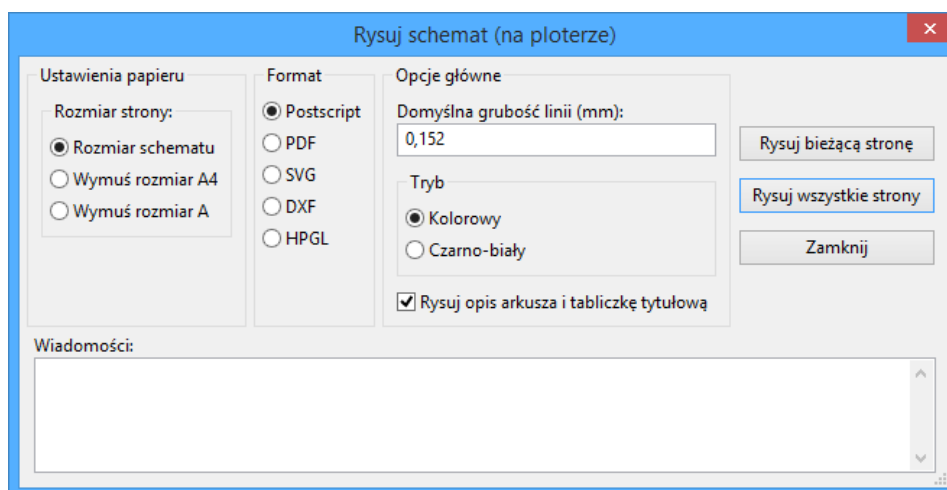
generuje plik wyjściowy wyłącznie dla bieżącego arkusza.

#### Rysuj wszystkie strony

pozwała na narysowanie całej hierarchii (pliki są generowane dla każdego z arkusza).

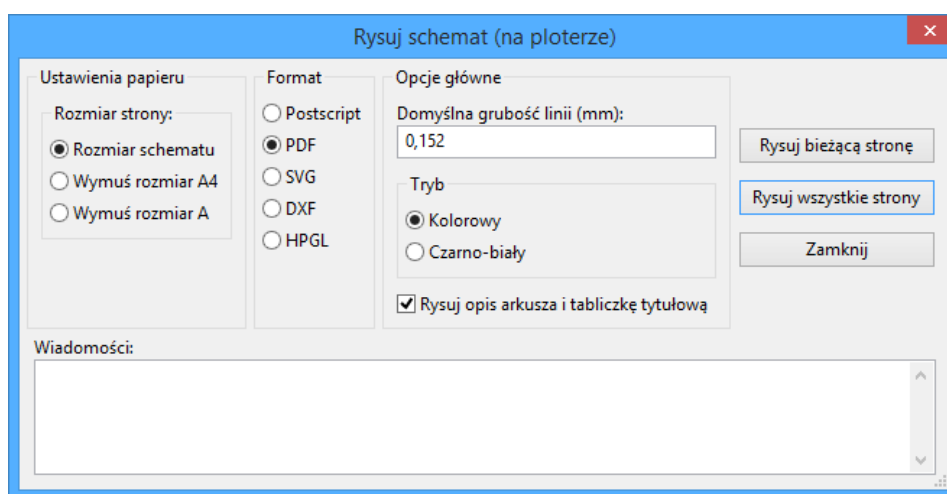
### 11.3 Rysuj w formacie PostScript

To polecenie pozwala na stworzenie rysunku schematu w formacie PostScript.



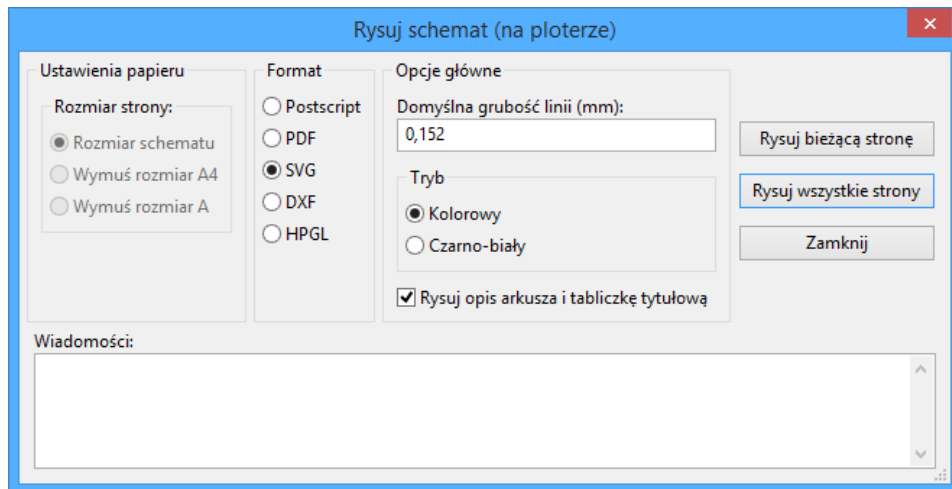
Nazwa pliku wyjściowego składała się będzie z nazwy arkusza i rozszerzenia **.ps**. Można dodatkowo odznaczyć opcję: „Rysuj opis arkusza i tabliczkę tytułową”. Jest to użyteczne w przypadku tworzenia pliku postscriptowego do późniejszej obróbki (format **.eps**), aby umożliwić wstawianie rysunków do procesora tekstu.

## 11.4 Rysuj w formacie PDF



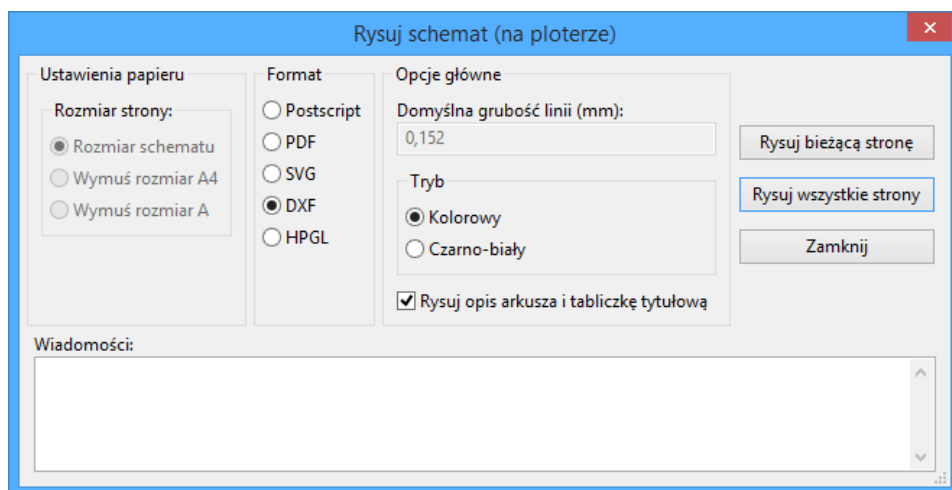
Pozwala na utworzenie rysunku schematu w formacie PDF. Nazwa pliku wyjściowego składała się będzie z nazwy arkusza i rozszerzenia **.pdf**.

## 11.5 Rysuj w formacie SVG



To polecenie pozwala na utworzenie plików, które zawierać będą skalowane rysunki wektorowe - SVG. Nazwa pliku wyjściowego składała się będzie z nazwy arkusza i rozszerzenia `.svg`.

## 11.6 Rysuj w formacie DXF



Pozwala na utworzenie plików z rysunkami CAD używając popularnego formatu DXF. Nazwa pliku wyjściowego składała się będzie z nazwy arkusza i rozszerzenia `.dxf`.

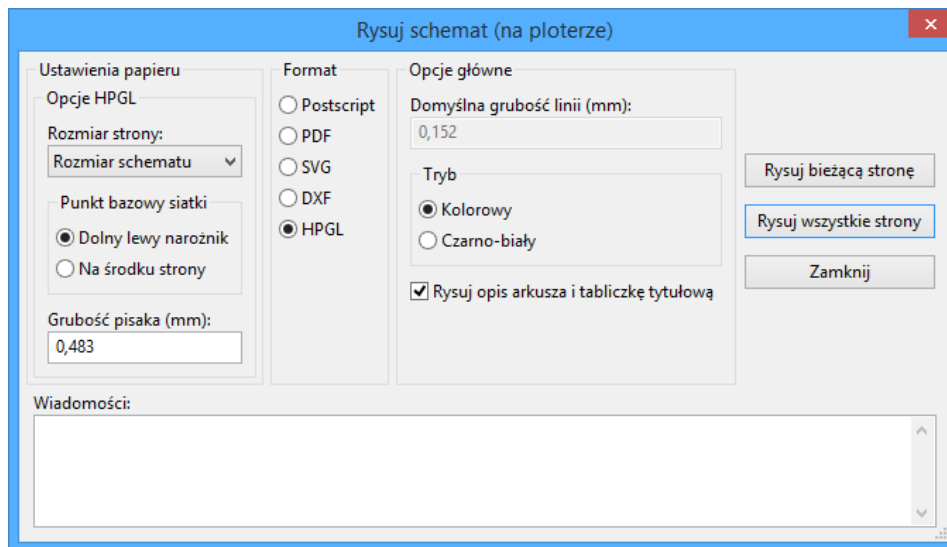
## 11.7 Rysowanie w formacie HPGL

Polecenie pozwala na stworzenie pliku dla plotera obsługującego format HPGL. W tym formacie można zdefiniować kilka parametrów dla plotera:

- Rozmiar arkusza.

- Punkt bazowy.
- Rozmiar pisaka (w mm).

Po wybraniu tego polecenia otworzy się następujące okno:



Nazwa pliku wyjściowego składała się będzie z nazwy arkusza i rozszerzenia .plt.

### 11.7.1 Wybór rozmiaru arkusza

Normalnie jest zaznaczona opcja *Rozmiar schematu*. W takim przypadku, rozmiar arkusza plotera będzie taki sam jak rozmiar arkusza określony w Eeschema, a skala będzie wynosić 1. Jeśli wybrano inny rozmiar arkusza docelowego (od A4 do A0, lub A do E), to skala zostanie automatycznie dobrana, tak aby rysunek wypełnił w pełni stronę plotera.

### 11.7.2 Ustawienie przesunięcia strony

Dla wszystkich standardowych rozmiarów, można przenieść punkt zerowy by wyrównać rysunek na środku strony. Niektóre plotery mogą posiadać punkt początkowy w centrum pola roboczego lub w dolnym lewym rogu, dlatego wymagane jest poprawne ustawienie tej opcji, zgodnie z możliwościami plotera.

Mówiąc ogólnie:


- Dla ploterów posiadających punkt początkowy na środku arkusza, przesunięcie musi być ujemne i ustawione w połowie rozmiaru arkusza.
- Dla ploterów posiadających punkt początkowy w lewym dolnym narożniku, przesunięcie musi być ustawione na 0.

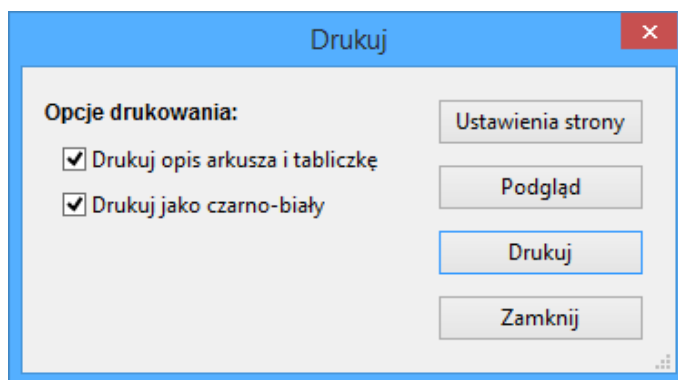
By ustawić przesunięcie należy:

- Wybrać rozmiar arkusza.
- Ustawić przesunięcie X oraz Y.
- Zaakceptować dane o przesunięciu.

## 11.8 Drukuj



To polecenie, podobne do polecenia  na głównym pasku narzędzi, pozwala na podgląd oraz utworzenie wydruków korzystając z normalnej drukarki.



Pierwsza opcja "Drukuj opis arkusza i tabliczkę" pozwala na wydrukowanie także odnośników arkuszy oraz tabliczki która znajduje się w prawym dolnym rogu.

Opcja "Drukuj jako czarno-biały" wymusza zaś wydruk monochromatyczny. Opcja ta zwykle jest stosowana, gdy do wydruków jest używana laserowa drukarka monochromatyczna, ponieważ większość drukarek dla jasnych kolorów korzysta z dość nieczytelnej symulacji pół-tonalnej. Stąd też połączenia, rysowane kolorem zielonym, mogłyby stać się mało widoczne.



## Rozdział 12

# Symbol Library Editor

### 12.1 General Information About Symbol Libraries

A symbol is a schematic element which contains a graphical representation, electrical connections, and fields defining the symbol. Symbols used in a schematic are stored in symbol libraries. Eeschema provides a symbol library 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 library editing tool provides a simple way to manage symbol library files.

### 12.2 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:

- Jego postaci graficznej (linie, okręgi, pola tekstowe) które pozwalają na zdefiniowanie symbolu.
- Wyprowadzeń, inaczej pinów (które muszą być narysowane w ściśle określony sposób używając powszechnie przyjętych standardów (zwykły pin, lub wejście zegarowe, lub aktywny niskim poziomem, albo aktywny zboczem.) opisując ich właściwości elektryczne, używane przez ERC.
- Pól (tekstowych) takich jak oznaczenie, wartość, nazwa footprintu potrzebna do wstawienia go na płytkę.
- Aliases used to associate a common symbol such as a 7400 with all of its derivatives such as 74LS00, 74HC00, and 7437. All of these aliases share the same library symbol.

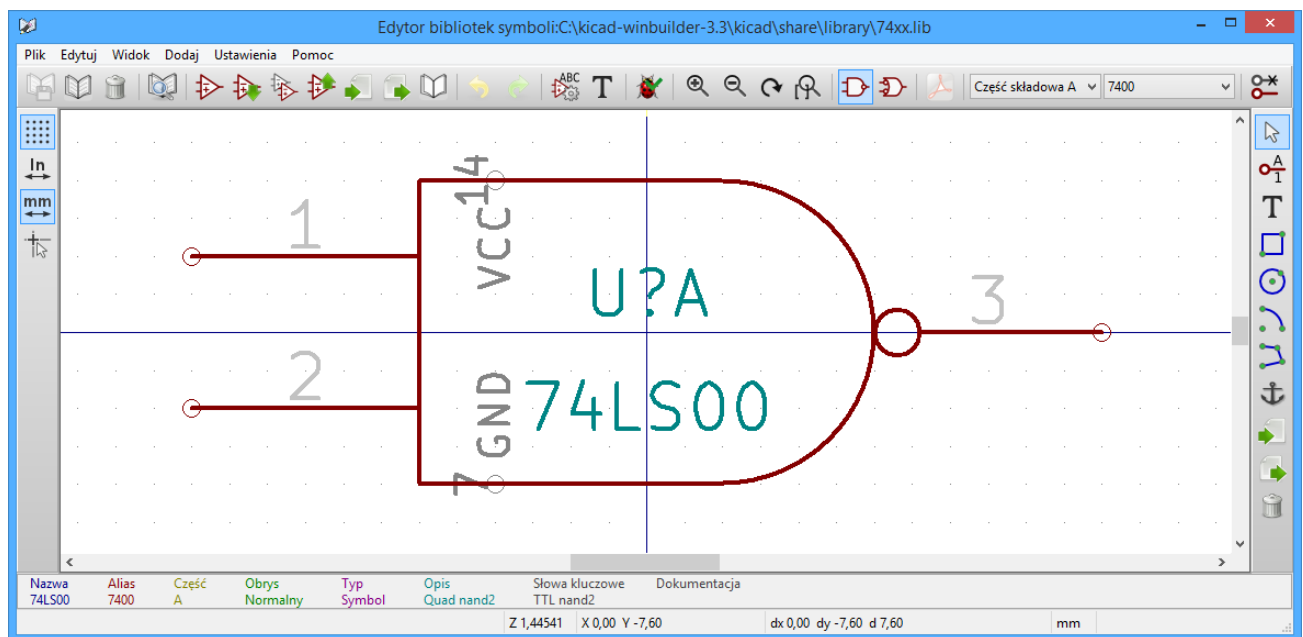
Proper symbol designing requires:

- 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.
  - Projektowanie wyglądu (z wyjątkiem pinów) z użyciem linii, prostokątów, okręgów, wielokątów i tekstów.
-

- Dodanie wyprowadzeń, dokładnie określając ich projekt graficzny, nazwę oraz numer pinu, a także ich właściwości elektryczne (wejście, wyjście, trzy-stanowe, port zasilania...).
- Adding an alias if other symbols have the same design and pin out or removing one if the symbol has been created from another symbol.
- Dodanie pól w razie potrzeby (jest to opcjonalne, nazwa modułu jest wykorzystywany przez oprogramowanie PCB i/lub określenie ich widoczności).
- Documenting the symbol by adding a description string and links to data sheets, etc.
- Zapisanie go do wybranej biblioteki.

## 12.3 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.










### 12.3.1 Główny pasek menu

The main tool bar typically located at the top of the main window shown below consists of the library management tools, undo/redo commands, zoom commands, and symbol properties dialogs.














	Save the currently selected library. The button will be disabled if no library is currently selected or no changes to the currently selected library have been made.
	Select the library to edit.
	Delete a symbol from the currently selected library or any library defined by the project if no library is currently selected.
	Open the symbol library browser to select the library and symbol to edit.
	Create a new symbol.
	Load symbol from currently selected library for editing.
	Create a new symbol from the currently loaded symbol.
	Save the current symbol changes in memory. The library file is not changed.
	Import one symbol from a file.
	Export the current symbol to a file.
	Create a new library file containing the current symbol. Note: new libraries are not automatically added to the project.
	Undo last edit.
	Redo last undo.
	Edit the current symbol properties.
	Edit the fields of current symbol.
	Test the current symbol for design errors.
	Zoom in.
	Zoom out.
	Refresh display.
	Zoom to fit symbol in display.

	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.
	Show the associated documentation. The button will be disabled if no documentation is defined for the current symbol.
	Select the unit to display. The drop down control will be disabled if the current symbol is not derived from multiple units.
	Select the alias. The drop down control will be disabled if the current symbol does not have any aliases.
	Pin editing: independent editing for pin shape and position for symbols with multiple units and alternate symbols.
	Show pin table.

### 12.3.2 Pasek narzędzi edycji symbolu





The vertical toolbar typically located on the right hand side of the main window allows you to place all of the elements required to design a symbol. The table below defines each toolbar button.

	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 center. Left-click again to define the first arc end point. Left-click again to define the second arc end point.


	Polygon tool. Left-click to begin drawing a new graphical polygon item in the current symbol. Left-click for each addition polygon line. Double-left-click to complete the polygon.
	Anchor tool. Left-click to set the anchor position of the symbol.
	Import a symbol from a file.
	Export the current symbol to a file.
	Delete tool. Left-click to delete an object from the current symbol.

### 12.3.3 Pasek opcji

Pasek narzędzi (zazwyczaj pionowy) znajdujący się po lewej stronie okna głównego pozwala na ustalenie pewnych opcji edycyjnych. Poniższa tabela opisuje każdy z przycisków na tym pasku narzędzi.



	Toggle grid visibility on and off.
	Set units to inches.
	Set units to millimeters.
	Toggle full screen cursor on and off.

## 12.4 Wybór biblioteki

The selection of the current library is possible via the  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 symbol is the contents of its value field.

