



**Eeschema**

**6 Nopember 2020**

---

# Daftar Isi

<b>1</b>	<b>Mengenai Eeschema</b>	<b>1</b>
1.1	Deskripsi	1
1.2	Tinjauan Teknis	1
<b>2</b>	<b>Perintah-perintah Umum Eeschema</b>	<b>2</b>
2.1	Perintah-perintah Tetikus	3
2.1.1	Perintah-perintah Dasar	3
2.1.2	Pekerjaan Blok	3
2.2	Tombol <i>Hotkey</i>	4
2.3	Grid	6
2.4	Pilihan Perbesaran/perkecilan	6
2.5	Menampilkan Koordinat Kursor	6
2.6	Bilah Menu Atas	7
2.7	Bilah Alat Bagian Atas	7
2.8	Bilah Alat Bagian Kanan	9
2.9	Bilah Alat Sebelah Kiri	10
2.10	Menu <i>Pop-up</i> dan Pengeditan Cepat	10
<b>3</b>	<b>Menu Utama</b>	<b>13</b>
3.1	Menu <i>File</i>	13
3.2	Menu <i>Preferences</i>	14
3.2.1	Mengelola Tabel Pustaka Simbol	15
3.2.1.1	Menambahkan Pustaka Baru	15
3.2.1.2	Menghapus Pustaka	15
3.2.1.3	Properti Pustaka	16
3.2.2	Menu <i>General Options</i>	16
3.2.2.1	Tab <i>Display</i>	16
3.2.2.2	Tab <i>Editing</i>	17
3.2.2.3	Tab <i>Controls</i>	18
3.2.2.4	Tab <i>Colors</i>	19
3.2.2.5	Tab <i>Default Fields</i>	19
3.3	Menu <i>Help</i>	20

<b>4</b>	<b>Bilah Atas Umum</b>	<b>21</b>
4.1	Manajemen Lembar Kerja	21
4.2	Alat Pencari	21
4.3	Alat <i>Netlist</i>	22
4.4	Alat Anotasi	23
4.5	Alat <i>Electrical Rules Check</i>	25
4.5.1	Kotak Dialog Utama ERC	25
4.5.2	Kotak Dialog Opsi ERC	26
4.6	Alat Daftar Komponen ( <i>Bill of Materials</i> )	27
4.7	Alat Pengaturan Atribut	30
4.7.1	Tricks to simplify fields filling	31
4.8	Peralatan Impor <i>Footprint</i>	32
4.8.1	Cara mengakses:	32
<b>5</b>	<b>Manajemen Pustaka Simbol</b>	<b>33</b>
5.1	Tabel Pustaka Simbol ( <i>Symbol Library Table</i> )	33
5.1.1	Tabel Pustaka Simbol Global	34
5.1.2	Tabel Pustaka Simbol Proyek	34
5.1.3	Konfigurasi Awal	35
5.1.4	Menambahkan Entri Tabel	35
5.1.5	Substitusi <i>Environment Variable</i>	35
5.1.6	Pola Pemakaian	36
5.1.7	Pemetaan Ulang Proyek <i>Legacy</i>	36
<b>6</b>	<b>Membuat dan Mengedit Skematik</b>	<b>37</b>
6.1	Pengenalan	37
6.2	Pertimbangan Umum	37
6.3	Rantai Pengembangan	38
6.4	Meletakkan dan Mengedit Simbol	38
6.4.1	Mencari dan Meletakkan Simbol	38
6.4.2	<i>Power Port</i>	40
6.4.3	Mengedit dan Modifikasi Simbol (untuk komponen yang telah diletakkan)	40
6.4.3.1	Modifikasi Simbol	40
6.4.3.2	Modifikasi Atribut Teks	41
6.5	<i>Wire, Bus, Label, Power Port</i>	41
6.5.1	Pengenalan	41
6.5.2	Koneksi ( <i>Wire</i> dan <i>Label</i> )	42
6.5.3	Koneksi ( <i>Bus</i> )	43
6.5.3.1	Anggota <i>Bus</i>	43

6.5.3.2	Koneksi Antar Anggota <i>Bus</i>	43
6.5.3.3	Koneksi Global Antar <i>Bus</i>	44
6.5.4	Koneksi <i>Port Power</i>	45
6.5.5	“No Connect” flag	46
6.6	Peralatan Pelengkap Gambar	46
6.6.1	Komentar Teks	46
6.6.2	Blok Judul pada Lembar Kerja	46
6.7	Menyelamatkan tembolok simbol	47
<b>7</b>	<b>Skematik Hirarkis</b>	<b>49</b>
7.1	Pengenalan	49
7.2	Penelusuran Hirarki	50
7.3	Label Lokal, Hirarkis, dan Global	50
7.3.1	Properti	50
7.4	Membuat Hirarki	50
7.5	Simbol Lembar Kerja	51
7.6	Koneksi - Pin Hirarkis	51
7.7	Koneksi - Label Hirarkis	52
7.7.1	Label, Label Hirarkis, Label Global, dan <i>Pin Power</i> Tersembunyi	53
7.7.1.1	Label Sederhana	54
7.7.1.2	Label Hirarkis	54
7.7.1.3	<i>Pin Power</i> Tersembunyi	54
7.7.2	Label Global	54
7.8	Hirarki Kompleks	54
7.9	Hirarki Datar	55
<b>8</b>	<b>Alat Anotasi Simbol</b>	<b>58</b>
8.1	Pengenalan	58
8.2	Contoh Penggunaan	59
8.2.1	Urutan Anotasi	59
8.2.2	Pilihan Anotasi	60
<b>9</b>	<b>Verifikasi Desain dengan <i>Electrical Rules Check</i></b>	<b>63</b>
9.1	Pengenalan	63
9.2	Penggunaan ERC	64
9.3	Contoh ERC	65
9.4	Menampilkan Hasil Pemeriksaan	65
9.5	Pin dan Bendera <i>Power</i>	66
9.6	Konfigurasi	67
9.7	Berkas Laporan ERC	67

<b>10</b>	<b>Membuat Netlist</b>	<b>68</b>
10.1	Gambaran Umum . . . . .	68
10.2	Format-format Netlist . . . . .	68
10.3	Contoh Netlist . . . . .	69
10.4	Catatan Mengenai Netlist . . . . .	72
10.4.1	Penamaan Netlist . . . . .	72
10.4.2	Netlist PSPICE . . . . .	72
10.5	Format Lain . . . . .	73
10.5.1	Memulai Kotak Dialog . . . . .	73
10.5.2	Format Baris Perintah . . . . .	74
10.5.3	Konverter dan Gaya Lembar Kerja ( <i>Plugin</i> ) . . . . .	74
10.5.4	Format Berkas Netlist Antara . . . . .	75
<b>11</b>	<b>Plot dan Cetak</b>	<b>76</b>
11.1	Pengenalan . . . . .	76
11.2	Perintah Pencetak Umum . . . . .	76
11.3	Plot dalam Format Postscript . . . . .	77
11.4	Plot dalam Format PDF . . . . .	78
11.5	Plot dalam Format SVG . . . . .	78
11.6	Plot dalam Format DXF . . . . .	79
11.7	Plot dalam Format HPGL . . . . .	79
11.7.1	Pemilihan Ukuran Lembar Kerja . . . . .	80
11.7.2	Pengaturan <i>Offset</i> . . . . .	80
11.8	Mencetak di Kertas . . . . .	81
<b>12</b>	<b>Editor Pustaka Simbol</b>	<b>82</b>
12.1	Informasi Umum . . . . .	82
12.2	Tentang Pustaka Simbol . . . . .	82
12.3	Tentang Editor Pustaka Simbol . . . . .	83
12.3.1	Bilah Alat Utama . . . . .	83
12.3.2	Bilah Alat Elemen . . . . .	84
12.3.3	Bilah Alat Opsi . . . . .	85
12.4	Pemilihan dan Pemeliharaan Pustaka . . . . .	85
12.4.1	Memilih dan Menyimpan Simbol . . . . .	86
12.4.1.1	Pemilihan Simbol . . . . .	86
12.4.1.2	Menyimpan Simbol . . . . .	86
12.4.1.3	Mentransfer Simbol ke Pustaka Lain . . . . .	87
12.4.1.4	Membatalkan Perubahan Simbol . . . . .	87
12.5	Membuat Simbol Pustaka . . . . .	87

---

12.5.1	Membuat Simbol Baru	87
12.5.2	Membuat Simbol dari Simbol yang Lain	88
12.5.3	Properti Simbol	89
12.5.4	Simbol dengan Representasi Simbolis Alternatif	90
12.6	Elemen Grafis	91
12.6.1	Keanggotaan Elemen Grafis	91
12.6.2	Elemen Teks Grafis	92
12.7	Simbol dengan Beberapa Bagian dan Gaya Bentuk Alternatif	92
12.7.1	Contoh Simbol dengan Beberapa Bagian dengan Simbol Berbeda:	93
12.7.1.1	Elemen Simbolis Grafis	94
12.8	Membuat dan Mengedit Pin	95
12.8.1	Tentang Pin	95
12.8.2	Properti Pin	96
12.8.3	Gaya Grafis Pin	96
12.8.4	Tipe Elektrikal Pin	97
12.8.5	Properti Global Pin	97
12.8.6	Mendefinisikan Pin untuk Banyak Bagian dan Representasi Simbolis Alternatif	98
12.9	Atribut-atribut Simbol	98
12.9.1	Mengedit Atribut Simbol	99
12.10	Simbol <i>Power</i>	100
<b>13</b>	<b>LibEdit - Simbol</b>	<b>102</b>
13.1	Gambaran Umum	102
13.2	Memposisikan Jangkar Simbol	103
13.3	Alias Simbol	103
13.4	Atribut-atribut Simbol	104
13.5	Dokumentasi Simbol	105
13.5.1	Kata Kunci Simbol	106
13.5.2	Dokumentasi Simbol (Doc)	106
13.5.3	Berkas Dokumen Terkait (DocFileName)	106
13.5.4	Penyaring <i>Footprint</i> untuk CvPcb	107
13.6	Pustaka Simbol	108
13.6.1	Mengekspor atau Membuat Simbol	109
13.6.2	Mengimpor Simbol	109
<b>14</b>	<b>Peramban Pustaka Simbol</b>	<b>110</b>
14.1	Pengenalan	110
14.2	Viewlib - Tampilan Utama	111
14.3	Bilah Alat Atas pada Peramban Pustaka Simbol	111

<b>15</b>	<b>Membuat Netlist dan Berkas BOM Kustom</b>	<b>113</b>
15.1	Berkas Netlist Antara	113
15.1.1	Contoh Skematik	113
15.1.2	Contoh Berkas Netlist Antara	113
15.2	Konversi ke Format Netlist yang Baru	117
15.3	Pendekatan XSLT	117
15.3.1	Membuat Berkas Netlist Pads-Pcb	117
15.3.2	Membuat Berkas Netlist Cadstar	119
15.3.3	Membuat Berkas Netlist OrcadPCB2	121
15.3.4	Antarmuka <i>Plugin</i> Eeschema	126
15.3.4.1	Membuka Kotak Dialog <i>Plugin</i>	126
15.3.4.2	Parameter Konfigurasi <i>Plugin</i>	126
15.3.4.3	Membuat Berkas Netlist dengan Baris Perintah	127
15.3.4.4	Format Baris Perintah: Contoh xsltproc	127
15.3.5	Pembuatan <i>Bill of Materials</i>	127
15.4	Format Baris Perintah: Contoh Skrip Python	128
15.5	Struktur Netlist Antara	128
15.5.1	Struktur Berkas Netlist Umum	130
15.5.2	Bagian <i>Header</i>	130
15.5.3	Bagian Komponen	130
15.5.3.1	Catatan Mengenai Tanda Waktu Komponen	131
15.5.4	Bagian <i>Libparts</i>	131
15.5.5	Bagian Pustaka	132
15.5.6	Bagian <i>Net</i>	132
15.6	Lebih Lanjut Tentang xsltproc	132
15.6.1	Pengenalan	133
15.6.2	Sinopsis	133
15.6.3	Opsi-opsi Baris Perintah	133
15.6.4	Nilai Kembali Xsltproc	134
15.6.5	Informasi Lebih Lanjut Tentang xsltproc	135
<b>16</b>	<b>Simulator</b>	<b>136</b>
16.1	Menerapkan Model	136
16.1.1	Tab <i>Passive</i>	137
16.1.2	Tab <i>Model</i>	138
16.1.3	Tab <i>Source</i>	139
16.2	Direktif Spice	140
16.3	Simulasi	140
16.3.1	Menu	141

---



---

16.3.1.1	Menu <i>File</i>	141
16.3.1.2	Simulasi	142
16.3.1.3	Menu <i>View</i>	142
16.3.2	Bilah Alat	142
16.3.3	Panel <i>Plot</i>	142
16.3.4	Konsol Keluaran	142
16.3.5	Daftar Sinyal	143
16.3.6	Daftar Kursor	143
16.3.7	Panel <i>Tune</i>	143
16.3.8	Alat <i>Tuner</i>	143
16.3.9	Alat <i>Probe</i>	143
16.3.10	Pengaturan Simulasi	144

---

## *Buku Panduan*

### **Hak Cipta**

*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.*

*All trademarks within this guide belong to their legitimate owners.*

### **Kontributor**

Jean-Pierre Charras, Fabrizio Tappero.

### **Penerjemah**

Triyan W. Nugroho.

### **Umpan balik**

Silakan mengirimkan laporan *bug*, saran atau versi baru ke alamat berikut:

- Tentang dokumentasi KiCad: <https://gitlab.com/kicad/services/kicad-doc/issues>
- Tentang perangkat lunak KiCad: <https://gitlab.com/kicad/code/kicad/issues>
- Tentang penerjemahan KiCad: <https://gitlab.com/kicad/code/kicad-i18n/issues>

### **Tanggal Publikasi dan Versi Perangkat Lunak**

Dipublikasikan pada tanggal 30 Mei 2015.

---

## Bab 1

# Mengenal Eeschema

### 1.1 Deskripsi

Eeschema adalah sebuah perangkat lunak untuk membuat skematik rangkaian elektronika yang didistribusikan sebagai bagian dari KiCad dan tersedia untuk sistem operasi berikut ini:

- Linux
- Apple macOS
- Windows

Apapun sistem operasi yang Anda gunakan, semua berkas Eeschema 100% kompatibel antar sistem operasi.

Eeschema adalah aplikasi yang terintegrasi, dimana semua fungsi untuk menggambar, mengontrol, mengatur tata letak, manajemen pustaka, dan akses ke perangkat lunak desain PCB dapat dilakukan di dalam Eeschema itu sendiri.

Eeschema didesain untuk bekerja sama dengan PcbNew, yaitu perangkat lunak desain papan sirkuit di KiCad. Eeschema juga bisa mengekspor berkas *netlist*, yang akan mendaftar semua koneksi elektrik, untuk paket perangkat lunak lainnya.

Eeschema menyertakan sebuah editor pustaka simbol, yang dapat membuat dan mengedit simbol dan melakukan pengaturan pustaka. Eeschema juga mengintegrasikan fungsi-fungsi tambahan namun penting yang diperlukan oleh perangkat lunak pembuat skematik yang modern, antara lain:

- *Electrical rules check* (ERC), sebagai kontrol otomatis untuk koneksi yang hilang atau tidak tepat
- Ekspor berkas-berkas plot dalam berbagai format (Postscript, PDF, HPGL, dan SVG)
- Pembuatan daftar komponen (*Bill of Materials*), melalui skrip Python atau XSLT, yang mampu menangani berbagai format fleksibel.

### 1.2 Tinjauan Teknis

Eeschema hanya dibatasi oleh memori yang tersedia. Oleh karena itu, tidak ada batasan nyata untuk jumlah komponen, pin-pin komponen, koneksi, atau lembar kerja. Untuk diagram dengan banyak lembar kerja, Eeschema akan menampilkannya secara hirarkis.

Eeschema bisa menggunakan diagram dengan banyak lembar kerja dalam beberapa cara:

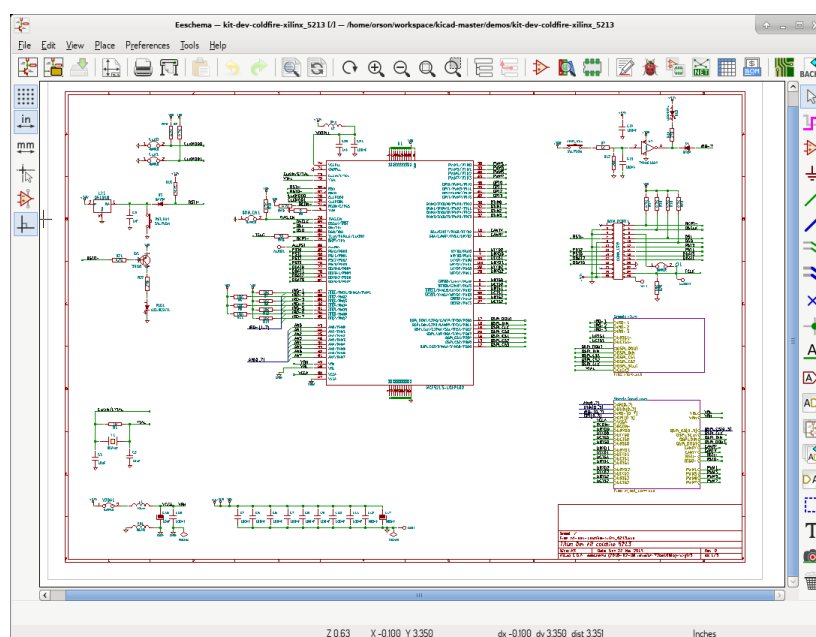
- Hirarki sederhana (setiap skematik hanya digunakan satu kali saja).
  - Hirarki kompleks (beberapa skematik digunakan lebih dari sekali dan dapat dimunculkan beberapa kali).
  - Hirarki datar (skematik tidak terhubung secara eksplisit di dalam diagram utama).
-

## Bab 2

# Perintah-perintah Umum Eeschema

Perintah-perintah yang ada di Eeschema dapat dijalankan dengan cara:

- Klik pada bilah menu (di bagian atas layar).
- Klik pada ikon di bagian atas layar (untuk perintah-perintah umum).
- Klik pada ikon di bagian kanan layar (untuk perintah atau "peralatan" khusus).
- Klik pada ikon di bagian kiri layar (untuk berbagai opsi tampilan).
- Menekan tombol tetikus (untuk perintah-perintah tambahan yang penting). Secara khusus, klik kanan pada tetikus akan membuka sebuah menu kontekstual untuk elemen yang berada di bawah kursor (Perbesar atau perkecil, pengaturan grid, dan pengeditan elemen).
- Tombol fungsi (tombol F1, F2, F3, F4, Insert dan Spasi). Secara spesifik, tombol **Escape** berfungsi untuk membatalkan perintah yang sedang dijalankan. Tombol **Insert** akan menduplikasi elemen terakhir yang telah dibuat.
- Menekan tombol *hotkey*. Tombol *hotkey* akan menjalankan perintah pemilihan peralatan, dan langsung menjalankan peralatan yang dipilih tersebut pada lokasi kursor saat ini. Daftar tombol *hotkey* yang tersedia dapat dilihat di menu *Help* → *List Hotkeys* atau dengan menekan tombol **Ctrl+F1**.



## 2.1 Perintah-perintah Tetikus

### 2.1.1 Perintah-perintah Dasar

#### Tombol kiri

- Klik tunggal: menampilkan karakteristik simbol atau teks yang berada di bawah kursor. Tampilan ini dapat dilihat di bilah status.
- Klik ganda: mengedit suatu simbol atau teks (jika elemennya bisa diedit).

#### Tombol kanan

- Membuka sebuah menu *pop-up*.

### 2.1.2 Pekerjaan Blok

Anda dapat memindah, menggeser, menyalin, dan menghapus area yang telah dipilih pada seluruh menu Eeschema.

Suatu area dapat dipilih dengan cara menggambar sebuah kotak di sekitarnya dengan menggunakan tombol kiri tetikus.

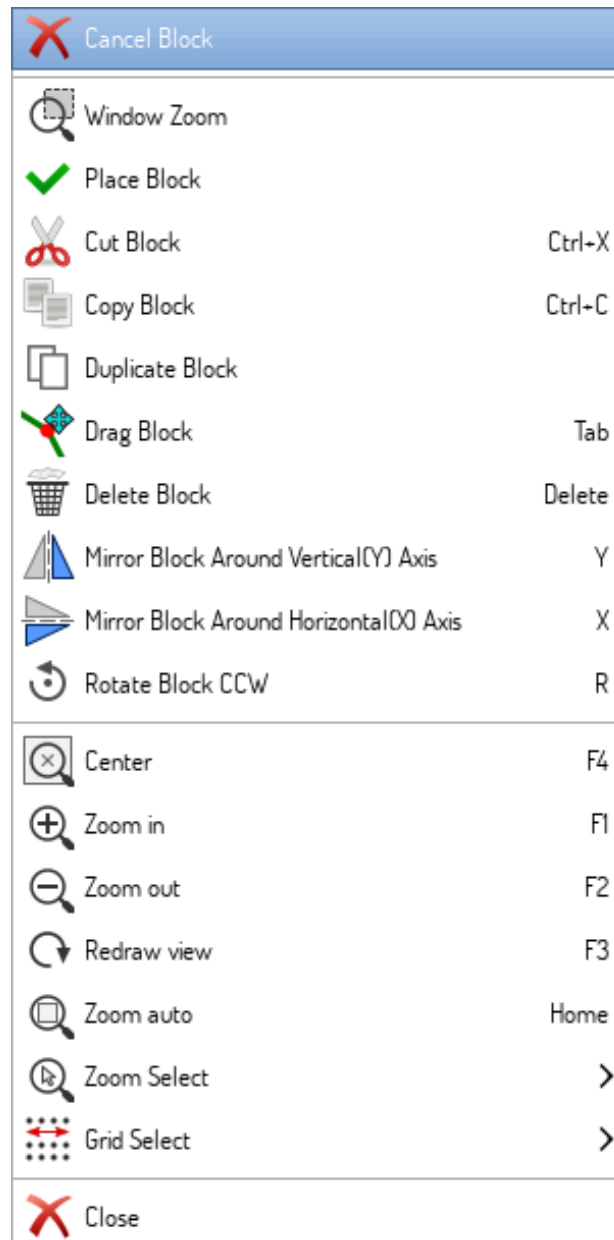
Holding “Shift”, “Ctrl”, or “Shift + Ctrl” during selection respectively performs copying, dragging and deletion:

Tombol kiri tetikus	Memindahkan elemen yang dipilih.
Shift + tombol kiri tetikus	Menyalin elemen yang dipilih.
Ctrl + tombol kiri tetikus	Menggeser elemen yang dipilih.
Ctrl + Shift + tombol kiri tetikus	Menghapus elemen yang dipilih.

Saat menggeser atau menyalin, Anda dapat melakukan:

- Klik sekali lagi untuk meletakkan elemen.
- Klik tombol kanan atau tekan tombol **Escape** untuk membatalkan perintah.

Jika perintah pemindahan blok sudah mulai dijalankan, Anda juga bisa memilih perintah lain untuk dijalankan dengan cara melakukan klik-kanan. Akan muncul sebuah menu *pop-up* berisi berbagai perintah yang bisa dijalankan.



## 2.2 Tombol *Hotkey*

- Tombol **Ctrl+F1** akan menampilkan daftar *hotkey* yang tersedia.
- Pengaturan tombol *hotkey* bisa dilakukan di kotak dialog *Schematic Editor Options*, pada tab *Controls*. Untuk mengaksesnya, buka menu *Preferences* → *General Options*.

Berikut ini adalah daftar tombol *hotkey* yang tersedia:

Bantuan	Ctrl+F1
Perbesar	F1
Perkecil	F2
Gambar Ulang Tampilan	F3
Tengahkan	F4
Tampilan Muat Satu Layar	Home
Perbesar/perkecil ke Pilihan	@

Reset Koordinat Lokal	Spasi
Edit Item	E
Hapus Item	Del
Putar Item	R
Geser Item	G
Urungkan ( <i>Undo</i> )	Ctrl+Z
Ulangi ( <i>Redo</i> )	Ctrl+Y
Klik Kiri Tetikus	Return
Klik Kiri Ganda Tetikus	End
Simpan Skematik	Ctrl+S
Muat Skematik	Ctrl+O
Cari Item	Ctrl+F
Cari Item Berikutnya	F5
Cari Penanda DRC Berikutnya	Shift+F5
Cari dan Ganti	Ctrl+Alt+F
Ulangi Item Terakhir	Ins
Pindahkan Blok → Geser Blok	Tab
Salin Blok	Ctrl+C
Rekat Blok	Ctrl+V
Potong Blok	Ctrl+X
Pindahkan Item Skematik	M
Gandakan Simbol atau Label	C
Tambahkan Simbol	A
Tambahkan <i>Power</i>	P
Cerminkan X	X
Cerminkan Y	Y
Hadapkan Simbol ke Orientasi Normal	N
Edit Nilai Simbol	V
Edit Referensi Simbol	U
Edit <i>Footprint</i> Simbol	F
Edit dengan <i>Symbol Editor</i>	Ctrl+E
Mulai <i>Wire</i>	W
Mulai <i>Bus</i>	B
Akhiri Garis <i>Wire Bus</i>	K
Tambahkan Label	L
Tambahkan Label Hirarkis	H
Tambahkan Label Global	Ctrl+L
Tambahkan <i>Junction</i>	J
Tambahkan Penanda "Tak Hubung"	Q
Tambahkan Lembar Kerja	S
Tambahkan Jalur Masuk <i>Wire</i>	Z
Tambahkan Jalur Masuk <i>Bus</i>	/
Tambahkan Garis Poligon	I
Tambahkan Teks	T
Perbarui PCB dari Skematik	F8
Mengisi Atribut Otomatis	O
Tinggalkan Lembar Kerja	Alt+BkSp
Hapus <i>Node</i>	BkSp
Sorot Koneksi	Ctrl+X

Semua tombol *hotkey* dapat diatur ulang dengan menggunakan editor *hotkey* (melalui menu *Preferences* → *General Options* → *Controls*).

Kita juga bisa melakukan impor/ekspor pengaturan *hotkey* melalui menu *Preferences* → *Import and Export* → *Import/Export Hotkeys*.

## 2.3 Grid

Di Eeschema, kursor selalu bergerak di atas grid. Grid dapat disesuaikan menurut keinginan Anda:

- Ukuran grid dapat diubah melalui menu *pop-up* atau menggunakan menu *Preferences/Options*.
- Warna grid dapat diubah melalui tab *Colors* pada kotak dialog *Schematic Editor Options* (dapat diakses melalui menu *Preferences* → *General Options*).
- Grid dapat ditampilkan atau disembunyikan dengan menggunakan tombol di bilah alat sebelah kiri.

Ukuran grid *default* adalah 50 mil (0.050") atau 1,27 milimeter.

Ukuran tersebut adalah ukuran yang direkomendasikan untuk meletakkan simbol dan *wire* di skematik, dan untuk meletakkan pin pada saat mendesain sebuah simbol di *Symbol Editor*.

Kita juga bisa menggunakan ukuran grid yang lebih kecil, dari 25 mil hingga 10 mil. Ukuran ini hanya dimaksudkan untuk digunakan pada saat mendesain bentuk simbol atau meletakkan teks dan komentar, dan tidak direkomendasikan untuk digunakan pada saat meletakkan pin dan *wire*.

## 2.4 Pilihan Perbesaran/perkecilan

Untuk mengubah tingkat perbesaran/perkecilan:

- Klik kanan untuk membuka menu *pop-up* dan pilih tingkat perbesaran/perkecilan yang diinginkan.
- Atau gunakan tombol fungsi:
  - F1: Perbesar
  - F2: Perkecil
  - F4 atau klik tombol tengah tetikus (tanpa menggerakkan tetikus): Tengahkan tampilan di sekitar posisi penunjuk kursor
- Perbesar/perkecil Jendela:
  - Roda tetikus: Perbesar/perkecil
  - Shift + Roda tetikus: Geser tampilan ke atas/bawah
  - Ctrl + Roda tetikus: Geser tampilan ke kiri/kanan

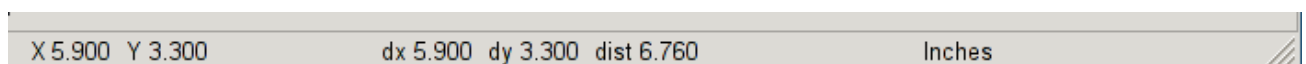
## 2.5 Menampilkan Koordinat Kursor

Satuan tampilan adalah dalam inci atau milimeter. Akan tetapi, Eeschema selalu menggunakan 0.001 inci (mil/thou) sebagai satuan internalnya.

Informasi berikut ini ditampilkan pada bagian kanan bawah jendela:

- Faktor perbesaran/perkecilan
- Posisi absolut kursor
- Posisi relatif kursor

Koordinat relatif dapat direset menjadi nol dengan menekan **Spasi**. Hal ini berguna untuk mengukur jarak antara dua titik atau untuk meluruskan suatu obyek.





## 2.6 Bilah Menu Atas

bilah menu atas digunakan untuk membuka dan menyimpan skematik, konfigurasi program, dan untuk melihat dokumentasi.

File Edit View Place Preferences Tools Help

## 2.7 Bilah Alat Bagian Atas







Fungsi-fungsi utama Eeschema dapat diakses melalui bilah alat atas.

Jika Eeschema dijalankan secara mandiri, berikut ini adalah set peralatan yang tersedia:



Jika KiCad dijalankan pada mode proyek, maka dua ikon pertama di atas tidak tersedia, karena ikon tersebut hanya bekerja dengan berkas-berkas individual.
















	Membuat skematik baru (hanya pada mode mandiri).
	Membuka skematik (hanya pada mode mandiri).
	Simpan keseluruhan proyek skematik.
	Pilih ukuran lembar kerja dan edit kotak judul pada lembar kerja.
	Buka kotak dialog pencetak.
	Rekatkan item atau blok yang telah disalin/dipotong ke lembar kerja yang sedang dibuka.
	<i>Undo</i> : Batalkan perubahan terakhir.
	<i>Redo</i> : Batalkan perintah <i>undo</i> terakhir.
	Tampilkan kotak dialog pencarian simbol dan teks pada skematik.
	Tampilkan kotak dialog cari dan ganti teks pada skematik.
	Muat ulang tampilan; tampilan muat satu layar.
	Perbesar dan perkecil.
	Lihat dan jelajahi pohon hirarki.
	Tinggalkan lembar kerja saat ini dan naik di hirarki.
	Jalankan <i>Symbol Library Editor</i> untuk melihat dan mengedit pustaka dan simbol.
	Jelajahi pustaka-pustaka simbol.
	Tambahkan anotasi simbol.
	<i>Electrical Rules Checker</i> (ERC), validasi koneksi-koneksi elektrik secara otomatis.