---

#### Notatka


- Należy wcześniej określić nazwy bibliotek dostępnych dla Eeschema, by mieć do nich dostęp.
  - Zawartość bieżącej biblioteki może zostać zapisana po modyfikacji, klikając na  z głównego paska narzędzi.
  - A symbol can be removed from any library by clicking on the .
-

## 12.4.1 Select and Save a Symbol

When you edit a symbol you are not really working on the symbol in its library but on a copy of it in the computer's memory. Any edit action can be undone easily. A symbol may be loaded from a local library or from an existing symbol.

### 12.4.1.1 Symbol Selection

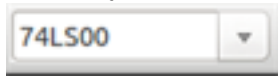


Clicking the  on the main tool bar displays the list of the available symbols that you can select and load from the currently selected library.

---

#### Notatka



If a symbol is selected by its alias, the name of the loaded symbol is displayed on the window title bar instead of the selected alias. The list of symbol aliases is always loaded with each symbol and can be edited. You can create a new

symbol by selecting an alias of the current symbol from the . The first item in the alias list is the root name of the symbol.

---

#### Notatka




Alternatively, clicking the  allows you to load a symbol which has been previously saved by the .


### 12.4.1.2 Save a Symbol

After modification, a symbol can be saved in the current library, in a new library, or exported to a backup file.




To save the modified symbol in the current library, click the . Please note that the update command only saves the symbol changes in the local memory. This way, you can make up your mind before you save the library.



To permanently save the symbol changes to the library file, click the  which will overwrite the existing library file with the symbol changes.



If you want to create a new library containing the current symbol, click the . You will be asked to enter a new library name.


---

#### Notatka

Nowe elementy w nowych bibliotekach będą widoczne dopiero po zmianie konfiguracji bibliotek w programie Eeschema. You must add any new library you wish to use in a schematic to the list of project libraries in Eeschema using the [Symbol Library Table dialog](#).






---



Click the  to create a file containing only the current symbol. This file will be a standard library file which will contain only one symbol. This file can be used to import the symbol into another library. In fact, the create new library command and the export command are basically identical.

### 12.4.1.3 Transfer Symbols to Another Library

You can very easily copy a symbol from a source library into a destination library using the following commands:

- Wybór biblioteki źródłowej poprzez kliknięcie na ikonę .
- Load the symbol to be transferred by clicking the . The symbol will be displayed in the editing area.
- Następnie wybrać docelową bibliotekę klikając w .
- Save the current symbol to the new library in the local memory by clicking the .
- Save the symbol in the current local library file by clicking the .


### 12.4.1.4 Discarding Symbol Changes

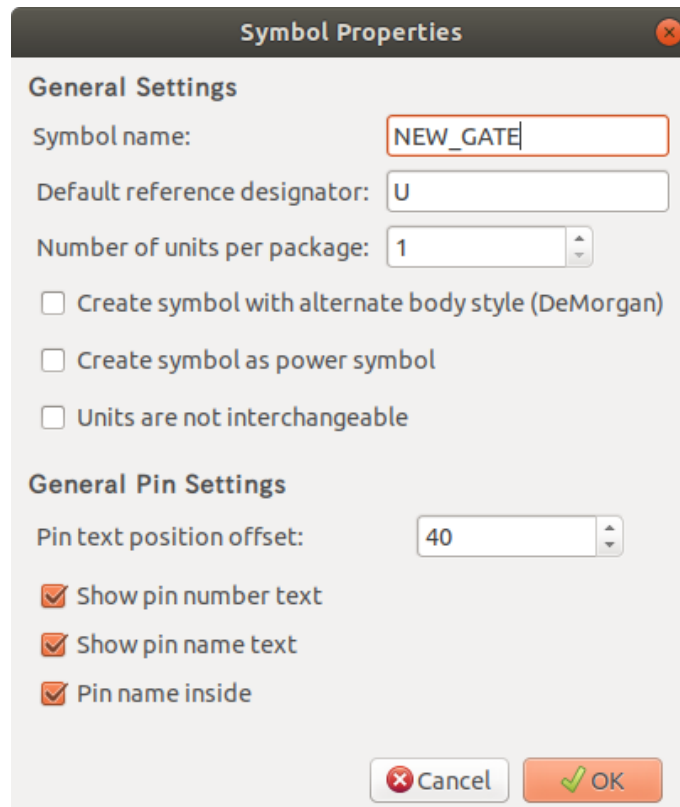
When you are working on a symbol, the edited symbol is only a working copy of the actual symbol in its library. This means that as long as you have not saved it, you can just reload it to discard all changes made. If you have already updated it in the local memory and you have not saved it to the library file, you can always quit and start again. Eeschema will undo all the changes.

## 12.5 Creating Library Symbols

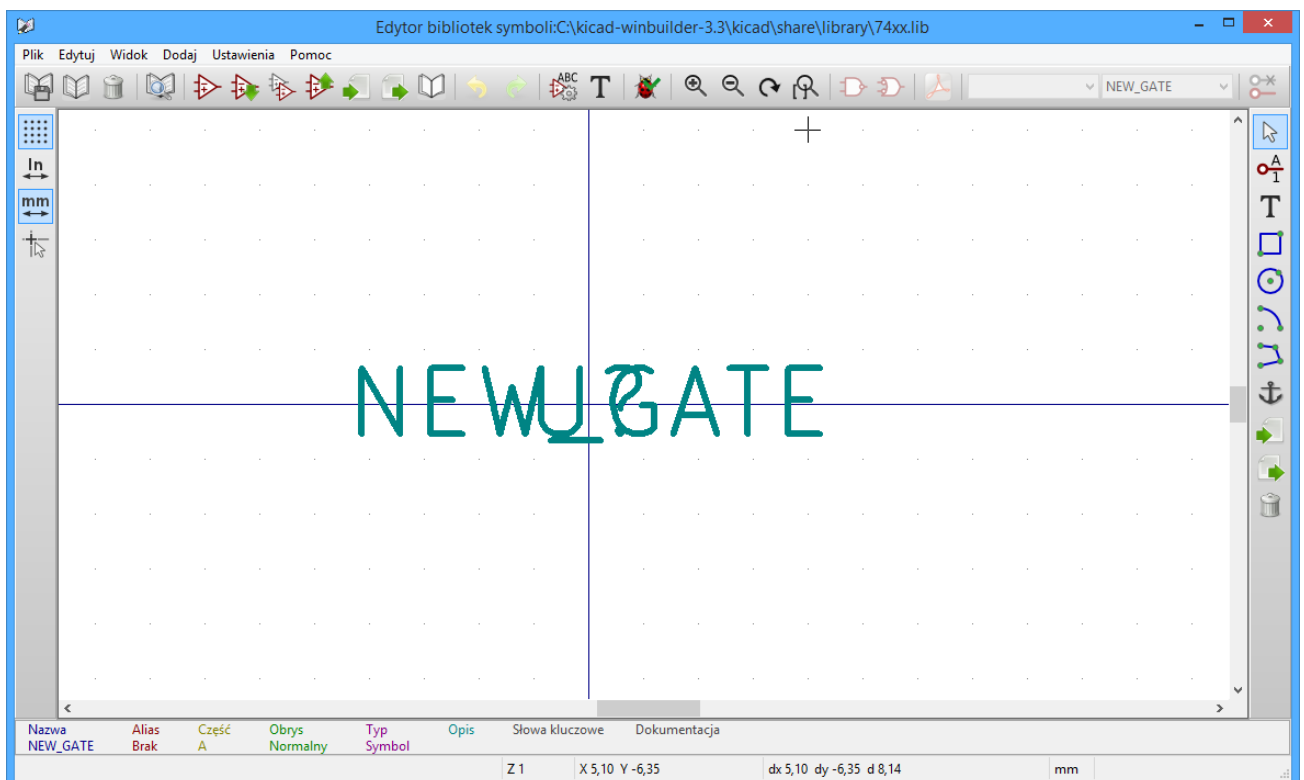
### 12.5.1 Create a New Symbol



A new symbol can be created by clicking the . You will be asked for a symbol name (this name is used as default value for the value field in the schematic editor), the reference designator (U, IC, R...), the number of units per package (for example a 7400 is made of 4 units per package) and if an alternate body style (sometimes referred to as DeMorgan) is desired. If the reference designator field is left empty, it will default to "U". These properties can be changed later, but it is preferable to set them correctly at the creation of the symbol.



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










### 12.5.2 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.


- Click on the  or modify its name by right-clicking on the value field and editing the text. If you chose to duplicate the current symbol, you will be prompted for a new symbol name.
- If the model symbol has aliases, you will be prompted to remove aliases from the new symbol which conflict with the current library. If the answer is no the new symbol creation will be aborted. Symbol libraries cannot have any duplicate names or aliases.
- Edit the new symbol as required.

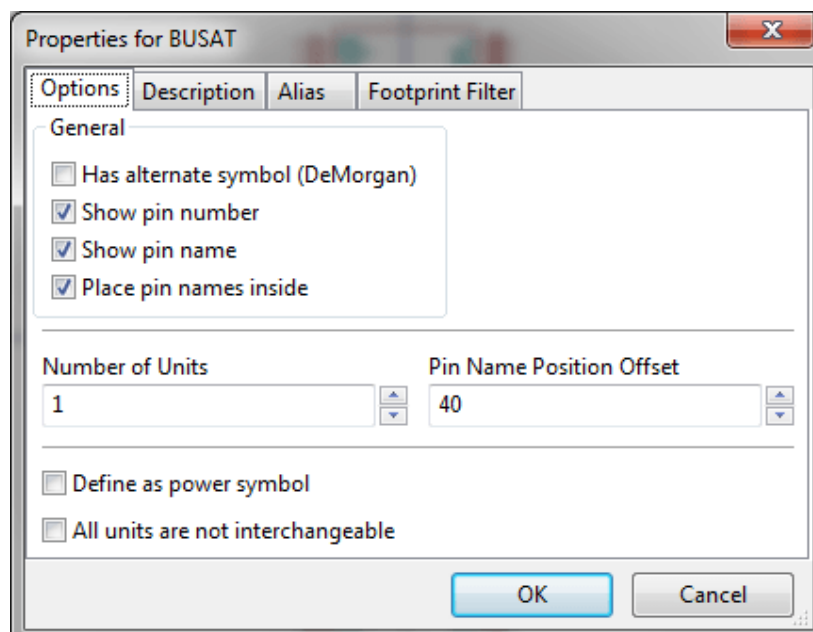
- Update the new symbol in the current library by clicking the  or save to a new library by clicking the   
or if you want to save this new symbol in an other existing library select the other library by clicking on the  and save the new symbol.

- Zapisać bieżącą bibliotekę na dysku klikając w ikonę .

### 12.5.3 Symbol Properties

Symbol properties should be carefully set during the symbol creation or alternatively they are inherited from the

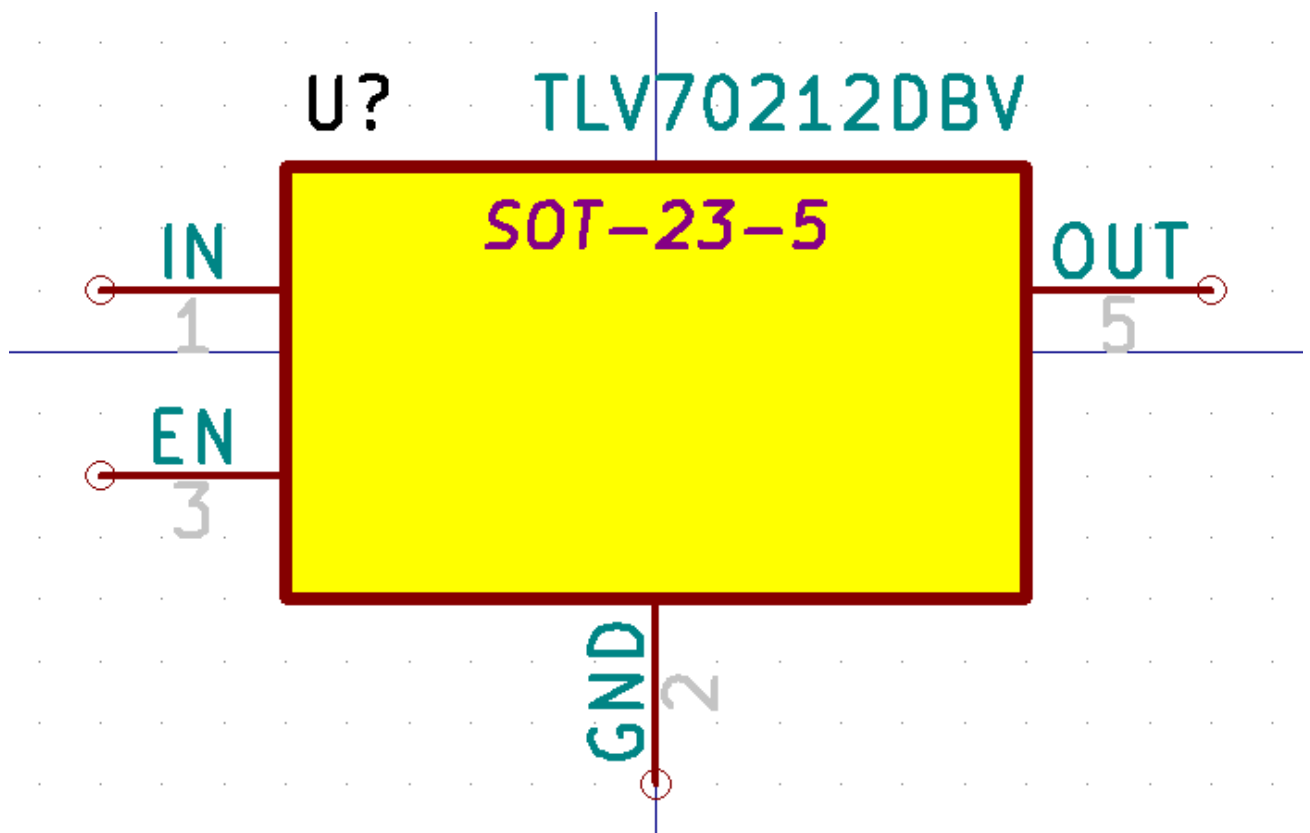
copied symbol. To change the symbol properties, click on the  to show the dialog below.



It is very 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 add the new unit pins and symbols. 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. This text will be visible if the corresponding options are checked. The option "Place pin names inside" defines the pin name position relative to the pin body. This text 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 30 to 40 (in 1/1000 inch) is reasonable.



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

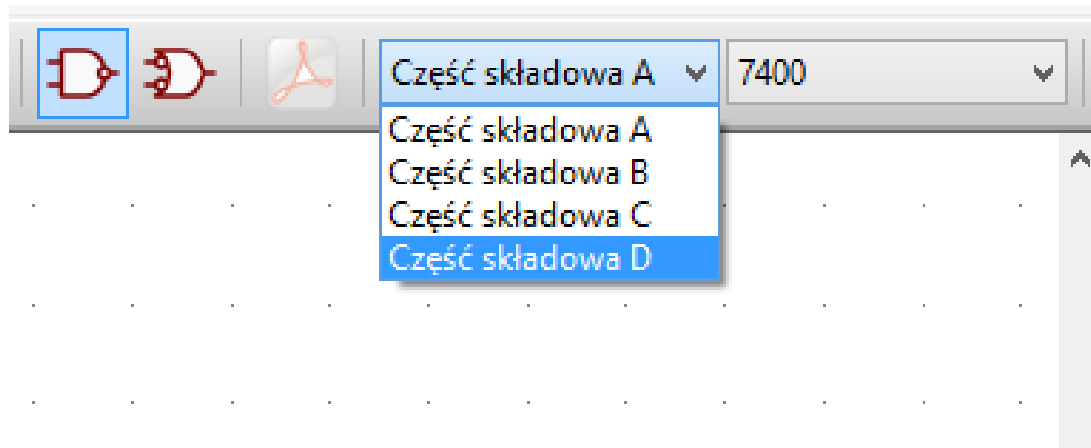


#### 12.5.4 Symbols with Alternate Symbolic Representation

If the symbol has more than one symbolic representation, you will have to select one representation to edit them. To

edit the normal representation, click the .

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



## 12.6 Elementy graficzne symbolu

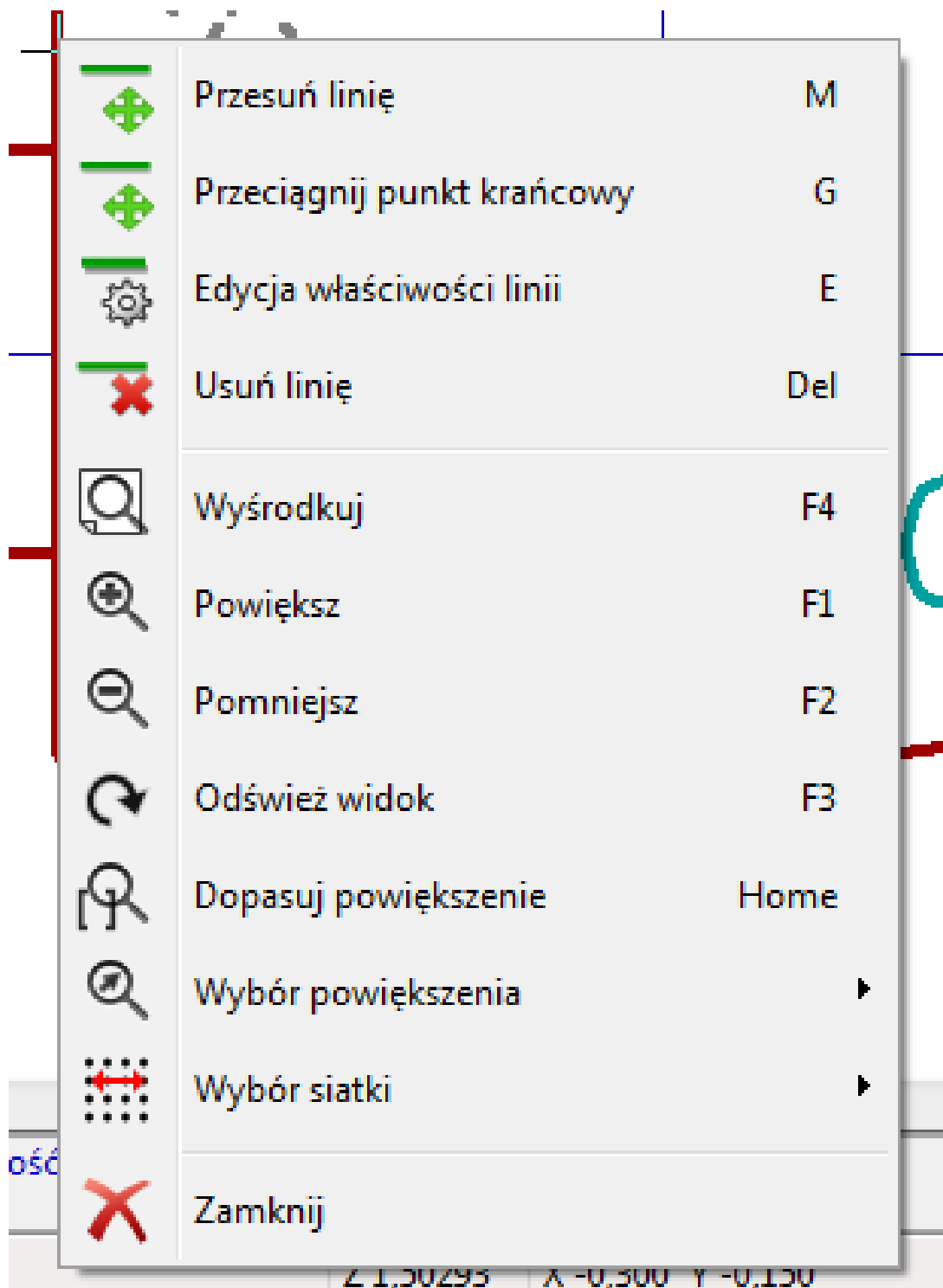
Graphical elements create the representation of a symbol and contain no electrical connection information. Their design is possible using the following tools:

- Linie i linie łamane są definiowane poprzez punkty startowe i końcowe.
- Prostokąty są definiowane przez punkty dwóch przeciwległych narożników.
- Okręgi są definiowane przez punkt centralny i promień.
- Łuki są definiowane przez punkt początkowy i końcowy łuku oraz ich punkt centralny. Kąt rozwarcia łuku może zawierać się w przedziale  $0^\circ$  to  $180^\circ$ .

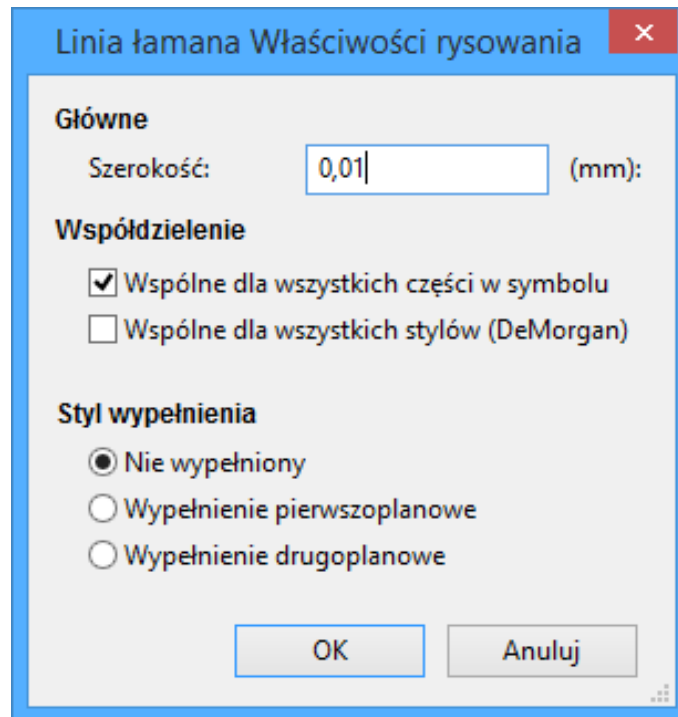
The vertical toolbar on the right hand side of the main window allows you to place all of the graphical elements required to design the representation of a symbol.

### 12.6.1 Przynależność elementów graficznych

Każdy z elementów graficznych (linia, łuk, okrąg, itd.) może być określona jako część wspólna dla wszystkich części składowych lub stylów, albo specyficzna dla nich. Opcje dotyczące przynależności elementu graficznego można łatwo wyświetlić klikając prawym klawiszem myszy nad wybranym elementem wywołując menu podręczne. Poniżej przykład menu dla elementu typu linia.



Można także kliknąć dwukrotnie na taki element by zmodyfikować jego właściwości. Poniżej przykład dla elementu typu linia łamana.



Głównymi właściwościami dla elementów graficznych są:

- Szerokość linii, która określa szerokość linii symbolu w obecnie wybranych jednostkach miary.
- The Common to all units in symbol setting defines 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.
- The Common by all body styles (DeMorgan) setting defines 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.
- Styl wypełnienia określa czy graficzna postać symbolu ma być rysowana jako niewypełniona, wypełniona kolorem tła lub wypełniona kolorem.

### 12.6.2 Tekst jako grafika w symbolu

**T**he **T** allows for the creation of graphical text. Graphical text is always readable, even when the symbol is mirrored. Please note that graphical text items are not fields.

## 12.7 Multiple Units per Symbol and Alternate Body Styles

Symbols can have two symbolic representations (a standard symbol and an alternate symbol often referred to as "DeMorgan") 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.

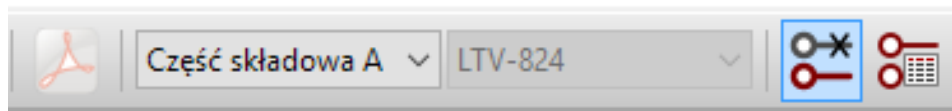
Consider for instance a relay with two switches which can be designed as a symbol with three different units: a coil, switch 1, and switch 2. Designing a symbol with multiple units per package and/or alternate body styles is very flexible. A pin or a body symbol item can be common to all units or specific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

Domyślnie, piny są specyficzne dla każdej reprezentacji każdej części, ponieważ ich liczba różni się dla każdej części, a ich konstrukcja jest inna dla każdej reprezentacji. Gdy pin jest wspólny wystarczy wyprowadzić go tylko raz (np. w przypadku pinów zasilania). Również w przypadku projektu, który jest prawie zawsze identyczny dla każdej części (ale różni się pomiędzy normalną a skonwertowaną reprezentacją).

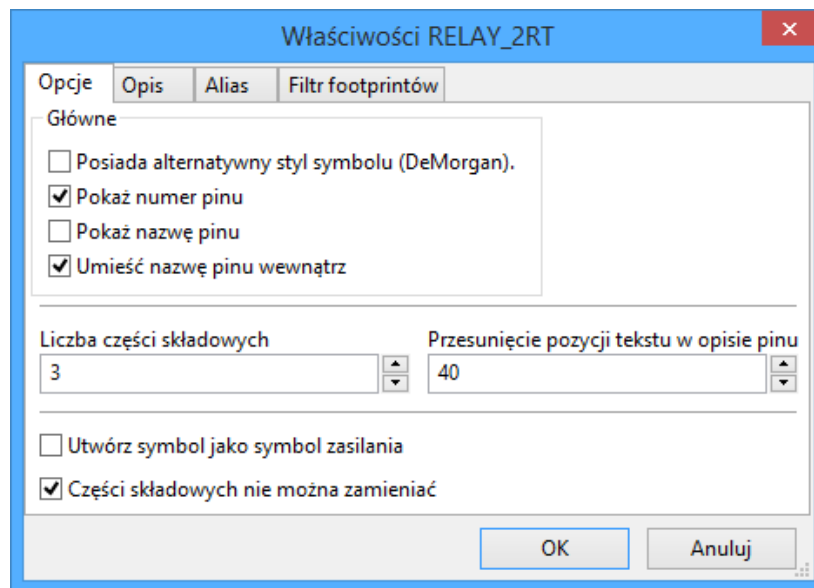
### 12.7.1 Example of a Symbol Having Multiple Units with Different Symbols:

Jest to przypadek przekaźnika mechanicznego, który posiada dwa zestawy styków oraz cewkę (trzy różne części):

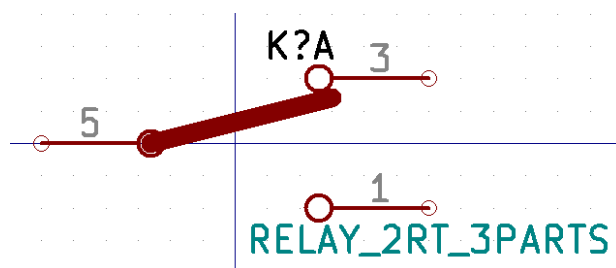
Opcja: piny nie są powiązane. Można dodawać lub edytować piny niezależnie dla każdej z części.



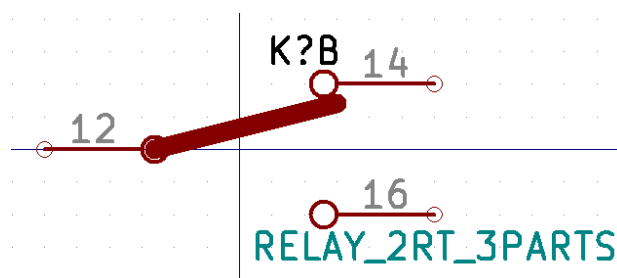
Musi być wybrana opcja informująca, że części składowe nie są wymienne między sobą.



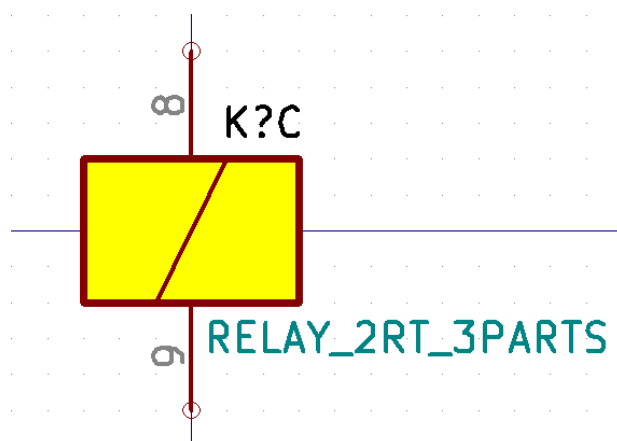
Część 1



## Część 2



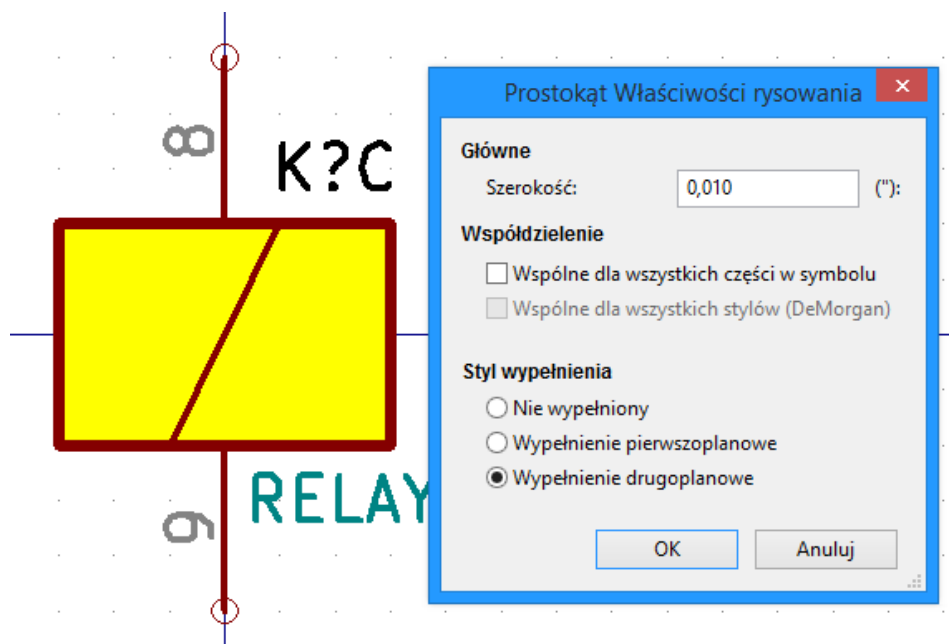
## Część 3



Ponieważ symbol nie posiada tej samej postaci graficznej dla wszystkich części, nie ma możliwości zamiany pomiędzy częściami A i C.

### 12.7.1.1 Elementy geometryczne w symbolach

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



## 12.8 Tworzenie i edycja wyprowadzeń (pinów)



Aby utworzyć i wstawić pin należy kliknąć w polecenie . Aby dokonać prostych edycji wyprowadzeń najlepiej jest kliknąć dwukrotnie na wybranym wyprowadzeniu, lub kliknąć prawym klawiszem by skorzystać z menu podręcznego. Piny muszą być tworzone starannie, ponieważ każdy błąd będzie mieć wpływ na projekt PCB. Każdy dodany już pin można ponownie edytować, usunąć, obrócić lub przenieść.

### 12.8.1 Wyprowadzenia - Informacje podstawowe

Pin jest zdefiniowany przez jego graficzną reprezentację, jego nazwę oraz numer”. Numer pinu jest określony przez 4 znaki i/lub cyfry. Dla narzędzia sprawdzania elektrycznych reguł projektowych (ERC) by było skuteczne, typ elektryczny”(wejście, wyjście, trójstanowy...) także musi zostać poprawnie określone. Gdyby typ pinu nie został określony poprawnie, wynik testu ERC byłby w gruncie rzeczy nieprawidłowy.

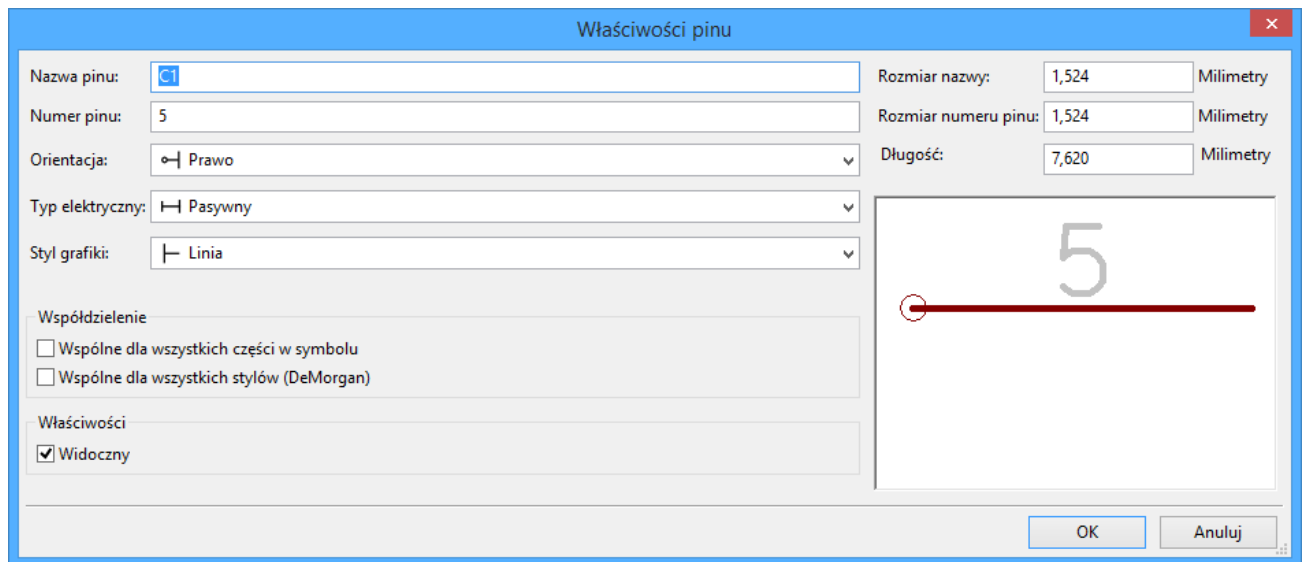
Ważne uwagi:

- Nie należy wstawiać znaków spacji w nazwach pinów i ich numeracji. Wstawienie spacji w nazwie spowoduje, że zostanie ona automatycznie zastąpiona przez znak dolnej kreski ”\_”.
- Nazwa pinu z sygnałem zanegowanym rozpoczyna się od znaku tyldy ~. Wtedy nad tekstem zostanie dorysowana pozioma kreska. Znak ten działa jak przełącznik, zatem możliwe jest również stosowanie go tylko dla wybranego fragmentu nazwy, np. PA0/~INT0~/PCI0, co w rezultacie da PA0/INT0/PCI0.
- Jeśli nie ma nazwy pinu to w pole nazwa należy wstawić jeden znak tyldy.
- Nazwy pinów rozpoczynające się od znaku *hash* #, są zarezerwowane dla symboli zasilania.



- Numer pinu może składać się z maksymalnie czterech liter lub liczb (również mieszanie). 1, 2, ...9999 to prawidłowe nazwy, ale prawidłowymi nazwami będą też: A1, B3 (standardowa notacja w układach PGA/BGA) lub Anod, Gnd, Wire.
- Duplicate pin numbers cannot exist in a symbol.

### 12.8.2 Właściwości wyprowadzeń



Okno z właściwościami pinu pozwala na zmiany charakterystycznych cech wyprowadzeń. Okno to ukazuje się zawsze podczas tworzenia nowego pinu, albo gdy w pin zostanie kliknięty dwukrotnie myszą. Jego zawartość pozwala na zdefiniowanie lub modyfikację parametrów takich jak:

- Nazwa i rozmiar tekstu nazwy.
- Numer i rozmiar tekstu numeru.
- Długość.
- Typ graficzny i elektryczny wyprowadzenia.
- Przynależność do części i alternatywnej reprezentacji.
- Widoczność.

### 12.8.3 Style graficzne pinów

Na poniższym rysunku w jednym elemencie zastosowano wszystkie rodzaje kształtów wyprowadzeń. Wybór formy ma wyłącznie znaczenie graficzne i nie ma żadnego znaczenia dla narzędzia ERC jak i dla funkcji związanych z tworzeniem list sieci.



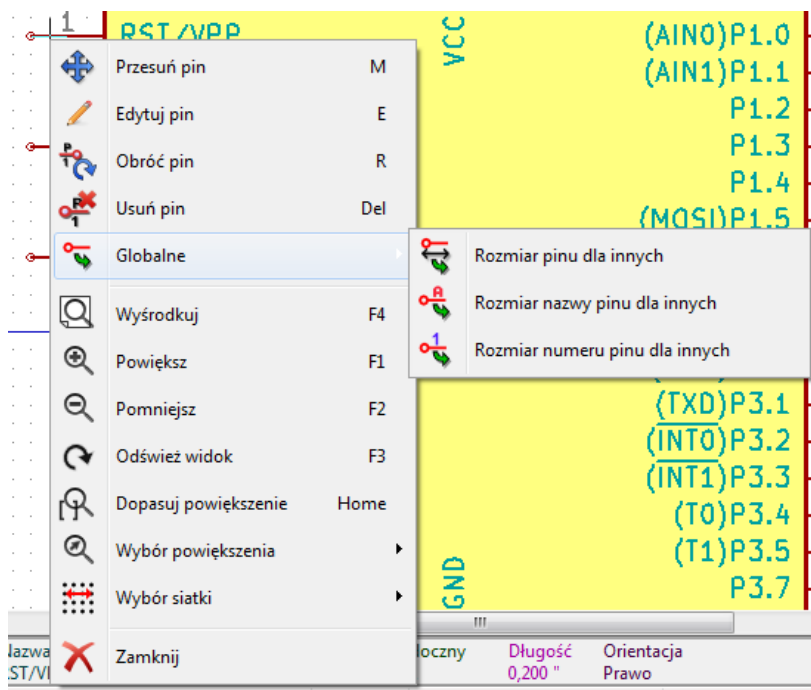
#### 12.8.4 Typy elektryczne

Wybór właściwego typu elektrycznego jest bardzo ważny dla narzędzia ERC. Zwykle typy elektryczne są definiowane jako:

- Dwukierunkowy, który oznacza, że piny mogą pracować jako wejścia lub wyjścia zależnie od konfiguracji (na przykład szyna danych mikroprocesora).
- Trójstanowy, posiadający zwykle trzy aktywne stany.
- Passive is used for passive symbol pins, resistors, connectors, etc.
- Nieokreślony, używany gdy sprawdzenie ERC nie ma znaczenia dla tego pinu.
- Power input is used for the symbol's power pins. Power pins are automatically connected to the other power input pins with the same name.
- Wyjścia zasilania są używane w przypadku regulatorów napięć.
- Otwarty emiter i otwarty kolektor, które można używać w przypadku wyjść logicznych lub w komparatorach dołączenia ich wyjść (tzw. suma na drucie).
- Not connected is used when a symbol has a pin that has no internal connection.