	Jalankan CvPcb untuk menerapkan <i>footprint</i> ke simbol.
	Ekspor <i>netlist</i> (Pcbnew, Spice, dan berbagai format lainnya).
	Mengedit atribut-atribut simbol.
	Membuat daftar komponen ( <i>Bill of Materials/BOM</i> ).
	Menjalankan Pcbnew untuk membuat tata letak PCB.
	Mengimpor-balik penerapan <i>footprint</i> (yang telah dipilih melalui CvPcb atau Pcbnew) ke atribut " <i>footprint</i> ".

## 2.8 Bilah Alat Bagian Kanan

Bilah alat ini berisi berbagai alat untuk:







- Meletakkan simbol, *wire*, *bus*, *junction*, label, teks, dsb.
- Membuat lembar kerja hirarkis dan simbol-simbol koneksi.

	Membatalkan perintah atau alat yang sedang aktif.
	Menyorot sebuah <i>net</i> dengan menandai <i>wire</i> dan label <i>net</i> dengan warna yang berbeda. Jika KiCad dijalankan dalam mode proyek, maka jalur tembaga yang terkait dengan <i>net</i> yang dipilih juga akan disorot di Pcbnew.
	Menampilkan kotak dialog pemilihan simbol untuk memilih simbol yang akan diletakkan di lembar kerja.
	Menampilkan kotak dialog pemilihan simbol <i>power</i> untuk memilih simbol <i>power</i> yang akan diletakkan di lembar kerja.
	Menggambar <i>wire</i> .
	Menggambar <i>bus</i> .
	Menggambar jalur masuk <i>wire-ke-bus</i> . Elemen ini hanya berupa tampilan grafis saja dan tidak akan menghubungkan apapun. Oleh karena itu elemen ini tidak bisa digunakan untuk membuat koneksi antar <i>wire</i> .
	Menggambar jalur masuk <i>bus-ke-bus</i> .
	Meletakkan penanda "Tak Hubung". Penanda tersebut harus diletakkan pada pin-pin simbol yang memang akan dibiarkan tidak terhubung. Tanda ini dipasang sebagai pemberitahuan bagi <i>Electrical Rules Checker</i> (ERC) bahwa tidak terhubungnya pin tersebut memang disengaja, sehingga tidak akan dilaporkan sebagai suatu kesalahan.
	Meletakkan <i>junction</i> untuk menghubungkan dua buah <i>wire</i> yang saling bersilang, atau menghubungkan <i>wire</i> ke pin, jika koneksi antara <i>wire</i> dan pin tersebut meragukan, apakah ujung <i>wire</i> atau pin tersebut tidak secara langsung terhubung ke ujung <i>wire</i> yang lain.
	Meletakkan sebuah label lokal. Label lokal menghubungkan item-item yang berada <b>pada lembar kerja yang sama</b> . Untuk koneksi antar dua lembar kerja yang berbeda, Anda harus menggunakan label global atau hirarkis.
	Meletakkan sebuah label global. Semua label global dengan nama yang sama akan saling terhubung, walaupun berada di lembar kerja yang berbeda.
	Meletakkan label hirarkis. Label hirarkis digunakan untuk membuat sebuah koneksi antara anak lembar kerja dengan lembar kerja utamanya.
	Meletakkan anak lembar kerja hirarkis. Anda harus menentukan nama berkas dari anak lembar kerja ini.
	Mengimpor sebuah pin hirarkis dari anak lembar kerja. Perintah ini dapat dijalankan hanya pada anak lembar kerja hirarkis. Perintah ini akan membuat pin-pin hirarkis sesuai dengan label hirarkis yang diletakkan pada anak lembar kerja yang dituju.

	Meletakkan sebuah pin hirarkis pada anak lembar kerja. Perintah ini hanya dapat dijalankan pada anak lembar kerja hirarkis. Perintah ini akan membuat pin-pin hirarkis sesuai keinginan kita, meskipun pin tersebut tidak ada pada anak lembar kerja yang dituju.
	Menggambar garis. Garis tersebut hanya berupa tampilan grafis saja dan tidak akan menghubungkan apapun.
	Meletakkan sebuah komentar teks.
	Meletakkan sebuah gambar bitmap.
	Menghapus elemen yang dipilih.

## 2.9 Bilah Alat Sebelah Kiri

Bilah alat ini berfungsi untuk melakukan pengaturan tampilan:

	Menampilkan atau menyembunyikan grid.
	Gunakan satuan inci.
	Gunakan satuan milimeter.
	Memilih bentuk kursor (layar penuh atau kecil).
	Menampilkan atau menyembunyikan pin-pin "tak tampak".
	Memilih cara peletakan <i>wire</i> dan <i>bus</i> , secara sudut bebas atau 90 derajat.

## 2.10 Menu *Pop-up* dan Pengeditan Cepat

Klik kanan tetikus akan membuka sebuah menu kontekstual untuk elemen yang dipilih. Menu tersebut berisi:

- Faktor perbesaran/perkecilan.
- Pengaturan grid.
- Parameter-parameter yang sering diedit untuk elemen yang dipilih.

Menu *pop-up* tanpa ada elemen yang dipilih.

 Center

 Zoom in

 Zoom out

 Redraw view

 Zoom auto

 Zoom Select

 Grid Select

 Close

F4

F1

F2

F3

Home

>

>

Mengedit label.

 Move

 Drag

 Duplicate

 Rotate Counterclockwise

 Edit

 Delete

 Change Type

 Center

 Zoom in

 Zoom out

 Redraw view

 Zoom auto

 Zoom Select

 Grid Select

 Close

M

G

C

R

E

Delete

>

F4

F1


F2


F3


Home

>

>

 Change to Hierarchical Label

 Change to Text

 Change to Global Label

 VRH

 VRL

 VCCA

 GND

49

48

50

47

VF

VF

VK

VS



Bab 3

Menu Utama

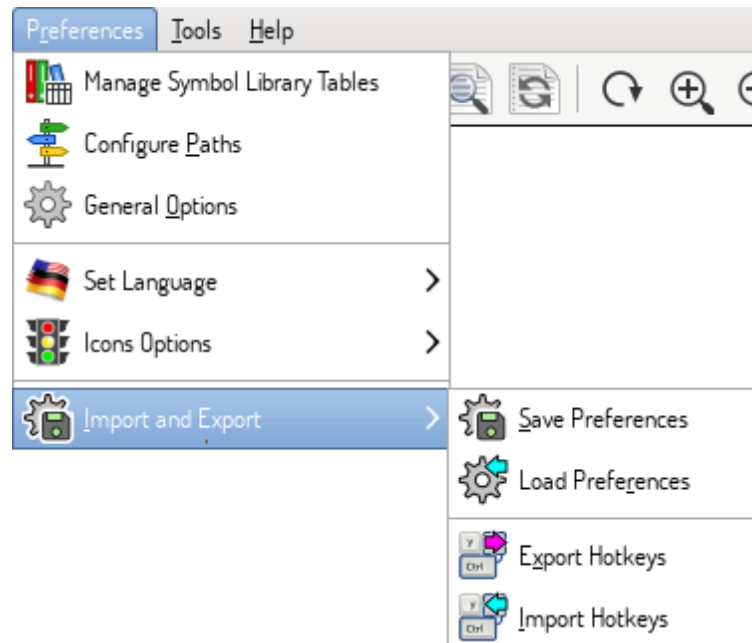
3.1 Menu *File*



New	Menutup skematik yang sedang dibuka dan membuat skematik baru (hanya dalam mode mandiri).
Open	Membuka sebuah proyek skematik (hanya dalam mode mandiri).
Open Recent	Membuka sebuah proyek skematik dari daftar berkas yang baru saja dibuka (hanya dalam mode mandiri).
Append Schematic Sheet	Memasukkan isi dari lembar kerja lain ke dalam lembar kerja yang sedang dibuka.
Import Non-Kicad Schematic File	Mengimpor sebuah proyek skematik dalam format berkas selain KiCad.
Save	Menyimpan lembar kerja yang sedang dibuka dan semua anak lembar kerjanya.
Save Current Sheet	Menyimpan hanya lembar kerja yang sedang dibuka saja, dan tidak menyimpan lembar kerja lain yang ada di dalam proyek tersebut.
Save Current Sheet As...	Menyimpan dengan nama lain lembar kerja yang sedang dibuka.
Page Settings	Mengatur ukuran halaman dan blok judul pada lembar kerja.
Print	Mencetak proyek skematik (Lihat juga bagian <a href="#">Plot dan Cetak</a> ).

Plot	Mengekspor ke format PDF, PostScript, HPGL atau SVG (Lihat bagian <a href="#">Plot dan Cetak</a> ).
Close	Menutup aplikasi.

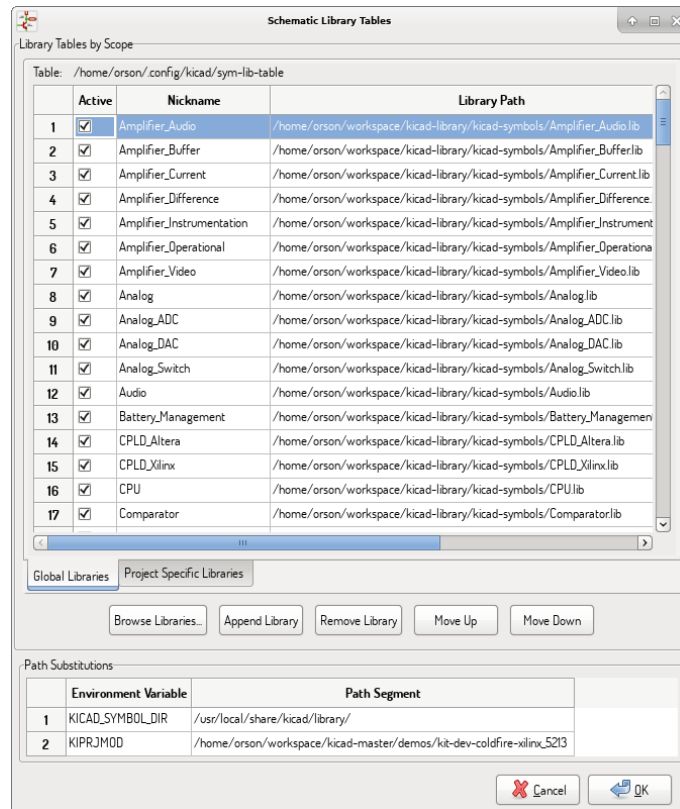
### 3.2 Menu *Preferences*



Manage Symbol Library Tables	Menambah/menghapus pustaka simbol.
Configure Paths	Mengatur <i>path</i> pencarian <i>default</i> .
General Options	Preferensi (satuan, ukuran grid, nama-nama atribut, dsb.).
Set Language	Memilih bahasa untuk antarmuka program.
Icons Options	Pengaturan tampilan ikon.
Import and Export	Pengaturan transfer dari/ke berkas.



### 3.2.1 Mengelola Tabel Pustaka Simbol



Eeschema menggunakan dua tabel pustaka untuk menyimpan daftar pustaka simbol yang tersedia, yang dibedakan berdasarkan cakupannya:

- Pustaka Global

Pustaka-pustaka yang berada di dalam tabel Pustaka Global (*Global Libraries*) tersedia untuk semua proyek. Berkas pustaka ini disimpan di **sym-lib-table** di dalam direktori rumah Anda (*path* yang tepat bergantung pada sistem operasi yang Anda gunakan; pastikan dari *path* yang ditampilkan di atas tabel).

- Pustaka Spesifik untuk Proyek

Pustaka-pustaka yang berada di dalam tabel Pustaka Spesifik untuk Proyek (*Project Specific Libraries*) tersedia hanya untuk proyek yang saat ini sedang dibuka. Berkas pustaka ini disimpan di **sym-lib-table** di dalam direktori proyek (pastikan dari *path* yang ditampilkan di atas tabel).

Anda bisa melihat kedua daftar tersebut dengan melakukan klik pada tab *Global Libraries* atau *Project Specific Libraries* di bawah tabel pustaka.

#### 3.2.1.1 Menambahkan Pustaka Baru

Tambahkan pustaka dengan cara klik pada tombol **Browse Libraries...** dan memilih berkas yang akan ditambahkan, atau klik pada **Append Library** dan mengetikkan *path* menuju berkas pustaka. Pustaka yang dipilih akan ditambahkan ke tabel pustaka yang sedang dibuka (pustaka global atau spesifik untuk proyek).

#### 3.2.1.2 Menghapus Pustaka

Hapus pustaka dengan memilih satu atau lebih pustaka dan klik tombol **Remove Library**.

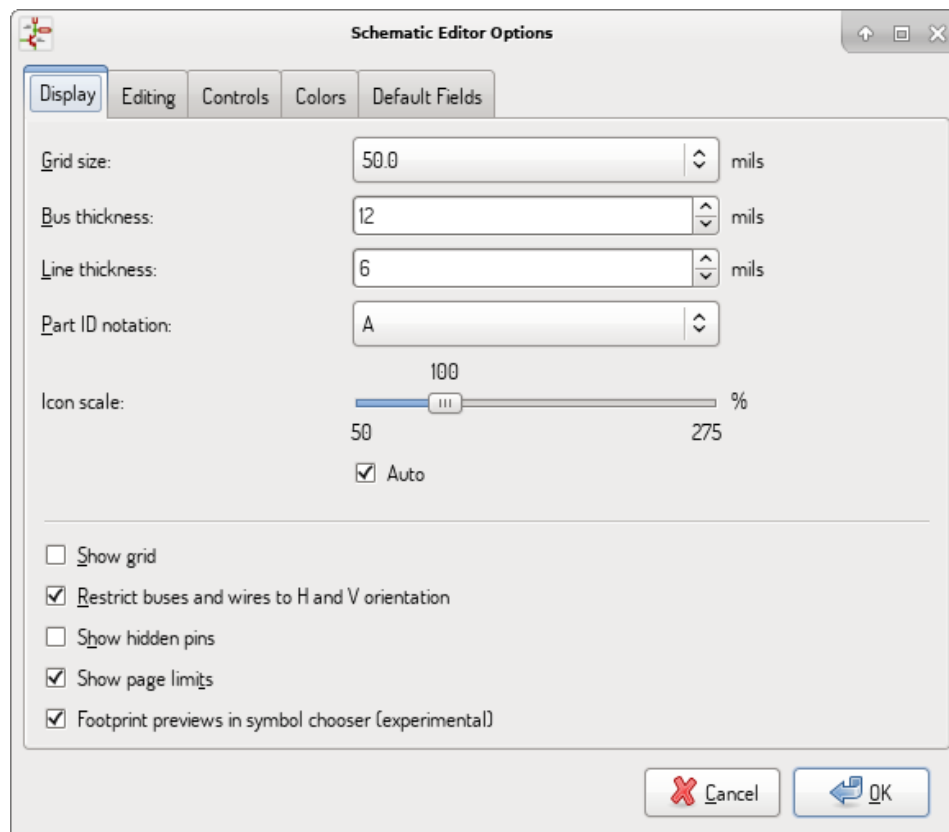
### 3.2.1.3 Properti Pustaka

Setiap baris dari tabel pustaka berisi beberapa atribut untuk suatu pustaka:

Active	Mengaktifkan atau menonaktifkan pustaka. Berguna untuk secara sementara mengurangi jumlah set pustaka yang dimuat.
Nickname	Tanda pengenal yang singkat dan unik untuk menetapkan simbol ke komponen. Simbol direpresentasikan oleh <i>string</i> <code>&lt;Library Nickname&gt;:&lt;Symbol Name&gt;</code> .
Library Path	<i>Path</i> menuju lokasi pustaka.
Plugin Type	Menunjukkan format berkas pustaka.
Options	Menyimpan opsi-opsi spesifik suatu pustaka, jika digunakan oleh <i>plugin</i> .
Description	Secara singkat menjelaskan isi pustaka.

## 3.2.2 Menu General Options

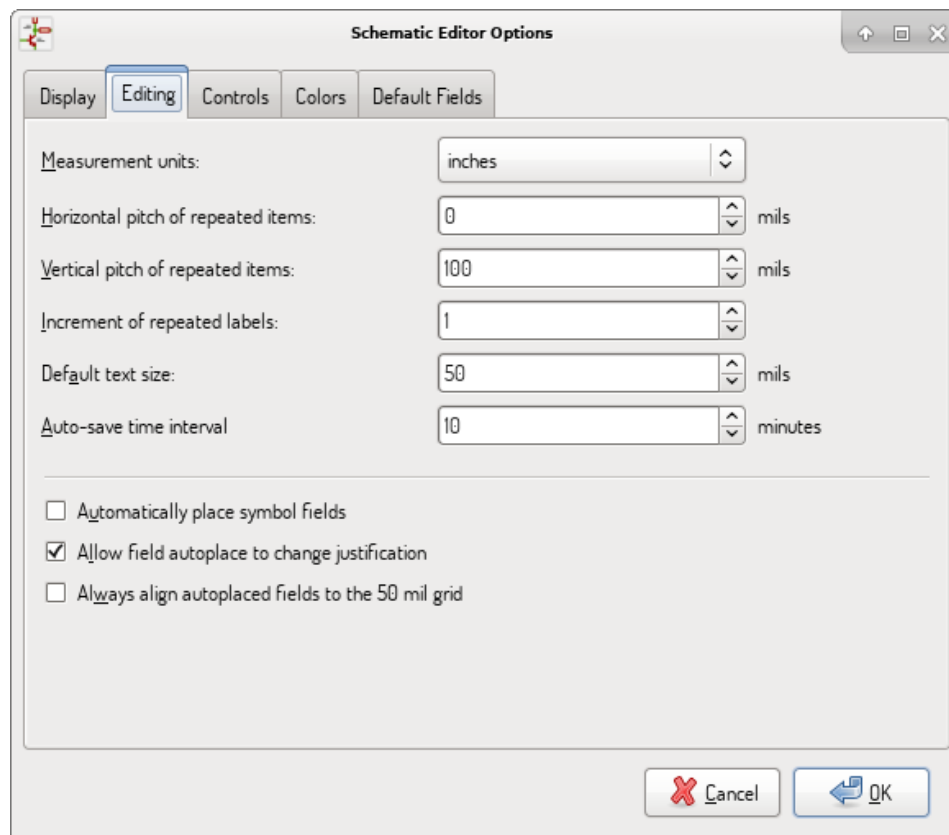
### 3.2.2.1 Tab Display



Grid Size	Pemilihan ukuran grid. <b>Direkomendasikan</b> untuk menggunakan ukuran grid normal (0.050 inci atau 1,27 mm) pada saat bekerja dengan Eeschema. Ukuran grid yang lebih kecil bisa digunakan untuk pembuatan komponen.
Bus thickness	Ukuran pena yang digunakan untuk menggambar <i>bus</i> .
Line thickness	Ukuran pena yang digunakan untuk menggambar suatu obyek yang tidak memiliki ukuran khusus.
Part ID notation	Gaya atau sufiks yang digunakan untuk menandakan unit simbol (U1A, U1.A, U1-1, dsb.)
Icon scale	Pengaturan ukuran ikon pada bilah alat.
Show Grid	Menampilkan atau menyembunyikan grid.

Restrict buses and wires to H and V orientation	Jika dipilih, <i>bus</i> dan <i>wire</i> akan digambar hanya dengan garis vertikal atau horisontal. Jika tidak dipilih, maka <i>bus</i> dan <i>wire</i> dapat diletakkan dengan orientasi apapun.
Show hidden pins:	Menampilkan pin-pin yang tidak tampak (atau <i>disembunyikan</i> ). Umumnya adalah pin-pin <i>power</i> .
Show page limits	Jika dipilih, menunjukkan batas halaman di layar.
Footprint previews in symbol chooser	Menampilkan pratinjau bingkai <i>footprint</i> dan pemilih <i>footprint</i> ketika meletakkan sebuah simbol. <b>Catatan:</b> hal ini mungkin akan menyebabkan masalah atau menjadi lambat, gunakan dengan mempertimbangkan resiko tersebut.

### 3.2.2.2 Tab Editing

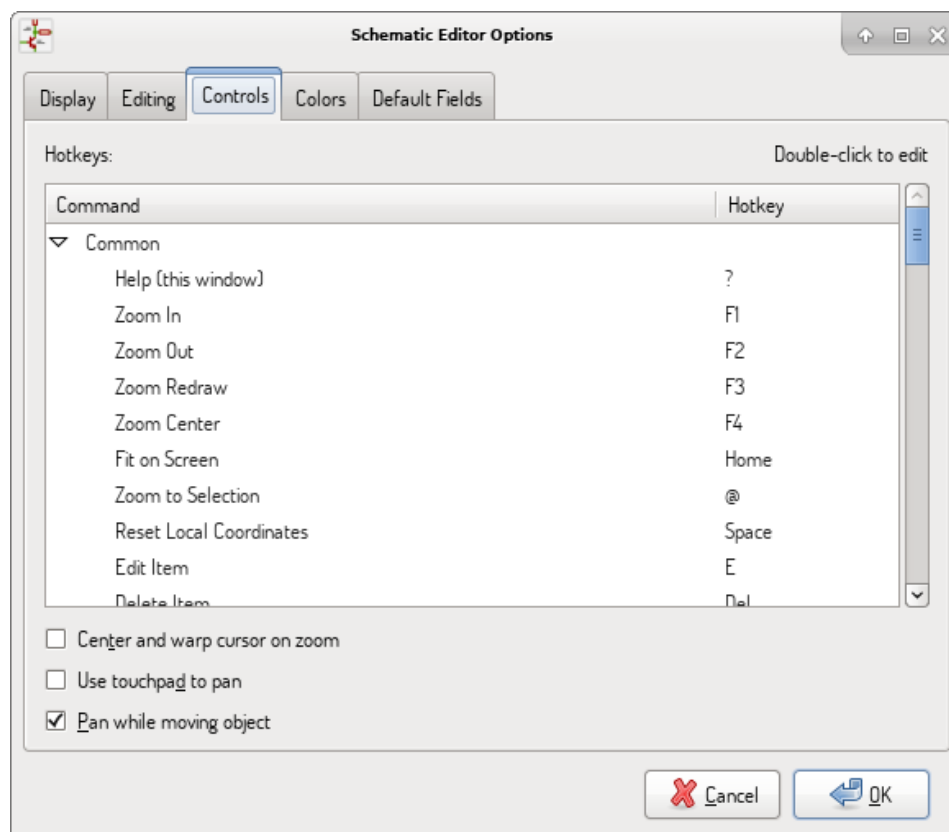


Measurement units	Memilih satuan yang digunakan untuk tampilan dan koordinat kursor (inci atau milimeter).
Horizontal pitch of repeated items	Penambahan nilai aksis X ketika melakukan duplikasi elemen ( <i>default</i> : 0) (setelah meletakkan sebuah item seperti simbol, label atau <i>wire</i> , duplikasi dapat dilakukan dengan tombol <i>Insert</i> )
Vertical pitch of repeated items	Penambahan nilai aksis Y ketika melakukan duplikasi elemen ( <i>default</i> : 0.100 inci atau 2,54 mm).
Increment of repeated labels	Penambahan nilai label ketika melakukan duplikasi teks yang diakhiri dengan angka, misalnya anggota <i>bus</i> (umumnya bernilai 1 atau -1).
Default text size	Ukuran teks yang digunakan pada saat membuat teks atau label baru.
Auto-save time interval	Waktu antara penyimpanan berkas cadangan (dalam menit).
Automatically place symbol fields	Jika dicentang, atribut-atribut simbol (misalnya nilai dan referensi) pada simbol yang sedang diletakkan mungkin akan dipindah posisinya agar tidak bertumpuk dengan item-item lain.

Allow field autoplace to change justification	Kelanjutan dari opsi <i>Automatically place symbol fields</i> . Mengaktifkan pengaturan justifikasi teks untuk atribut-atribut simbol pada saat meletakkan suatu item.
Always align autoplaced fields to the 50 mil grid	Kelanjutan dari opsi <i>Automatically place symbol fields</i> . Jika dicentang, atribut akan diletakkan secara otomatis dengan menggunakan ukuran grid 50 mil. Jika tidak dicentang, atribut tersebut akan diletakkan secara bebas.

### 3.2.2.3 Tab Controls

Mengatur *hotkey* dan perilaku antarmuka pengguna.



Anda bisa mengubah tombol *hotkey* dengan cara klik-ganda pada aksi yang diinginkan. Klik-kanan pada salah satu aksi tersebut akan menampilkan menu *pop-up* berikut ini:

Edit	Mengubah tombol <i>hotkey</i> untuk suatu aksi (sama seperti melakukan klik-ganda).
Undo Changes	Membatalkan perubahan <i>hotkey</i> yang baru saja diterapkan.
Restore Default	Mengembalikan tombol <i>hotkey</i> untuk suatu aksi ke pengaturan <i>default</i> .
Undo All Changes	Membatalkan semua perubahan <i>hotkey</i> yang baru saja diterapkan.
Restore All to Default	Mengembalikan keseluruhan <i>hotkey</i> ke pengaturan <i>default</i> .

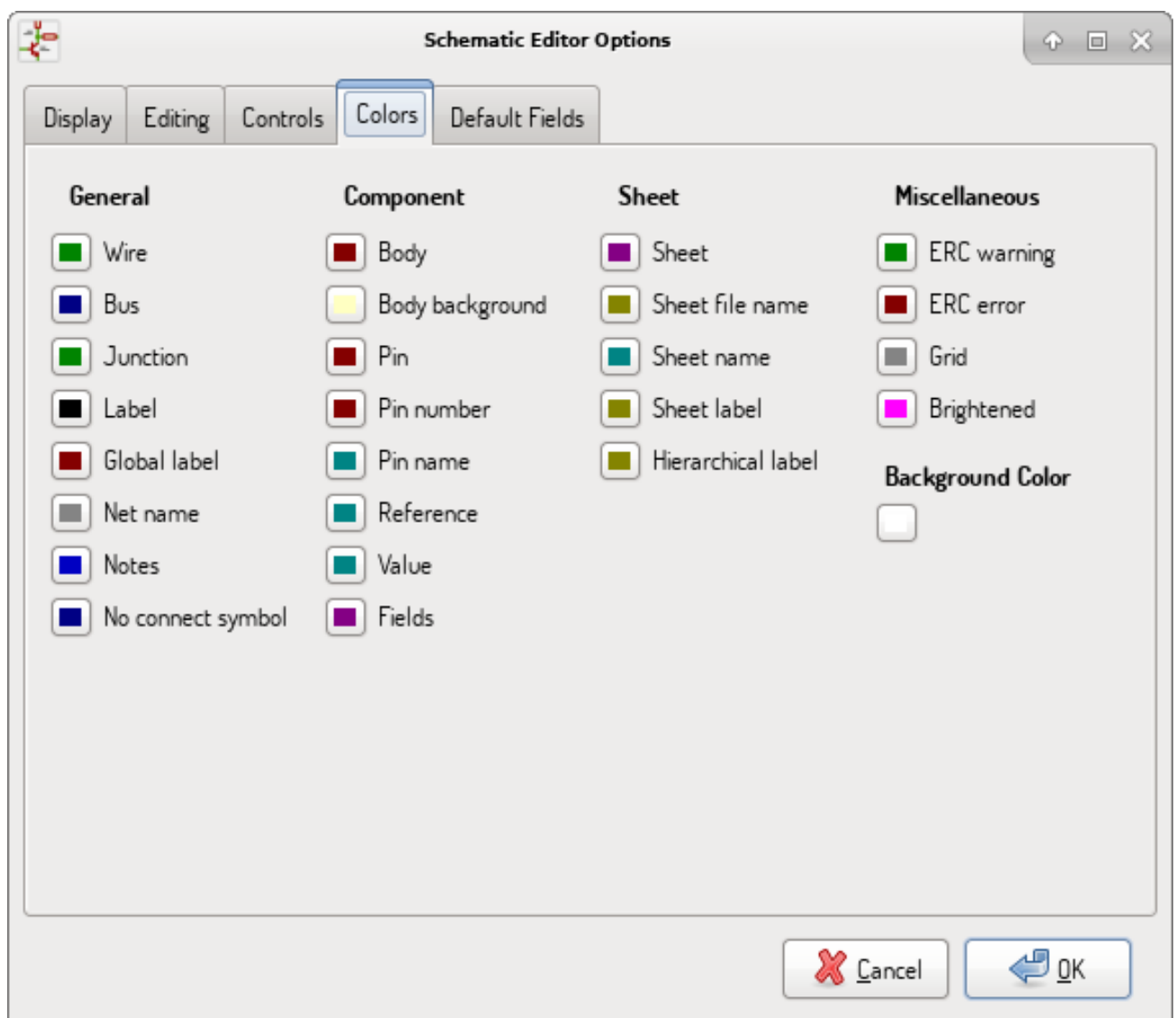
Disediakan pula beberapa pilihan sebagai berikut:

Center and warp cursor on zoom	Jika dicentang, lokasi yang ditunjuk oleh kursor akan diposisikan ke tengah layar pada saat melakukan perbesaran/perkecilan.
--------------------------------	--

Use touchpad to pan	Jika diaktifkan, tampilan layar akan digeser jika kita memutar roda tetikus (atau menggunakan gestur <i>touchpad</i> ), dan untuk melakukan perbesaran/perkecilan, kita harus menekan <i>Ctrl</i> sambil memutar roda tetikus. Jika opsi ini tidak diaktifkan, roda tetikus digunakan untuk melakukan perbesaran/perkecilan tampilan, dan untuk menggeser tampilan layar, gunakan tombol <i>Ctrl</i> atau <i>Shift</i> .
Pan while moving object	Jika dicentang, tampilan layar akan digeser secara otomatis jika kursor keluar dari tampilan jendela pada saat menggambar atau memindahkan suatu obyek.