### 12.8.5 Wyprowadzenia - Zmiany globalne

You can modify the length or text size of the name and/or number of all the pins using the Global command entry of the pin context menu. Click on the parameter you want to modify and type the new value which will then be applied to all of the current symbol's pins.



### 12.8.6 Wyprowadzenia - Symbole wieloczęściowe i podwójna reprezentacja

Symbols with multiple units and/or graphical representations are particularly problematic when creating and editing pins. The majority of pins are specific to each unit (because their pin number is specific to each unit) and to each symbolic representation (because their form and position is specific to each symbolic representation). The creation and the editing of pins can be problematic for symbols with multiple units per package and alternate symbolic representations. 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 both representations for symbols with an alternate symbolic representation.


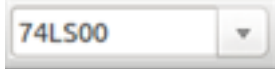
Jedynym wyjątkiem od tego to graficzny typ pinów i nazwa. Ta zależność została wprowadzona by umożliwić łatwiejsze tworzenie pinów i ich edycję w najczęstszych przypadkach. Zależność może być przełączona za pomocą  na głównym pasku narzędzi. Pozwala to na całkowicie niezależne tworzenie pinów dla każdej części i reprezentacji.

A symbol can have two symbolic representations (representation known as 'DeMorgan') and can be made up of more than one unit as in the case of symbols with logic gates. For certain symbols, you may want several different graphic elements and pins. Like the relay sample shown in the [previous section](#), a relay can be represented by three distinct units: a coil, switch contact 1, and switch contact 2.

The management of the symbols with multiple units and symbols with alternate symbolic representations is flexible. A pin can be common or specific to different units. A pin can also be common to both symbolic representations or specific to each symbolic representation.

Domyślnie piny są specyficzne dla każdej reprezentacji każdej części, ponieważ ich ilość może być różna dla każdej części, a ich konstrukcja jest inna dla każdej reprezentacji. Gdy pin jest wspólny dla wszystkich urządzeń, należy go narysować tylko raz jak w przypadku pinów zasilania.

An example is the output pin 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

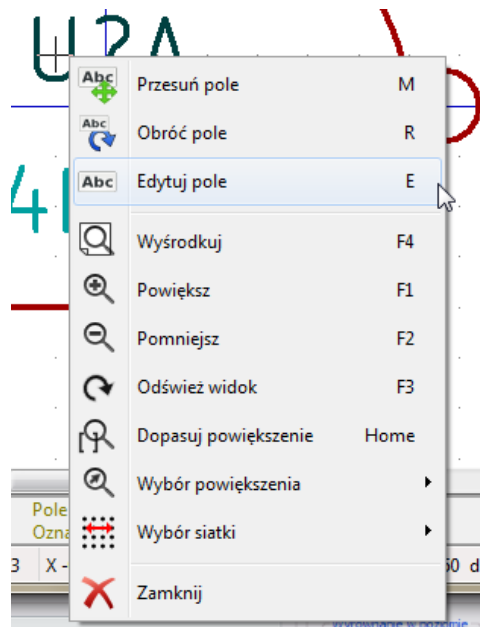
alternate symbolic representation, it must first be enabled by clicking the  button on the tool bar. To edit the pin number for each unit, select the appropriate unit using the  drop down control.

## 12.9 Symbol Fields

All library symbols are defined with four default fields. The reference designator, value, footprint assignment, and documentation file link fields are created whenever a symbol is created or copied. Only the reference designator and value fields are required. For existing fields, you can use the context menu commands by right-clicking on the pin. Symbols defined in libraries are typically defined with 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.

### 12.9.1 Editing Symbol Fields

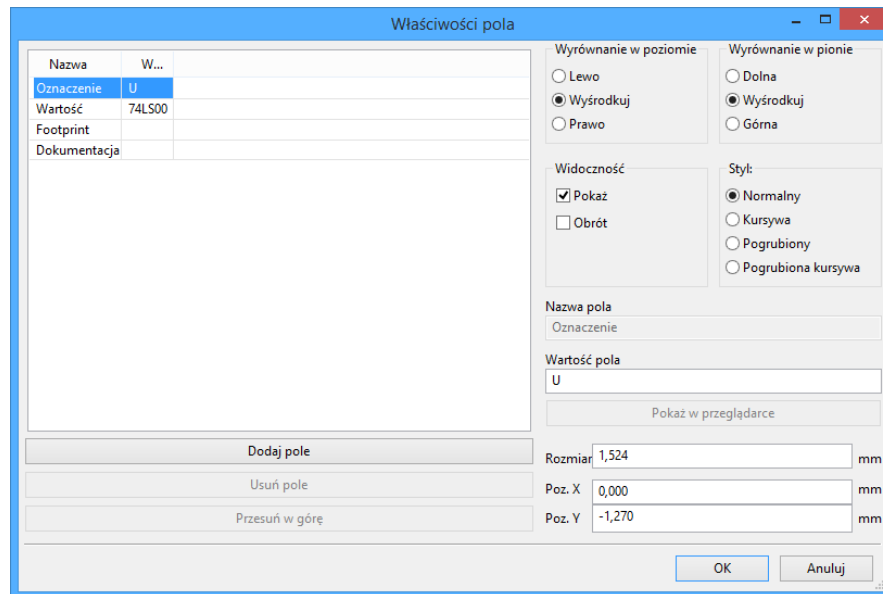
To edit an existing symbol field, right-click on the field text to show the field context menu shown below.



To edit undefined fields, add new fields, or delete optional fields



on the main tool bar to open the field properties dialog shown below.



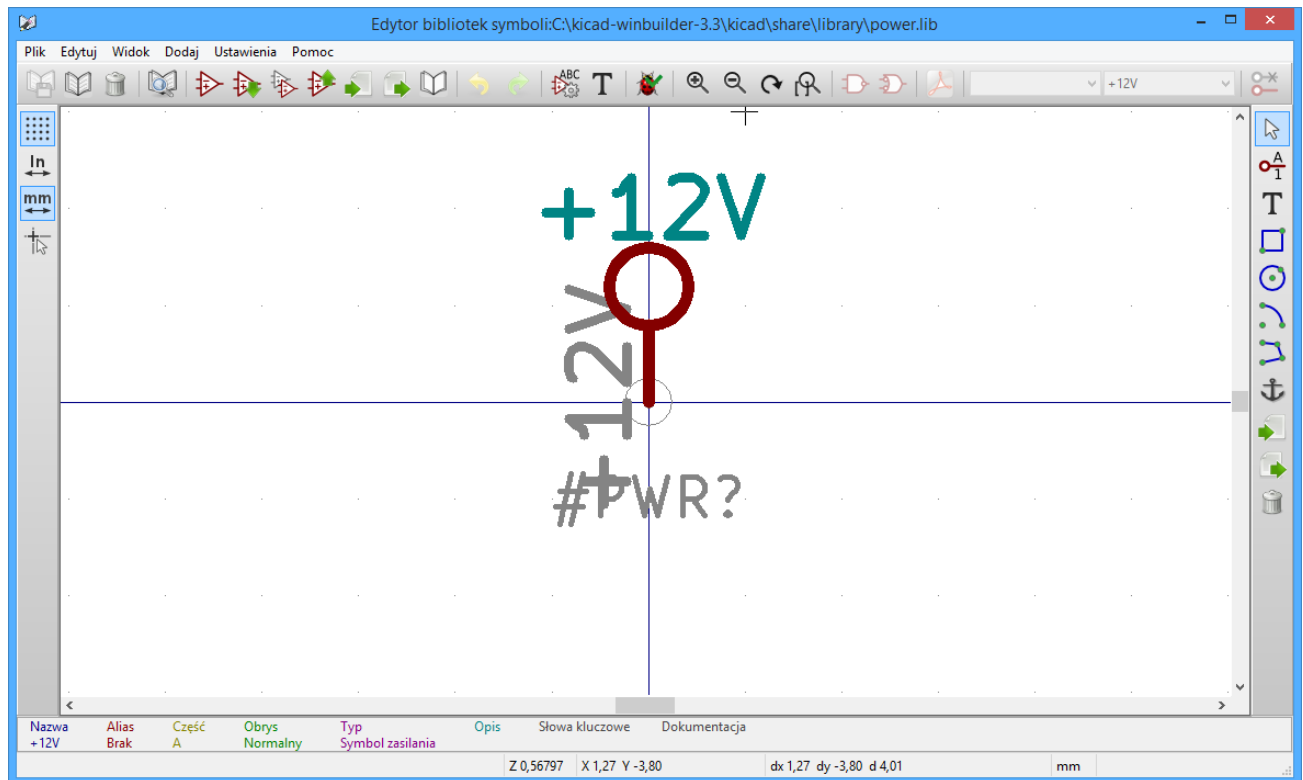
Fields are text sections associated with the symbol. Do not confuse them with the text belonging to the graphic representation of this symbol.

Ważne uwagi:

- Modifying value fields effectively creates a new symbol using the current symbol as the starting point for the new symbol. This new symbol has the name contained in the value field when you save it to the currently selected library.
- Aby edytować niewidoczne lub puste pola trzeba skorzystać z okna edycji przedstawionego powyżej.
- Pole Footprint zawiera bezwzględny odnośnik do footprintu w postaci LIBNAME:FPNAME, gdzie LIBNAME to nazwa biblioteki footprintów zdefiniowana w tabeli bibliotek (zobacz rozdział "Tabela Bibliotek Footprintów" w podręczniku Pcbnew) oraz FPNAME, która jest nazwą footprintu w bibliotece LIBNAME.

## 12.10 Tworzenie symboli zasilania

Power symbols are created the same way as normal symbols. It may be useful to place them in a dedicated library such as power.lib. Power symbols consist of a graphical symbol and a pin of the type "Power Invisible". Power port symbols are handled like any other symbol by the schematic capture software. Some precautions are essential. Below is an example of a power +5V symbol.



By utworzyć port zasilania, należy wykonać następujące kroki:

- Dodać pin typu "Wejście zasilania" nazwany +12V (Ważne, ponieważ połączenie pomiędzy poszczególnymi portami odbywa się po nazwie), z numerem pinu 1 (numer nie jest tak istotny), z długością 0 i stylem graficznym Linia".
- Umieścić na przykład mały okrąg i jeden segment od niego jak pokazano powyżej.
- Ustawić punkt zaczepienia w miejscu pinu.
- The symbol value is +5V.
- The symbol reference is \#+5V. The reference text is not important except the first character which must be # to indicate that the symbol is a power symbol. By convention, every symbol in which the reference field starts with a # will not appear in the symbol list or in the netlist and the reference is declared as invisible.

Łatwiejszą metodą tworzenia nowego portu zasilania jest użycie innego portu jako wzorca:

- Załadować istniejący symbol zasilania.
- Zmienić nazwę pinu razem z nazwą nowego symbolu zasilania.
- Dokonać edycji pola Wartość by jego zawartość była zgodna z nazwą pinu, jeśli chcemy by była ona wyświetlana.
- Save the new symbol.

## Rozdział 13

# LibEdit - Symbols

### 13.1 Przegląd zagadnień

A symbol consist of the following elements

- Jego postaci graficznej (kształty geometryczne, teksty).
- Wyprowadzenia (piny).
- Fields or associated text used by the post processors: netlist, symbols list.

Two fields are to be initialized: reference and value. The name of the design associated with the symbol, and the name of the associated footprint, the other fields are the free fields, they can generally remain empty, and could be filled during schematic capture.

However, managing the documentation associated with any symbol facilitates the research, use and maintenance of libraries. The associated documentation consists of

- Linie komentarza.
- Linie ze słowami kluczowymi takimi jak np.: TTL CMOS NAND2, oddzielonymi spacjami
- Linie z nazwą pliku zewnętrznej dokumentacji (np. notę aplikacyjną, notę katalogową.).

Domyślnie pliki te są wyszukiwane w następujących katalogach:

kicad/share/library/doc

Jeśli nie znaleziono:

kicad/library/doc

W systemie Linux:

/usr/local/kicad/share/library/doc

/usr/share/kicad/library/doc

/usr/local/share/kicad/library/doc

Key words allow you to selectively search for a symbol according to various selection criteria. Comments and key words are displayed in various menus, and particularly when you select a symbol from the library.

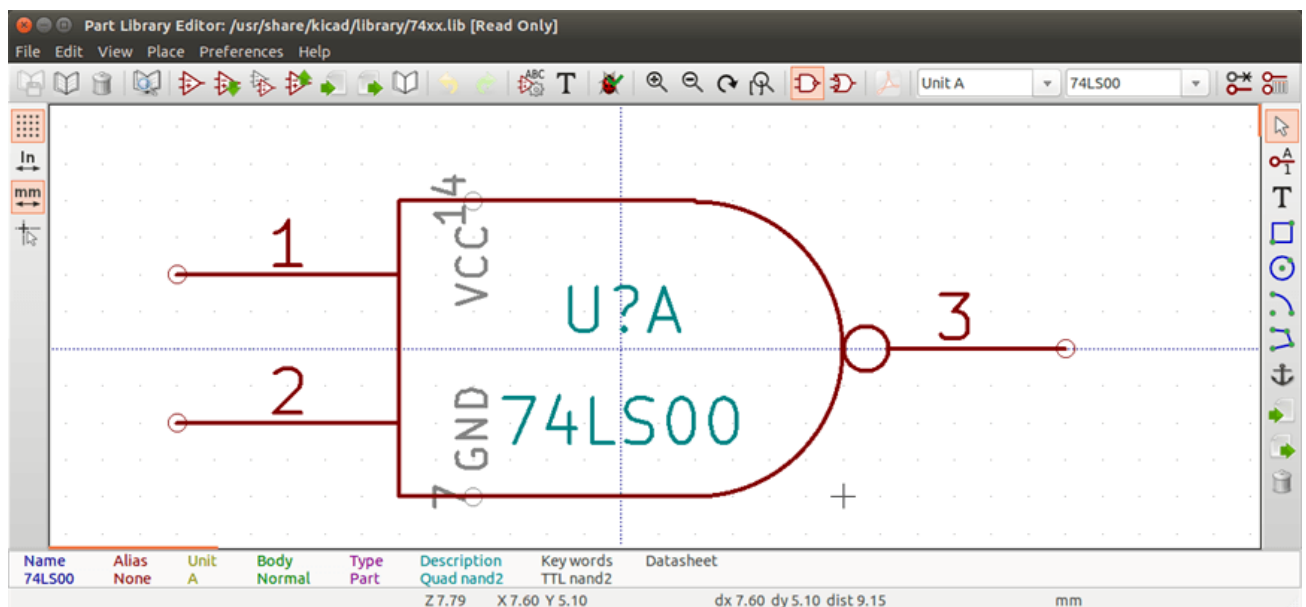
The symbol also has an anchoring point. A rotation or a mirror is made relative to this anchor point and during a placement this point is used as a reference position. It is thus useful to position this anchor accurately.


A symbol can have aliases, i.e. equivalent names. This allows you to considerably reduce the number of symbols that need to be created (for example, a 74LS00 can have aliases such as 74000, 74HC00, 74HCT00...).

Finally, the symbols are distributed in libraries (classified by topics, or manufacturer) in order to facilitate their management.

## 13.2 Position a symbol anchor

Domyślnie punkt zaczepienia znajduje się na pozycji (0, 0) i jest pokazywany jako dwie krzyżujące się osie.

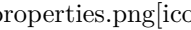


Punkt zaczepienia może być przeniesiony wybierając polecenie  i klikając w jego nowym punkcie położenia. Rysunek zostanie automatycznie wycelowany wobec nowego punktu zaczepienia.

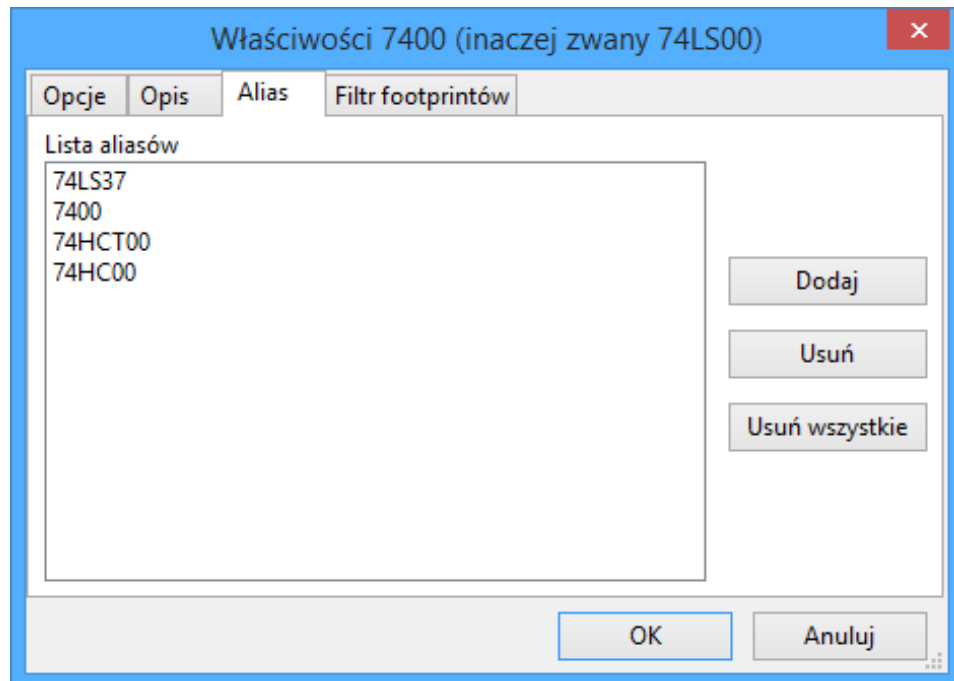
## 13.3 Symbol aliases

An alias is another name corresponding to the same symbol in the library. Symbols with similar pin-out and representation can then be represented by only one symbol, having several aliases (e.g. 7400 with alias 74LS00, 74HC00, 74LS37).

Korzystanie z aliasów pozwala na tworzenie kompletnych bibliotek symboli znacznie szybciej. Oprócz tego biblioteki takie są znacznie bardziej zwarte i są ładowane szybciej.

Aby zmodyfikować listę aliasów, należy wybrać w głównym oknie edycji narzędzie  i wybrać zakładkę *Alias*:





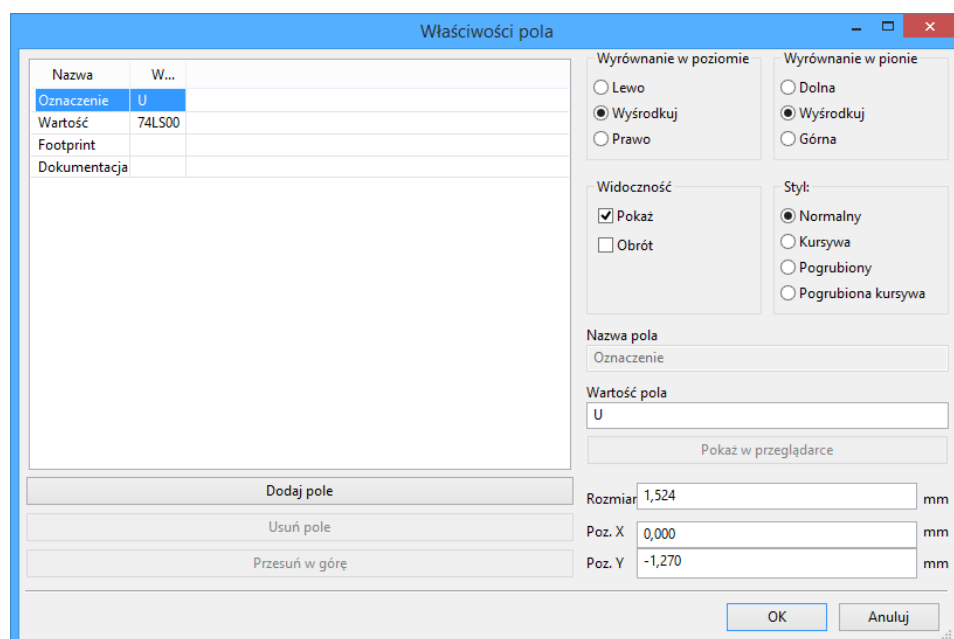
Za pomocą klawiszy obok listy można dodawać lub usuwać aliasy. Bieżący alias nie może być oczywiście zmieniony ponieważ jest edytowany.