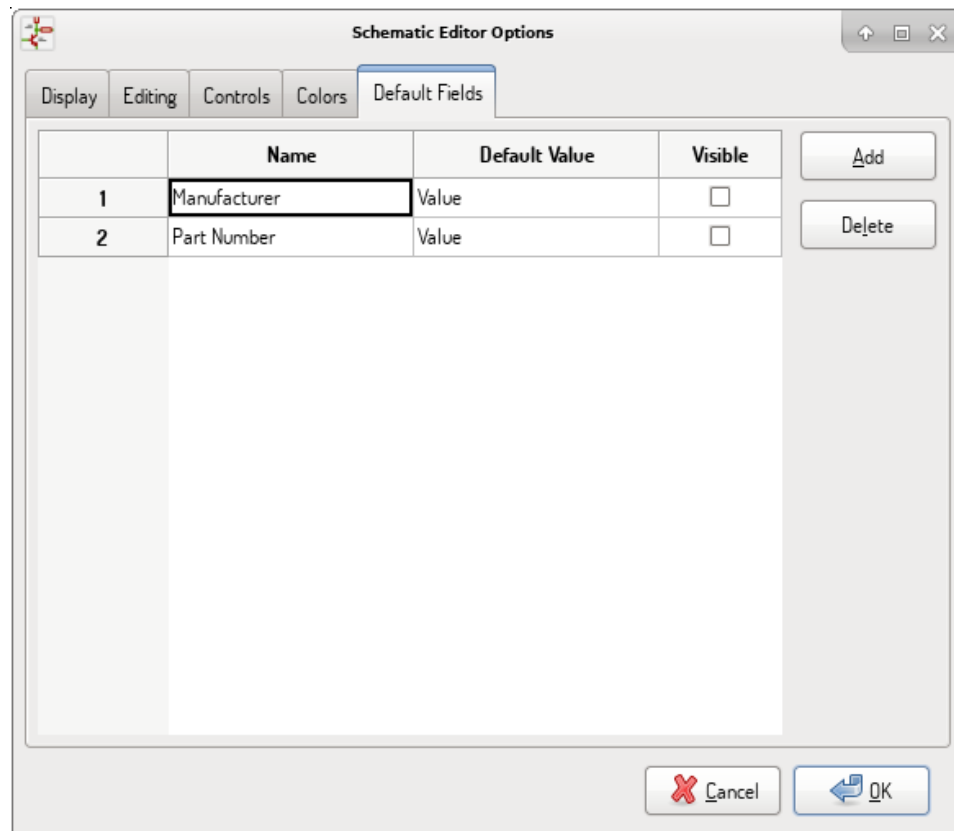
### 3.2.2.4 Tab Colors

Tab ini berisi skema warna untuk berbagai elemen grafis. Klik pada kotak warna untuk memilih warna baru untuk elemen yang diinginkan.



### 3.2.2.5 Tab Default Fields

Tab ini digunakan untuk membuat atribut-atribut tambahan yang akan muncul pada simbol yang baru diletakkan.



### 3.3 Menu *Help*


Akses ke bantuan *on-line* (dokumen ini) yang berisi tutorial KiCad.

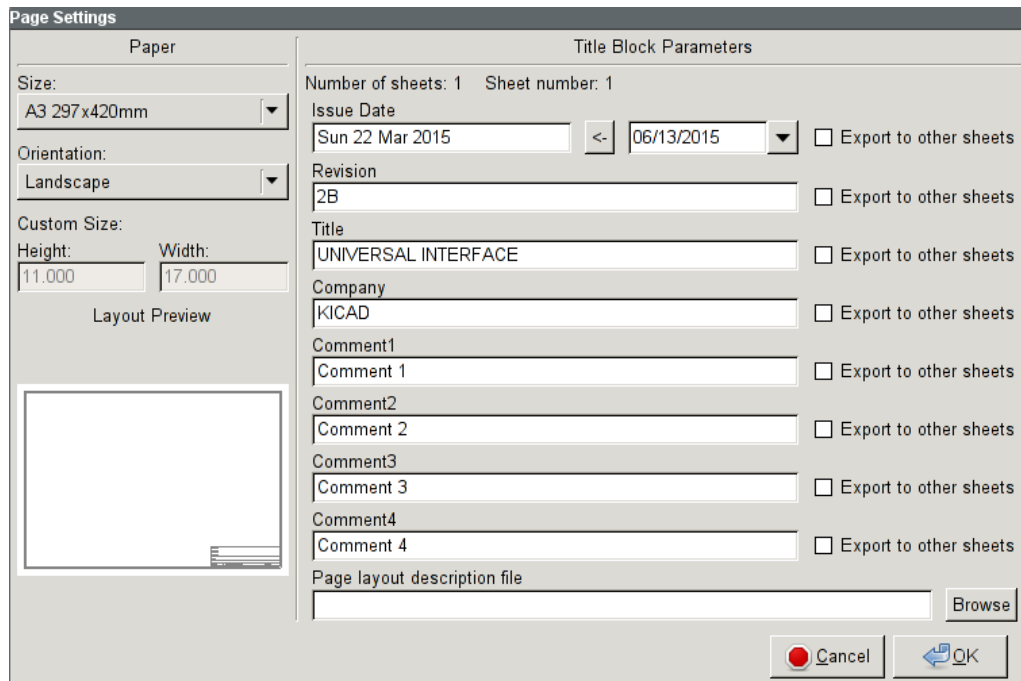
Jika Anda ingin mengirimkan laporan kutu, gunakan tombol “\_Copy Version Information\_” untuk mengidentifikasi sistem yang Anda gunakan.

## Bab 4

# Bilah Atas Umum


### 4.1 Manajemen Lembar Kerja

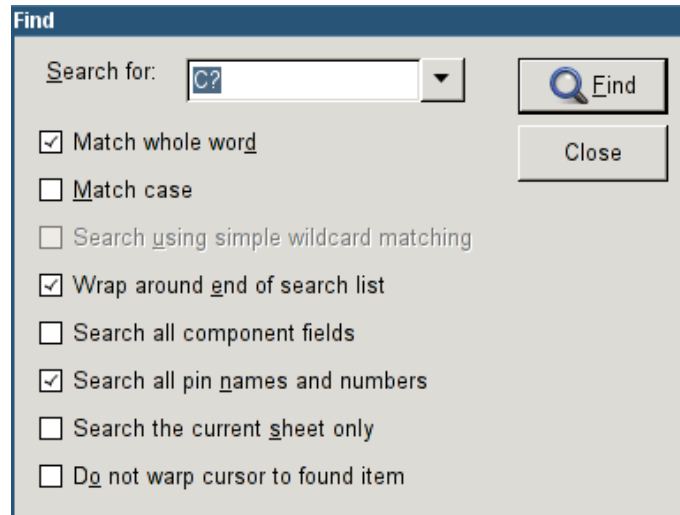
Ikon *Sheet Settings* () digunakan untuk melakukan pengaturan ukuran lembar kerja dan isi blok judul.



Penomoran lembar kerja diperbarui secara otomatis. Anda dapat mengatur tanggal agar menampilkan tanggal saat ini dengan menekan tombol panah ke kiri pada kotak "Issue Date", tetapi tanggal ini tidak akan berubah secara otomatis.

### 4.2 Alat Pencari


Ikon *Find* () digunakan untuk mengakses alat pencari.



Anda dapat melakukan pencarian referensi, nilai atau teks pada lembar kerja yang sedang dibuka atau pada keseluruhan hirarki. Ketika elemen yang dicari telah ditemukan, kursor akan diposisikan pada elemen tersebut di dalam anak lembar kerja yang sesuai.

### 4.3 Alat *Netlist*



Ikon *Netlist* (  ) digunakan untuk membuka peralatan pembuat *netlist*.

Alat ini akan membuat sebuah berkas yang berisi penjelasan koneksi pada keseluruhan hirarki.

Pada hirarki dengan banyak lembar kerja, label lokal hanya akan ditampilkan di dalam lembar kerja dimana ia berada. Sebagai contoh, label LABEL1 pada lembar kerja 3 berbeda dengan LABEL1 pada lembar kerja 5 (jika tidak ada koneksi yang dibuat untuk menghubungkan kedua label tersebut). Hal ini karena nama lembar kerja terhubung secara internal dengan label lokal.

---

#### Catatan

Meskipun tidak ada batasan panjang teks untuk suatu label di Eeschema, harap dipertimbangkan bahwa program lain yang digunakan untuk membaca berkas *netlist* yang dihasilkan oleh Eeschema mungkin memiliki batasan tertentu.

---

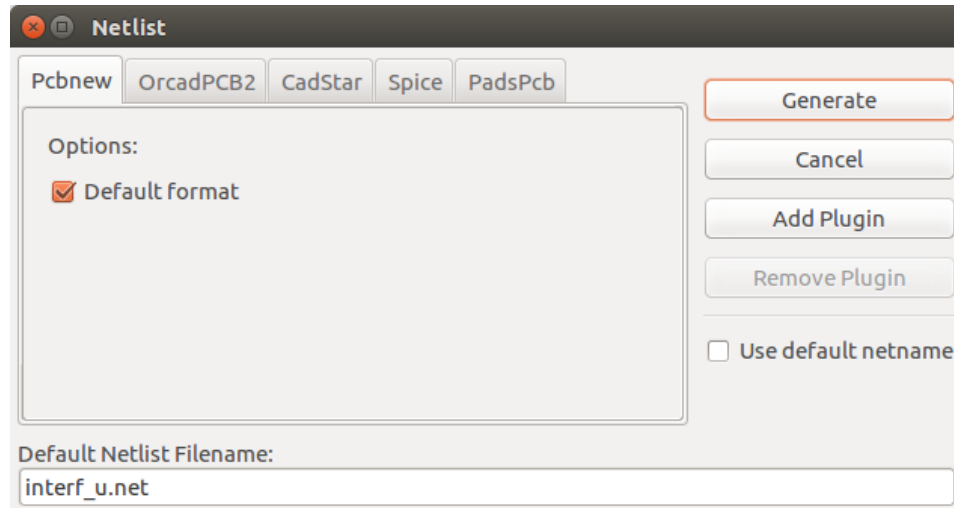
---

#### Catatan

Hindari penggunaan spasi pada label, karena label tersebut akan muncul sebagai kata-kata yang terpisah pada berkas yang dihasilkan. Hal ini bukanlah kekurangan Eeschema, namun karena kebanyakan format *netlist* mengasumsikan bahwa sebuah label tidak menggunakan spasi.

---





Opsi:

Default Format	Jika dicentang, maka Pcbnew akan digunakan sebagai format <i>default</i> .
----------------	--

Eeschema juga bisa menghasilkan berkas dengan beberapa format berikut ini:


- Orcad PCB2
- CadStar
- Spice (simulator)

*Plugin* eksternal juga bisa ditambahkan untuk mendukung format *netlist* yang lain (Pada gambar di atas, ditambahkan *plugin* PadsPcb).

Informasi lebih lanjut mengenai pembuatan *netlist* dapat dilihat pada bagian [Membuat Netlist](#).

## 4.4 Alat Anotasi



Gunakan ikon  untuk menjalankan alat anotasi. Alat ini digunakan untuk memberikan referensi pada komponen.

Untuk komponen-komponen *multi-part* (misalnya 7400 TTL yang memiliki 4 *gate*), kita perlu mengalokasikan sufiks *multi-part* (maka U3 untuk 7400 TTL akan dibagi menjadi U3A, U3B, U3C, dan U3D).

Anda bisa memberikan anotasi pada keseluruhan komponen atau hanya untuk komponen-komponen baru, yaitu komponen yang belum diberikan anotasi sebelumnya.

Annotate Schematic

Scope

☒ Use the entire schematic

☐ Use the current page only

☒ Keep existing annotation


☐ Reset existing annotation

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

Annotation Order

☐ Sort components by X position

☒ Sort components by Y position



Annotation Choice

☒ Use first free number in schematic

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

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

Dialog

☐ Automatically close this dialog

☐ Silent mode

Close

Clear Annotation

Annotate

Pilihan Scope

Use the entire schematic	Semua lembar kerja akan dilakukan anotasi ulang ( <i>default</i> ).
Use the current page only	Hanya lembar kerja yang saat ini dibuka saja yang akan dilakukan anotasi ulang (opsi ini hanya digunakan pada kasus-kasus khusus, misalnya untuk menghitung jumlah resistor yang diperlukan pada lembar kerja saat ini.).
Keep existing annotation	Anotasi kondisional, hanya komponen-komponen baru saja yang akan diberikan anotasi ( <i>default</i> ).
Reset existing annotation	Anotasi tidak kondisional, keseluruhan komponen akan dilakukan anotasi ulang (opsi ini perlu digunakan jika terdapat referensi ganda).
Reset, but do not swap any annotated multi-unit parts	Saat melakukan anotasi ulang, komponen <i>multi_unit</i> tetap dikelompokkan ke dalam satu grup (misalnya U2A, U2B).

Urutan Anotasi (*Annotation Order*)

Pilih urutan penomoran komponen (secara horisontal atau vertikal).

Pilihan Anotasi (*Annotation Choice*)

Pilih format referensi yang akan diterapkan.

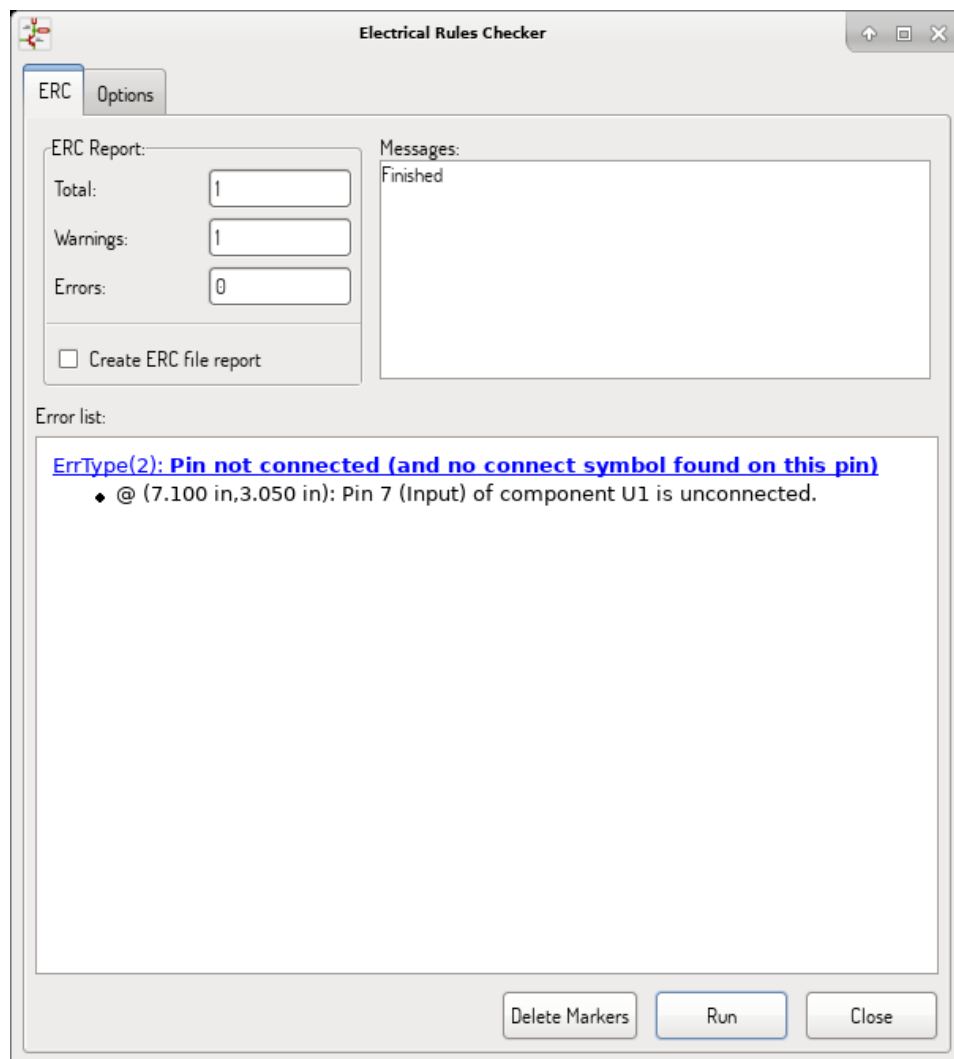
## 4.5 Alat *Electrical Rules Check*

Untuk menjalankan peralatan *Electrical Rules Check* (ERC), klik ikon .

Peralatan ini digunakan untuk melakukan verifikasi desain dan mampu mendeteksi koneksi yang belum terhubung, serta inkonsistensi.

Setelah ERC dijalankan, Eeschema akan menandai setiap permasalahan yang ditemukan. Penjelasan kesalahan akan ditampilkan setelah kita melakukan klik kiri pada tanda tersebut. Kita juga bisa membuat berkas laporan kesalahan.

### 4.5.1 Kotak Dialog Utama ERC



Kesalahan yang ditemukan akan ditampilkan pada kotak dialog ERC, beserta:

- Jumlah kesalahan dan peringatan.
- Jumlah kesalahan.
- Jumlah peringatan.

Opsi:

Create ERC file report	Centang opsi ini untuk membuat berkas laporan ERC.
------------------------	--

Perintah:

Delete Markers	Hapus semua penanda kesalahan/peringatan ERC.
Run	Menjalankan <i>Electrical Rules Check</i> .
Close	Tutup kotak dialog.

- Klik pada pesan kesalahan akan menampilkan penanda yang terkait di skematik.

4.5.2 Kotak Dialog Opsi ERC



Tab ini digunakan untuk mendefinisikan aturan konektivitas antar pin; Anda dapat memilih salah satu dari pilihan berikut, untuk setiap kasus:

- Tidak ada kesalahan
- Peringatan
- Kesalahan

Setiap kotak pada matriks dapat dimodifikasi dengan melakukan klik pada kotak tersebut.

Opsi:


Test similar labels	Melaporkan label yang sama namun menggunakan huruf besar/kecil yang berbeda (contohnya label/Label/LaBeL). Nama <i>net</i> bersifat <i>case-sensitive</i> , sehingga label dengan penggunaan huruf besar/kecil yang berbeda akan diperlakukan sebagai <i>net</i> yang terpisah.
Test unique global labels	Melaporkan label global yang muncul hanya satu kali saja untuk suatu <i>net</i> . Seharusnya diperlukan setidaknya dua buah label untuk membuat satu koneksi.

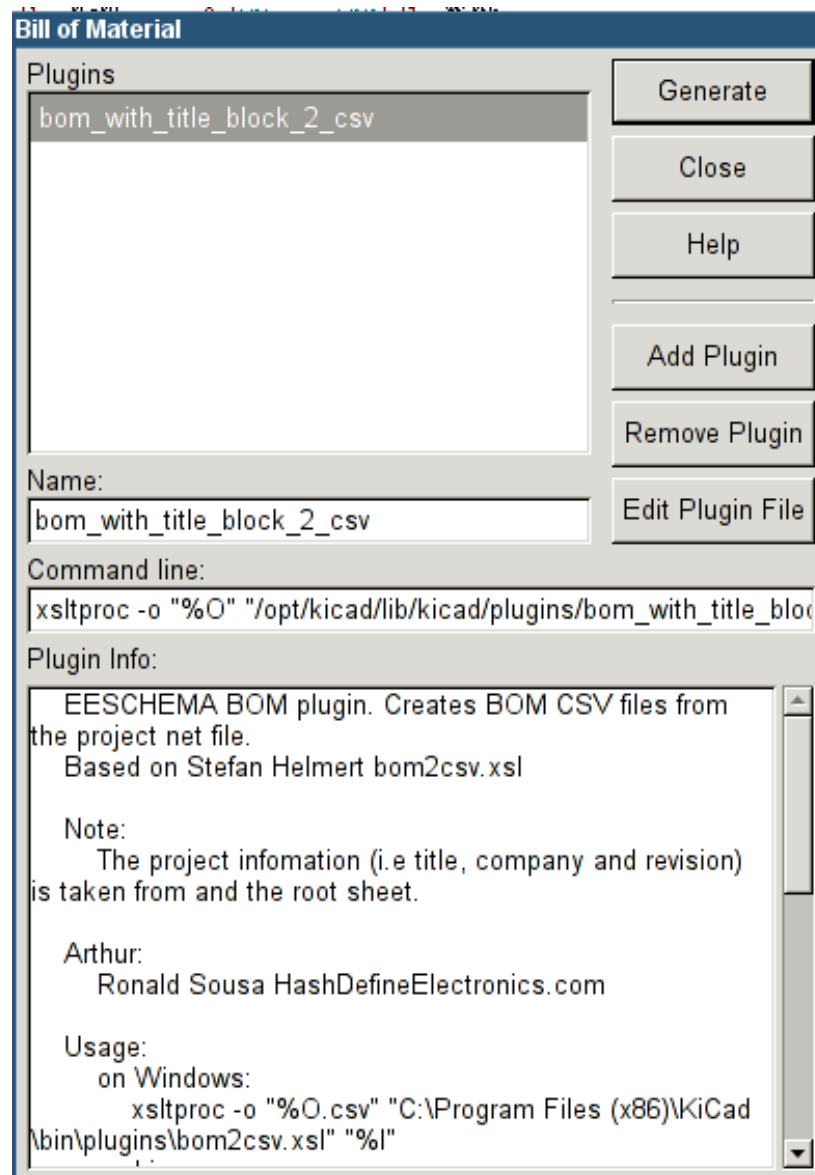
Perintah:

Initialize to Default	Mengembalikan pengaturan ke awal.
-----------------------	-----------------------------------

## 4.6 Alat Daftar Komponen (*Bill of Materials*)



Ikon  akan menjalankan alat pembuat daftar komponen (*Bill of Materials*/BOM). Alat ini akan membuat sebuah berkas yang berisi daftar komponen dan/atau koneksi-koneksi hirarkis (label global).

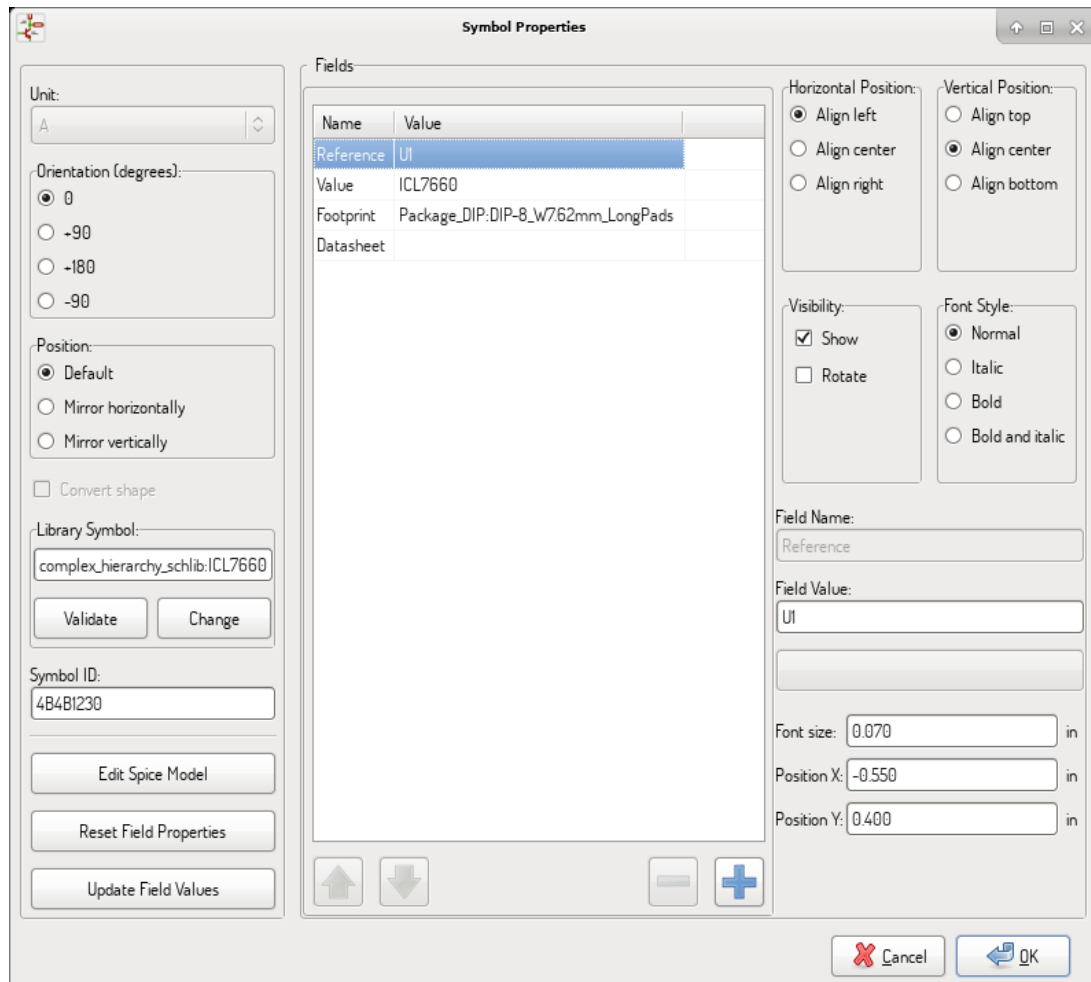


Pembuat BOM yang dimiliki Eeschema memanfaatkan *plugin-plugin* eksternal, menggunakan skrip Python atau XSLT. Terdapat beberapa contoh yang terinstal di dalam direktori berkas program KiCad.

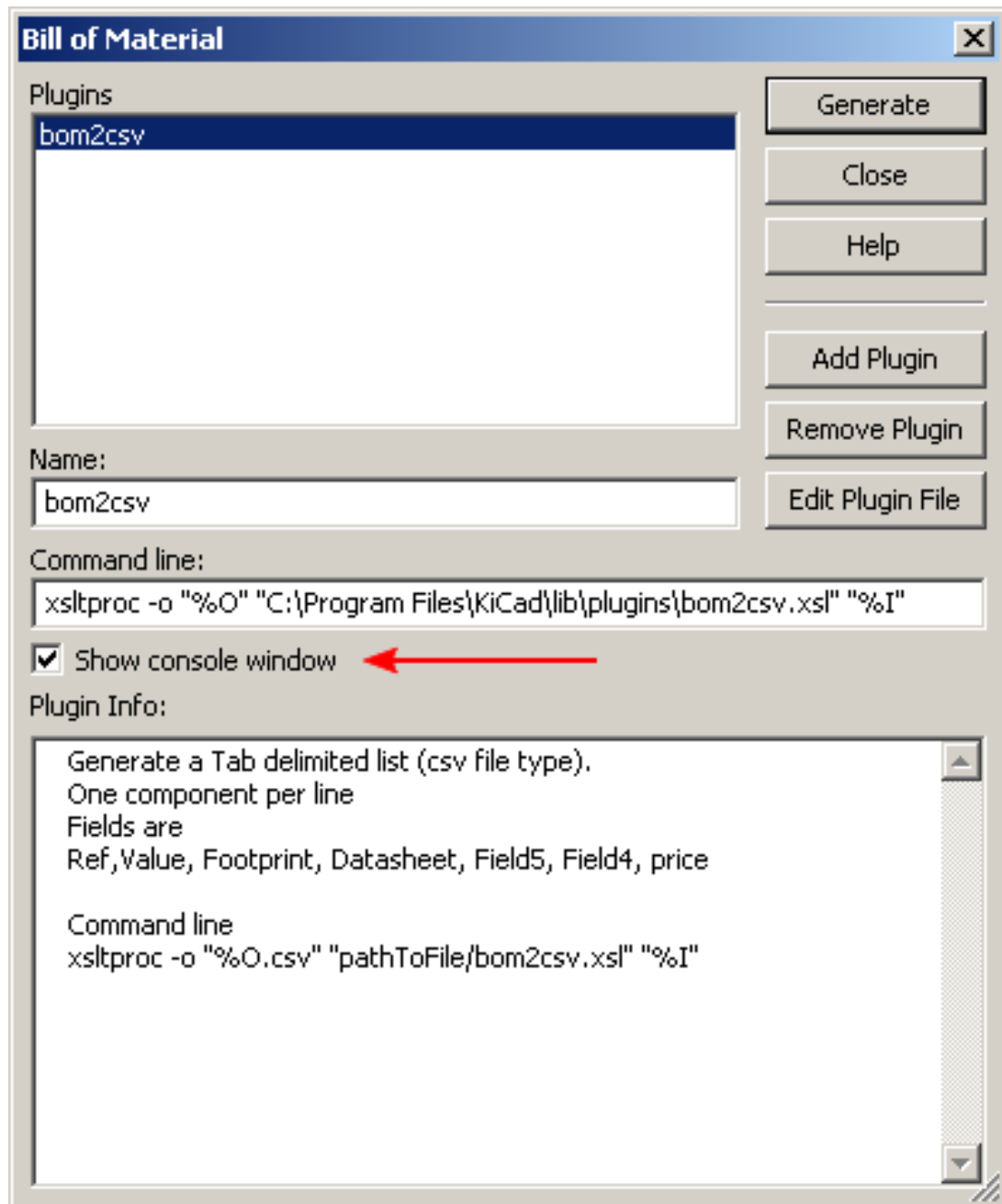
Daftar properti komponen yang diperlukan untuk membuat BOM antara lain:

- *Value* - nama unik untuk setiap komponen yang digunakan.
- *Footprint* - bisa dibuat secara manual atau dari anotasi-mundur (lihat di bawah).
- *Field1* - nama manufaktur.
- *Field2* - nomor *part* manufaktur.
- *Field3* - nomor *part* distributor.


Contoh:



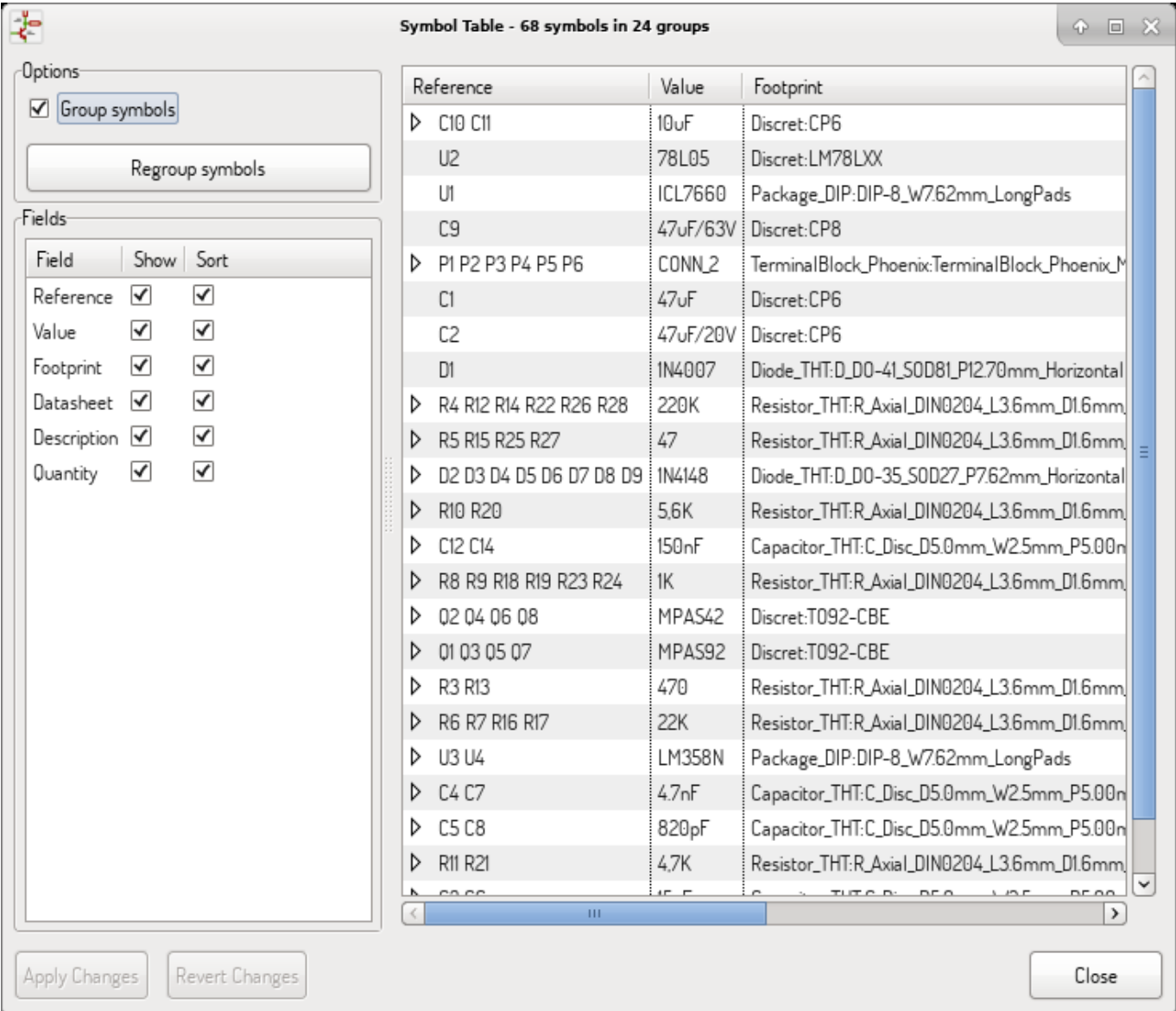
Pada **MS Windows**, kotak dialog pembuat BOM memiliki opsi khusus (ditunjukkan oleh panah berwarna merah) untuk menampilkan jendela *plugin* eksternal. + Secara *default*, perintah pembuat BOM dieksekusi oleh jendela konsol secara tersembunyi dan hasilnya akan diteruskan ke kotak *Plugin info*. Centang pilihan ini untuk menampilkan jendela perintah yang sedang berjalan. Hal ini mungkin diperlukan jika *plugin* menyediakan antarmuka grafis.



## 4.7 Alat Pengaturan Atribut

Ikon  digunakan untuk melihat dan memodifikasi nilai-nilai atribut simbol.





Jika Anda sudah selesai memodifikasi nilai suatu atribut, Anda harus menerima perubahan dengan klik tombol *Apply* atau membatalkan modifikasi dengan klik tombol *Revert*.

4.7.1 Tricks to simplify fields filling

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

These methods are illustrated below.

Copy (Ctrl+C)	Selection	Paste (Ctrl+V)

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

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

4.8 Peralatan Impor *Footprint*

4.8.1 Cara mengakses:



Ikon **BACK** akan menjalankan anotasi-mundur (*back-annotate*).  
Peralatan ini digunakan agar perubahan *footprint* yang dilakukan di PcbNew akan diimpor balik ke Eeschema.

## Bab 5

# Manajemen Pustaka Simbol

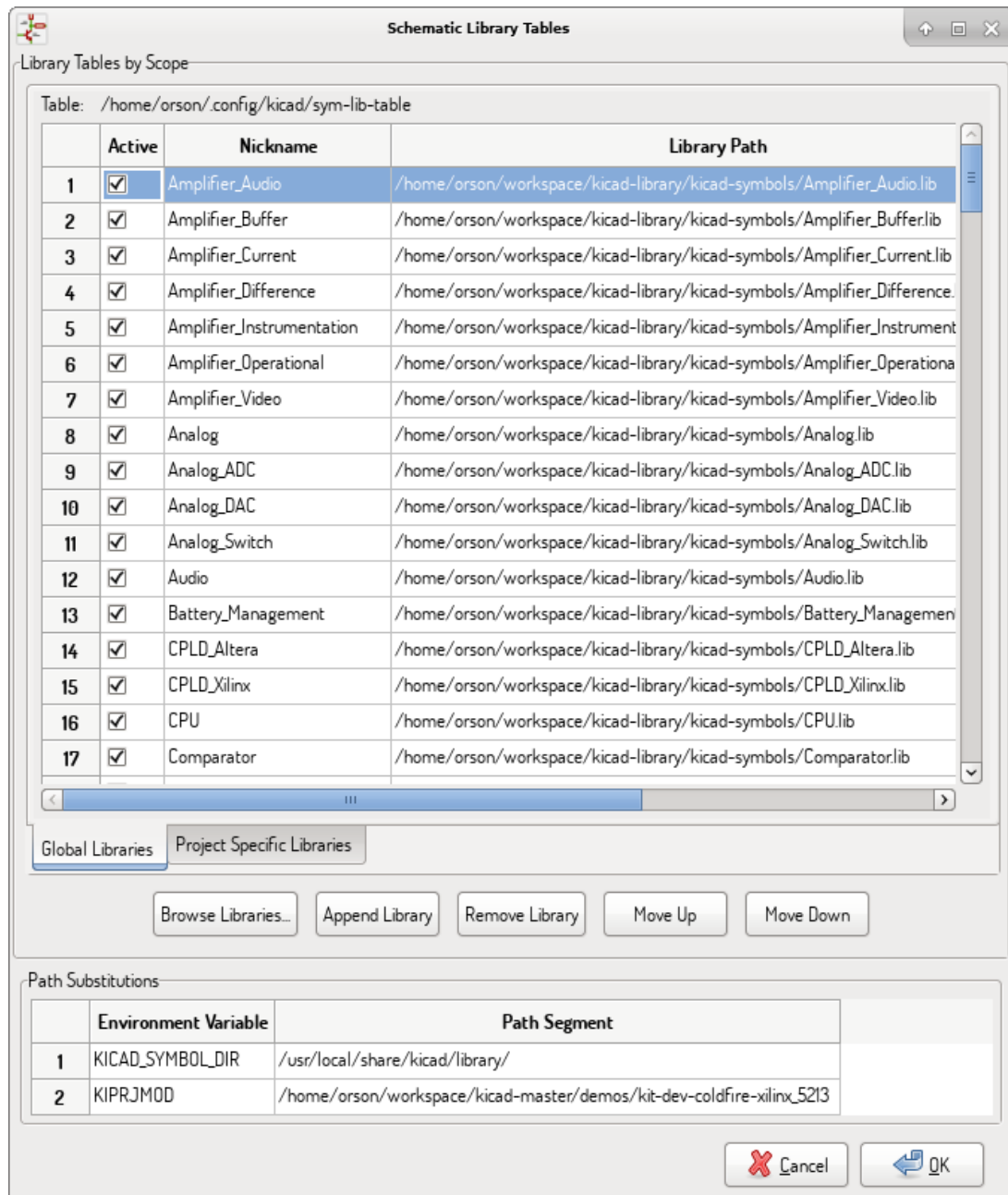
Symbol libraries hold collections of symbols used when creating schematics. Each symbol in a schematic is uniquely identified by a full name that is composed of a library nickname and a symbol name. An example is `Audio:AD1853`.

### 5.1 Tabel Pustaka Simbol (*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.

Gambar di bawah ini menunjukkan kotak dialog pengaturan tabel pustaka simbol yang bisa dibuka melalui *Manage Symbol Library Tables* pada menu *Preferences*.



### 5.1.1 Tabel Pustaka Simbol Global

Tabel pustaka simbol global berisi daftar pustaka yang selalu tersedia tanpa bergantung kepada berkas proyek yang saat ini dibuka. Tabel ini disimpan di dalam berkas *sym-lib-table* di direktori rumah pengguna. Lokasi direktori ini berbeda-beda, tergantung pada sistem operasi yang digunakan.

### 5.1.2 Tabel Pustaka Simbol Proyek

Tabel pustaka simbol yang spesifik terhadap proyek berisi daftar pustaka yang tersedia secara khusus untuk berkas proyek yang saat ini sedang dibuka. Tabel pustaka simbol proyek ini hanya dapat diedit ketika dibuka bersama dengan berkas proyek. Jika tidak ada berkas proyek yang sedang dibuka, atau jika tidak ada berkas tabel pustaka simbol di *path* proyek yang saat ini sedang dibuka, maka akan dibuat sebuah tabel kosong yang bisa diedit dan disimpan kemudian bersama dengan berkas proyek.

### 5.1.3 Konfigurasi Awal

Saat pertama kali Eeschema dijalankan dan berkas tabel pustaka simbol global **sym-lib-table** tidak ditemukan di direktori rumah pengguna, Eeschema akan mencoba menyalin berkas tabel pustaka simbol *default sym-lib-table* yang tersimpan di dalam direktori sistem templat KiCad menuju ke berkas *sym-lib-table* di dalam direktori rumah pengguna. Jika berkas *default* tidak dapat ditemukan, akan muncul sebuah kotak dialog untuk memilih lokasi berkas *sym-lib-table* alternatif. Jika tidak ditemukan berkas *sym-lib-table* atau kotak dialog tersebut diabaikan oleh pengguna, akan dibuat sebuah tabel pustaka simbol kosong pada direktori rumah pengguna. Jika ini terjadi, pengguna bisa menyalin *sym-lib-table* secara manual atau melakukan konfigurasi tabel kosong tersebut secara langsung.

---

#### Catatan

Tabel pustaka simbol *default* berisi keseluruhan pustaka simbol yang terinstal sebagai bagian dari KiCad. Hal ini bisa menjadi keuntungan maupun kerugian, tergantung dari penggunaan dan kecepatan sistem. Jumlah waktu yang dibutuhkan untuk memuat pustaka simbol proporsional terhadap jumlah pustaka di dalam tabel pustaka simbol. Jika waktu muat suatu pustaka simbol terlalu lama, hapuslah pustaka yang tidak pernah atau jarang digunakan dari tabel pustaka global dan tambahkan pustaka tersebut ke tabel pustaka proyek jika diperlukan.

---

### 5.1.4 Menambahkan Entri Tabel

Untuk dapat menggunakan sebuah pustaka simbol, sebelumnya pustaka tersebut harus ditambahkan ke tabel global atau tabel proyek. Tabel proyek hanya bisa digunakan jika Anda sedang membuka sebuah berkas proyek.

**Setiap entri pustaka harus memiliki nama panggilan yang unik.**

Nama panggilan ini tidak harus berhubungan dengan nama berkas atau *path* dari berkas pustaka sebenarnya. Tanda : dan / tidak bisa digunakan sebagai nama panggilan pustaka. Setiap entri pustaka harus memiliki *path* dan/atau nama berkas yang valid tergantung dari tipe pustaka tersebut. *Path* dapat ditulis secara absolut, relatif, atau menggunakan substitusi *environment variable* dari sistem Anda (lihat bagian di bawah).

Tipe *plugin* yang sesuai harus dipilih agar pustaka tersebut dapat dibaca dengan benar. KiCad saat ini hanya mendukung *plugin* pustaka simbol *legacy*.

Terdapat kolom *Description* untuk menambahkan penjelasan mengenai entri pustaka tersebut. Kolom *Option* saat ini tidak digunakan dan tidak ada efek saat memuat pustaka.

- Ingat bahwa Anda tidak diperkenankan memiliki nama panggilan yang sama di dalam satu tabel. Namun, dimungkinkan untuk memiliki nama panggilan yang sama di dalam tabel pustaka global dan proyek. Entri pada tabel pustaka proyek akan lebih didahulukan daripada entri pada tabel pustaka global jika ada nama panggilan yang sama. Ketika kita memasukkan suatu entri pada tabel pustaka proyek, sebuah berkas *sym-lib-table* yang berisi entri tersebut akan dituliskan ke dalam folder berkas proyek yang saat ini dibuka. - -

### 5.1.5 Substitusi *Environment Variable*

Salah satu fitur yang sangat berguna dari tabel pustaka simbol adalah substitusi *environment variable*. Kita bisa mendefinisikan *path* sesuai keinginan kita yang menuju ke lokasi penyimpanan pustaka simbol di *environment variable*. Substitusi *environment variable* didukung dengan menggunakan sintaks `${ENV_VAR_NAME}` pada *path* pustaka.

Secara *default*, pada saat dijalankan KiCad mendefinisikan **dua *environment variable***:

- *environment variable* **KIPRJMOD** yang selalu menunjuk ke direktori proyek yang sedang dibuka. **KIPRJMOD** tidak dapat dimodifikasi.
- *environment variable* **KICAD\_SYMBOL\_DIR** menunjuk ke *path* di mana pustaka simbol *default* terinstal dengan KiCad.

Anda dapat mengganti **KICAD\_SYMBOL\_DIR** dengan mendefinisikannya melalui menu *Preferences* → *Configure Path*. Anda bisa memasukkan pustaka Anda sendiri untuk menggantikan pustaka simbol *default* KiCad.

Dengan *environment variable* **KIPRJMOD**, Anda bisa menyimpan pustaka pada *path* proyek tanpa harus mendefinisikan *path* absolut (yang tidak selalu diketahui) menuju pustaka di dalam tabel pustaka simbol proyek.

---

### 5.1.6 Pola Pemakaian

Pustaka-pustaka simbol dapat didefinisikan secara global atau secara spesifik hanya untuk proyek yang sedang dibuka. Pustaka-pustaka simbol yang didefinisikan di tabel global akan selalu tersedia dan disimpan pada berkas *sym-lib-table* di dalam direktori rumah pengguna. Sedangkan tabel pustaka proyek akan aktif hanya untuk berkas proyek yang sedang dibuka.

Ada keuntungan dan kerugian untuk masing-masing metode tersebut. Mendefinisikan keseluruhan pustaka di dalam tabel global berarti pustaka-pustaka tersebut akan selalu tersedia saat dibutuhkan. Namun kerugiannya adalah waktu yang dibutuhkan untuk memuat pustaka-pustaka tersebut akan bertambah.

Mendefinisikan keseluruhan pustaka simbol di dalam tabel proyek berarti bahwa Anda hanya memiliki pustaka yang dibutuhkan oleh proyek saja, sehingga akan mengurangi waktu yang dibutuhkan untuk memuat pustaka-pustaka tersebut. Namun kekurangannya adalah Anda harus selalu mengingat untuk menambahkan setiap pustaka simbol yang dibutuhkan untuk setiap proyek.

Maka pola pemakaian yang baik adalah mendefinisikan secara global pustaka-pustaka yang umum digunakan. Sedangkan untuk pustaka-pustaka yang spesifik yang hanya dibutuhkan oleh proyek, kita definisikan di dalam tabel pustaka proyek. Tidak ada batasan bagi suatu pustaka untuk didefinisikan ke tabel manapun.

### 5.1.7 Pemetaan Ulang Proyek *Legacy*

Ketika kita membuka sebuah skematik yang dibuat sebelum diimplementasikannya tabel pustaka simbol, Eeschema akan berusaha memetakan ulang tautan pustaka simbol di skematik menuju ke tabel pustaka simbol yang sesuai. Keberhasilan dari proses ini bergantung pada beberapa faktor:

- pustaka asli yang digunakan di skematik masih tersedia dan belum berubah dari saat simbol ditambahkan ke dalam skematik.
- semua operasi penyelamatan telah dilakukan ketika terdeteksi untuk membuat sebuah pustaka penyelamat atau menjaga agar pustaka penyelamat yang ada tetap dalam kondisi terkini.
- integritas pustaka tembolok simbol proyek tidak dalam kondisi rusak.

**Awis**

Proses pemetaan ulang akan membuat cadangan untuk keseluruhan berkas yang berubah selama proses pemetaan, di dalam folder *rescue-backup* di folder proyek. Selalu buat cadangan secara manual untuk proyek Anda sebelum melakukan pemetaan ulang, untuk berjaga-jaga jika pemetaan tidak berjalan sebagaimana mestinya.

**Awis**

Operasi penyelamatan dilakukan walaupun telah dinonaktifkan, untuk memastikan bahwa simbol-simbol yang tepat telah tersedia untuk dipetakan ulang. Jangan membatalkan operasi ini atau pemetaan ulang akan gagal untuk memetakan simbol-simbol skematik secara benar. Setiap tautan simbol yang rusak harus diperbaiki secara manual.

---

**Catatan**

Jika pustaka asli telah dihapus dan tidak dilakukan penyelamatan, pustaka tembolok dapat digunakan sebagai pustaka penyelamat terakhir. Salin pustaka tembolok ke nama berkas yang baru dan tambahkan berkas pustaka ke bagian paling atas daftar pustaka dengan menggunakan aplikasi Eeschema versi sebelum implementasi tabel pustaka simbol.

---

## Bab 6

# Membuat dan Mengedit Skematik

### 6.1 Pengenalan

Sebuah skematik dapat direpresentasikan oleh satu lembar kerja saja. Namun jika skematik tersebut berukuran besar, akan dibutuhkan beberapa lembar kerja.

Skematik yang direpresentasikan dengan beberapa lembar kerja adalah bersifat hirarkis, dan keseluruhan lembar kerja (tiap lembar kerja direpresentasikan sebagai satu berkas tersendiri) menjadi satu kesatuan proyek Eeschema. Manipulasi dari skematik hirarkis akan dijelaskan pada bagian [Skematik Hirarkis](#).

### 6.2 Pertimbangan Umum

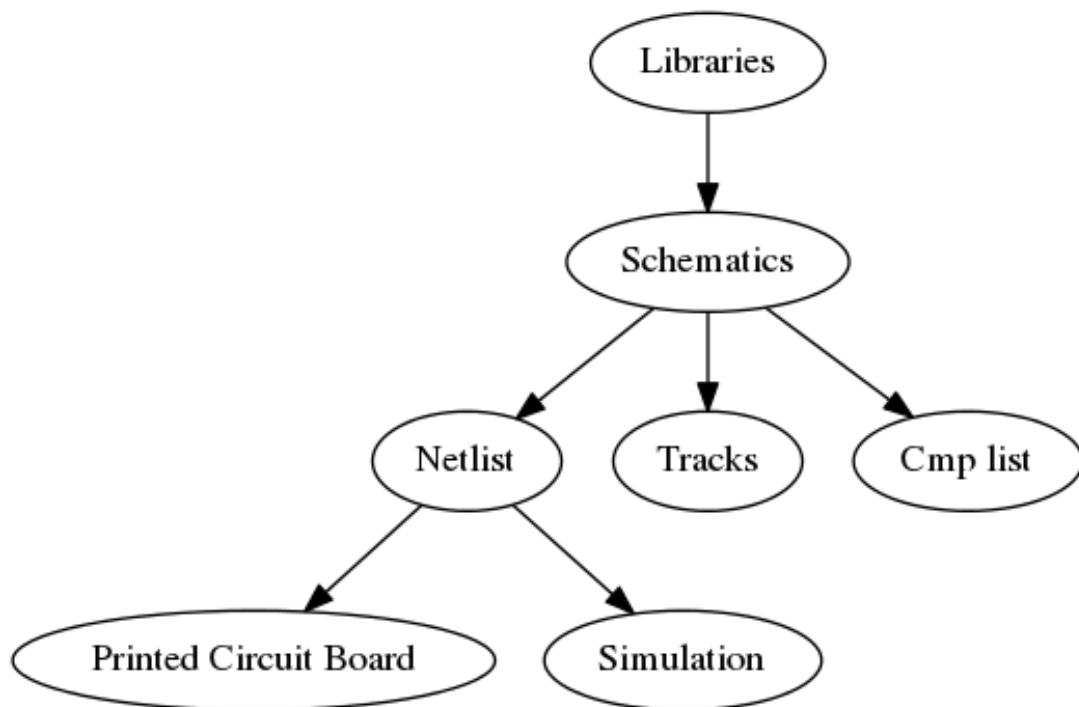
Sebuah skematik yang didesain dengan Eeschema adalah lebih dari sekedar representasi grafis sederhana suatu divais elektronika. Skematik menjadi titik mulai dari rantai pengembangan yang memungkinkan kita:

- Melakukan validasi berdasarkan suatu aturan tertentu ([Electrical Rules Check](#)) untuk mendeteksi kesalahan dan kelalaian.
- Pembuatan daftar komponen ([BOM](#)) secara otomatis.
- [Pembuatan netlist](#) untuk perangkat lunak simulasi seperti SPICE.
- [Pembuatan netlist](#) untuk mentransfer tata letak PCB.

Sebuah skematik berisi sejumlah simbol, *wire*, label, *junction*, *bus*, dan *power port*. Agar skematik lebih mudah dipahami, Anda bisa meletakkan elemen-elemen grafis tambahan seperti jalur masuk *bus*, komentar, dan garis poligon.

---


## 6.3 Rantai Pengembangan



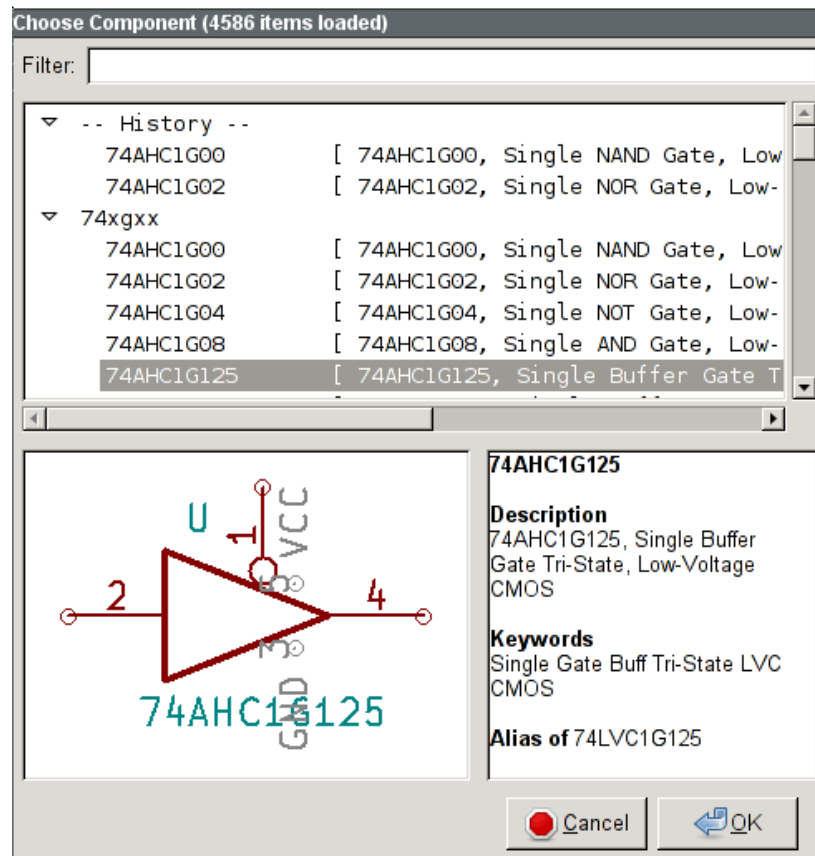
Simbol ditambahkan ke dalam skematik dari pustaka simbol. Setelah skematik selesai dibuat, akan dibuat sebuah *netlist*, yang nantinya akan digunakan untuk mengimpor koneksi dan *footprint* dari skematik ke PcbNew.

## 6.4 Meletakkan dan Mengedit Simbol

### 6.4.1 Mencari dan Meletakkan Simbol

Untuk memuat sebuah simbol ke dalam skematik, Anda bisa menggunakan ikon . Sebuah kotak dialog akan muncul, di mana kita bisa mengetikkan nama dari simbol yang akan dimuat.





Kotak Dialog *Choose Symbols* akan menyaring simbol berdasarkan nama, kata kunci, dan deskripsi menurut apa yang Anda ketikkan ke dalam kotak pencarian. Penyaringan lebih jauh dapat digunakan dengan cara mengetikkan ke dalam kotak pencarian:

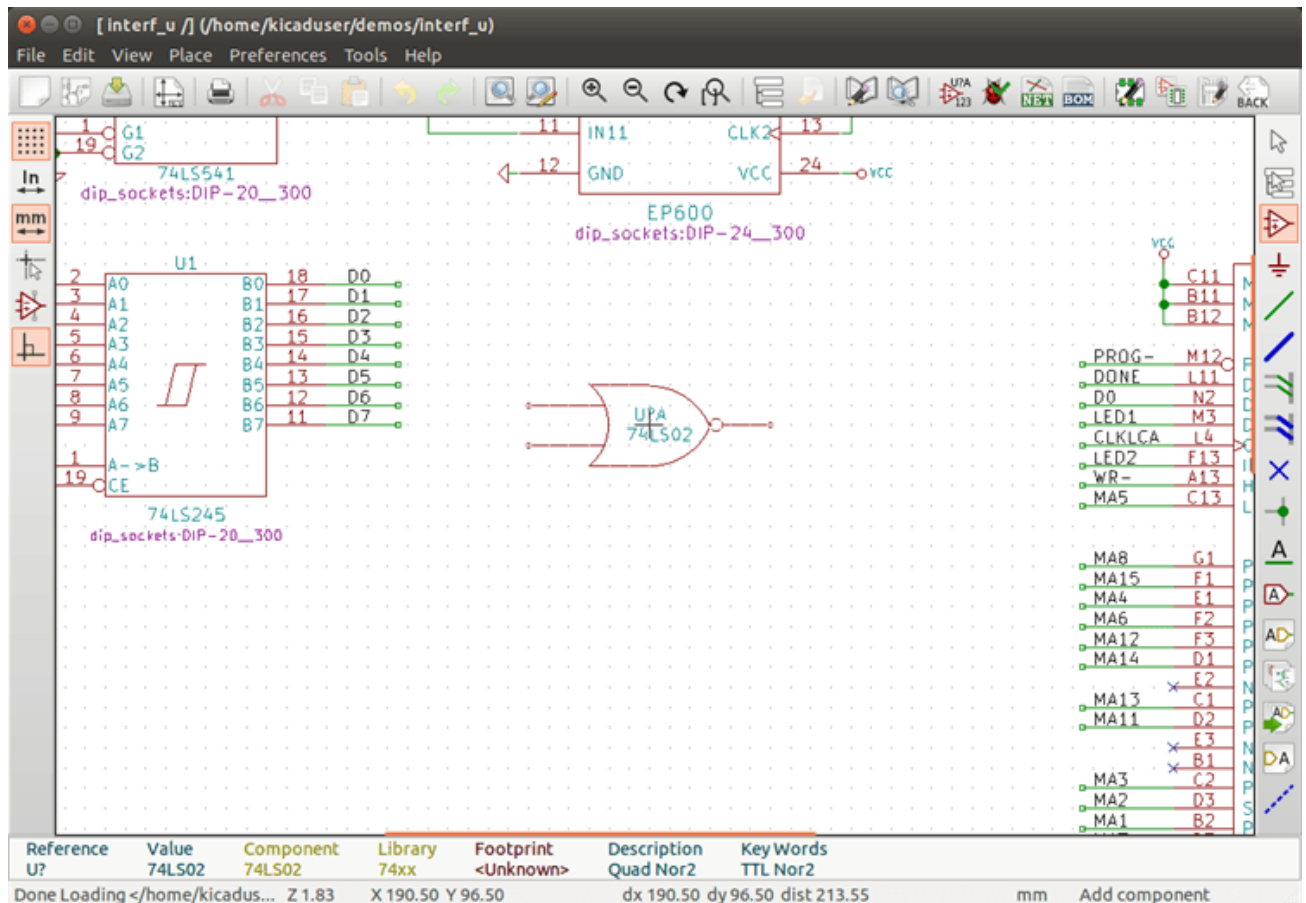
- **Wildcards:** gunakan karakter ? dan \* yang berarti "karakter apapun" dan "karakter sepanjang apapun".
- **Relasional:** jika sebuah deskripsi atau kata kunci memiliki tagar dengan format "*Key:123*", Anda bisa melakukan pencarian relatif terhadap tagar tersebut dengan mengetikkan "*Key>123*" (lebih besar dari), "*Key<123*" (kurang dari), dsb. Angka bisa terdiri dari salah satu sufiks berikut ini: (huruf besar dan huruf kecil dianggap sama)  

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

$$|===|k|l|mi|gi|ti|2^{10}|2^{20}|2^{30}|2^{40}|===$$
- **Regular expression:** jika Anda biasa menggunakan *regular expression*, Anda bisa menggunakannya di sini. Gaya *regular expression* yang digunakan adalah **wxWidgets Advanced Regular Expression style**, yang mirip dengan *regular expression* Perl.

Sebelum meletakkan simbol di skematik, Anda bisa melakukan rotasi, mencerminkan, dan mengedit atribut-atributnya, dengan menggunakan tombol *hotkey* atau melalui menu konteks dengan klik-kanan tetikus. Anda juga bisa melakukan aksi tersebut setelah meletakkan simbol.

Berikut ini contoh peletakan simbol:



## 6.4.2 Power Port

Simbol *power port* adalah sebuah simbol (simbol-simbol tersebut dikelompokkan di dalam pustaka "*power*"), sehingga kita bisa meletakkannya menggunakan kotak dialog pencarian simbol. Namun, karena peletakan *power* sering dilakukan, disediakan alat



. Alat ini mirip dengan alat peletakan simbol lain, namun pencarian dilakukan secara langsung di dalam pustaka "*power*".

## 6.4.3 Mengedit dan Modifikasi Simbol (untuk komponen yang telah diletakkan)

Ada dua cara untuk mengedit sebuah simbol:

- Modifikasi pada simbol itu sendiri: posisi, orientasi, pemilihan satuan pada simbol dengan pilihan satuan.
- Modifikasi pada atribut simbol: referensi, nilai, *footprint*, dsb.

Setelah simbol diletakkan pada lembar kerja, Anda mungkin perlu untuk mengubah nilainya (khususnya untuk resistor, kapasitor, dsb.), namun Anda tidak perlu segera memberikan nomor referensi, atau memilih satuan (kecuali untuk komponen dengan satuan terkunci, di mana Anda harus memilih secara manual). Hal ini dapat dilakukan secara otomatis dengan menggunakan fungsi anotasi.