To remove all aliases, you have firstly to select the root symbol. The first symbol in the alias list in the window of selection of the main toolbar.

## 13.4 Symbol fields

The field editor is called via the icon .


There are four special fields (texts attached to the symbol), and configurable user fields

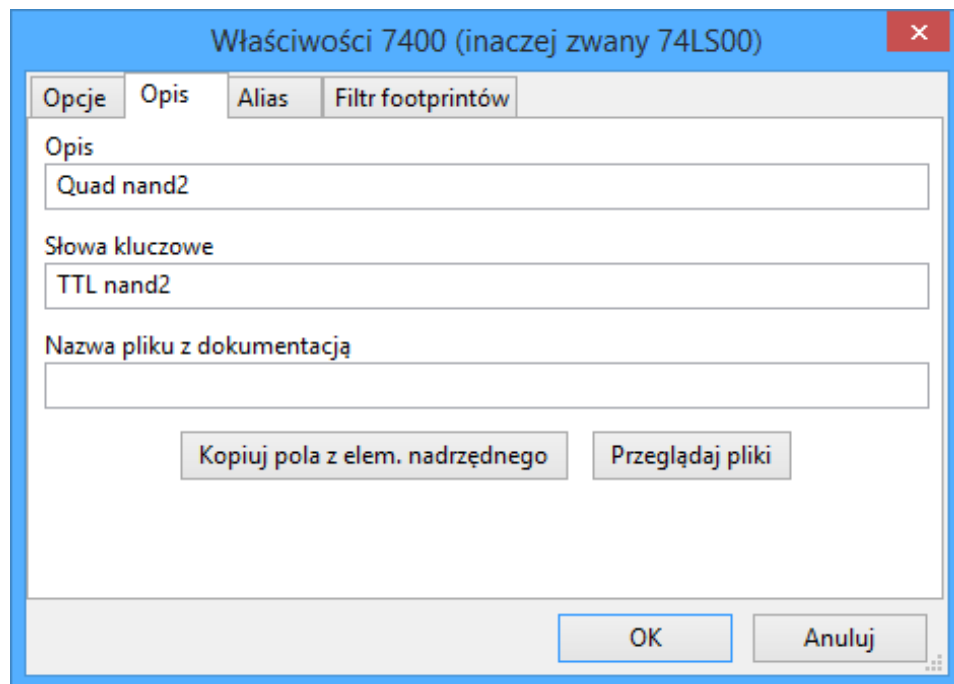


Pola specjalne to:

- Odnosićnik.
- Value. It is the symbol name in the library and the default value field in schematic.
- Footprint: nazwa modułu używanego na PCB. Nie jest zbyt potrzebna jeśli używamy CvPcb do ustalania listy footprintów.
- Dokumentacja: zarezerwowane (tutaj nie używane).

## 13.5 Symbol documentation

To edit documentation information, it is necessary to call the main editing window of the symbol via the icon  and to select the document folder.



Be sure to select the right alias, or the root symbol, because this documentation is the only characteristic which differs between aliases. The "Copy Doc" button allows you to copy the documentation information from the root symbol towards the currently edited alias.

### 13.5.1 Symbol keywords

Keywords allow you to search in a selective way for a symbol according to specific selection criteria (function, technological family, etc.)

Wielkość liter nie jest istotna, bo narzędzie zawarte w Eeschema nie rozróżnia wielkość liter. Najbardziej popularne słowa kluczowe użyte w bibliotekach to:

- CMOS TTL dla rodzin układów logicznych
- AND2 NOR3 XOR2 INV ...dla bramek (AND2 = 2 wejściowa bramka AND, NOR3 = 3 wejściowa bramka NOR).
- JKFF, DFF ...dla przerzutników typu JK lub D
- ADC, DAC, MUX ...
- OpenCol for the gates with open collector output. Thus if in the schematic capture software, you search the symbol: by keywords NAND2 OpenCol Eeschema will display the list of symbols having these 2 key words.

### 13.5.2 Symbol documentation (Doc)

The line of comment (and keywords) is displayed in various menus, particularly when you select a symbol in the displayed symbols list of a library and in the ViewLib menu.

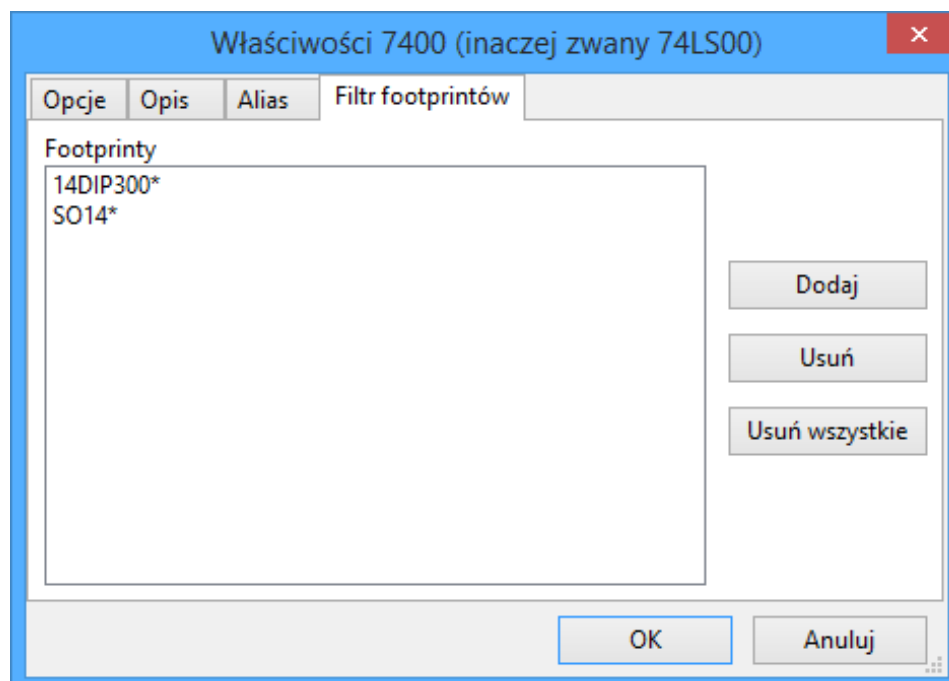
If this Doc. file exists, it is also accessible in the schematic capture software, in the pop-up menu displayed by right-clicking on the symbol.

### 13.5.3 Dołączony plik dokumentacji (\_Nazwa pliku z dokumentacją\_)

Wskazuje na dostępny plik z dokumentacją symbolu lub jego schematem aplikacyjnym. Może to być plik w formacie PDF (zwykle stosowany w takim przypadku) dostępny lokalnie na dysku twardym, ale można stosować również ścieżki URL by umożliwić dostęp do zasobów zdanych (np. na stronie WWW producenta).

### 13.5.4 Filtrowanie footprintów dla CvPcb

You can enter a list of allowed footprints for the symbol. This list acts as a filter used by CvPcb to display only the allowed footprints. A void list does not filter anything.



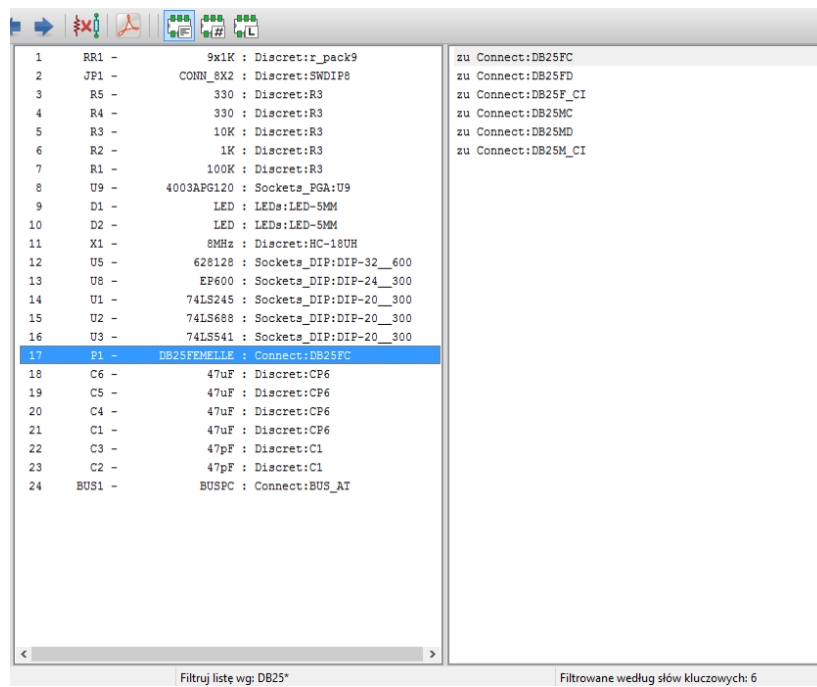
Na liście można stosować szeroko stosowane znaki maskujące.

S014\* pozwala CvPcb na wyświetlenie wszystkich footprintów, których nazwa rozpoczyna się znakami SO14.

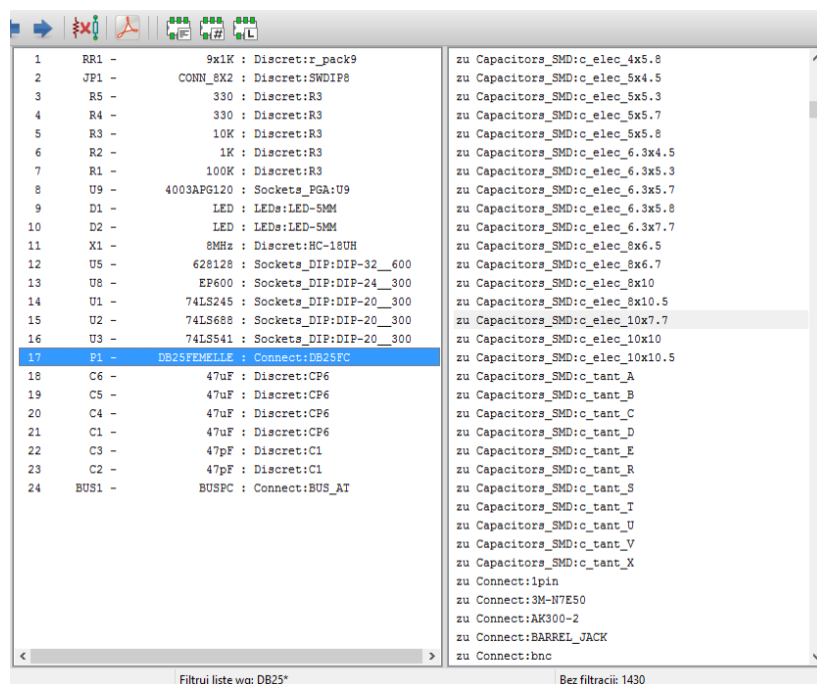
Zaś R? Pozwala na wyświetlenie tylko tych footprintów, których nazwa będzie składać się z dwóch znaków, przy czym nazwa taka będzie musiała się rozpoczynać znakiem R.

Poniżej przedstawiono widok okna CvPcb z włączoną jak i wyłączoną filtracją:

Z filtrowaniem



Bez filtracji




## 13.6 Biblioteka wzorców

You can easily compile a graphic symbols library file containing frequently used symbols. This can be used for the creation of symbols (triangles, the shape of AND, OR, Exclusive OR gates, etc.) for saving and subsequent re-use.

These files are stored by default in the library directory and have a *.sym* extension. These symbols are not gathered in libraries like the normal symbols because they are generally not so many.


### 13.6.1 Eksport/Tworzenie wzorca



A symbol can be exported with the button . You can generally create only one graphic, also it will be a good idea to delete all pins, if they exist.

### 13.6.2 Importowanie wzorca




Importing allows you to add graphics to a symbol you are editing. A symbol is imported with the button . Imported graphics are added as they were created in existing graphics.

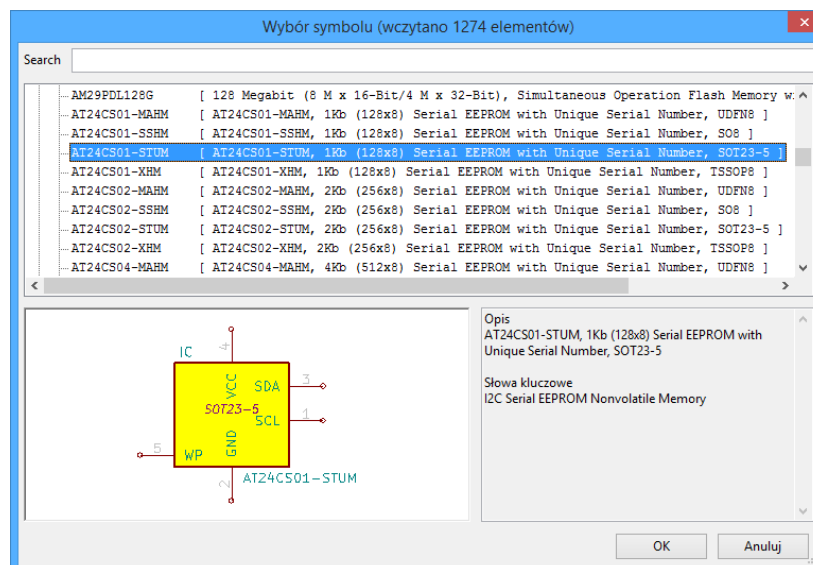
## Rozdział 14

# Symbol Library Browser

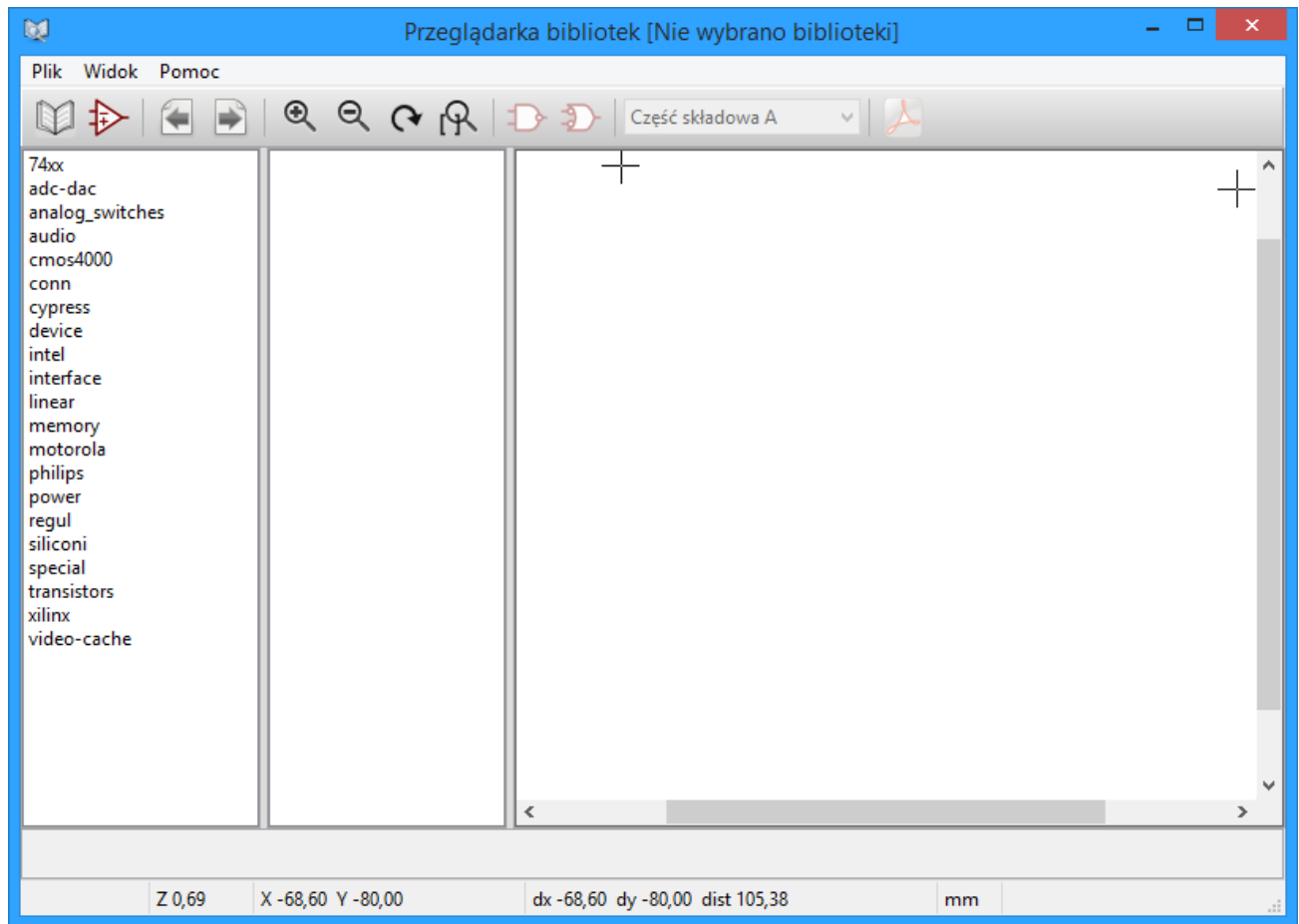
### 14.1 Wprowadzenie

The Symbol Library Browser allows you to quickly examine the content of symbol libraries. The Symbol Library

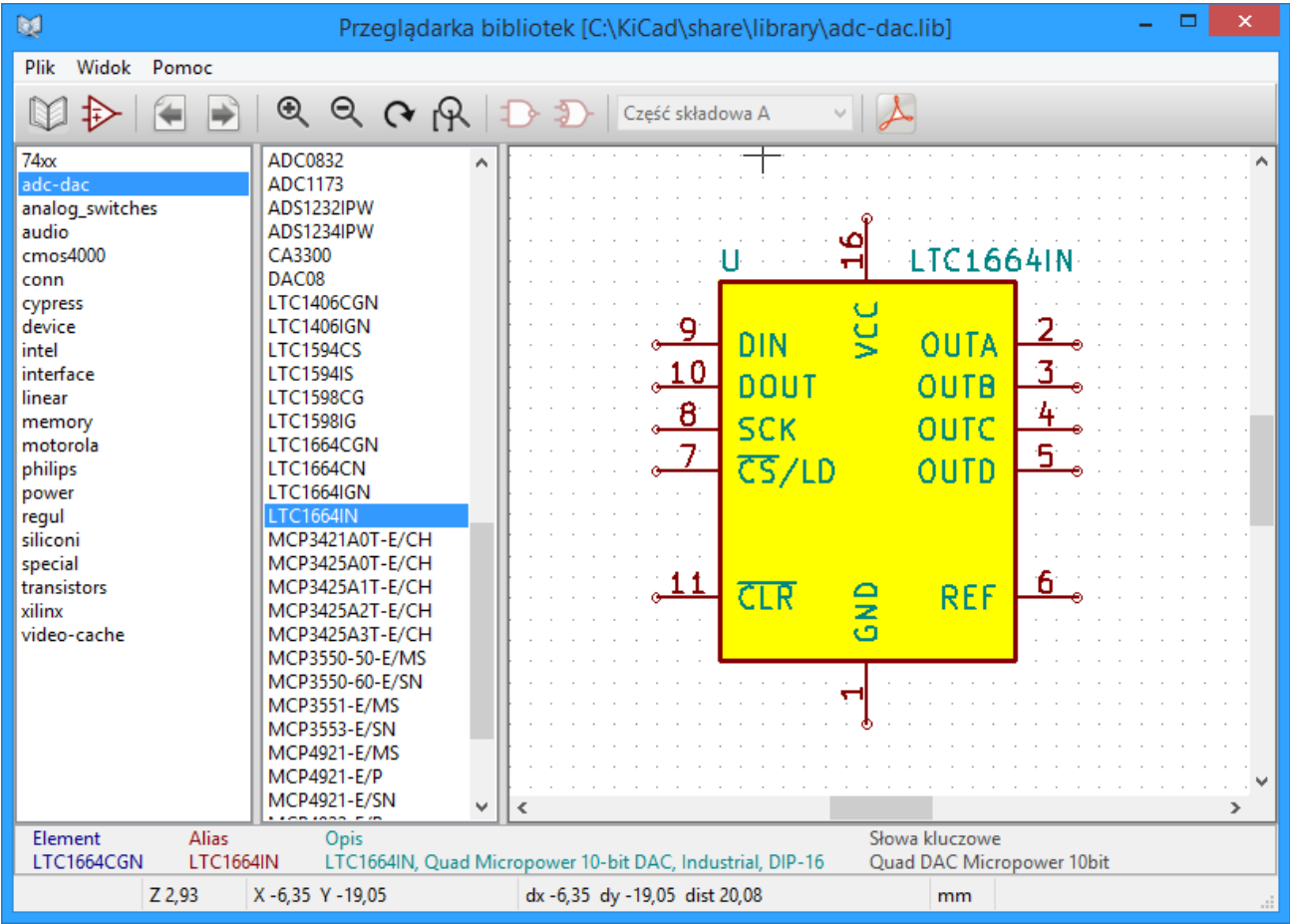
Viewer can be accessed by clicking  icon on the main toolbar, selecting Library Browser entry in the "View" menu or double clicking symbol image on "Choose Symbol" window.