### 6.4.3.1 Modifikasi Simbol

Untuk memodifikasi fitur-fitur dari sebuah simbol, letakkan kursor pada simbol, dan lakukan salah satu dari aksi berikut:

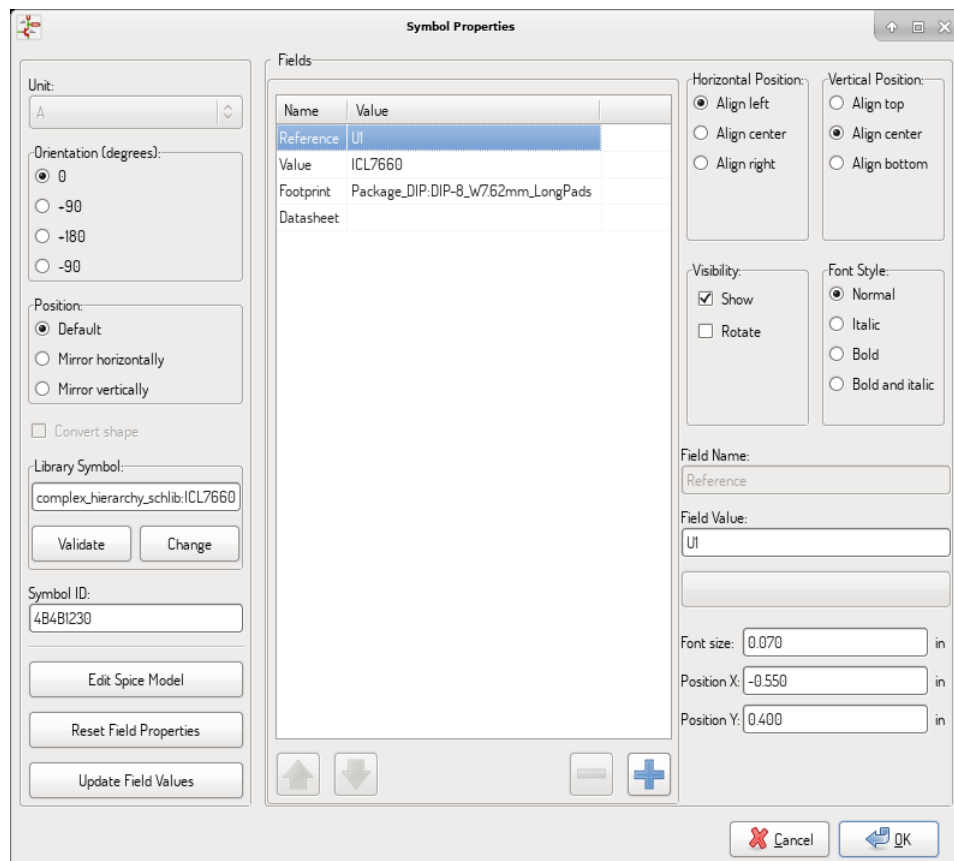
- Klik-ganda pada simbol untuk membuka kotak dialog editor simbol.
- Klik-kanan untuk membuka menu konteks dan gunakan salah satu dari perintah berikut: Pindahkan, Putar, Edit, Hapus, dsb.

### 6.4.3.2 Modifikasi Atribut Teks

Anda dapat memodifikasi referensi, nilai, posisi, orientasi, ukuran teks, dan tampilan dari atribut dengan cara berikut:

- Klik-ganda pada atribut teks untuk melakukan modifikasi.
- Klik-kanan untuk membuka menu konteks dan gunakan salah satu dari perintah berikut: Pindahkan, Putar, Edit, Hapus, dsb.

Untuk modifikasi lanjutan, atau untuk membuat suatu atribut, klik-ganda pada simbol untuk membuka kotak dialog *Symbol Properties*.



Setiap atribut dapat ditampilkan atau disembunyikan, dan dapat ditampilkan secara vertikal atau horisontal. Posisi yang dipilih selalu diindikasikan untuk simbol yang ditampilkan secara normal (tidak diputar atau dicerminkan) dan relatif terhadap titik jangkar dari simbol.

Opsi "*Reset to Library Defaults*" akan mengembalikan simbol ke orientasi awal, dan mengembalikan opsi-opsi, ukuran serta posisi dari setiap atribut. Akan tetapi, atribut teks tidak dimodifikasi karena bisa menyebabkan skematik menjadi rusak.

## 6.5 Wire, Bus, Label, Power Port

### 6.5.1 Pengenalan

Semua elemen gambar berikut juga dapat diletakkan dengan menggunakan peralatan pada bilah alat bagian kanan.

Elemen-elemen tersebut antara lain:

- **Wire**: koneksi antar simbol.

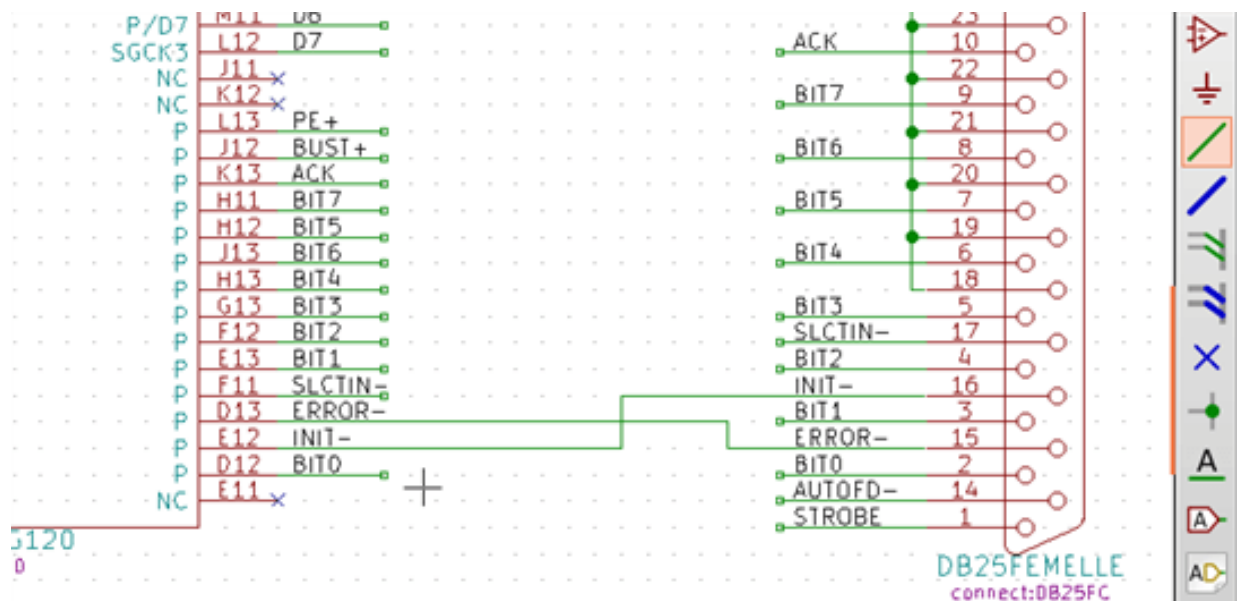
- **Bus:** untuk menghubungkan label *bus* dalam tampilan grafis.
- **Garis poligon:** untuk tampilan gambar grafis.
- **Junction:** untuk membuat koneksi pada persimpangan *wire* atau *bus*.
- **Bus entries:** to show connections between wires and buses.
- **Label:** untuk memberikan label atau membuat koneksi.
- **Label global:** untuk koneksi antar lembar kerja.
- **Teks:** untuk memberikan komentar atau anotasi.
- **Bendera "tak-hubung":** untuk menandai sebuah pin yang tidak memerlukan koneksi.
- **Lembar kerja hirarkis,** dan koneksi pin-pinnya.

### 6.5.2 Koneksi (*Wire* dan *Label*)

Ada dua cara untuk membuat suatu koneksi:

- *Wire* dari pin ke pin.
- Label.

Gambar berikut menunjukkan dua metode tersebut:



**Catatan 1:**

Titik “kontak” dari sebuah label adalah pada sudut kiri bawah dari huruf pertama label. Titik ini ditampilkan dengan kotak kecil ketika tidak terhubung.

Titik ini harus bersentuhan dengan *wire*, atau ditumpuk di ujung suatu pin sehingga label akan tampak terhubung.

**Catatan 2:**

Untuk membuat suatu koneksi, satu segmen dari *wire* ujungnya harus dihubungkan ke segmen yang lain atau ke sebuah pin.

Jika terjadi *wire* yang saling melintas (jika sebuah *wire* melewati sebuah pin, namun tidak terhubung ke ujung pin), maka koneksi tidak terjadi.

**Catatan 3:**

*Wire-wire* yang saling bersilang tidak terhubung secara implisit. Kita perlu menghubungkannya dengan sebuah titik *junction* jika ingin menghubungkan persilangan tersebut.

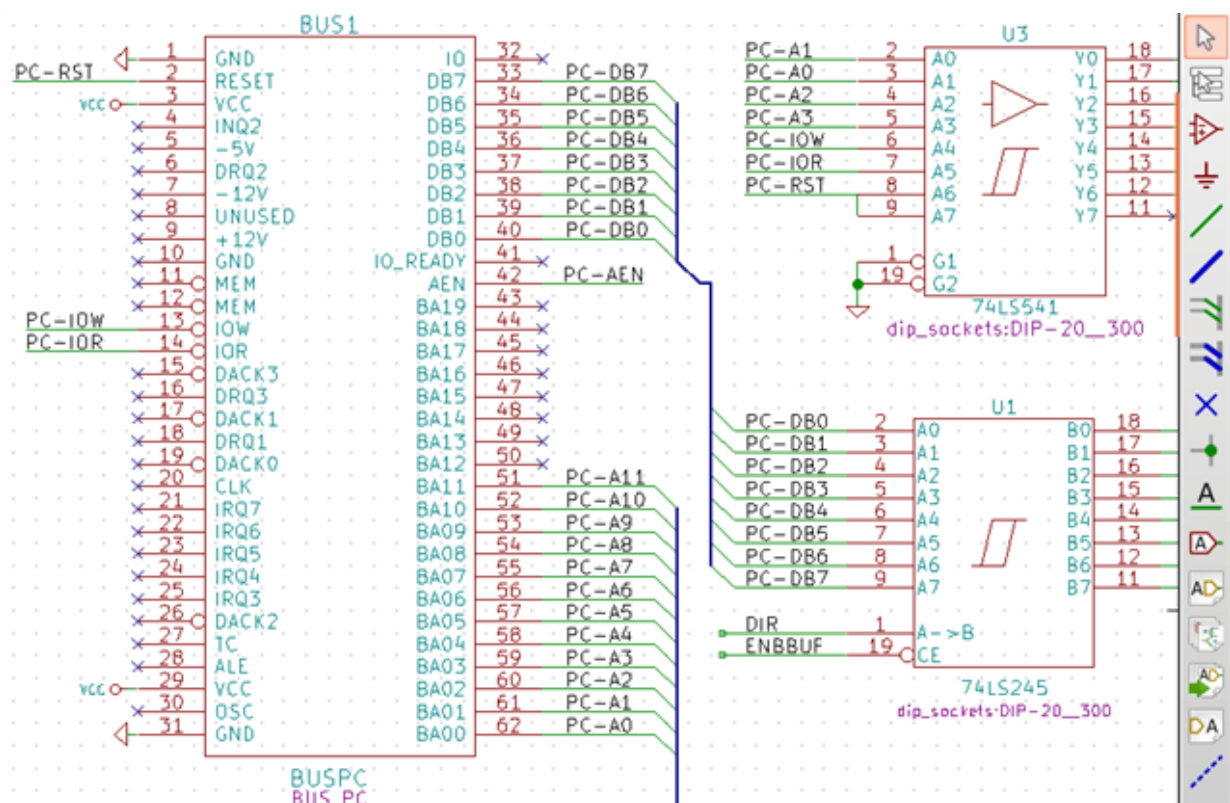
Gambar sebelumnya (*wire* yang terhubung ke DB25FEMALE pin 22, 21, 20, 19) menunjukkan koneksi menggunakan simbol *junction*.

**Catatan 4:**

Jika ada dua label yang berbeda diletakkan pada *wire* yang sama, maka mereka akan saling terhubung dan menjadi sama: semua elemen lain yang terhubung ke salah satu label, juga akan terhubung ke semua label tersebut.

**6.5.3 Koneksi (Bus)**

Pada skematik berikut, ada beberapa pin yang terhubung ke *bus*.

**6.5.3.1 Anggota Bus**

Dari sudut pandang skematik, *bus* adalah sekumpulan sinyal, dimulai dari prefiks umum, dan diakhiri dengan angka. Sebagai contoh, PCA0, PCA1, dan PCA2 adalah anggota dari *bus* PCA.

*Bus* yang lengkap memiliki nama PCA[N..m], dimana N dan m adalah nomor *wire* awal dan akhir pada *bus*. Jika PCA memiliki 20 anggota dari 0 sampai 19, maka nama *bus* yang lengkap ditulis dengan PCA[0..19]. Sedangkan untuk sekumpulan sinyal dengan nama PCA0, PCA1, PCA2, WRITE, READ tidak dapat dimasukkan ke dalam satu *bus*.

**6.5.3.2 Koneksi Antar Anggota Bus**

Pin-pin yang terhubung antar anggota yang sama dari sebuah *bus* harus dihubungkan dengan menggunakan label. Tidak dimungkinkan untuk menghubungkan sebuah pin secara langsung ke *bus*; koneksi seperti ini akan diabaikan oleh Eeschema.

Pada contoh di atas, koneksi dibuat dengan menggunakan label yang diletakkan pada *wire* yang terhubung ke pin. Jalur masuk *bus* (*wire* dengan sudut 45 derajat) hanya bersifat grafis, dan tidak diperlukan untuk membuat koneksi logis.

Dengan menggunakan perintah repetisi (tombol *Insert*), koneksi dapat dibuat dengan cepat dengan cara berikut, jika pin-pin komponen sejajar dalam urutan naik (umumnya pada komponen seperti memori, mikroprosesor, dsb.):

- Letakkan label pertama (misalnya PCA0)
- Gunakan perintah repetisi sebanyak yang diperlukan untuk meletakkan anggota *bus*. Eeschema akan secara otomatis membuat label berikutnya (PCA1, PCA2...) berurutan secara vertikal di atas label sebelumnya.
- Tariklah *wire* di bawah label pertama. Lalu gunakan perintah repetisi untuk meletakkan *wire* berikutnya di bawah label yang sudah dibuat.
- Jika diperlukan, letakkan jalur masuk *bus* dengan cara yang sama (Letakkan jalur masuk yang pertama, lalu gunakan perintah repetisi).

---

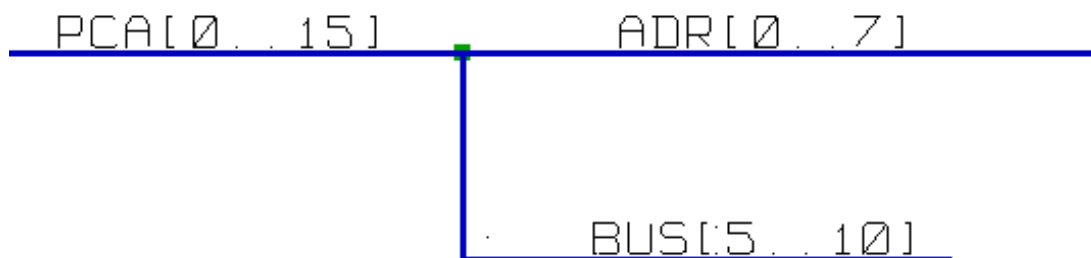
#### Catatan

Pada menu *Preferences* → *General Options*, Anda bisa mengatur parameter repetisi:

- Langkah vertikal.
  - Langkah horisontal.
  - Kenaikan label (bisa diatur untuk naik 2, 3, atau menurun)
- 

#### 6.5.3.3 Koneksi Global Antar *Bus*

Anda mungkin membutuhkan koneksi antar *bus*, untuk menghubungkan dua *bus* dengan nama yang berbeda, atau dalam kasus hirarki, untuk membuat koneksi antar lembar kerja yang berbeda. Anda dapat membuat koneksi ini dengan cara berikut.



*Bus* PCA [0..15], ADR [0..7] dan BUS [5..10] terhubung bersama (perhatikan adanya *junction* di sini karena *bus* vertikal tersambung di tengah segmen bus horisontal).

Lebih jelasnya, anggota *bus* berikut ini adalah terhubung: PCA0 terhubung dengan ADR0 (begitu juga PCA1 dan ADR1 ... PCA7 dan ADR7).

Demikian pula, PCA5, BUS5 dan ADR5 juga terhubung (seperti juga PCA6, BUS6 dan ADR6, serta PCA7, BUS7 dan ADR7). PCA8 dan BUS8 juga terhubung (seperti juga PCA9 dan BUS9, PCA10 dan BUS10).

---

### 6.5.4 Koneksi *Port Power*

Jika suatu simbol memiliki pin-pin *power* yang ditampilkan, maka pin-pin tersebut harus dihubungkan, seperti halnya sinyal-sinyal yang lain.


Simbol seperti *gate* dan *flip-flop* mungkin memiliki pin-pin *power* yang tersembunyi. Kita perlu memperhatikan pin-pin tersebut karena:

- Anda tidak bisa menghubungkan *wire*, karena pin-pin tersebut tidak ditampilkan.
- Anda tidak mengetahui nama pin-pin tersebut.

Namun, tidak disarankan untuk menampilkan pin-pin yang tersembunyi dan menghubungkannya seperti pin-pin yang lain, karena skematik akan menjadi sulit dipahami dan tidak sesuai dengan konvensi umum.

---

#### Catatan

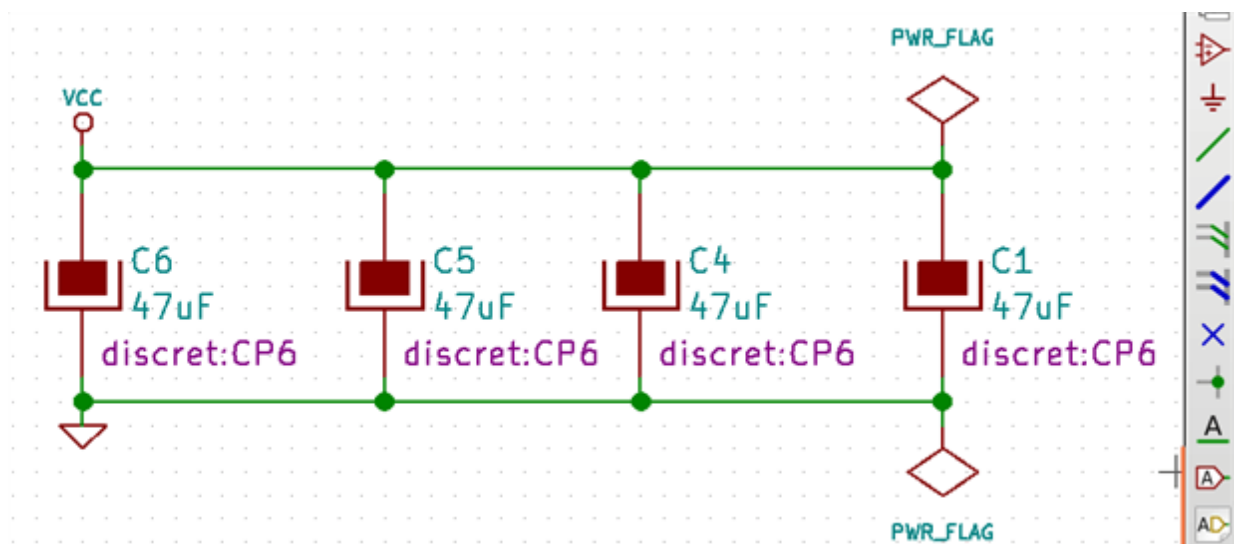
Jika Anda tetap ingin menampilkan pin-pin *power* yang tersembunyi, Anda harus mengaktifkan opsi "*Show invisible power pins*" pada menu *Preferences* → *General Options*, atau melalui ikon  pada bilah alat sebelah kiri.

---

Eeschema akan secara otomatis menghubungkan pin-pin *power* yang tersembunyi, yang memiliki nama sama dengan *net power*. Terkadang Anda perlu menyambungkan *net power* jika namanya tidak sama (misalnya, "GND" pada komponen TTL dan "VSS" pada komponen MOS); gunakan *port power* untuk menghubungkannya.

Tidak direkomendasikan untuk menggunakan label untuk koneksi *power*. Label hanya memiliki jangkauan koneksi "lokal", dan tidak akan bisa menghubungkan pin-pin *power* yang tersembunyi.

Gambar di bawah ini menampilkan sebuah contoh koneksi *port power*.



Pada contoh di atas, *ground* (GND) dihubungkan ke *port power* VSS, dan *port power* VCC dihubungkan ke VDD.


Tampak dua buah simbol PWR\_FLAG. Simbol tersebut mengindikasikan bahwa *port power* VCC dan GND terhubung ke catu daya. Tanpa simbol tersebut, ERC akan menampilkan peringatan bahwa *port power* tidak terhubung ke catu daya: *Warning: power port not powered*.

Semua simbol tersebut berada di pustaka simbol "*power*".

---



### 6.5.5 “No Connect” flag

These symbols are very useful to avoid undesired ERC warnings. The electrical rules check ensures that no connection has been accidentally left unconnected.

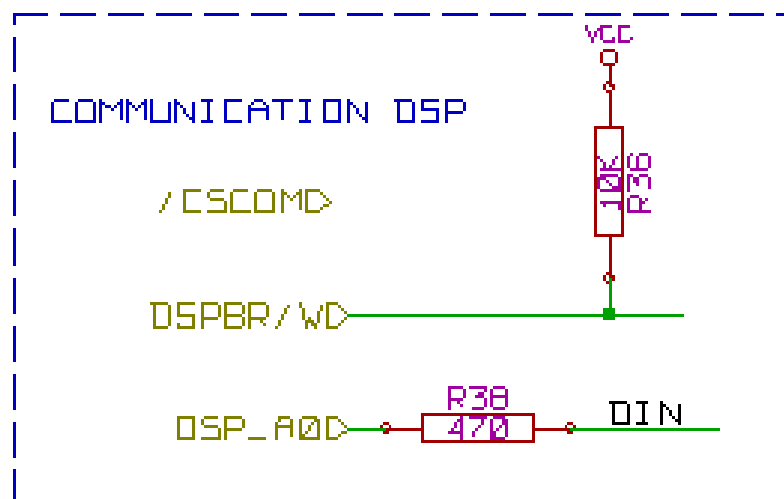
Jika suatu pin memang harus tidak terhubung, maka kita perlu meletakkan sebuah bendera "Tak-Hubung" (ikon ) pada pin tersebut. Simbol ini tidak akan berpengaruh pada berkas Netlist yang dihasilkan.

## 6.6 Peralatan Pelengkap Gambar


### 6.6.1 Komentar Teks

Terkadang kita perlu meletakkan anotasi, misalnya teks atau bingkai, untuk memperjelas suatu skematik. Teks (ikon ) dan garis poligon (ikon ) disediakan untuk hal ini. Peralatan ini berbeda dengan label atau *wire* yang merupakan elemen suatu koneksi.

Berikut adalah contoh suatu bingkai dengan komentar tekstual.



### 6.6.2 Blok Judul pada Lembar Kerja

Blok Judul dapat diedit melalui ikon .



**Page Settings**

**Paper**

Size: A3 297x420mm

Orientation: Landscape

Custom Size:  
Height: 11.000 Width: 17.000

Layout Preview

**Title Block Parameters**

Number of sheets: 1 Sheet number: 1

Issue Date: Sun 22 Mar 2015 <- 06/13/2015 ☐ Export to other sheets

Revision: 2B ☐ Export to other sheets

Title: UNIVERSAL INTERFACE ☐ Export to other sheets

Company: KICAD ☐ Export to other sheets

Comment1: Comment 1 ☐ Export to other sheets

Comment2: Comment 2 ☐ Export to other sheets

Comment3: Comment 3 ☐ Export to other sheets

Comment4: Comment 4 ☐ Export to other sheets

Page layout description file:  Browse

Cancel OK

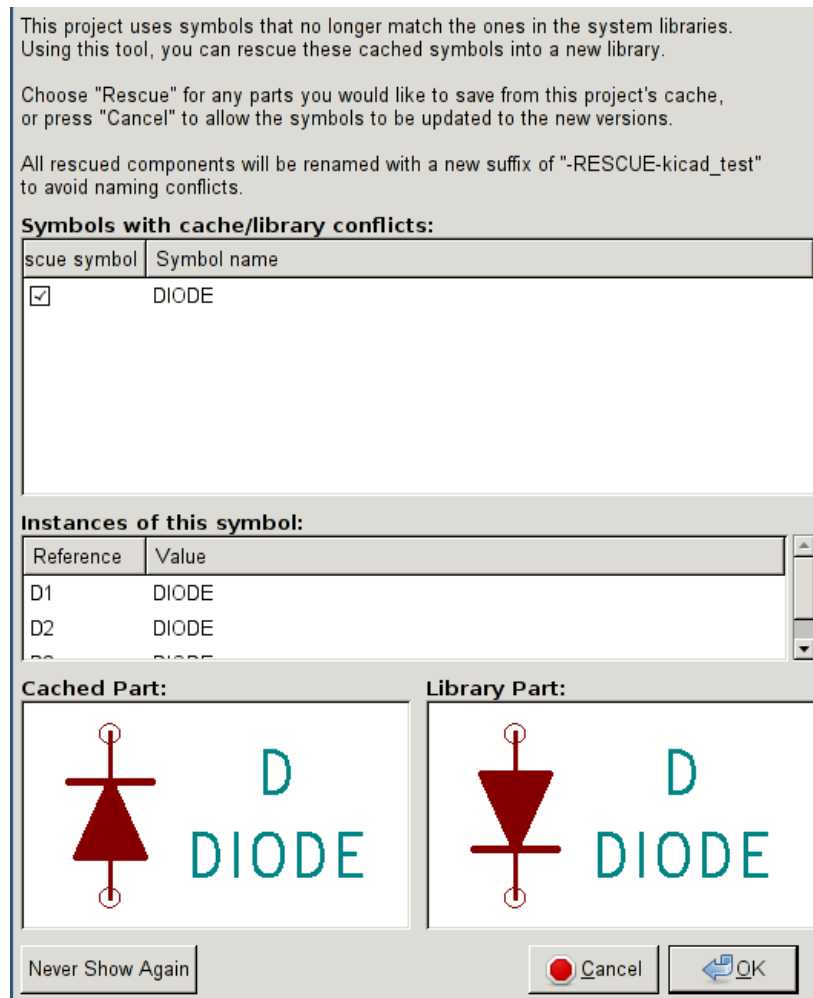
Comment 4		
Comment 3		
Comment 2		
Comment 1		
KICAD		
Sheet: /		
File: interf_u.sch		
<b>Title: UNIVERSAL INTERFACE</b>		
Size: A3	Date: 2015-10-03	Rev: 2B
KiCad E.D.A. eeschema 4.0.0-rc1-stable		Id: 1/1

Penomoran lembar kerja (*Sheet X/Y*) akan diperbarui secara otomatis.

## 6.7 Menyelamatkan tembolok simbol

Secara *default*, Eeschema akan memuat simbol dari pustaka proyek bergantung dari urutan *path* pustaka. Hal ini bisa menimbulkan suatu masalah saat memuat proyek yang sudah sangat tua: jika simbol di dalam pustaka telah berubah, atau sudah dihapus, atau jika pustaka yang digunakan sudah tidak ada lagi, maka simbol di dalam proyek akan diganti dengan versi yang baru. Simbol versi baru mungkin tidak sesuai, atau mungkin memiliki orientasi yang berbeda, yang akan menyebabkan skematik menjadi rusak.

Ketika suatu proyek disimpan, tembolok (*cache*) pustaka beserta isi dari pustaka simbol yang saat ini digunakan akan disimpan bersama dengan skematik. Dengan demikian proyek tersebut dapat didistribusikan tanpa harus menyertakan keseluruhan pustaka. Jika Anda memuat suatu proyek di mana simbol tersedia baik di tembolok maupun di pustaka sistem, Eeschema akan melakukan pemindaian pustaka untuk menemukan adanya konflik. Setiap konflik yang ditemukan akan ditampilkan pada kotak dialog berikut:



Anda bisa melihat pada contoh di atas bahwa proyek yang awalnya menggunakan sebuah dioda dengan katoda berada di atas, namun pustaka yang tersedia sekarang berisi dioda dengan katoda berada di bawah. Perubahan ini akan membuat skematik menjadi rusak! Jika kita menekan tombol OK, maka pustaka tembolok akan disimpan ke dalam pustaka khusus "rescue" dan semua simbol akan dinamai ulang untuk mencegah konflik dalam penamaan.

Jika Anda menekan *Cancel*, Eeschema tidak akan membuat pustaka "rescue", sehingga secara *default* Eeschema akan memuat semua komponen yang baru. Jika Anda menyimpan skematik ini, maka tembolok Anda akan ditimpa dan simbol-simbol lama tidak dapat diselamatkan. Jika sebelumnya Anda sudah menyimpan skematik, Anda masih bisa kembali dan menjalankan fungsi penyelamatan dengan memilih "Rescue Cached Components" di menu "Tools" untuk menampilkan kembali kotak dialog penyelamatan.

Jika Anda tidak ingin menampilkan kotak dialog ini, Anda bisa memilih tombol "Never show Again". Secara *default*, Eeschema tidak akan melakukan apapun dan akan memuat seluruh komponen baru. Pilihan ini dapat diubah kembali pada menu "Preferences".

## Bab 7

# Skematik Hirarkis


### 7.1 Pengenalan

Representasi hirarkis secara umum menjadi solusi yang baik untuk proyek yang lebih besar dari beberapa lembar kerja. Jika Anda ingin menangani proyek semacam ini, Anda perlu untuk:

- Menggunakan lembar kerja yang besar, yang akan menyebabkan skematik menjadi sulit ditangani dan dicetak.
- Menggunakan beberapa lembar kerja, yang akan menjadikan skematik terstruktur secara hirarkis.

Skematik yang lengkap akan terdiri dari satu lembar kerja utama, yang disebut lembar kerja *root*, dan anak lembar kerja yang menyusun hirarki. Pembagian desain menjadi beberapa lembar kerja yang terpisah bisa memudahkan pembacaan.

Dari lembar kerja utama, Anda akan menemukan semua anak lembar kerja. Manajemen skematik hirarkis sangat mudah di-

lakukan di Eeschema, berkat adanya "navigator hirarki" yang dapat diakses melalui ikon  pada bilah atas bagian atas.

Ada dua tipe hirarki, di mana keduanya bisa digunakan bersamaan: tipe pertama telah umum digunakan. Tipe kedua terdiri dari pembuatan simbol-simbol pada pustaka yang tampak seperti simbol biasa di skematik, tetapi sebenarnya berhubungan dengan sebuah skematik yang menjelaskan struktur internalnya.

Tipe yang kedua digunakan untuk mengembangkan *Integrated Circuit*, karena dalam kasus ini Anda harus menggunakan pustaka fungsi di dalam skematik.

Eeschema saat ini tidak mendukung tipe kedua.

Sebuah hirarki bisa:

- sederhana: satu lembar kerja hanya digunakan satu kali saja
- kompleks: satu lembar kerja dapat digunakan lebih dari satu kali (beberapa kali kemunculan)
- rata: adalah hirarki yang sederhana, namun koneksi antar lembar kerja tidak digambarkan.


Eeschema mendukung semua hirarki di atas.

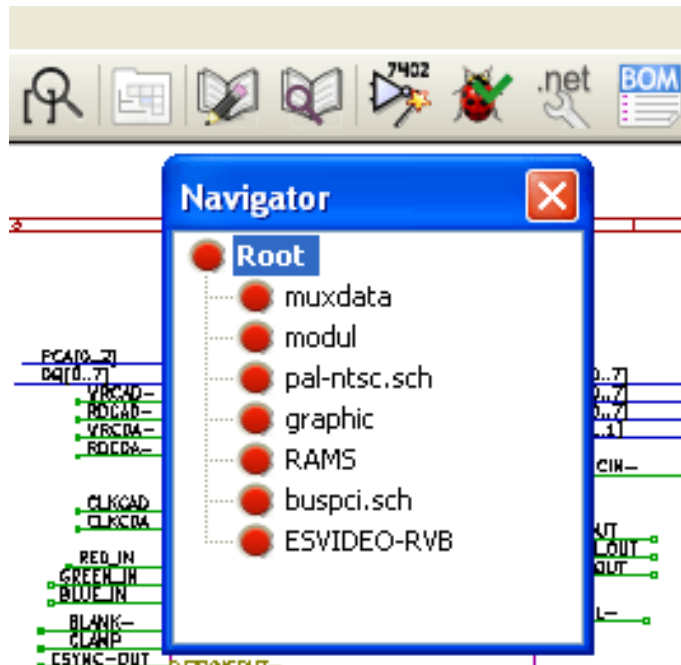
Pembuatan skematik hirarkis sangat mudah, keseluruhan hirarki ditangani mulai dari skematik utama, seperti jika anda hanya memiliki satu skematik.

Dua langkah penting untuk dipahami antara lain:

- Bagaimana cara membuat anak lembar kerja.
  - Bagaimana cara membangun koneksi-koneksi elektrik antar anak lembar kerja.
-

## 7.2 Penelusuran Hirarki

Penelusuran antara anak lembar kerja dapat dilakukan dengan menggunakan peralatan navigasi yang dapat diakses melalui tombol  pada bilah alat bagian atas.






Untuk memilih lembar kerja, klik pada nama lembar kerja tersebut. Untuk membuka, klik-kanan pada nama lembar kerja, dan pilih *Enter Sheet* atau klik-ganda dalam batas lembar kerja.

Untuk keluar dari lembar kerja yang sedang dibuka menuju ke lembar kerja utama, klik-kanan pada skematik, dan pilih *Leave Sheet* pada menu konteks, atau tekan *Alt+Backspace*.

## 7.3 Label Lokal, Hirarkis, dan Global

### 7.3.1 Properti

Label lokal, ikon , menghubungkan sinyal-sinyal di dalam satu lembar kerja. Sedangkan label hirarkis (ikon ) menghubungkan sinyal-sinyal di dalam satu lembar kerja dan ke pin hirarkis yang berada di lembar kerja utama.

Label global (ikon ) menghubungkan sinyal-sinyal antar semua hirarki. Pin *power* yang tersembunyi (jenis *power in* dan *power out*) sama dengan label global karena pin-pin tersebut semuanya terhubung antar hirarki.

---

#### Catatan

Di dalam satu hirarki (sederhana atau kompleks), kita bisa menggunakan label hirarkis maupun label global.

---

## 7.4 Membuat Hirarki

Anda perlu melakukan:


---

- Meletakkan simbol hirarki "*sheet symbol*" di lembar kerja utama.
- Masuk ke skematik baru (anak lembar kerja) dengan menggunakan navigator, dan menggambar skematik seperti biasa.
- Menggambar koneksi-koneksi elektrik antar kedua skematik dengan meletakkan label global (HLabels) pada skematik yang baru (anak lembar kerja), dan label yang memiliki nama yang sama di lembar kerja utama, disebut SheetLabels. SheetLabels akan terhubung ke simbol lembar kerja dari lembar kerja utama ke elemen-elemen yang lain dari skematik, seperti pin-pin simbol standar.

## 7.5 Simbol Lembar Kerja

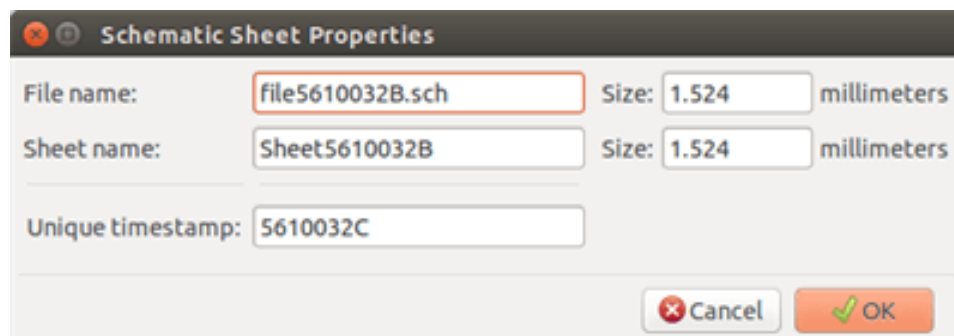
Kita perlu membuat sebuah persegi yang akan menggambarkan anak lembar kerja.

Ukuran dari persegi ini harus cukup untuk meletakkan label-label dan pin hirarkis, yang berhubungan dengan label global (HLabels) pada anak lembar kerja.

Label tersebut mirip seperti pin simbol biasa. Pilih ikon .

Klik untuk meletakkan sudut kiri atas persegi. Klik sekali lagi untuk meletakkan sudut kanan bawah, dan pastikan ukuran persegi cukup besar.

Akan muncul kotak dialog untuk mengatur nama berkas dan nama lembar kerja yang dibuat. Nama lembar kerja ini digunakan untuk mengakses skematik menggunakan navigator hirarki.



Anda harus menuliskan setidaknya sebuah nama berkas. Jika tidak ada nama lembar kerja, maka nama berkas akan digunakan sebagai nama lembar kerja.

## 7.6 Koneksi - Pin Hirarkis

Anda akan membuat titik koneksi (pin-pin hirarki) untuk simbol yang telah dibuat.

Titik-titik koneksi tersebut mirip dengan pin simbol normal, namun bisa menghubungkan sebuah *bus* yang lengkap dengan hanya satu titik koneksi.

Ada dua cara untuk membuatnya:

- Letakkan pin-pin yang berbeda sebelum menggambar anak lembar kerja (peletakan manual).
- Letakkan pin-pin yang berbeda setelah menggambar anak lembar kerja dan label global (peletakan semi-otomatis).

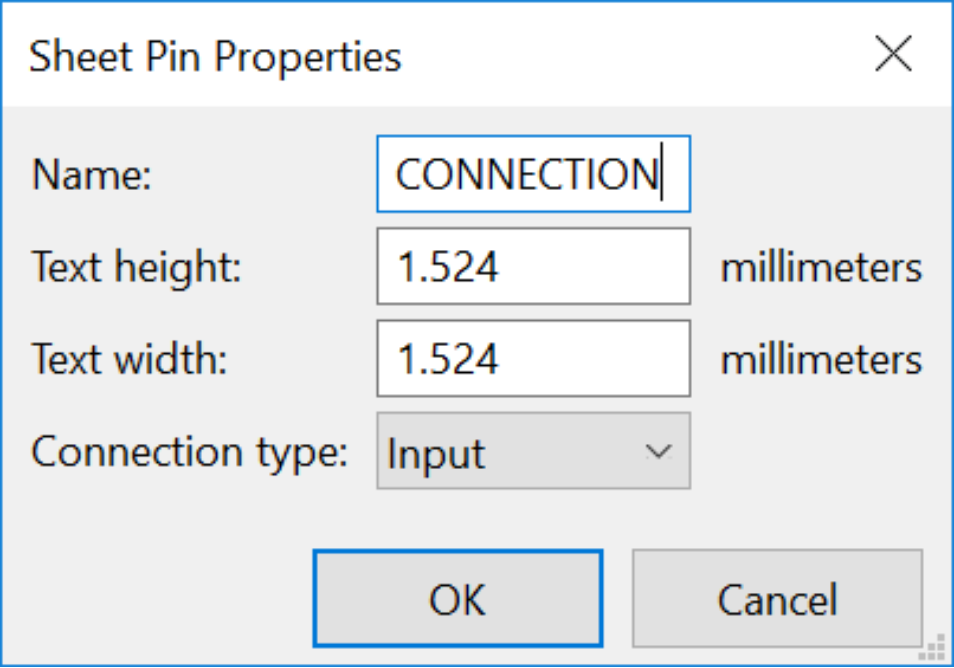
Lebih disarankan untuk menggunakan cara yang kedua.

**Peletakan manual:**

- Pilih ikon .

- Klik pada simbol hirarki di mana Anda ingin meletakkan pin.


Lihat gambar di bawah ini untuk contoh pembuatan pin hirarkis dengan nama "CONNECTION".

A screenshot of the 'Sheet Pin Properties' dialog box. The dialog has a title bar with a close button (X). Inside, there are four labeled input fields: 'Name:' with the text 'CONNECTION', 'Text height:' with the value '1.524', 'Text width:' with the value '1.524', and 'Connection type:' with a dropdown menu showing 'Input'. To the right of the 'Text height' and 'Text width' fields is the label 'millimeters'. At the bottom of the dialog are two buttons: 'OK' and 'Cancel'.

Anda bisa menentukan nama, ukuran dan arah pin pada saat pembuatan atau nanti, dengan klik-kanan pada pin dan pilih "*Edit Sheet Pin*" pada menu *pop-up*.

Di dalam lembar kerja, kita harus membuat sebuah label hirarkis dengan nama yang sama dengan pin hirarkis. Penamaan harus dilakukan secara manual, dan harus dipastikan bahwa kedua nama tersebut sama. Itulah sebabnya mengapa lebih direkomendasikan untuk menggunakan metode kedua, seperti di bawah ini.


#### Peletakan secara otomatis:

- Pilih ikon .
- Klik pada simbol hirarki dari mana Anda ingin mengimpor pin-pin yang terkait dengan label global yang diletakkan pada skematik yang sesuai. Akan muncul sebuah pin hirarkis, jika ada label global baru. Pin ini tidak terkait dengan pin yang telah ada.
- Klik di mana Anda ingin meletakkan pin ini.

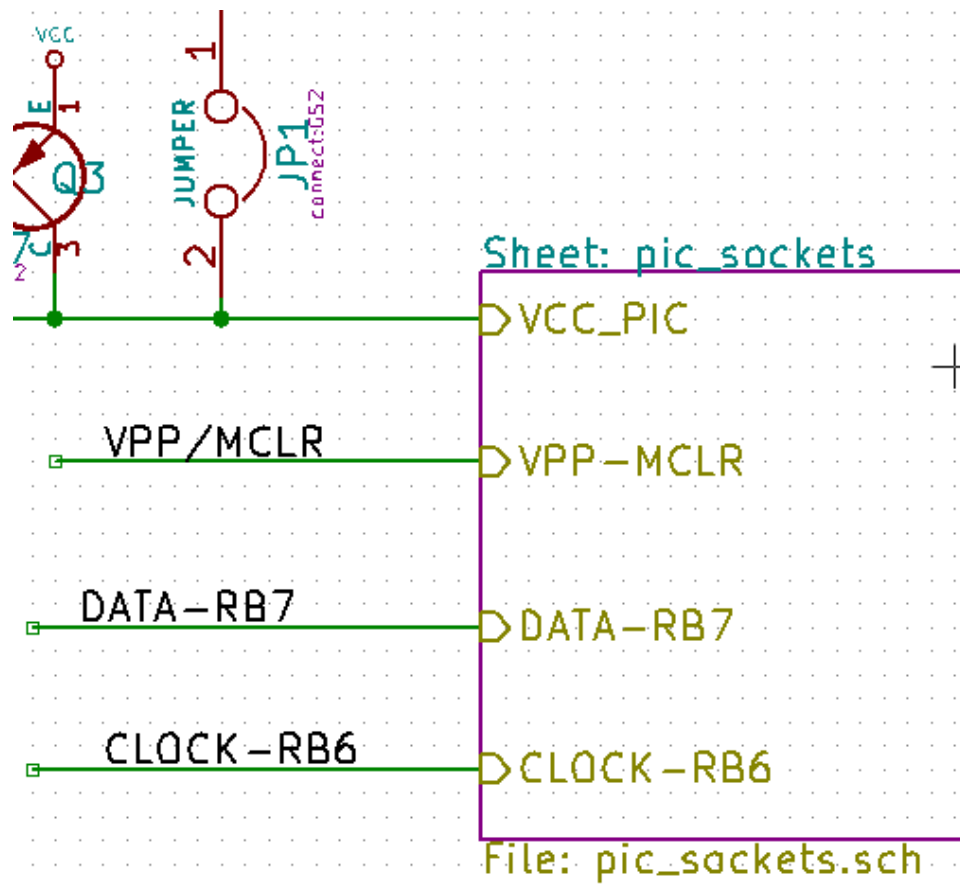
Dengan demikian, semua pin yang diperlukan dapat diletakkan dengan cepat dan tanpa kesalahan. Aspek mereka sesuai dengan label global yang terkait.

## 7.7 Koneksi - Label Hirarkis

Setiap pin dari simbol lembar kerja yang kita buat, harus terkait ke sebuah label yang disebut label hirarkis pada anak lembar kerja. Label hirarkis mirip dengan label biasa, namun label hirarkis bisa menghubungkan antara anak lembar kerja dengan lembar kerja utama. Representasi grafis dari dua label komplementer (pin dan HLabel) memiliki kemiripan. Pembuatan label hirarkis

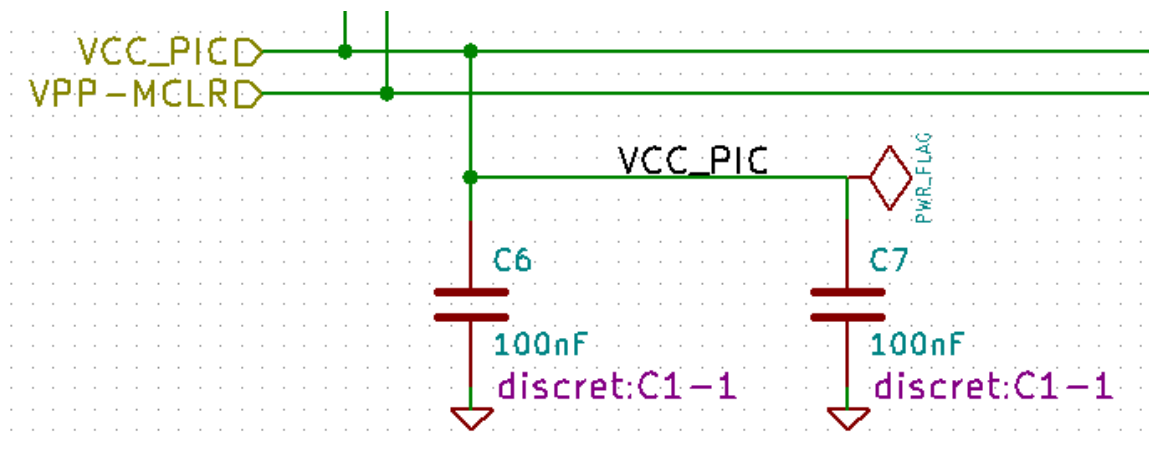
dilakukan dengan menggunakan ikon .

Lihat contoh lembar kerja utama di bawah ini:



Perhatikan pin VCC\_PIC, terhubung ke konektor JP1.

Berikut adalah koneksi yang terkait pada anak lembar kerja:



Kembali Anda temukan dua label hirarkis yang terkait, yang menghubungkan antar dua lembar kerja hirarkis.

#### Catatan

Anda bisa menggunakan label hirarkis dan pin hirarkis untuk menghubungkan dua *bus*, menurut sintaks (Bus [N..m]) yang telah dijelaskan sebelumnya.

### 7.7.1 Label, Label Hirarkis, Label Global, dan *Pin Power* Tersembunyi

Berikut beberapa komentar mengenai berbagai cara untuk membuat suatu koneksi, selain koneksi *wire*.

### 7.7.1.1 Label Sederhana

Label sederhana memiliki koneksi dengan kapasitas lokal, yaitu terbatas pada lembar kerja skematik di mana label tersebut diletakkan. Hal ini karena:

- Setiap lembar kerja memiliki nomor lembar kerja.
- Nomor lembar kerja ini terkait dengan label.

Maka, jika Anda meletakkan label "TOTO" pada lembar kerja n° 3, sebenarnya nama label ini adalah "TOTO\_3". Jika Anda juga meletakkan sebuah label "TOTO" pada lembar kerja n° 1 (lembar kerja utama), Anda sebenarnya meletakkan label "TOTO\_1", berbeda dari "TOTO\_3". Hal ini berlaku meskipun hanya ada satu lembar kerja saja.

### 7.7.1.2 Label Hirarkis

Sifat-sifat label sederhana juga berlaku untuk label hirarkis.

Dengan demikian pada lembar kerja yang sama, sebuah label hirarkis "TOTO" akan terhubung ke label lokal "TOTO", tetapi tidak terhubung ke label hirarkis atau label "TOTO" di lembar kerja lain.

Label hirarkis akan terhubung ke simbol pin lembar kerja yang sesuai pada simbol hirarkis yang diletakkan di lembar kerja utama.

### 7.7.1.3 Pin Power Tersembunyi

*Pin power* yang tersembunyi akan saling terhubung jika memiliki nama yang sama. Maka *pin power* tersembunyi dengan nama VCC akan terhubung dengan semua *pin power* tersembunyi lainnya yang juga bernama VCC, namun hanya di dalam satu lembar kerja saja.

Artinya, jika Anda meletakkan label VCC pada suatu anak lembar kerja, label tersebut tidak akan terhubung ke pin VCC, karena nama sebenarnya dari label ini adalah VCC\_n, di mana n adalah nomor lembar kerja.

Jika Anda ingin supaya label VCC tersebut terhubung dengan label VCC di keseluruhan skematik, Anda harus menghubungkannya secara eksplisit ke *pin power* tersembunyi melalui simbol *power VCC*.

## 7.7.2 Label Global

Label global yang memiliki nama yang sama akan terhubung di keseluruhan hirarki.

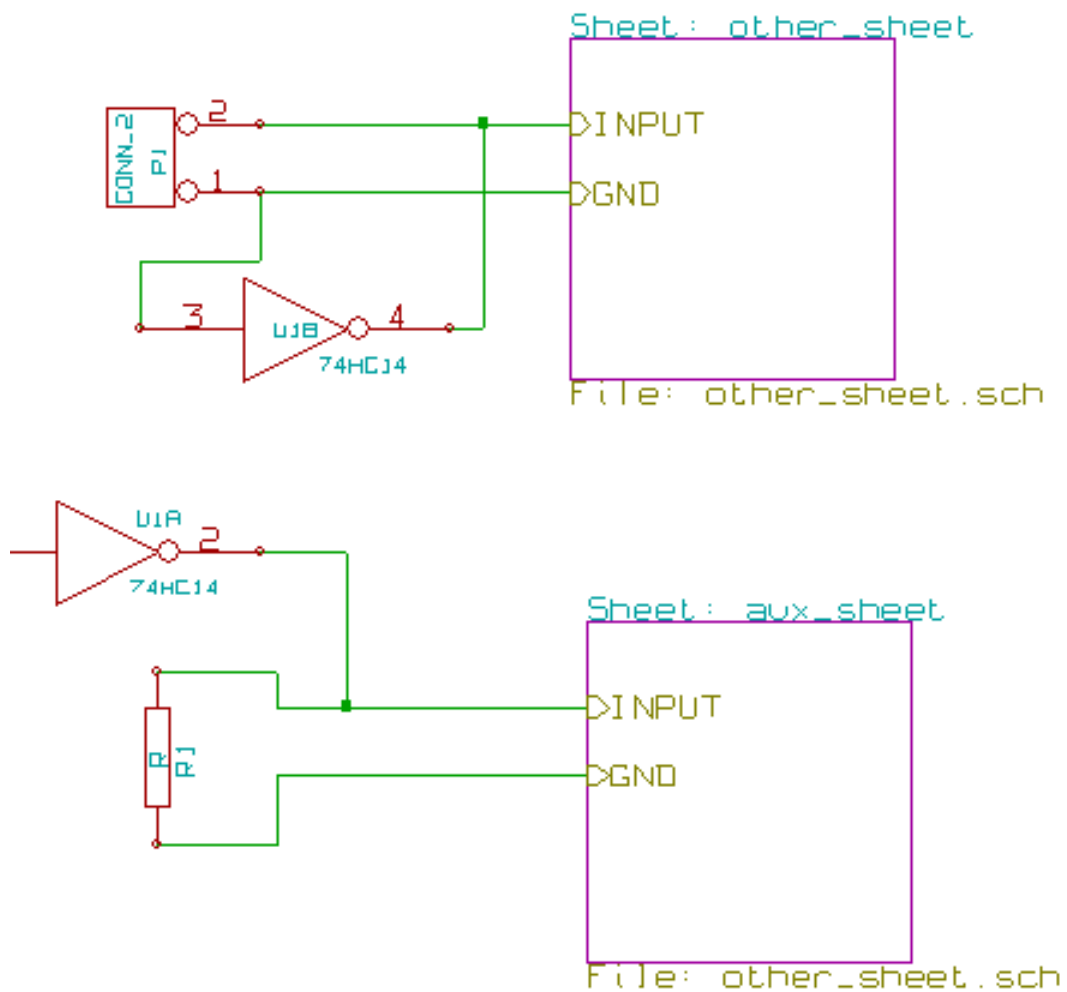
(label *power* seperti vcc ... adalah label global)

## 7.8 Hirarki Kompleks

Berikut adalah suatu contoh. Skematik yang sama digunakan sebanyak dua kali (dua kemunculan). Dua buah lembar kerja berbagi skematik yang sama karena kedua lembar kerja tersebut menggunakan nama berkas yang sama ("other\_sheet.sch"). Sedangkan nama lembar kerja harus unik.

---



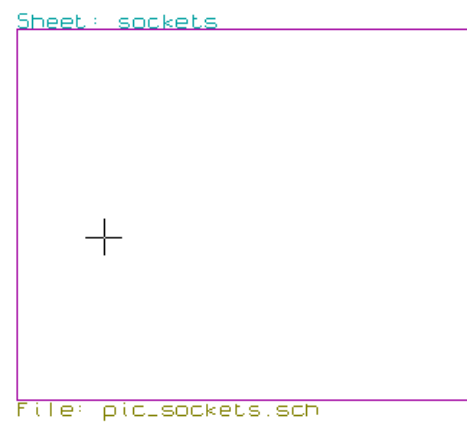
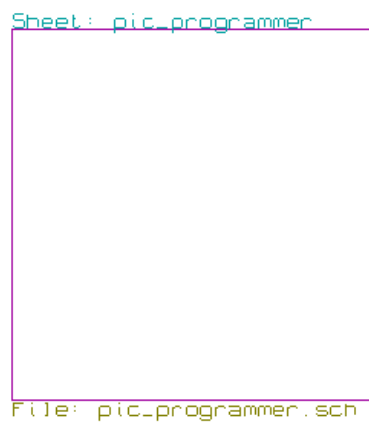


## 7.9 Hirarki Datar

Anda bisa membuat sebuah proyek menggunakan banyak lembar kerja tanpa harus membuat koneksi antar lembar kerja tersebut (hirarki datar) jika Anda mengikuti aturan berikut:

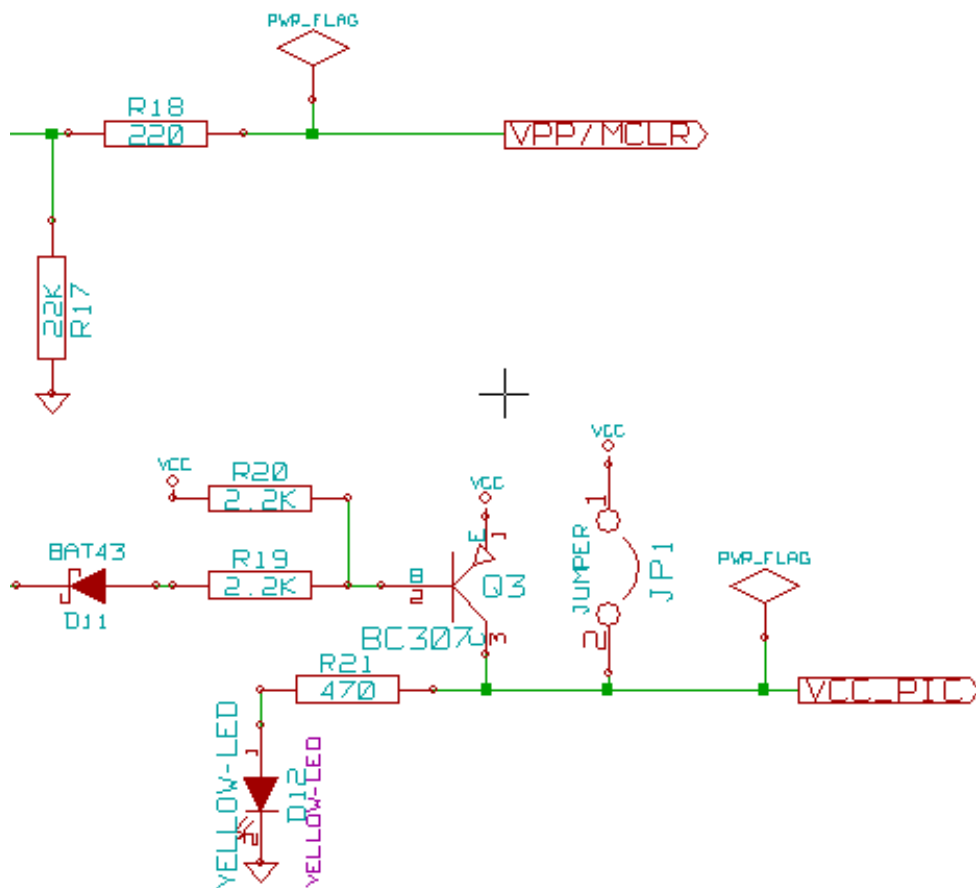
- Buat sebuah lembar kerja utama berisi lembar kerja yang lain, yang berfungsi sebagai tautan antara lembar kerja.
- Tidak diperlukan koneksi eksplisit.
- Gunakan label global pada semua lembar kerja, dan jangan menggunakan label hirarkis.

Berikut adalah contoh lembar kerja utama.

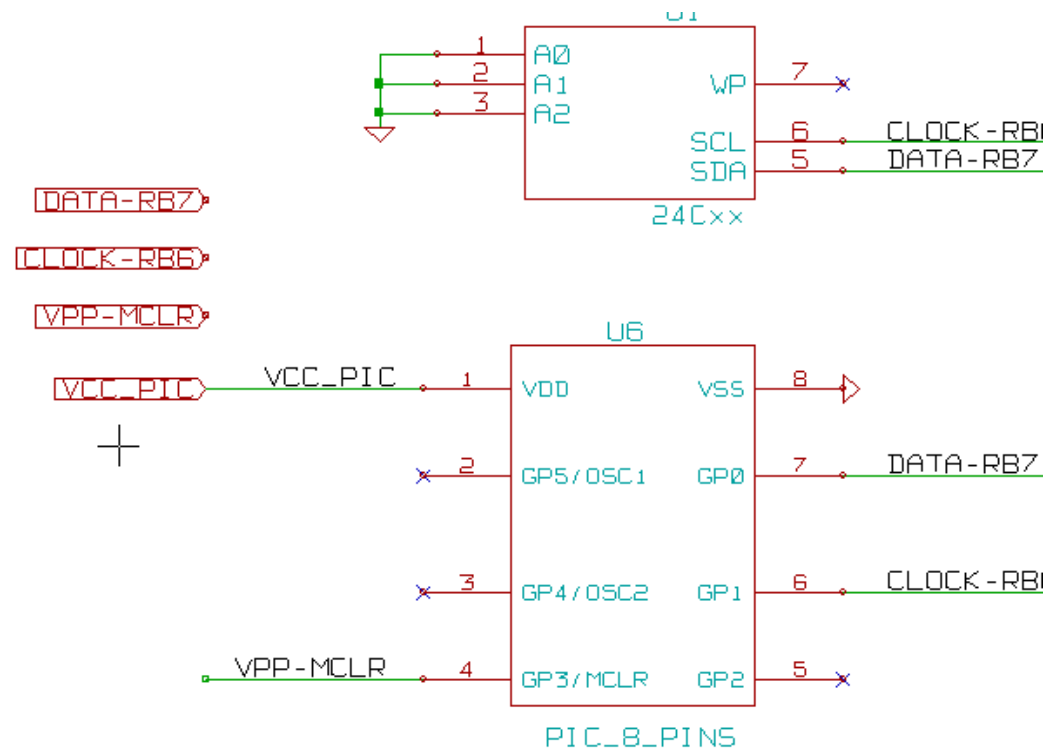


Berikut ini dua halaman yang terhubung oleh label global.

Berikut adalah pic\_programmer.sch.



Berikut adalah pic\_sockets.sch.




Perhatikan label global yang digunakan.

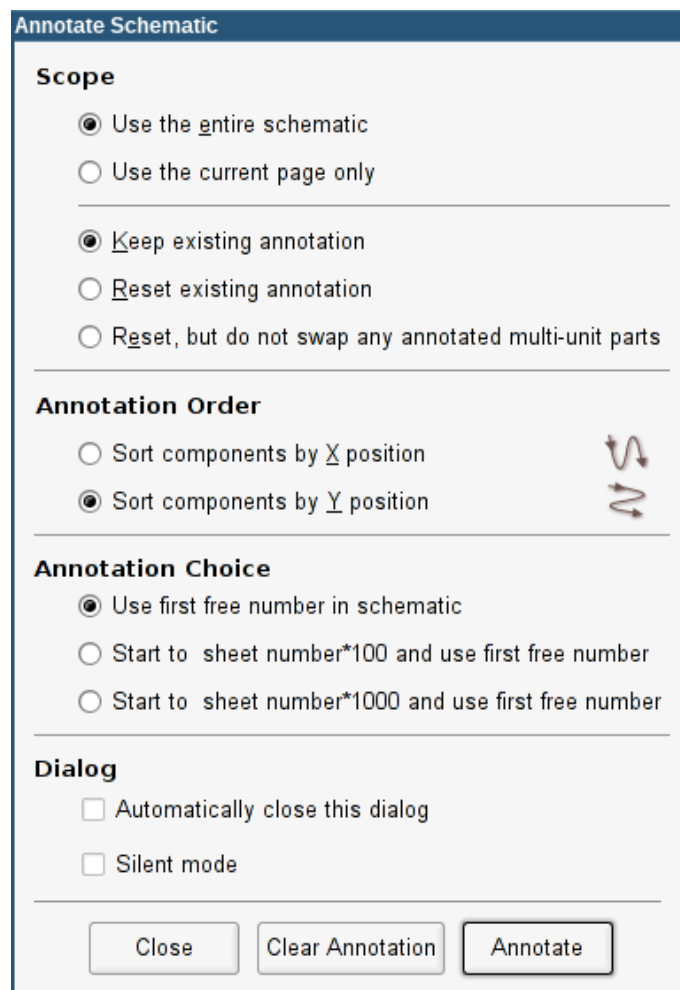


## Bab 8

# Alat Anotasi Simbol

### 8.1 Pengenalan

Alat anotasi digunakan untuk memberikan sebuah penunjuk secara otomatis ke simbol-simbol yang ada pada skematik. Memberikan anotasi pada simbol-simbol yang terdiri dari beberapa bagian akan menghasilkan sufiks yang unik untuk meminimalkan jumlah simbol tersebut. Alat anotasi dapat diakses melalui ikon . Gambar berikut menunjukkan tampilan jendela utamanya.