## 14.2 Ekran główny



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.








14.3 Symbol Library Browser Top Toolbar





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



Dostępne polecenia to:

	Selection of the desired library which can be also selected in the displayed list.
	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 converted) if exist.
	Selection of the unit for symbols that contain multiple units.
	If it exist, display the associated documents. Exists only when called by the place symbol dialog frame from Eeschema.
	Close the browser and place the selected symbol in Eeschema. This icon is only displayed when browser has been called from Eeschema (double click on a symbol in the component chooser).

## Rozdział 15

# Tworzenie własnych list sieci i plików BOM

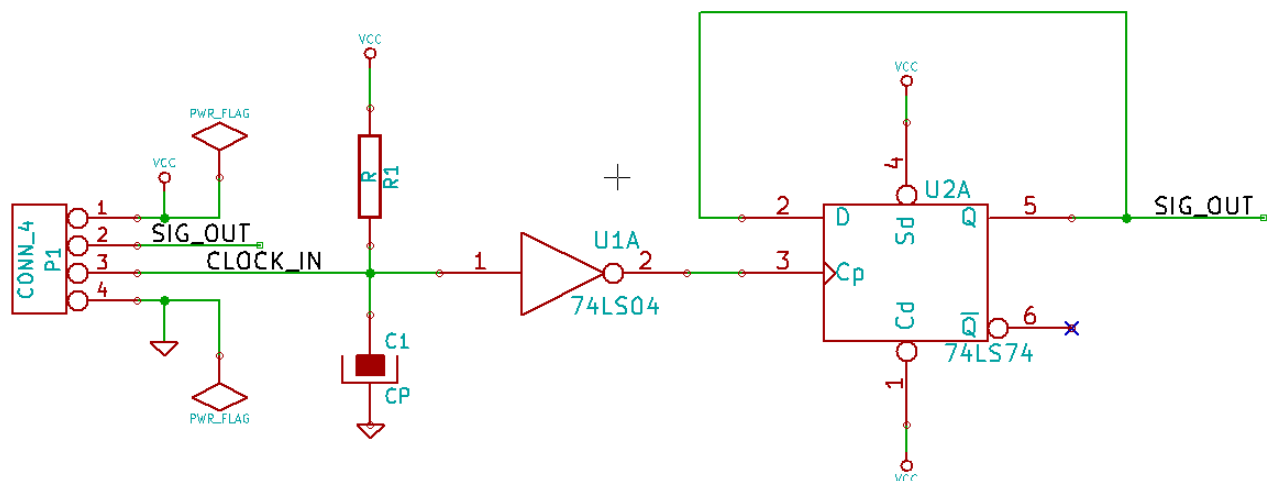
### 15.1 Plik pośredniej listy sieci

Lista materiałowa BOM i lista sieci może być skonwertowana z formatu pośredniego tworzonego przez Eeschema.

Plik ten jest zbudowany z tagów XML i nosi nazwę pliku pośredniego listy sieci. Ponieważ zawiera on jednak wiele danych na temat komponentów schematu, może być też wykorzystywany do tworzenia list materiałowych BOM lub innych raportów - nie tylko list sieci.

W zależności od formatu wyjściowego (BOM, nowe listy sieci), tylko niektóre sekcje tego pliku będą wykorzystywane w trakcie przetwarzania.

#### 15.1.1 Przykładowy schemat



### 15.1.2 Przykład pośredniej listy sieci

Odpowiednia dla przedstawionego schematu pośrednia lista sieci (używając składni XML) jest pokazana poniżej.

```
<?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>
      </footprints>
    </libpart>
  </libparts>
</export>
```

```
<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>
    <fp>R?-*</fp>
    <fp>SM1206</fp>
  </footprints>
  <fields>
    <field name="Reference">R</field>
    <field name="Value">R</field>
  </fields>
  <pins>
    <pin num="1" name="~" type="passive"/>
    <pin num="2" name="~" type="passive"/>
  </pins>
</libpart>
<libpart lib="conn" part="CONN_4">
  <description>Symbole general de connecteur</description>
  <fields>
    <field name="Reference">P</field>
    <field name="Value">CONN_4</field>
  </fields>
  <pins>
    <pin num="1" name="P1" type="passive"/>
    <pin num="2" name="P2" type="passive"/>
    <pin num="3" name="P3" type="passive"/>
    <pin num="4" name="P4" type="passive"/>
  </pins>
</libpart>
<libpart lib="74xx" part="74LS04">
  <description>Hex Inverseur</description>
  <fields>
    <field name="Reference">U</field>
    <field name="Value">74LS04</field>
```

```
</fields>
<pins>
  <pin num="1" name="~" type="input"/>
  <pin num="2" name="~" type="output"/>
  <pin num="3" name="~" type="input"/>
  <pin num="4" name="~" type="output"/>
  <pin num="5" name="~" type="input"/>
  <pin num="6" name="~" type="output"/>
  <pin num="7" name="GND" type="power_in"/>
  <pin num="8" name="~" type="output"/>
  <pin num="9" name="~" type="input"/>
  <pin num="10" name="~" type="output"/>
  <pin num="11" name="~" type="input"/>
  <pin num="12" name="~" type="output"/>
  <pin num="13" name="~" type="input"/>
  <pin num="14" name="VCC" type="power_in"/>
</pins>
</libpart>
<libpart lib="74xx" part="74LS74">
  <description>Dual D FlipFlop, Set & Reset</description>
  <docs>74xx/74hc_hct74.pdf</docs>
  <fields>
    <field name="Reference">U</field>
    <field name="Value">74LS74</field>
  </fields>
  <pins>
    <pin num="1" name="Cd" type="input"/>
    <pin num="2" name="D" type="input"/>
    <pin num="3" name="Cp" type="input"/>
    <pin num="4" name="Sd" type="input"/>
    <pin num="5" name="Q" type="output"/>
    <pin num="6" name="~Q" type="output"/>
    <pin num="7" name="GND" type="power_in"/>
    <pin num="8" name="~Q" type="output"/>
    <pin num="9" name="Q" type="output"/>
    <pin num="10" name="Sd" type="input"/>
    <pin num="11" name="Cp" type="input"/>
    <pin num="12" name="D" type="input"/>
    <pin num="13" name="Cd" type="input"/>
    <pin num="14" name="VCC" type="power_in"/>
  </pins>
</libpart>
</libparts>
<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>
<library logical="74xx">
  <uri>F:\kicad\share\library\74xx.lib</uri>
</library>
</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>
  <net code="5" name="/SIG_OUT">
    <node ref="P1" pin="2"/>
    <node ref="U2" pin="5"/>
    <node ref="U2" pin="2"/>
  </net>
  <net code="6" name="/CLOCK_IN">
    <node ref="R1" pin="2"/>
    <node ref="C1" pin="1"/>
    <node ref="U1" pin="1"/>
    <node ref="P1" pin="3"/>
  </net>
</nets>
</export>
```

## 15.2 Konwersja na nowy format listy sieci

Poprzez zastosowanie filtra dla tego pliku pośredniego z listą sieci, można wygenerować listę sieci w wielu innych formatach oraz listę materiałową BOM. Ponieważ jest to tylko transformacja jednej postaci tekstu na drugą postać,

filtr ten można łatwo napisać w języku PYTHON lub XSLT.

XSLT sam w sobie jest językiem XML przeznaczonym do definiowania transformacji dla XML. W przypadku użycia XSLT, można użyć programu `xsltproc.exe` do odczytu pliku XML z zastosowaniem przy tym arkusza stylu (tzw. "filtr") i zapisaniu danych wynikowych. Dlatego użytkownik musi wcześniej utworzyć odpowiedni arkusz stylu pliku wyjściowego używając konwencji przyjętych w XSLT. Cały proces konwersji jest przejrzysty i obsługiwany przez Eeschema zaraz jak tylko zostanie skonfigurowany.

## 15.3 Konwerter XSLT

Dokumentacja XSL Transformations (XSLT) znajduje się pod adresem:

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

### 15.3.1 Przykład tworzenia pliku z listą sieci PADS-PCB

Format PADS-PCB posiada dwie sekcje pliku listy sieci.

- Listę footprint-ów
- Listę sieci (zgrupowane wyprowadzenia według sieci)

Poniżej znajduje się przykład arkusza stylów, na podstawie którego można skonwertować plik pośredni listy do jej odpowiednika w formacie akceptowanym przez 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>
  <xsl:value-of select="@ref"/>
  <xsl:text>.</xsl:text>
  <xsl:value-of select="@pin"/>
  <xsl:text>&nl;</xsl:text>
</xsl:template>

</xsl:stylesheet>
```



Finalny plik wyjściowy po zastosowaniu tego arkusza jako filtra dla 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*
```

Polecenie które dokonało takiej konwersji wygląda następująco:

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

### 15.3.2 Przykład tworzenia listy sieci programu Cadstar

Format Cadstar składa się z dwóch sekcji.

- Listę footprint-ów

- Listę sieci (zgrupowane wyprowadzenia według sieci)

Tutaj znajduje się przykład pliku z arkuszem stylu do przeprowadzenie tej konwersji:

```
<?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">
      <xsl:text>"</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:variable>
    <xsl:apply-templates select="node" mode="first"/>
    <xsl:value-of select="$netname"/>
    <xsl:apply-templates select="node" mode="others"/>
  </xsl:if>
</xsl:template>

<!-- for each node -->
<xsl:template match="node" mode="first">
  <xsl:if test="position()=1">
    <xsl:text>.ADD_TER </xsl:text>
    <xsl:value-of select="@ref"/>
    <xsl:text>.</xsl:text>
    <xsl:value-of select="@pin"/>
    <xsl:text> </xsl:text>
  </xsl:if>
</xsl:template>

<xsl:template match="node" mode="others">
  <xsl:choose>

```

```
<xsl:when test='position()=1'>
</xsl:when>
<xsl:when test='position()=2'>
    <xsl:text>.TER    </xsl:text>
</xsl:when>
<xsl:otherwise>
    <xsl:text>        </xsl:text>
</xsl:otherwise>
</xsl:choose>
<xsl:if test="position()>1">
    <xsl:value-of select="@ref"/>
    <xsl:text>.</xsl:text>
    <xsl:value-of select="@pin"/>
    <xsl:text>&nl;</xsl:text>
</xsl:if>
</xsl:template>

</xsl:stylesheet>
```

Poniżej znajduje się plik wyjściowy dla programu Cadstar.

```
.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
```

### 15.3.3 Przykład tworzenia listy sieci programu OrcadPCB2

Ten format posiada tylko jedną sekcję - listę footprintów. Każdy z footprintów zawiera swoją listę wyprowadzeń z odnośnikami do właściwych sieci.

Arkusz stylów wymagany do przeprowadzenia tej konwersji:

```

<?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 $noname U1 74LS125
      ... pin list ...
    )
    and calls "create_pin_list" template to build the pin list
-->
<xsl:template match="comp">
    <xsl:text> ( </xsl:text>
    <xsl:choose>
        <xsl:when test = "tstamp != '' ">
            <xsl:apply-templates select="tstamp"/>
        </xsl:when>
        <xsl:otherwise>
            <xsl:text>00000000</xsl:text>
        </xsl:otherwise>
    </xsl:choose>
    <xsl:text> </xsl:text>
    <xsl:choose>
        <xsl:when test = "footprint != '' ">
            <xsl:apply-templates select="footprint"/>
        </xsl:when>
        <xsl:otherwise>
            <xsl:text>$noname</xsl:text>
        </xsl:otherwise>
    </xsl:choose>
    <xsl:text> </xsl:text>
    <xsl:value-of select="@ref"/>
    <xsl:text> </xsl:text>
    <xsl:choose>
        <xsl:when test = "value != '' ">
```

```

        <xsl:apply-templates select="value"/>
    </xsl:when>
    <xsl:otherwise>
        <xsl:text>"~"</xsl:text>
    </xsl:otherwise>
</xsl:choose>
<xsl:text>&nl;</xsl:text>
<xsl:call-template name="Search_pin_list" >
    <xsl:with-param name="cmplib_id" select="libsource/@part"/>
    <xsl:with-param name="cmp_ref" select="@ref"/>
</xsl:call-template>
<xsl:text> )&nl;</xsl:text>
</xsl:template>

<!--
    This template search for a given lib component description in list
    lib component descriptions are in /export/libparts,
    and each description start at ./libpart
    We search here for the list of pins of the given component
    This template has 2 parameters:
        "cmplib_id" (reference in libparts)
        "cmp_ref" (schematic reference of the given component)
-->
<xsl:template name="Search_pin_list" >
    <xsl:param name="cmplib_id" select="0" />
    <xsl:param name="cmp_ref" select="0" />
    <xsl:for-each select="/export/libparts/libpart">
        <xsl:if test = "@part = $cmplib_id ">
            <xsl:apply-templates name="build_pin_list" select="pins/pin">
                <xsl:with-param name="cmp_ref" select="$cmp_ref"/>
            </xsl:apply-templates>
        </xsl:if>
    </xsl:for-each>
</xsl:template>

<!--
    This template writes the pin list of a component
    from the pin list of the library description
    The pin list from library description is something like
        <pins>
            <pin num="1" type="passive"/>
            <pin num="2" type="passive"/>
        </pins>
    Output pin list is ( <pin num> <net name> )
    something like
        ( 1 VCC )
        ( 2 GND )

```

```

-->
<xsl:template name="build_pin_list" match="pin">
    <xsl:param name="cmp_ref" select="0" />

    <!-- write pin numner and separator -->
    <xsl:text> ( </xsl:text>
    <xsl:value-of select="@num"/>
    <xsl:text> </xsl:text>

    <!-- search net name in nets section and write it: -->
    <xsl:variable name="pinNum" select="@num" />
    <xsl:for-each select="/export/nets/net">
        <!-- net name is output only if there is more than one pin in net
            else use "?" as net name, so count items in this net
        -->
        <xsl:variable name="pinCnt" select="count(node)" />
        <xsl:apply-templates name="Search_pin_netname" select="node">
            <xsl:with-param name="cmp_ref" select="$cmp_ref"/>
            <xsl:with-param name="pin_cnt_in_net" select="$pinCnt"/>
            <xsl:with-param name="pin_num"> <xsl:value-of select="$pinNum"/>
            </xsl:with-param>
        </xsl:apply-templates>
    </xsl:for-each>

    <!-- close line -->
    <xsl:text> )&nl;</xsl:text>
</xsl:template>

<!--
    This template writes the pin netname of a given pin of a given component
    from the nets list
    The nets list description is something like
    <nets>
        <net code="1" name="GND">
            <node ref="J1" pin="20"/>
            <node ref="C2" pin="2"/>
        </net>
        <net code="2" name="">
            <node ref="U2" pin="11"/>
        </net>
    </nets>
    This template has 2 parameters:
        "cmp_ref"    (schematic reference of the given component)
        "pin_num"    (pin number)
-->

<xsl:template name="Search_pin_netname" match="node">
    <xsl:param name="cmp_ref" select="0" />

```



```

<xsl:param name="pin_num" select="0" />
<xsl:param name="pin_cnt_in_net" select="0" />

<xsl:if test = "@ref = $cmp_ref ">
  <xsl:if test = "@pin = $pin_num">
    <!-- net name is output only if there is more than one pin in net
         else use "?" as net name
    -->
    <xsl:if test = "$pin_cnt_in_net>1">
      <xsl:choose>
        <!-- if a net has a name, use it,
             else build a name from its net code
        -->
        <xsl:when test = "../@name != '' ">
          <xsl:value-of select="../@name"/>
        </xsl:when>
        <xsl:otherwise>
          <xsl:text>$N-0</xsl:text><xsl:value-of select="../@code"/>
        </xsl:otherwise>
      </xsl:choose>
    </xsl:if>
    <xsl:if test = "$pin_cnt_in_net <2">
      <xsl:text>?</xsl:text>
    </xsl:if>
  </xsl:if>
</xsl:if>

</xsl:template>

</xsl:stylesheet>