Berikut skema anotasi yang tersedia:

- Memberikan anotasi pada semua simbol (mereset pilihan anotasi sebelumnya)
- Memberikan anotasi pada semua simbol, namun jangan melakukan anotasi ulang pada simbol yang terdiri dari beberapa bagian, yang telah mendapatkan anotasi sebelumnya.
- Memberikan anotasi hanya pada simbol yang belum mendapatkan anotasi sebelumnya. Simbol yang belum mendapatkan anotasi akan memiliki penunjuk yang berakhir dengan tanda "?".
- Memberikan anotasi pada keseluruhan hirarki (gunakan opsi *Use the entire schematic*).
- Memberikan anotasi hanya pada lembar kerja yang sedang dibuka (gunakan opsi *Use the current page only*).

Opsi *Reset, but do not swap any annotated multi-unit parts* akan mempertahankan semua asosiasi yang telah ada pada simbol yang terdiri dari beberapa bagian. Sebagai contoh, U2A dan U2B bisa mendapatkan anotasi ulang ke U1A dan U1B, namun mereka tidak akan mendapatkan anotasi ulang menjadi U1A dan U2A, ataupun ke U2B dan U2A. Hal ini berguna jika Anda ingin memastikan agar pengelompokan pin tetap terjaga.

Kita bisa memilih urutan anotasi untuk mengatur nomor referensi di dalam setiap lembar kerja pada hirarki.

Kecuali untuk kasus-kasus tertentu, anotasi otomatis dapat diterapkan ke keseluruhan proyek (semua lembar kerja) dan ke komponen baru, jika Anda tidak ingin mengubah hasil anotasi sebelumnya.

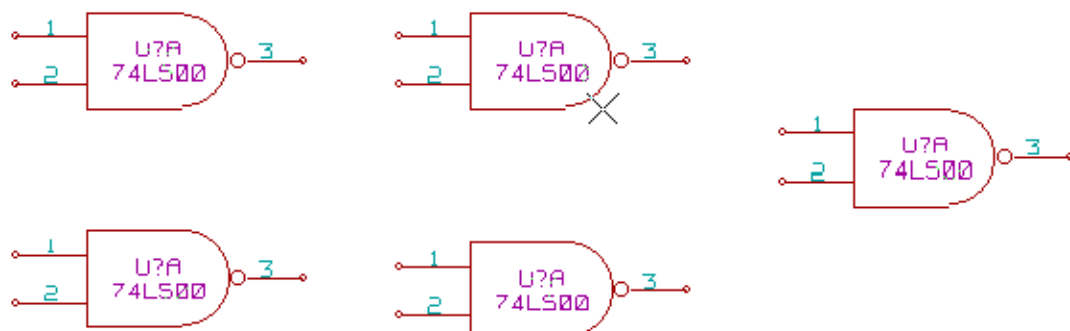
Kita bisa memilih metode yang digunakan untuk melakukan kalkulasi referensi:

- *Use first free number in schematic*: komponen akan diberikan anotasi mulai dari angka 1 (untuk tiap prefiks referensi). Jika sudah ada anotasi sebelumnya, maka nomor yang akan diberikan hanya nomor yang belum digunakan.
- *Start to sheet number\*100 and use first free number*: anotasi akan dimulai dari 101 untuk lembar kerja 1, dari 201 untuk lembar kerja 2, dst. Jika terdapat lebih dari 99 item yang memiliki prefiks referensi yang sama (U, R) di dalam lembar kerja 1, maka alat anotasi akan menggunakan nomor 200 dan seterusnya, dan anotasi untuk lembar kerja 2 akan dimulai dari nomor bebas berikutnya.
- *Start to sheet number\*1000 and use first free number*: anotasi dimulai dari 1001 untuk lembar kerja 1, dari 2001 untuk lembar kerja 2.

## 8.2 Contoh Penggunaan

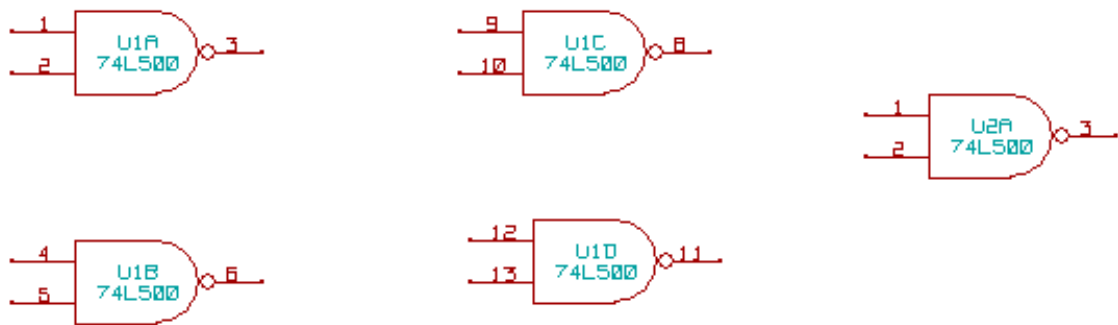
### 8.2.1 Urutan Anotasi

Contoh berikut menunjukkan 5 buah elemen yang telah diletakkan, namun belum dilakukan anotasi.

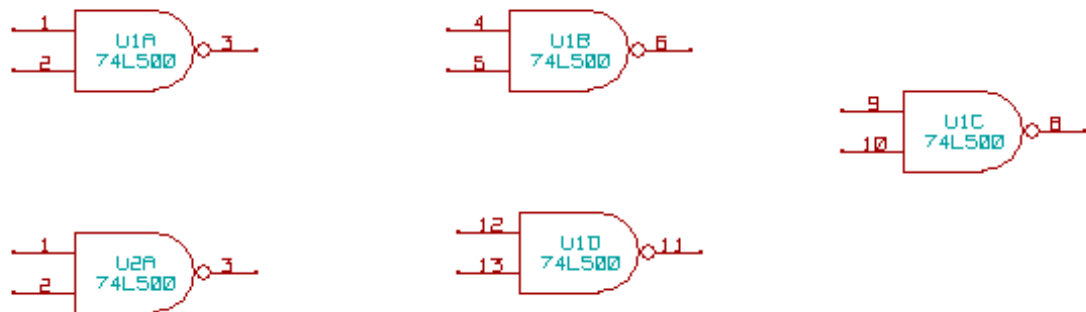


Setelah alat anotasi dijalankan, didapatkan hasil berikut.

Diurutkan berdasarkan sumbu X.



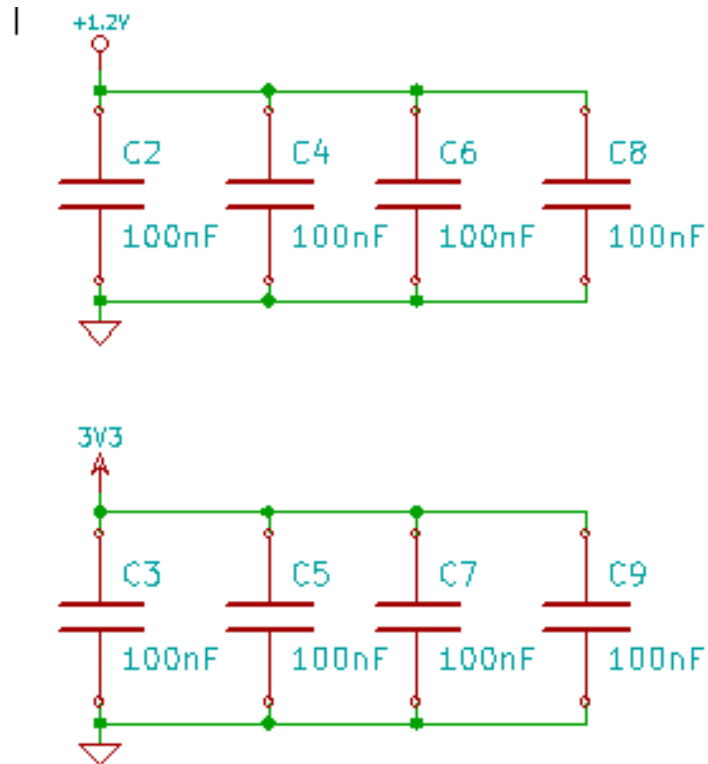
Diurutkan berdasarkan sumbu Y.



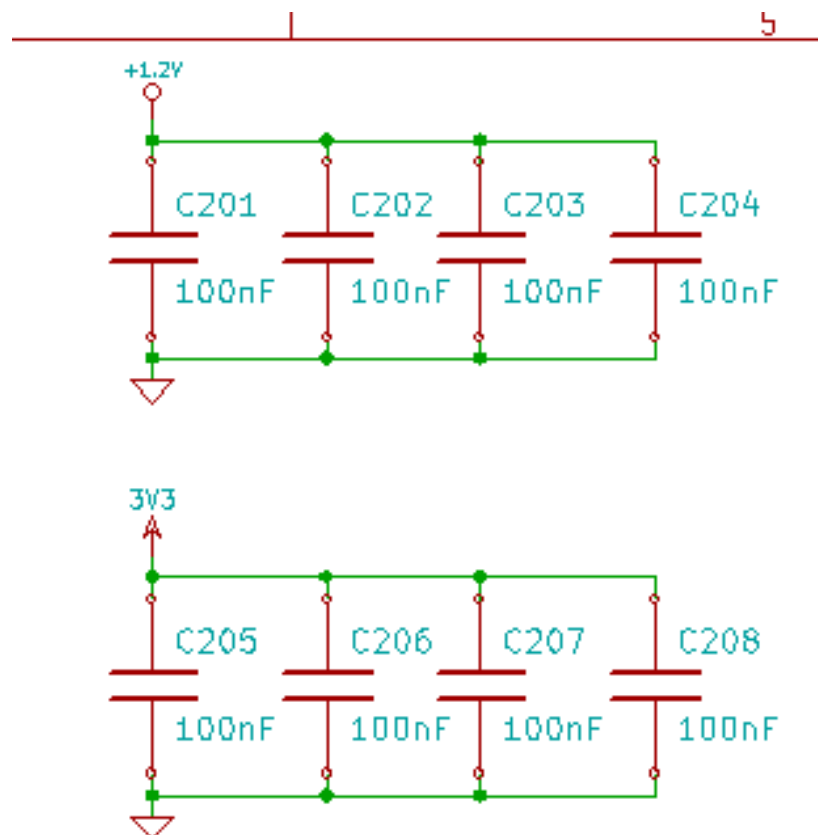
Amati bahwa keempat *gate* 74LS00 didistribusikan sebagai paket U1, sedangkan 74LS00 yang kelima mendapatkan penomoran berikutnya, yaitu U2.

### 8.2.2 Pilihan Anotasi

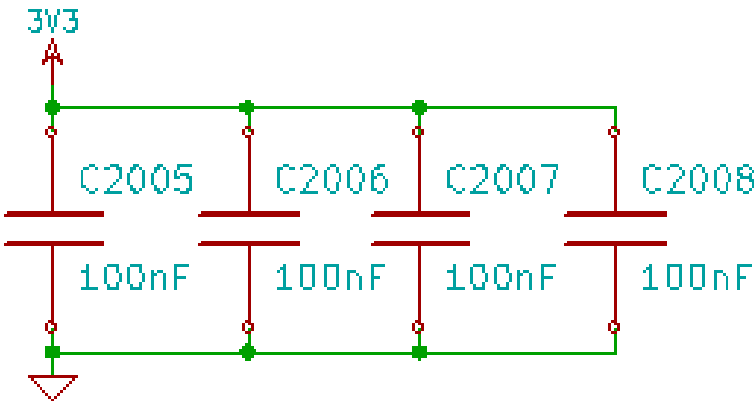
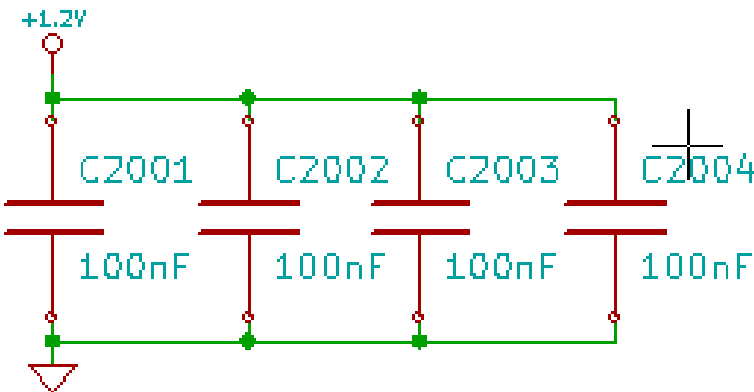
Berikut adalah anotasi pada lembar kerja 2 di mana digunakan opsi *Use first free number in schematic*.



Opsi start to sheet number\*100 and use first free number memberikan hasil berikut.



Opsi Start to sheet number\*1000 and use first free number memberikan hasil berikut.





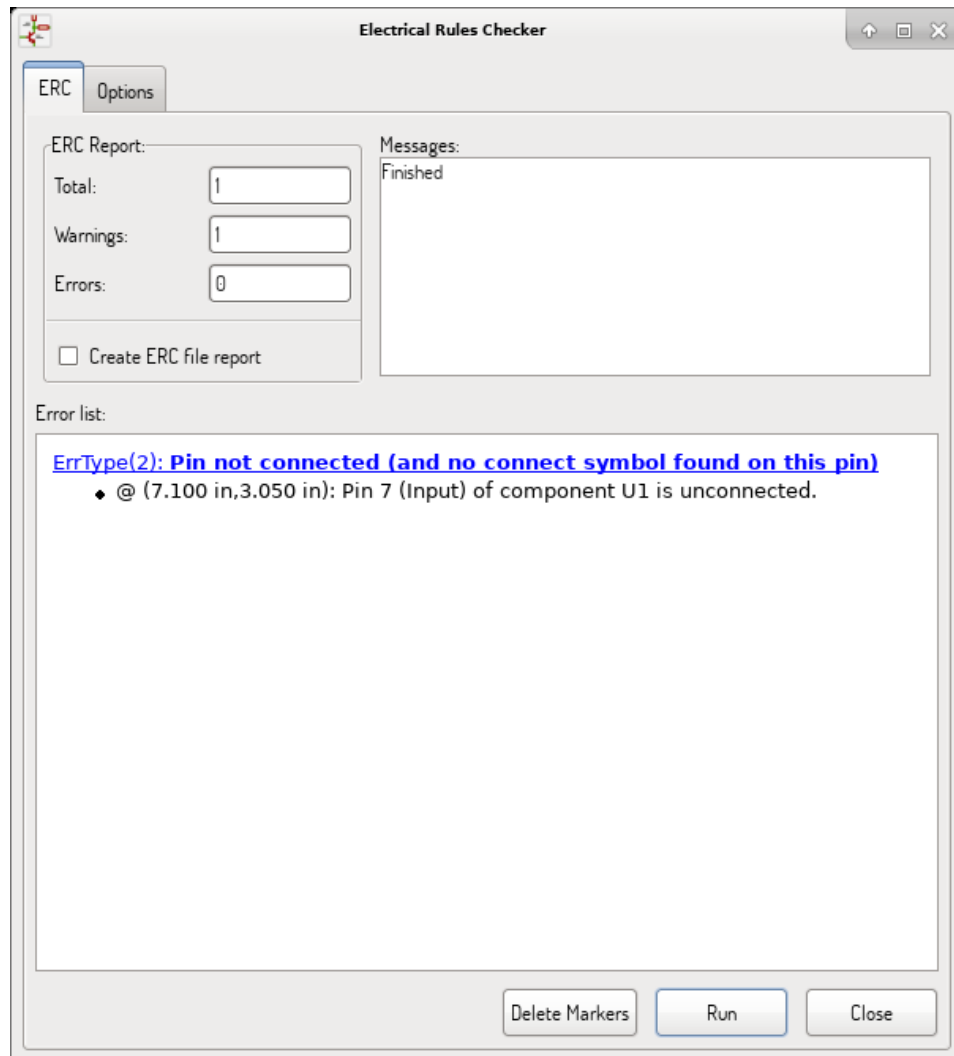
## Bab 9

# Verifikasi Desain dengan *Electrical Rules Check*


### 9.1 Pengenalan

*Electrical Rules Check* (ERC) bertugas melakukan pemeriksaan otomatis pada skematik Anda. ERC akan memeriksa adanya kesalahan pada lembar kerja Anda, seperti pin yang belum terhubung, simbol hirarkis yang belum terhubung, keluaran yang mengalami hubung-singkat, dsb. Namun perlu diingat bahwa pemeriksaan otomatis tidak bisa memeriksa secara sempurna, dan perangkat lunak yang mampu mendeteksi semua kesalahan desain belum sepenuhnya lengkap. Pemeriksaan seperti ini sangat berguna, karena Anda bisa mendeteksi kekeliruan dan kesalahan-kesalahan kecil dengan mudah.

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 Penggunaan ERC

ERC dapat diakses melalui ikon .

Akan muncul penanda pada elemen skematik yang menyebabkan kesalahan ERC (pin atau label).

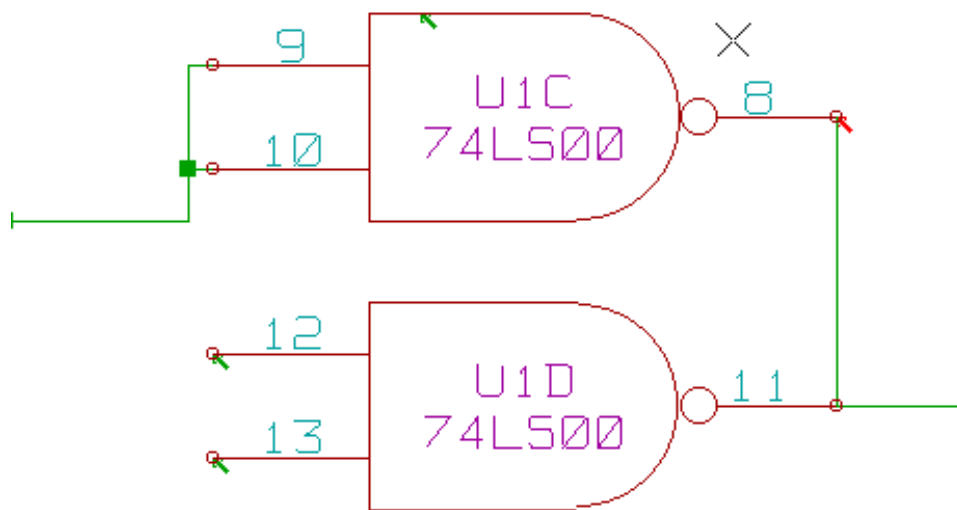
---

### Catatan

- Pada kotak dialog ini, saat Anda melakukan klik pada pesan kesalahan, Anda akan diarahkan ke penanda yang terkait pada skematik.
  - Pada skematik, klik-kanan pada penanda untuk mengakses pesan hasil pemeriksaan yang terkait.
- 

Anda juga bisa menghapus penanda kesalahan dari kotak dialog.

### 9.3 Contoh ERC

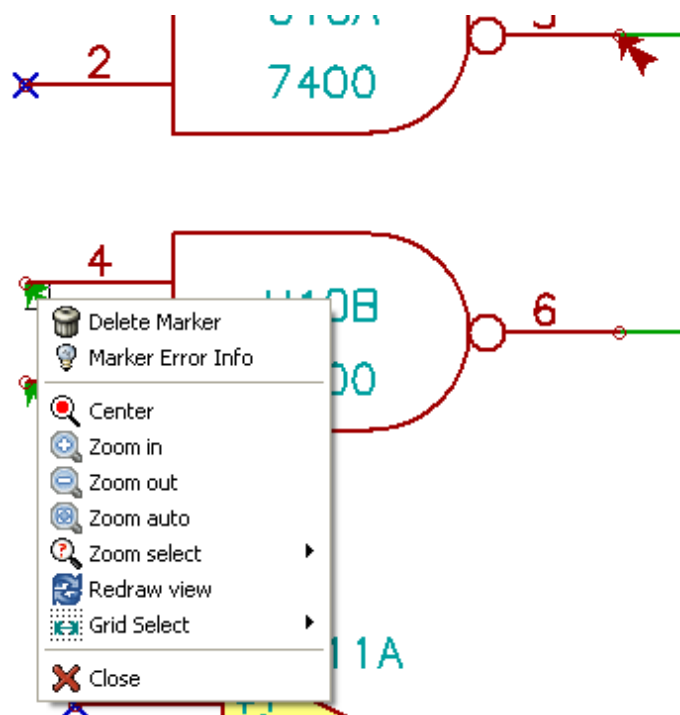


Pada gambar di atas, Anda bisa melihat adanya empat kesalahan:

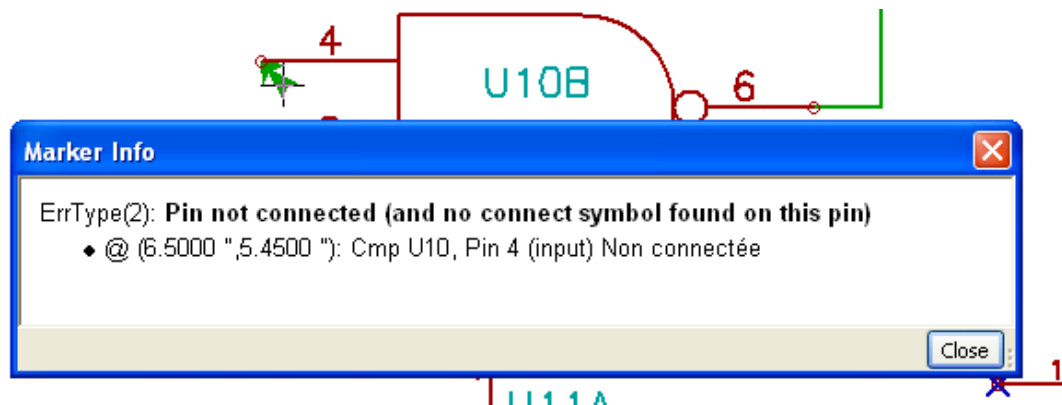
- Dua keluaran telah tersambung secara tidak sengaja (tanda panah warna merah).
- Dua masukan belum tersambung (tanda panah warna hijau).
- Terdapat kesalahan pada *port power* tersembunyi, dimana tidak ada bendera *power* (tanda panah warna hijau di atas).

### 9.4 Menampilkan Hasil Pemeriksaan

Klik-kanan pada penanda kesalahan. Akan muncul menu *pop-up* untuk mengakses kotak dialog hasil pemeriksaan ERC.



Klik *Marker Error Info* untuk menampilkan deskripsi kesalahan.

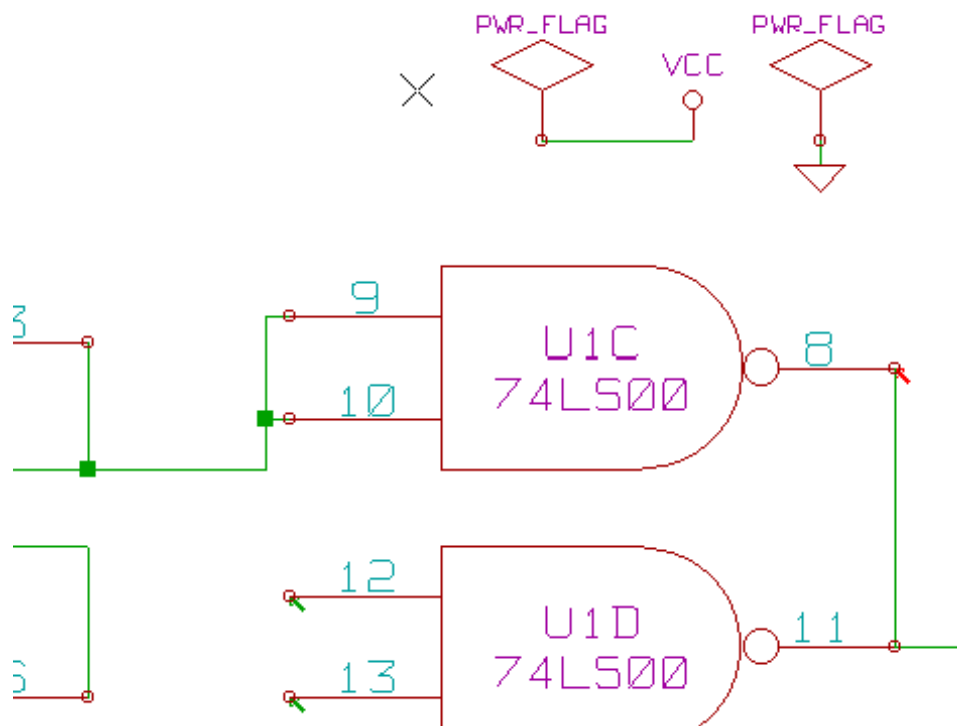


## 9.5 Pin dan Bendera *Power*

Kita akan sering menjumpai kesalahan atau peringatan pada pin-pin *power*, meskipun sepertinya semuanya tampak normal. Perhatikan contoh di atas. Hal ini terjadi karena, pada kebanyakan desain, *power* disediakan oleh konektor yang bukan merupakan sumber tenaga (seperti keluaran regulator, yang dinyatakan sebagai *Power out*).

Maka, ERC tidak akan menemukan adanya pin *Power out* untuk mengendalikan *wire* ini, dan akan menyatakan bahwa *wire* tersebut tidak terhubung ke sumber tenaga.

Untuk menghindari peringatan ini, Anda harus meletakkan sebuah "PWR\_FLAG" pada *port power* tersebut. Perhatikan contoh berikut:

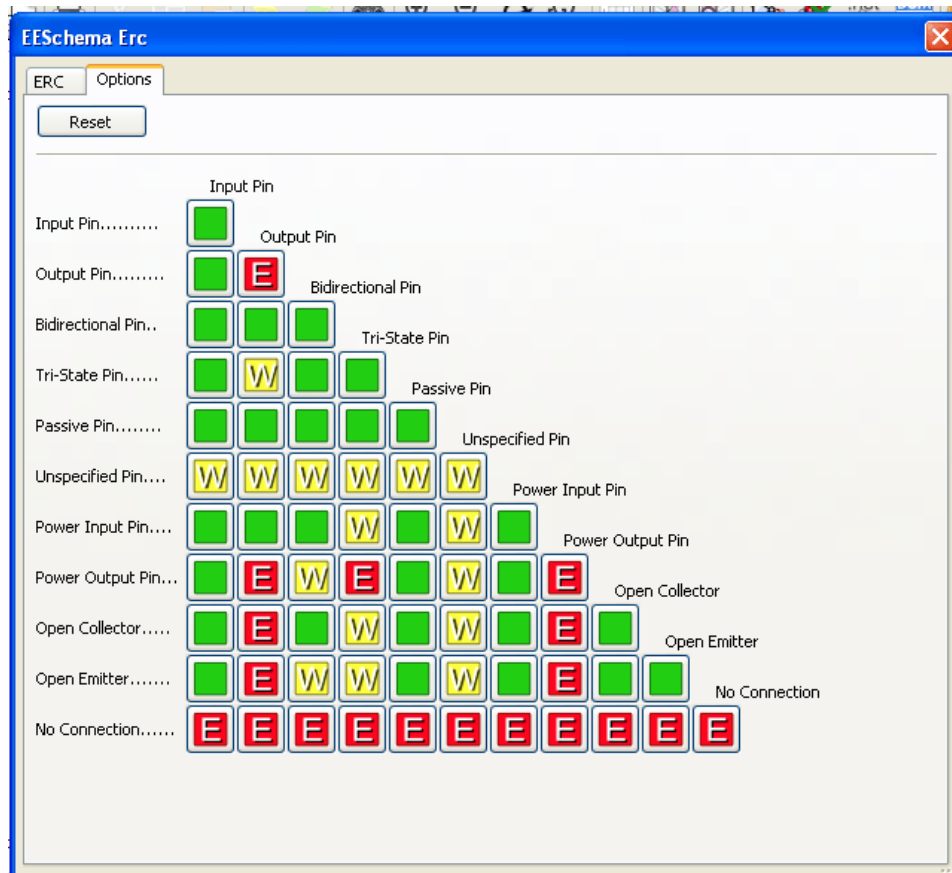


Penanda kesalahan akan menghilang.

Seringkali, PWR\_FLAG harus dihubungkan ke GND, karena regulator memiliki keluaran yang dinyatakan sebagai *Power out*, akan tetapi pin *ground* bukanlah *Power out* (atribut normalnya adalah *Power in*), sehingga tanpa adanya simbol PWR\_FLAG, *ground* tidak akan tampak terhubung ke sumber tenaga.

## 9.6 Konfigurasi

Tab *Options* berguna untuk melakukan konfigurasi aturan konektivitas sebagai acuan pemeriksaan kondisi elektrik dari adanya kesalahan atau peringatan.



Aturan dapat diubah dengan melakukan klik pada kotak yang diinginkan pada matriks. Pilihan yang tersedia adalah *normal*, *warning*, dan *error*.

## 9.7 Berkas Laporan ERC

Berkas laporan ERC dapat dibuat dan disimpan dengan memilih opsi *Write ERC report*. Ekstensi berkas untuk laporan ERC adalah *.erc*. Berikut adalah contoh dari berkas laporan ERC.