```

Poniżej znajduje się plik wyjściowy programu 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 ? )

```

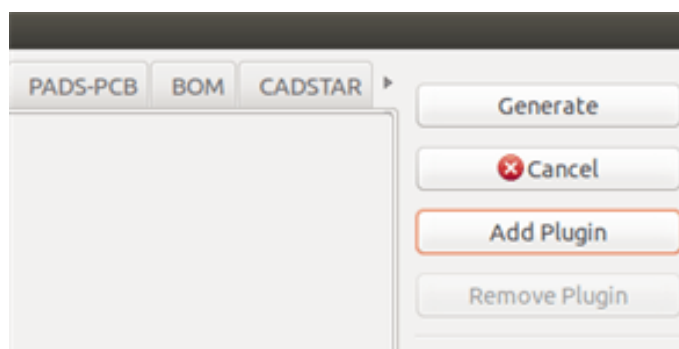
```
( 7 GND )
( 14 VCC )
)
( 4C6E20A6 $noname U1 74LS04
( 1 /CLOCK_IN )
( 2 N-04 )
( 7 GND )
( 14 VCC )
)
( 4C6E2094 $noname C1 CP
( 1 /CLOCK_IN )
( 2 GND )
)
( 4C6E208A $noname R1 R
( 1 VCC )
( 2 /CLOCK_IN )
)
)
*
```

### 15.3.4 Używanie systemu wtyczek Eeschema

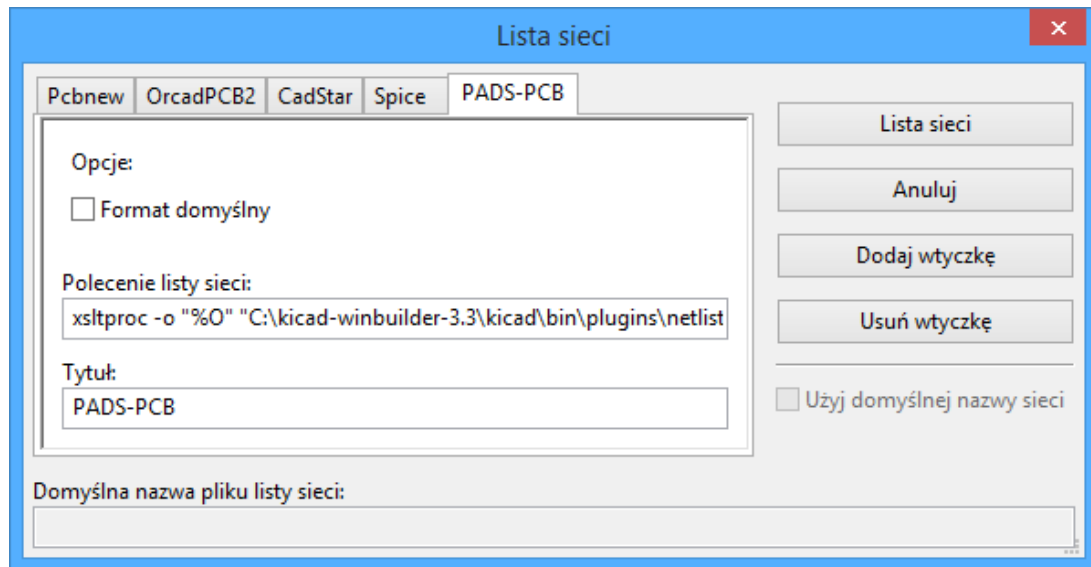
Konwertery pośrednich list sieci mogą być wywoływane bezpośrednio z Eeschema. Wystarczy tylko odpowiednio skonfigurować narzędzie do tworzenia list sieci.

#### 15.3.4.1 Inicjowanie okna dialogowego

Jeśli uruchomimy narzędzie do tworzenia list sieci, to z prawej strony znajdziemy przycisk "Dodaj wtyczkę", służący do konfigurowania i inicjowania nowych zakładek w tym oknie dialogowym.



Poniższy obrazek ukazuje skonfigurowaną wtyczkę PADS-PCB:



#### 15.3.4.2 Ustawienia nowych wtyczek

Zakładka z ustawieniami wymaga następujących informacji:

- Tytułu zakładki (określająca również nazwę formatu wyjściowego listy sieci)
- Lini poleceń, której przekazanie do systemu operacyjnego uruchomi konwersję.

Przy aktywacji klawisza Lista sieci na takiej zakładce:

1. Eeschema tworzy pośredni plik z listą sieci \*.xml, na przykład test.xml.
2. Eeschema uruchamia wtyczkę, która czyta plik test.xml i tworzy plik test.net.

#### 15.3.4.3 Generowanie list sieci za pomocą linii poleceń

Zakładając, że zostanie użyty program *xsltproc.exe* by zastosować filtr z arkusza stylów na pliku pośrednim listy sieci, *xsltproc.exe* będzie potrzebował odpowiednio skonstruowaną listę parametrów, zgodnie ze wzorcem:

*xsltproc.exe -o <plik wyjściowy> <plik arkusza stylów> <plik wejściowy XML do konwersji>*

Tak więc, używając systemu Windows linia poleceń przekazana do systemu będzie miała postać:

*f:/kicad/bin/xsltproc.exe -o "%O" f:/kicad/bin/plugins/netlist\_form\_pads-pcb.xsl "%I"*

Używając systemu Linux polecenie będzie miało postać:

*xsltproc -o "%O" /usr/local/kicad/bin/plugins/netlist\_form\_pads-pcb.xsl "%I"*

Plik *netlist\_form\_pads-pcb.xsl* jest w tym przypadku arkuszem stylu. Należy pamiętać by nazwy plików zawierające (lub mogące zawierać) znaki spacji zamknąć w cudzysłowach.

Linia poleceń akceptuje parametry dla nazw plików:

Jako parametry obsługiwane są sekwencje:

- %B nazwa bazowa pliku oraz ścieżka do wybranego pliku wyjściowego, pozbawiona ścieżki oraz rozszerzenia.
- %I kompletna nazwa pliku oraz ścieżka do tymczasowego pliku wejściowego.
- %O kompletna nazwa pliku oraz ścieżka do wybranego przez użytkownika pliku wyjściowego.

%I zostanie zastąpione przez nazwę aktywnego w danej chwili pliku pośredniego.

%O zostanie zastąpiony przez nazwę aktywnego w danej chwili pliku wyjściowego (końcowy plik z listą sieci).

#### 15.3.4.4 Format linii poleceń: przykład z xsltproc

Format linii poleceń dla xsltproc jest następujący:

```
<ścieżka do xsltproc> xsltproc <parametry>
```

W systemie Windows:

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

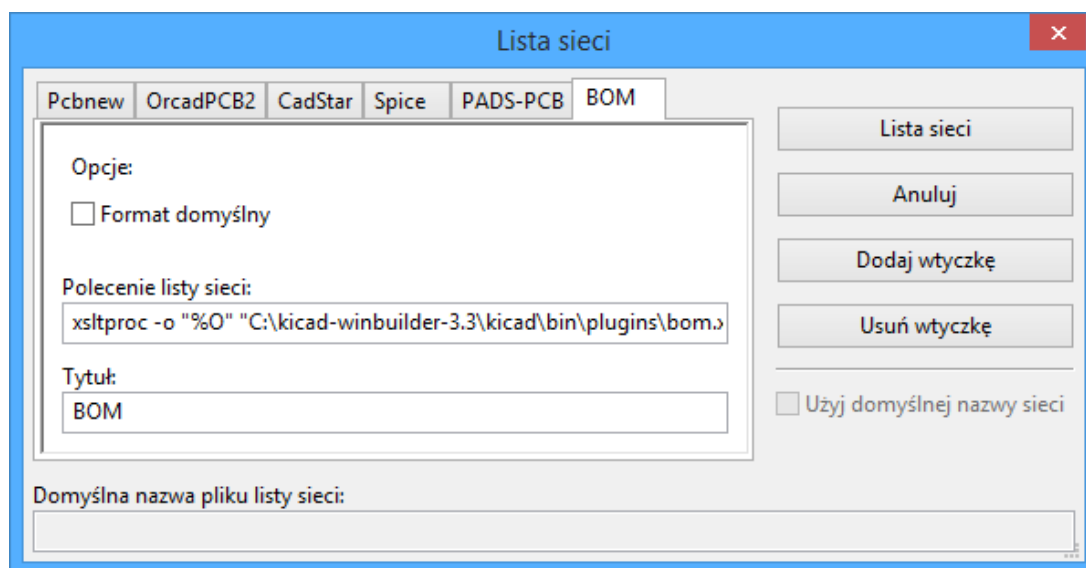
W systemie Linux:

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

Zakładając, że xsltproc został zainstalowany na komputerze (w systemie Windows, wszystkie pliki są w kicad/bin).

#### 15.3.5 Generowanie list materiałowych (BOM)

Ponieważ lista pośrednia sieci zawiera wszystkie informacje o zastosowanych komponentach, można na jej podstawie utworzyć listę materiałową (BOM). Poniżej znajduje się okno z ustawieniami (w systemie Linux) pozwalające utworzyć własny plik BOM:



Ścieżka do arkusza stylu bom2csv.xml jest zależna od systemu operacyjnego. Obecnie najlepszym arkuszem stylu XSLT do generowania plików BOM jest *bom2csv.xml*. Można go zmodyfikować do własnych potrzeb, a jeśli będzie on użyteczny można zaproponować by stał się częścią projektu KiCad.

## 15.4 Format polecenia: Przykład skryptu Python

Linia poleceń dla Python-a wygląda następująco:

```
python <plik_skryptu> <nazwa_pliku_wejściowego> <nazwa_pliku_wyjściowego>
```

W systemie Windows:

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

W systemie Linux:

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

Zakładając, że Python jest zainstalowany.

## 15.5 Plik pośredni listy sieci

Poniższy przykład ukazuje ideę samego pliku pośredniego.

```
<?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="/" />
    </comp>
  </components>
</export>
```

```
<tstamp>4C6E2094</tstamp>
<comp ref="R1">
  <value>R</value>
  <libsource lib="device" part="R"/>
  <sheetpath names="/" tstamp="/" />
  <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>
  <net code="5" name="/SIG_OUT">
    <node ref="P1" pin="2"/>
    <node ref="U2" pin="5"/>
    <node ref="U2" pin="2"/>
  </net>
  <net code="6" name="/CLOCK_IN">
    <node ref="R1" pin="2"/>
    <node ref="C1" pin="1"/>
    <node ref="U1" pin="1"/>
    <node ref="P1" pin="3"/>
  </net>
</nets>
</export>
```

### 15.5.1 Struktura ogólna

Plik pośredni listy sieci posiada 5 sekcji:

- Sekcja nagłówka.
- Sekcja komponentów.
- Sekcja elementów bibliotecznych.
- Sekcja bibliotek.
- Sekcja sieci połączeń.

Cały plik został objęty w tag <export>

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

### 15.5.2 Sekcja nagłówka

Nagłówek znajduje się w tagu <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>
```

Sekcja ta może być widoczna jako komentarze.

### 15.5.3 Sekcja komponentów

Sekcja komponentów zawiera się w tagu <components>

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

Jest to lista na której znajdują się poszczególne komponenty schematu. Każdy komponent jest opisany w następujący sposób:

```

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

```

<b>libsource</b>	name of the lib where this component was found.
<b>part</b>	component name inside this library.
<b>sheetpath</b>	path of the sheet inside the hierarchy: identify the sheet within the full schematic hierarchy.
<b>tstamps (time stamps)</b>	time stamp of the schematic file.
<b>tstamp (time stamp)</b>	time stamp of the component.

### 15.5.3.1 Uwagi na temat odcisków czasowych dla komponentów

Aby zidentyfikować składnik listy sieci (a także na płytce), jest używane jego oznaczenie i jest ono unikalne dla każdego z komponentów. Jednak KiCad udostępnia pomocniczą informację by jednoznacznie zidentyfikować komponent i odpowiadający mu moduł na płytce. Pozwala to na ponowną numerację symboli na schemacie w taki sposób by nie utracić powiązań pomiędzy komponentem i jego modulem.

Znacznik czasowy jest unikalnym identyfikatorem dla każdego składnika lub arkusza schematu w projekcie. Jednak w złożonych hierarchiach, w tym samym arkuszu składnik może być używany więcej niż raz, a zatem arkusz ten zawiera elementy o tym samym znaczniku czasowym.

Dany arkusz (wewnątrz złożonej hierarchii) ma jednak unikalny identyfikator: jego ścieżka wewnętrzna. Dany składnik zaś (wewnątrz złożonej hierarchii) ma unikalny identyfikator: ścieżka wewnętrzna + jego odcisk czasowy.

### 15.5.4 Sekcja elementów bibliotecznych

Sekcja elementów bibliotecznych znajduje się w tagu <libparts>, a dane w tej sekcji są zdefiniowane w bibliotekach schematu. Dla każdego komponentu sekcja ta zawiera dane:

- Nazwy dozwolonych footprintów (nazwy używają masek), zawarte w tagu <fp>.
- Pola zdefiniowane w bibliotece, zawarte w tagu <fields>.
- Lista pinów, zawarte w tagu <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>

```

Linie jak `<pin num="1" type="passive"/>` określają również typ elektryczny pinów. Dostępne są typy:

Input	Zwykły pin wejściowy
Output	Zwykły pin wyjściowy
Bidirectional	Wejście lub wyjście
Tri-state	Wejście lub wyjście trójstanowe
Passive	Pasywny, zwykle w komponentach biernych
Unspecified	Nieznany
Power input	Wejście zasilania dla komponentu
Power output	Wyjście zasilania z komponentu
Open collector	Otwarty kolektor
Open emitter	Otwarty emiter
Not connected	Musi być pozostawiony niepodłączony

### 15.5.5 Sekcja bibliotek

Sekcja bibliotek znajduje się w tagu `<libraries>`. Dostarcza ona listę bibliotek używanych w danym projekcie schematu.

```

<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>

```

### 15.5.6 Sekcja sieci

Sekcja sieci znajduje się w tagu `<nets>`. Zawiera ona listę wszystkich połączeń na schemacie.

```

<nets>
  <net code="1" name="GND">
    <node ref="U1" pin="7"/>
    <node ref="C1" pin="2"/>
  </net>

```

```

    <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>

```

Sekcja ta zawiera wszystkie sieci na schemacie.

Poszczególne sieci są pogrupowane wewnątrz tagu `<net>`:

```

<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	Jest to wewnętrzny identyfikator danej sieci
name	Jest to nazwa danej sieci
node	Zawiera odnośniki do poszczególnych pinów w danej sieci

## 15.6 Więcej informacji na temat xsltproc

Aby uzyskać więcej informacji na temat programu, polecamy zajrzeć na stronę [www: http://xmlsoft.org/XSLT/xsltproc.html](http://xmlsoft.org/XSLT/xsltproc.html)

### 15.6.1 Wprowadzenie

xsltproc jest narzędziem uruchamianym z linii poleceń do filtrowania za pomocą arkuszy stylów XSLT dokumentów XML. Jest on częścią libxslt, biblioteki XSLT C Library przeznaczonej dla GNOME. Chociaż powstała ona jako część projektu GNOME, może również działać niezależnie od GNOME.

xsltproc jest wywoływany z linii poleceń z podaną nazwą arkusza stylów do wykorzystania, a następnie z nazwą pliku lub plików, do którego arkusz stylów ma być zastosowany. Jeśli nazwa pliku wejściowego nie będzie podana, czyli parametr `-i` nie zostanie użyty, będzie wykorzystane standardowe wejście.

Jeśli arkusz stylów jest wbudowany w dokument XML z instrukcjami Style-sheet Processing Instruction, nie będzie trzeba dodatkowo podawać nazwy arkusza stylów w linii poleceń. xsltproc automatycznie wykryje i użyje zawartych stylów. Domyślnie dane wyjściowe zostaną skierowane na `stdout`. Można jednak określić plik wyjściowy przy użyciu opcji `-o`.

## 15.6.2 Synopsis

```
xsltproc [[-V] | [-v] | [-o *plik* ] | [--timing] | [--repeat] |  
[--debug] | [--novalid] | [--noout] | [--maxdepth *wart* ] | [--html] |  
[--param *nazwa* *wart* ] | [--stringparam *nazwa* *wart* ] | [--nonet] |  
[--path *ścieżki* ] | [--load-trace] | [--catalogs] | [--xinclude] |  
[--profile] | [--dumpextensions] | [--nowrite] | [--nomkdir] |  
[--writesubtree] | [--nodtdattr]] [ *arkuszystylu* ] [ *plik1* ] [ *plik2* ]  
[ *....* ]
```

## 15.6.3 Opcje linii poleceń

*-V* lub *--version*

Pokazuje używaną wersję libxml i libxslt.

*-v* lub *--verbose*

Pokazuje każdy krok wykonany przez xsltproc podczas przetwarzania arkusza stylów i dokumentów.

*-o* lub *--output file*

Przekierowuje wyjście do pliku o nazwie *plik*. Dla wyjść wielokrotnych, zwanych także jako “chunking”, *-o* folder/ przekierowuje pliki wyjściowe do określonego katalogu. Katalog ten musi być wcześniej utworzony.

*--timing*

Pokazuje czas zużyty na przetworzenie arkusza stylów, przetworzenia dokumentu oraz zastosowania arkusza stylów, a także czas zapisu danych wynikowych. Wartości pokazywane są milisekundach.

*--repeat*

Uruchamia transformację 20 razy. Używane przy testach czasowych.

*--debug*

Pokazuje drzewo XML transformowanego dokumentu w celu usuwania usterek w oprogramowaniu.

*--novalid*

Opuszcza ładowanie dokumentów DTD.

*--noout*

Nie generuje danych wyjściowych.

*--maxdepth value*

Określa maksymalną głębokość stosu wzorców, przed stwierdzeniem o wejściu libxslt do nieskończonej pętli. Domyślnie jest to 500.

*--html*

Dokument wejściowy jest plikiem HTML.

*--param name value*

Przekazuje parametr *nazwa* i wartość *wartość* do arkusza stylów. Można przekazać wiele par *nazwa*/*wartość*, jednak nie więcej niż 32. Jeśli wartość przekazywana jest łańcuchem a nie identyfikatorem węzła, należy użyć `--stringparam` zamiast tej opcji.

`--stringparam name value`

Przekazuje parametr *nazwa* i wartość *wartość* gdzie *wartość* jest łańcuchem znaków a nie identyfikatorem węzła. (Uwaga : Ciąg musi posiadać znaki kodowane w UTF-8.)

`--nonet`

Zabrania użycia sieci Internet w celu pobrania DTD, podmiotów lub dokumentów.

`--path paths`

Używa listy (separowanej za pomocą spacji lub przecinków) ścieżek systemu plików określonych przez *paths* w celu załadowania DTD, podmiotów lub dokumentów.

`--load-trace`

Wysyła na stderr wszystkie dokumenty ładowane podczas przetwarzania.

`--catalogs`

Używa katalogu SGML określonego w `SGML_CATALOG_FILES` by określić lokalizację zewnętrznych podmiotów. Domyślnie, `xsltproc` zagląda do katalogu określonego w `XML_CATALOG_FILES`. Jeśli nie jest to określone, używa `etc/xml/catalog`.

`--xinclude`

Przetwarza dokumenty wejściowe używając specyfikacji Xinclude. Więcej szczegółów na ten temat można znaleźć na stronie Web specyfikacji Xinclude: <http://www.w3.org/TR/xinclude/>

`--profile --norman`

Zwraca sprofilowane informacje na temat czasu spędzonego w każdej części arkusza stylów. Jest to przydatne w optymalizacji wydajności arkuszy stylów.

`--dumpextensions`

Zwraca listę wszystkich zarejestrowanych rozszerzeń na stdout.

`--nowrite`

Odrzuca polecenia tworzenia plików lub zasobów.

`--nomkdir`

Odrzuca polecenia utworzenia katalogów.

`--writesubtree path`

Pozwala na zapis tylko do wybranej podgałęzi *path*.

`--nodtdattr`

Nie stosuje domyślnych atrybutów pochodzących z dokumentów DTD.

---

### 15.6.4 Zwracane wartości

xsltproc zwraca także kody błędów, których można użyć w przypadku wywołań programu wewnątrz skryptów:

0 : normalne zakończenie

1 : brak argumentu

2 : za dużo parametrów

3 : opcja nieznana

4 : niepowodzenie przy parsowaniu arkusza stylów

5 : błąd arkusza stylu

6 : błąd w jednym z dokumentów

7 : nieobsługiwana metoda xsl:output

8 : parametry w postaci ciągów zawierają zarówno znaki apostrofów jak i cudzysłowów

9 : błąd wewnętrzny

10 : przetwarzanie zostało zatrzymane przez komunikat o przerwaniu

11 : nie można zapisać danych wyjściowych do pliku wyjściowego

### 15.6.5 Więcej informacji na temat xsltproc

Strona WEB libxml: <http://www.xmlsoft.org/>

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

---

## Rozdział 16

# Simulator

Eeschema 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.

### 16.1 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.

---

#### Notatka

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:

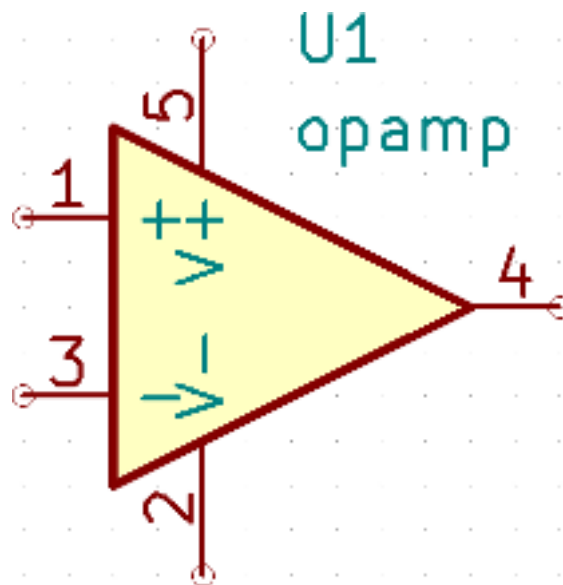
Disable symbol for simulation	When checked the component is excluded from simulation.
-------------------------------	---

## Alternate node sequence

Allows one to override symbol pin to model node mapping. To define a different mapping, specify pin numbers in order expected by the model.

*Example:*

```
* connections:
* 1:non-inverting input
* 2:inverting input
* 3:positive power supply
* 4:negative power supply
* 5:output
.subckt tl071 1 2 3 4 5
```



To match the symbol pins to the Spice model nodes shown above, one needs to use an alternate node sequence option with value: "1{nbsp}3{nbsp}5{nbsp}2{nbsp}4". It is a list of pin numbers corresponding to the Spice model nodes order.

### 16.1.1 Passive

*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.

**Notatka**

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

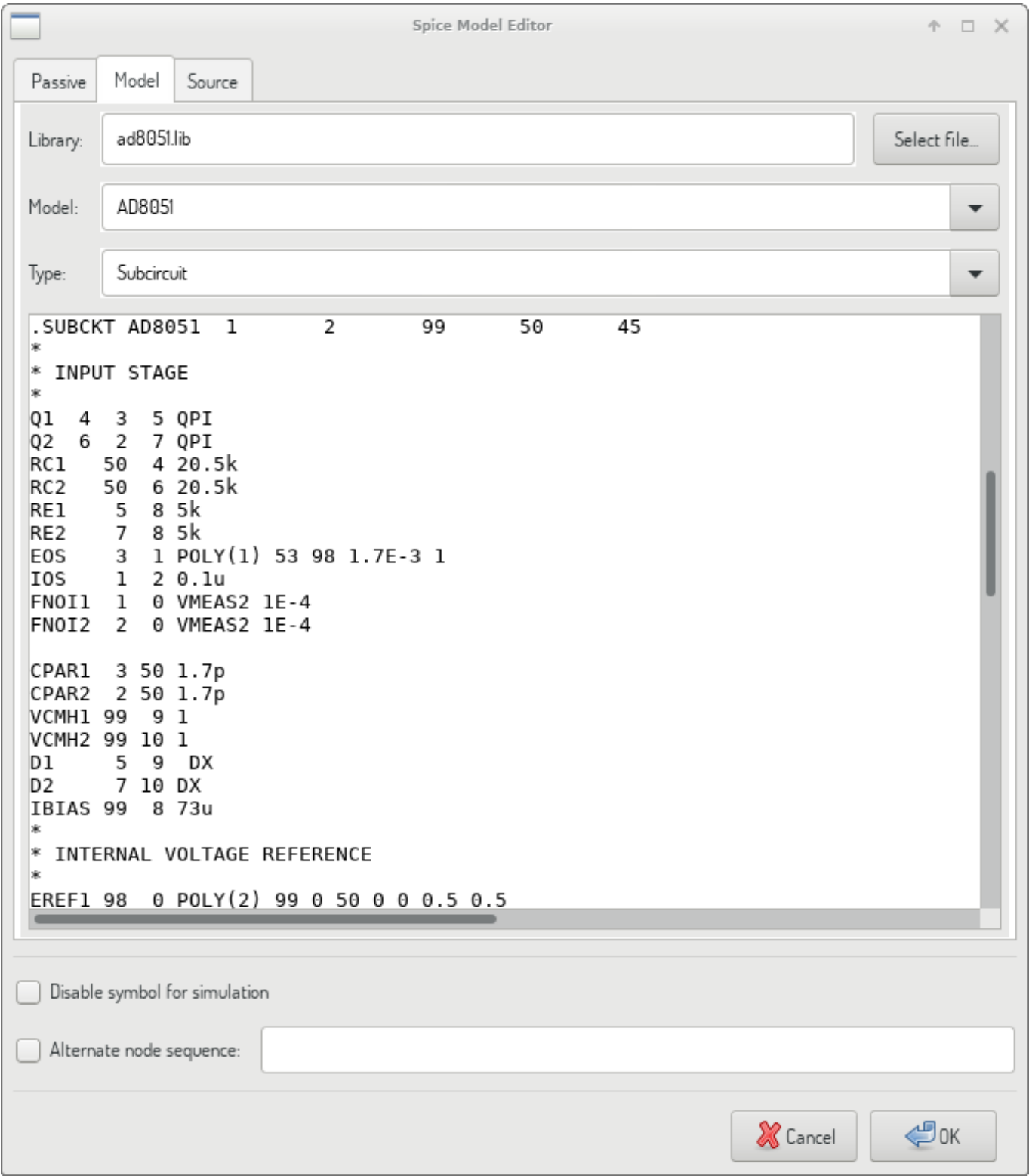


Type	Selects the device type (resistor, capacitor or inductor).
Value	Defines the device property (resistance, capacitance or inductance). The value may use common Spice unit prefixes (as listed below the text input field) and should use point as the decimal separator. Note that Spice does not correctly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

### 16.1.2 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.



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.
------	---

**16.1.3 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.

Spice Model Editor

Passive Model **Source**

DC/AC analysis:

DC:  Volts/Amps

AC magnitude:  Volts/Amps AC phase:  radians

Transient analysis:

Pulse Sinusoidal Exponential Piece-wise Linear

Initial value:  Volts/Amps

Pulsed value:  Volts/Amps

Delay time:  seconds

Rise time:  seconds

Fall time:  seconds

Pulse width:  seconds

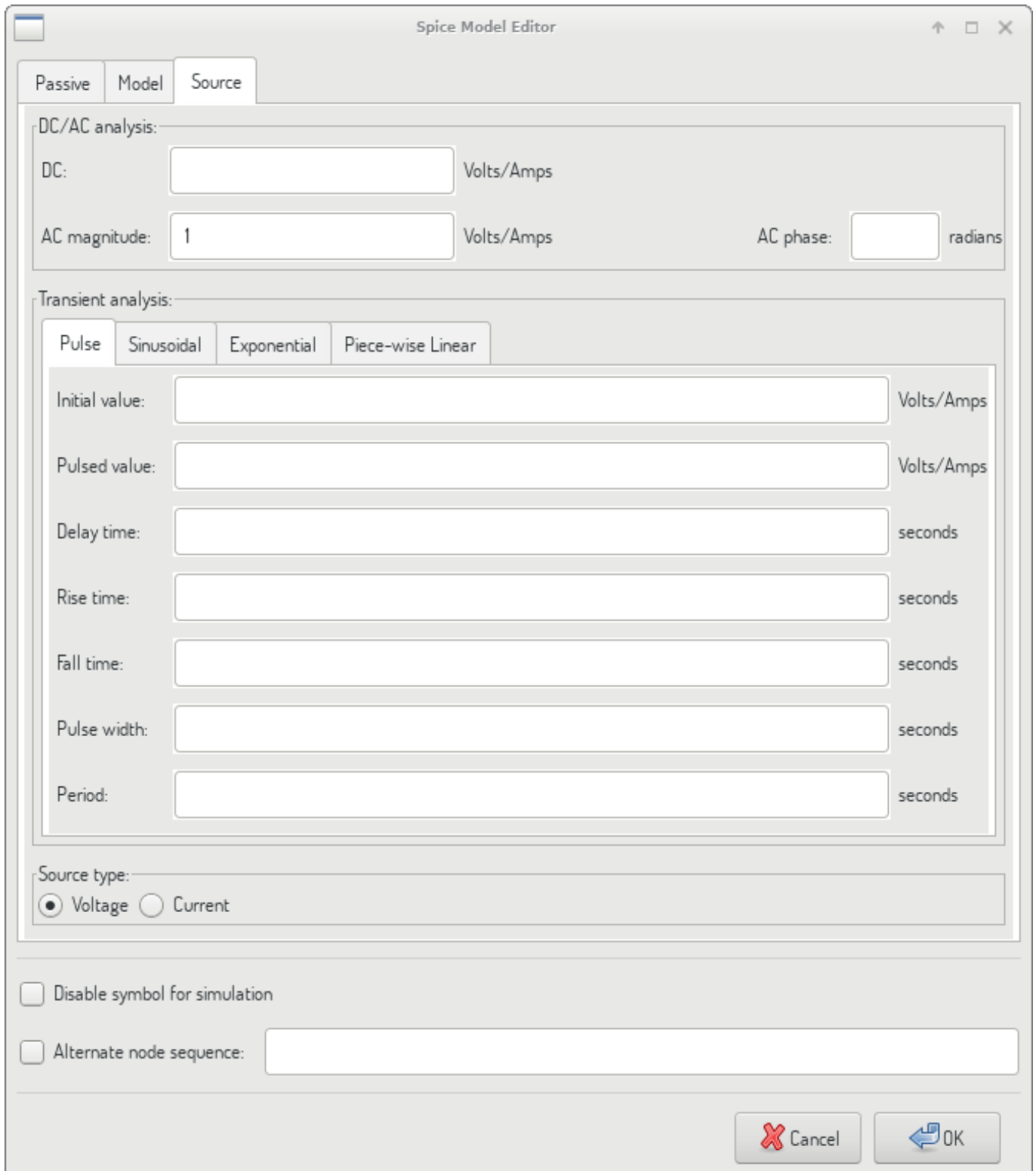

Period:  seconds

Source type:

☒ Voltage ☐ Current

☐ Disable symbol for simulation

☐ Alternate node sequence:

 Cancel  OK

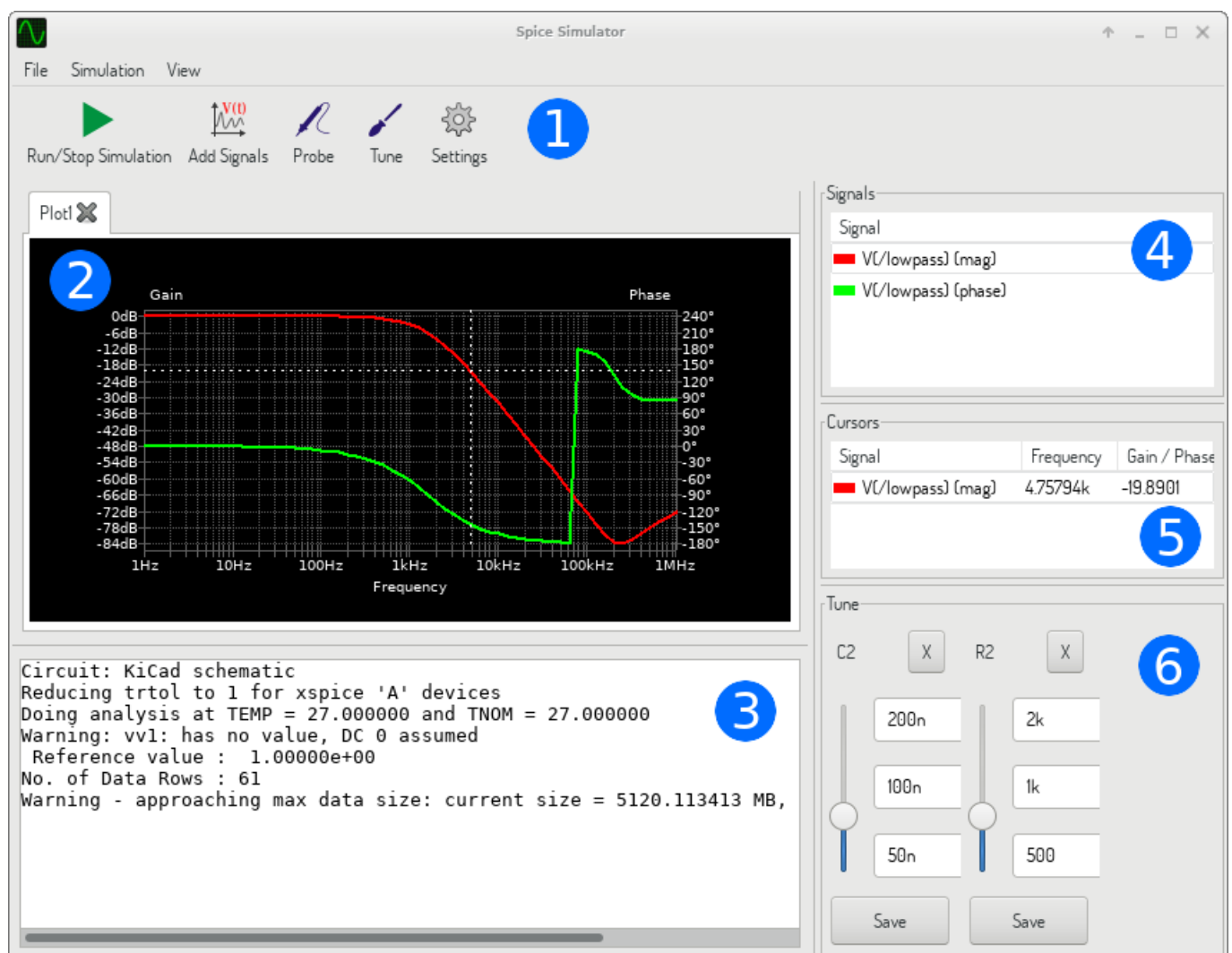
Refer to the [ngspice documentation](#), chapter 4 (Voltage and Current Sources) for more details about sources.

## 16.2 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.

## 16.3 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](#)

16.3.1 Menu

16.3.1.1 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.

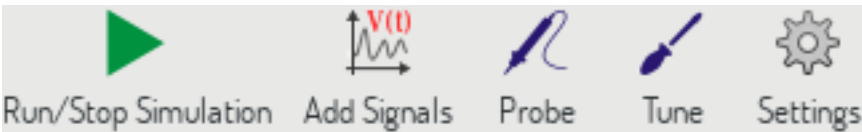
16.3.1.2 Simulation

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 <a href="#">Probe</a> tool.
Tune component value	Start the <a href="#">Tuner</a> tool.
Show SPICE Netlist...	Open a dialog showing the generated netlist for the simulated circuit.
Settings...	Open the <a href="#">simulation settings dialog</a> .

16.3.1.3 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.

16.3.2 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 <a href="#">Probe</a> tool.
Tune	Start the <a href="#">Tuner</a> tool.
Settings	Open the <a href="#">simulation settings dialog</a> .

### 16.3.3 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

### 16.3.4 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.

### 16.3.5 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

### 16.3.6 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.

---

### 16.3.7 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.

### 16.3.8 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.

### 16.3.9 Probe tool

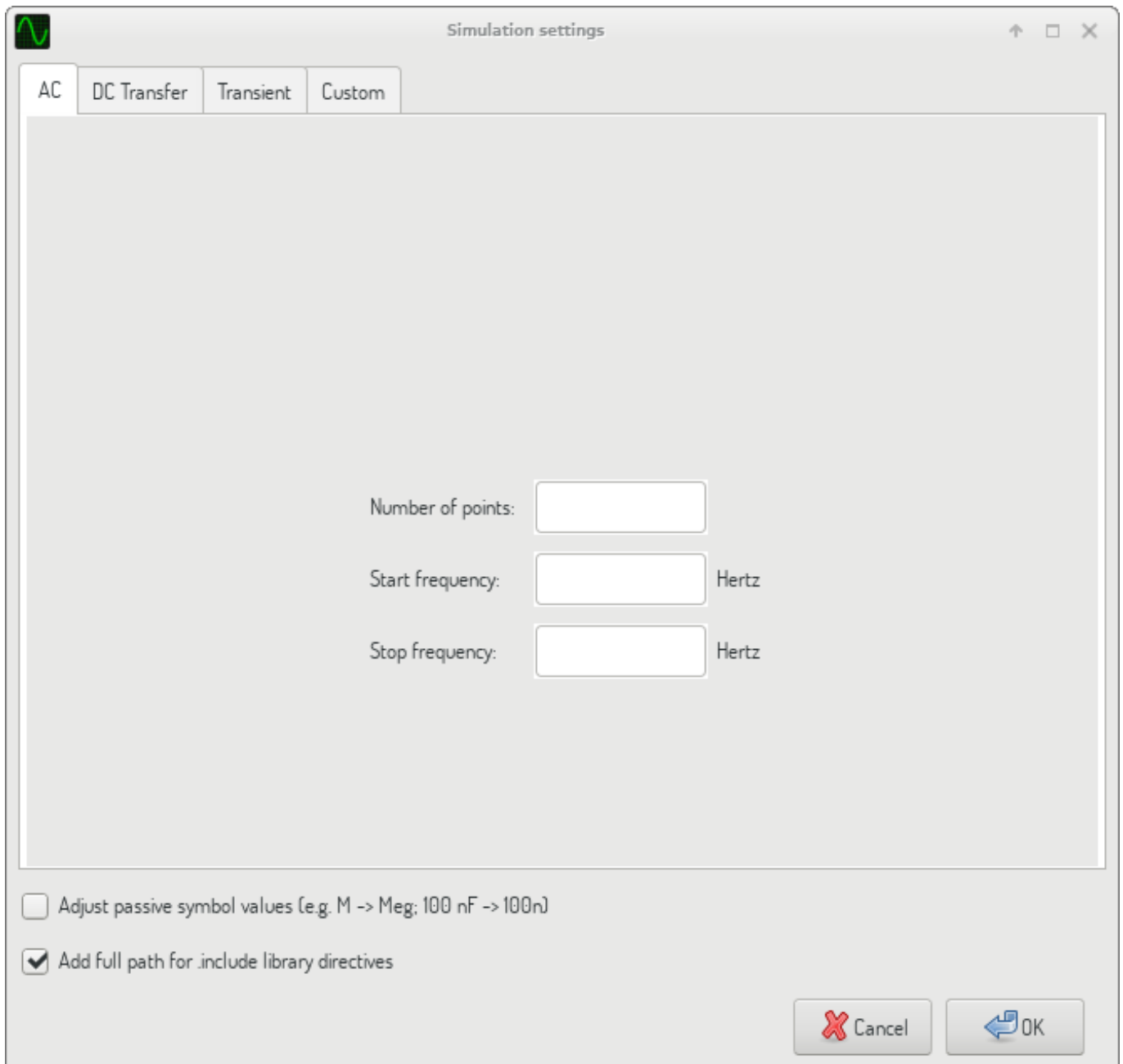
Probe tool provides an user-friendly way of selecting signals for plotting.

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

---



### 16.3.10 Simulation settings



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.