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

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

>> Errors ERC: 4
```

## Bab 10

# Membuat Netlist

### 10.1 Gambaran Umum

Netlist adalah sebuah berkas yang berisi penjelasan koneksi elektrik antar simbol. Koneksi tersebut disebut sebagai *net*. Pada berkas Netlist, Anda akan menemukan:

- Daftar simbol
- Daftar koneksi (*net*) antar simbol.

Ada beragam format Netlist yang berbeda. Terkadang, daftar simbol dan daftar *net* dibuat menjadi dua berkas yang terpisah. Netlist ini sangat penting dalam penggunaan perangkat lunak pembuat skematik, karena Netlist akan menjadi penghubung dengan perangkat lunak CAD elektronik lainnya seperti:


- Perangkat lunak pembuat tata letak PCB.
- Simulator skematik dan sinyal elektrik.
- Kompiler CPLD (dan IC terprogram lainnya).

Eeschema mendukung beberapa format Netlist:

- Format PCBNEW (papan PCB).
- Format PCB2 ORCAD (papan PCB).
- Format CADSTAR (papan PCB).
- Format Spice, untuk berbagai simulator (format Spice juga digunakan untuk berbagai simulator lain).

### 10.2 Format-format Netlist



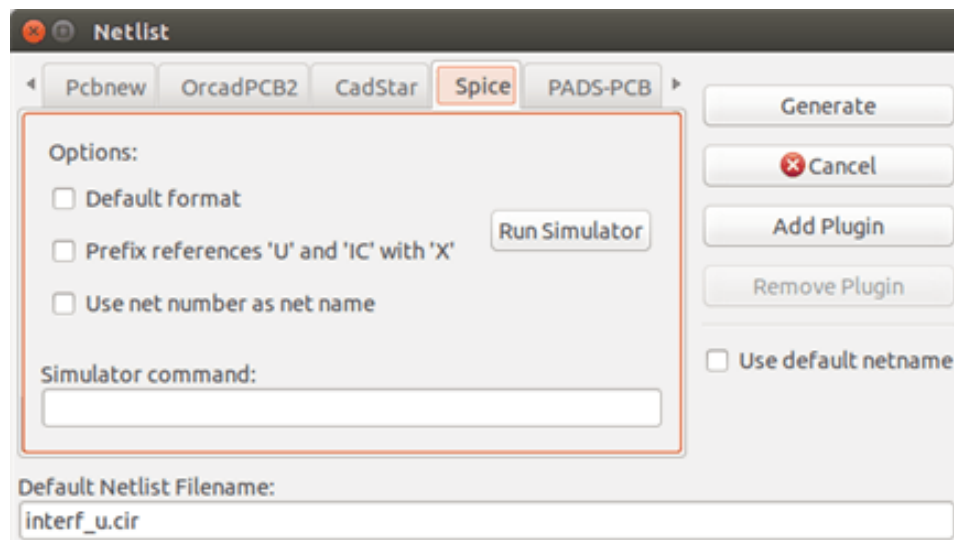
Pilih ikon  untuk membuka kotak dialog pembuatan Netlist.

Tab Pcbnew

---



Tab Spice



Anda dapat memilih format yang diinginkan dengan memilih tab yang sesuai. Pada format Spice, Anda dapat membuat Netlist dengan menggunakan nama *net* yang akan menjadikan berkas SPICE lebih mudah dibaca, atau menggunakan nomor *net* yang digunakan oleh Spice versi lama. Setelah menekan tombol Netlist, Anda harus menuliskan nama Netlist yang diinginkan.

---

**Catatan**

Pembuatan Netlist bisa memerlukan waktu hingga beberapa menit untuk skematik yang besar.

---

## 10.3 Contoh Netlist

Contoh berikut ini adalah sebuah desain skematik menggunakan pustaka PSPICE:



#### Contoh berkas Netlist PCBNEW:

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



```
(1 EMET_1)
(2 0)
)
(32CFC254 $noname R4 1K {Lib=R}
(1 +12V)
(2 VOUT_1)
)
(32CFC24C $noname R3 1K {Lib=R}
(1 +12V)
(2 N-000006)
)
(32CFC230 $noname Q2 Q2N2222 {Lib=NPN}
(1 VOUT_1)
(2 N-000008)
(3 EMET_1)
)
(32CFC227 $noname Q1 Q2N2222 {Lib=NPN}
(1 N-000006)
(2 INPUT_1)
(3 EMET_1)
)
)
# End
```

Berikut adalah Netlist dalam format PSPICE:

```
* 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 Catatan Mengenai Netlist

### 10.4.1 Penamaan Netlist

Kebanyakan perangkat lunak yang menggunakan Netlist tidak menerima penggunaan spasi pada nama komponen, pin, *net*, atau informasi-informasi lainnya. Hindari penggunaan spasi pada label, atau atribut nama atau nilai pada komponen atau pin-pinnya, untuk memastikan kompatibilitas maksimum.

Karakter khusus selain huruf dan angka juga bisa menyebabkan masalah. Pembatasan penggunaan karakter ini bukan dari Eeschema, akan tetapi dari format Netlist yang bisa menjadikannya tidak dapat dibaca oleh perangkat lunak yang akan menggunakan berkas Netlist tersebut.

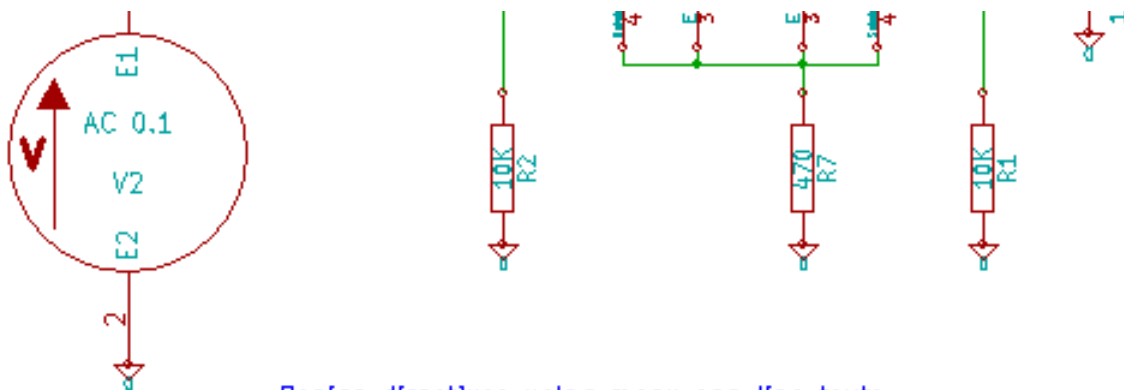
### 10.4.2 Netlist PSPICE

Untuk simulator Pspice, Anda harus memasukkan beberapa baris perintah pada Netlist itu sendiri (.PROBE, .AC, dsb).

Setiap baris teks yang ada pada diagram skematik yang dimulai dari kata kunci **-pspice** atau **-gnuap** akan dimasukkan (tanpa kata kunci) pada bagian atas Netlist.

Setiap baris teks yang ada pada diagram skematik yang dimulai dari kata kunci **+pspice** atau **+gnuap** akan dimasukkan (tanpa kata kunci) pada bagian akhir netlist.

Berikut adalah contoh penggunaan beberapa teks satu baris dan satu teks dengan lebih dari satu baris:



Pspice directives using many one line texts

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

Pspice directives using one multiline text:

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

Sebagai contoh, jika Anda menuliskan teks berikut (jangan menggunakan label!):

```
-PSPICE .PROBE
```

Sebuah baris `.PROBE` akan dimasukkan ke dalam Netlist.

Pada contoh sebelumnya, ada tiga baris yang dimasukkan di awal Netlist, dan dua di akhir Netlist dengan teknik ini.

Jika Anda menggunakan teks dengan lebih dari satu baris, kata kunci **+pspice** atau **+gnuap** hanya diperlukan satu kali saja:

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

menghasilkan empat baris:

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

Perhatikan juga bahwa *net* GND harus dinamai 0 (*zero*) untuk Pspice.

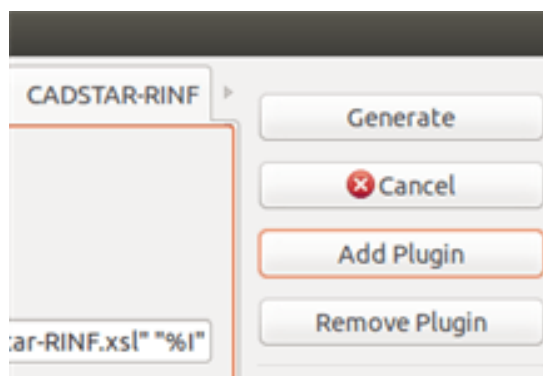
## 10.5 Format Lain

Untuk format Netlist yang lain, Anda bisa menambahkan konverter Netlist dalam bentuk *plugin*. Konverter tersebut akan dijalankan secara otomatis oleh Eeschema. Bab 14 membahas beberapa penjelasan dan contoh konverter.

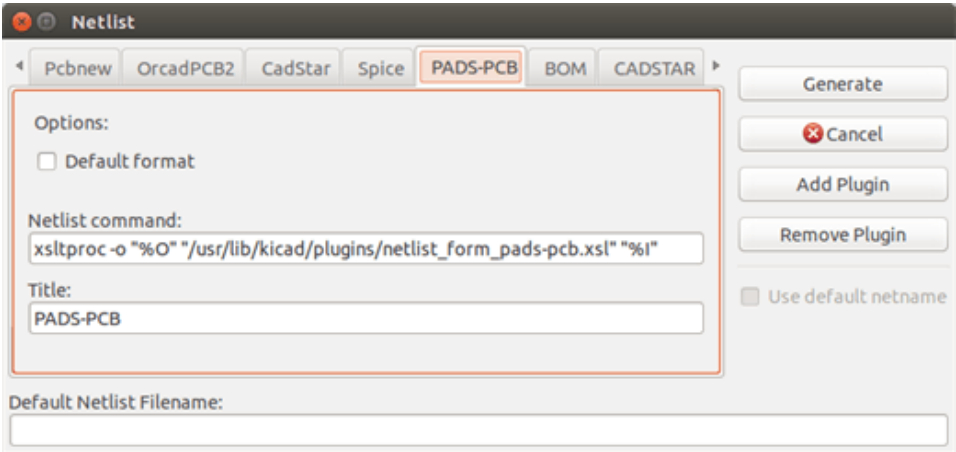
Sebuah konverter adalah berupa berkas teks (format `xsl`), namun kita juga bisa menggunakan bahasa lain seperti Python. Ketika menggunakan format `xsl`, sebuah alat (`xsltproc.exe` atau `xsltproc`) membaca berkas antara yang dibuat oleh Eeschema, dan berkas konverter untuk menghasilkan berkas keluaran. Pada kasus ini, berkas konverter (dalam gaya lembar kerja) berukuran sangat kecil dan sangat mudah ditulis.

### 10.5.1 Memulai Kotak Dialog

Anda bisa menambahkan sebuah *plugin* Netlist melalui tombol *Add Plugin*.



Berikut tampilan pengaturan *plugin* PadsPcb:



Pengaturan ini membutuhkan:

- Sebuah judul (sebagai contoh, nama dari format Netlist).
- *Plugin* yang akan dijalankan.

Ketika Netlist telah berhasil dibuat:

1. Eeschema membuat sebuah berkas antara \*.tmp, sebagai contoh test.tmp.
2. Eeschema menjalankan *plugin*, yang akan membaca test.tmp dan kemudian membuat test.net.

### 10.5.2 Format Baris Perintah

Berikut adalah contoh penggunaan `xsltproc.exe` sebagai alat untuk mengkonversi berkas `.xml`, dan sebuah berkas `netlist_form_pads-pcb.xml` sebagai gaya lembar kerja konverter:

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

Dengan:

f:/kicad/bin/xsltproc.exe	Alat untuk membaca dan mengonversi berkas xml
-o %O.net	Berkas keluaran: %O akan menentukan berkas keluaran.
f:/kicad/bin/plugins/netlist_form_pads-pcb.xml	Konverter nama berkas (gaya lembar kerja, dalam format xml).
%I	Akan digantikan oleh berkas antara yang dibuat oleh Eeschema (*.tmp).

Untuk sebuah skematik dengan nama test.sch, baris perintahnya adalah:

f:/kicad/bin/xsltproc.exe -o test.net f:/kicad/bin/plugins/netlist\_form\_pads-pcb.xml test.tmp.

### 10.5.3 Konverter dan Gaya Lembar Kerja (*Plugin*)

Ini adalah perangkat lunak yang sangat sederhana, karena tugasnya hanya untuk mengkonversi sebuah berkas teks masukan (berkas teks antara) menjadi berkas teks yang lain. Lebih lanjut lagi, dari berkas teks antara, Anda dapat membuat sebuah daftar BOM.

Ketika menggunakan `xsltproc` sebagai konverter, hanya gaya lembar kerja yang akan dibuat.

### 10.5.4 Format Berkas Netlist Antara

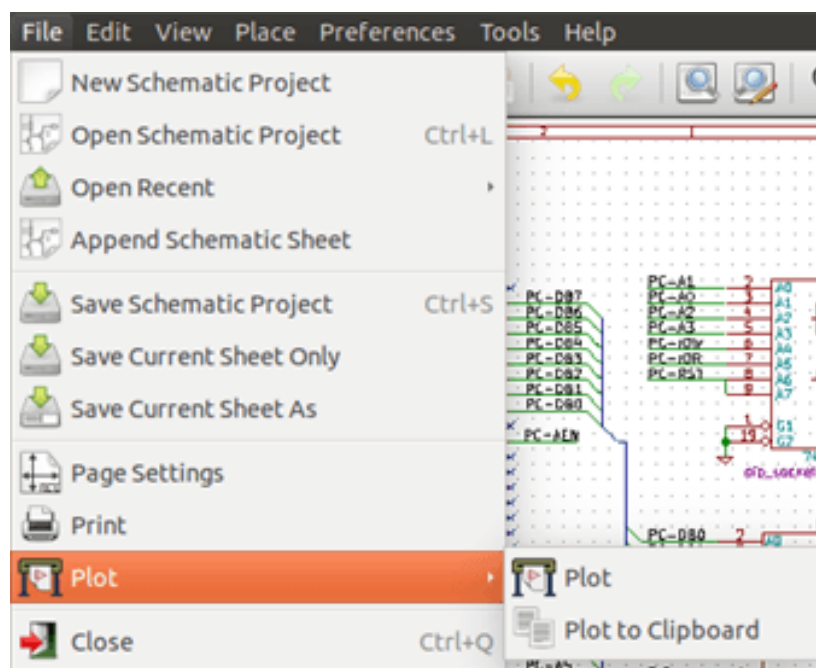
Lihat Bab 14 untuk mendapatkan penjelasan lebih mengenai `xsltproc`, penjelasan mengenai format berkas antara, dan beberapa contoh dari gaya lembar kerja untuk konverter.

## Bab 11

# Plot dan Cetak

### 11.1 Pengenalan

Anda bisa mengakses perintah cetak dan *plot* melalui menu *File*.



Format keluaran yang didukung adalah Postscript, PDF, SVG, DXF, dan HPGL. Anda juga bisa mencetak secara langsung ke mesin pencetak Anda.

### 11.2 Perintah Pencetak Umum

#### *Plot Current Page*

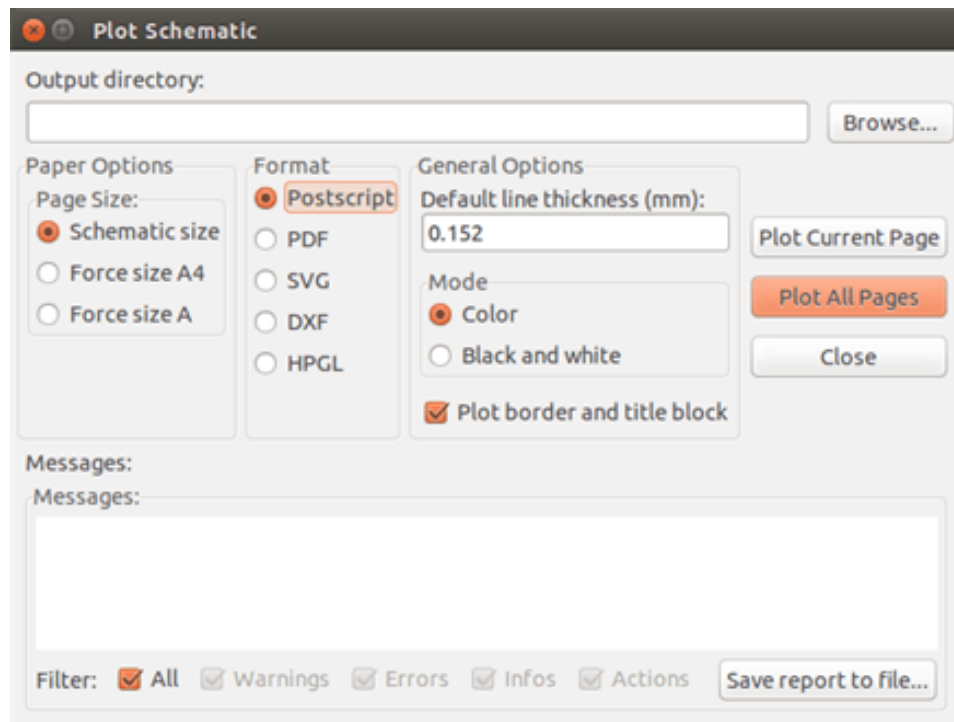
Mencetak hanya satu berkas untuk lembar kerja yang saat ini dibuka.

#### *Plot All Pages*

Mencetak *plot* untuk keseluruhan hirarki (akan dibuat satu berkas cetak untuk setiap lembar kerja).

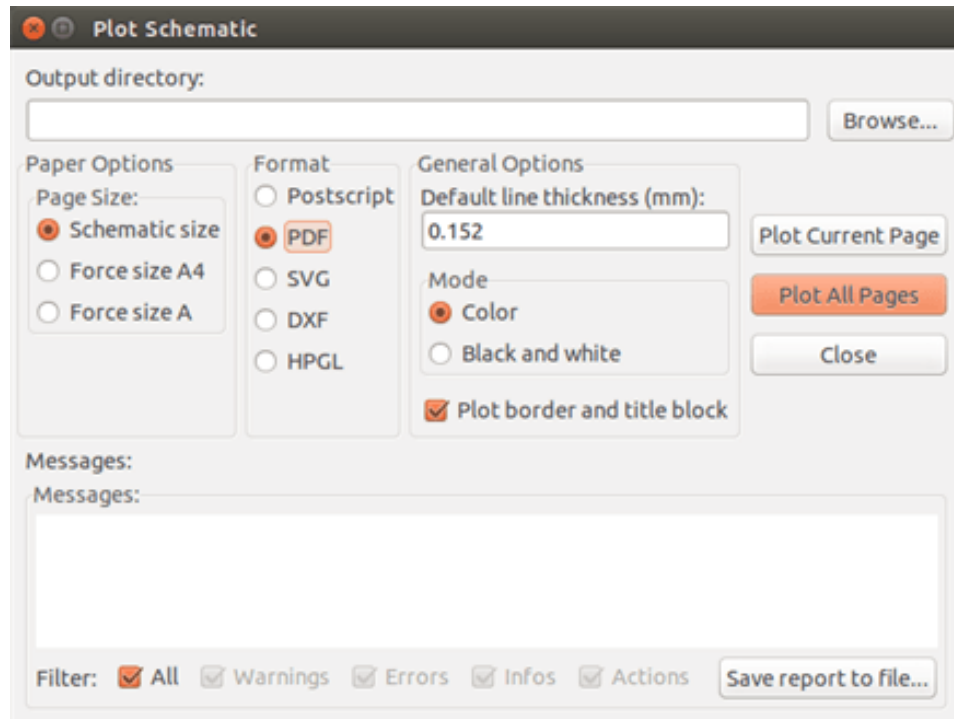
### 11.3 *Plot* dalam Format Postscript

Perintah berikut akan menghasilkan berkas PostScript.



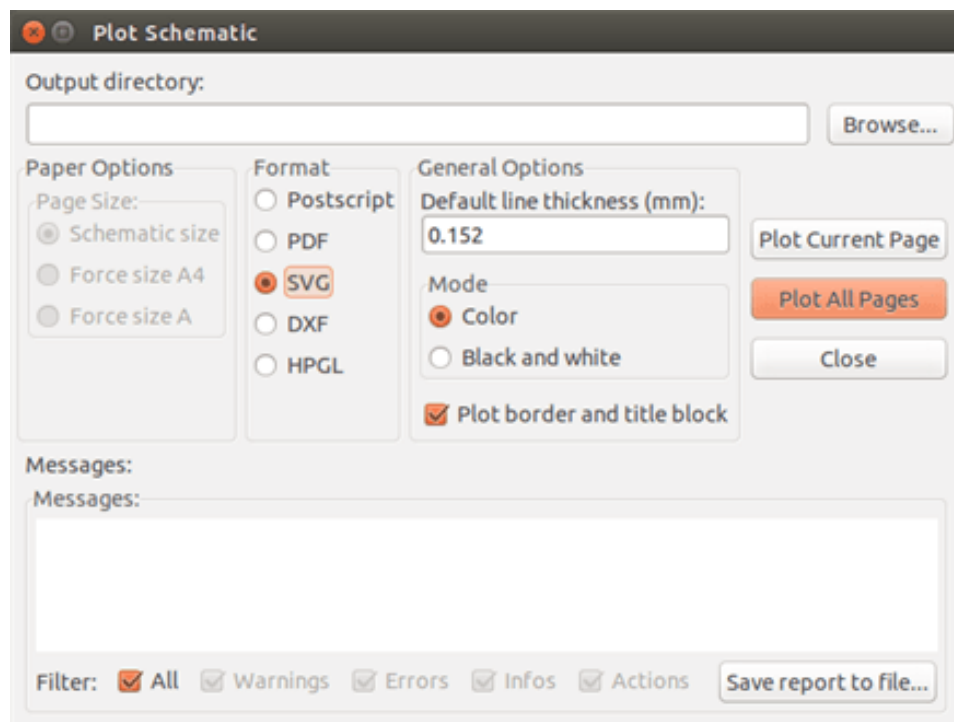
Nama berkas yang dihasilkan akan menggunakan nama lembar kerja dengan ekstensi `.ps`. Anda bisa menonaktifkan opsi "*Plot border and title block*". Hal ini berguna jika Anda ingin membuat sebuah berkas Postscript sebagai enkapsulasi (format `.eps`) yang kadang digunakan untuk memasukkan sebuah diagram ke dalam perangkat lunak pengolah kata. Kotak *Messages* akan menampilkan nama berkas yang berhasil dibuat.

## 11.4 *Plot* dalam Format PDF



Anda bisa membuat berkas *plot* menggunakan format PDF. Nama berkas yang dihasilkan akan menggunakan nama lembar kerja, dengan ekstensi `.pdf`.

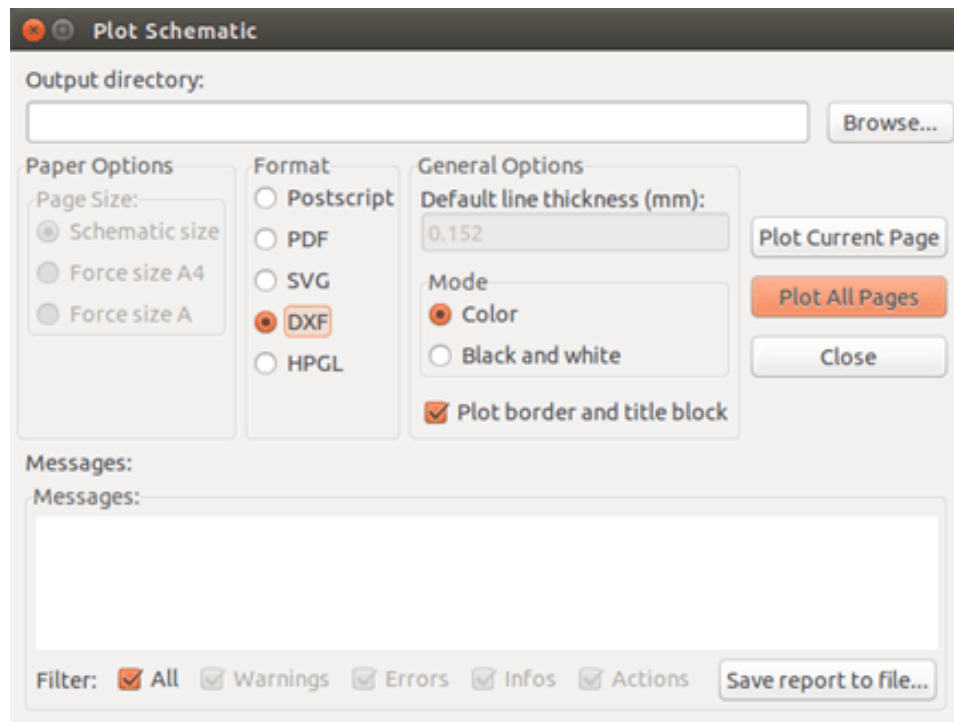
## 11.5 *Plot* dalam Format SVG





Anda bisa membuat berkas *plot* menggunakan format vektor SVG. Nama berkas yang dihasilkan akan menggunakan nama lembar kerja, dengan ekstensi `.svg`.

## 11.6 *Plot* dalam Format DXF



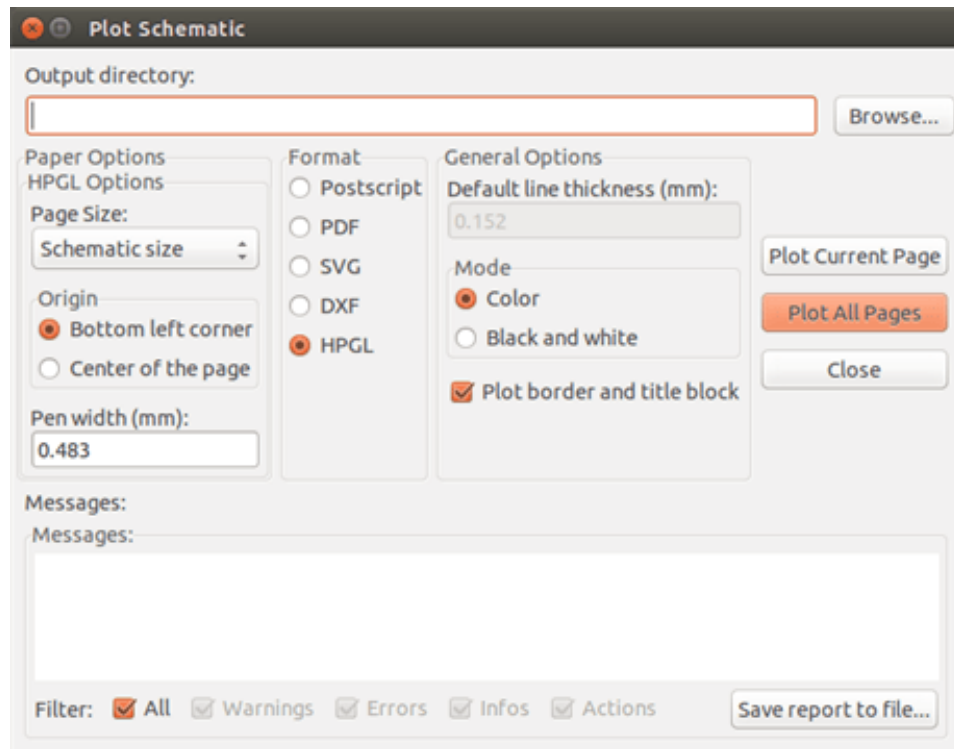
Anda bisa membuat berkas *plot* menggunakan format DXF. Nama berkas yang dihasilkan akan menggunakan nama lembar kerja dengan ekstensi `.dxf`.

## 11.7 *Plot* dalam Format HPGL

Perintah ini akan menghasilkan sebuah berkas HPGL. Dalam format ini, Anda dapat menentukan:

- Ukuran halaman.
- Pangkal halaman (*origin*).
- Lebar pena (dalam mm).

Berikut tampilan kotak dialog pengaturan *plotter*:



Nama berkas yang dihasilkan akan menggunakan nama lembar kerja dengan ekstensi `.plt`.

### 11.7.1 Pemilihan Ukuran Lembar Kerja

Umumnya, ukuran halaman akan menyesuaikan ukuran lembar kerja. Pada kasus ini, akan digunakan ukuran lembar kerja yang dipilih pada pengaturan blok judul, dan skala yang dipilih adalah 1. Jika dipilih ukuran lembar kerja yang lain (A4 dengan A0, atau A dengan E), maka skala akan diatur secara otomatis untuk mengisi halaman.

### 11.7.2 Pengaturan *Offset*

Untuk semua ukuran standar, Anda dapat mengatur *offset* untuk memposisikan gambar seakurat mungkin di posisi tengah. Karena mesin *plotter* memiliki titik pangkal (*origin*) di tengah atau di sudut kiri bawah lembar kerja, kita perlu mengatur *offset* agar hasil *plot* sesuai keinginan kita.


Secara umum:

- Untuk mesin *plotter* yang memiliki titik pangkal di tengah lembar kerja, maka *offset* harus bernilai negatif dan diatur setengah dari ukuran lembar kerja.
- Untuk mesin *plotter* yang memiliki titik pangkal di sudut kiri bawah lembar kerja, maka *offset* harus bernilai 0.

Untuk mengatur *offset*:

- Pilih ukuran lembar kerja.
- Atur *offset X* dan *offset Y*.
- Klik untuk menerapkan pengaturan.

## 11.8 Mencetak di Kertas

Perintah yang dapat diakses melalui ikon  akan memvisualisasikan dan menghasilkan berkas desain untuk mesin cetak standar.



Opsi "*Print sheet reference and title block*" akan mengaktifkan atau menonaktifkan referensi lembar kerja dan blok judul.

Opsi "*Print in black and white*" akan mengatur hasil cetak dengan warna monokrom. Opsi ini diperlukan jika Anda menggunakan mesin cetak laser hitam putih, karena warna selain itu akan dicetak secara buram atau abu-abu, sehingga terkadang sulit dibaca.

## Bab 12

# Editor Pustaka Simbol

### 12.1 Informasi Umum

Simbol adalah elemen skematik yang berisi representasi grafis, koneksi elektrik, dan atribut-atribut yang mendefinisikan simbol tersebut. Simbol yang digunakan pada skematik disimpan di dalam pustaka simbol. Eeschema menyediakan sebuah alat editor pustaka simbol yang dapat digunakan untuk membuat pustaka, mengeksport simbol ke suatu berkas, dan mengimpor simbol dari suatu berkas. Alat editor pustaka ini menggunakan cara yang sederhana untuk melakukan manajemen berkas pustaka simbol.

### 12.2 Tentang Pustaka Simbol

Suatu pustaka simbol memiliki satu atau lebih simbol. Secara umum, simbol dikelompokkan berdasarkan fungsi, jenis, dan/atau manufaktur.

Sebuah simbol terdiri dari:

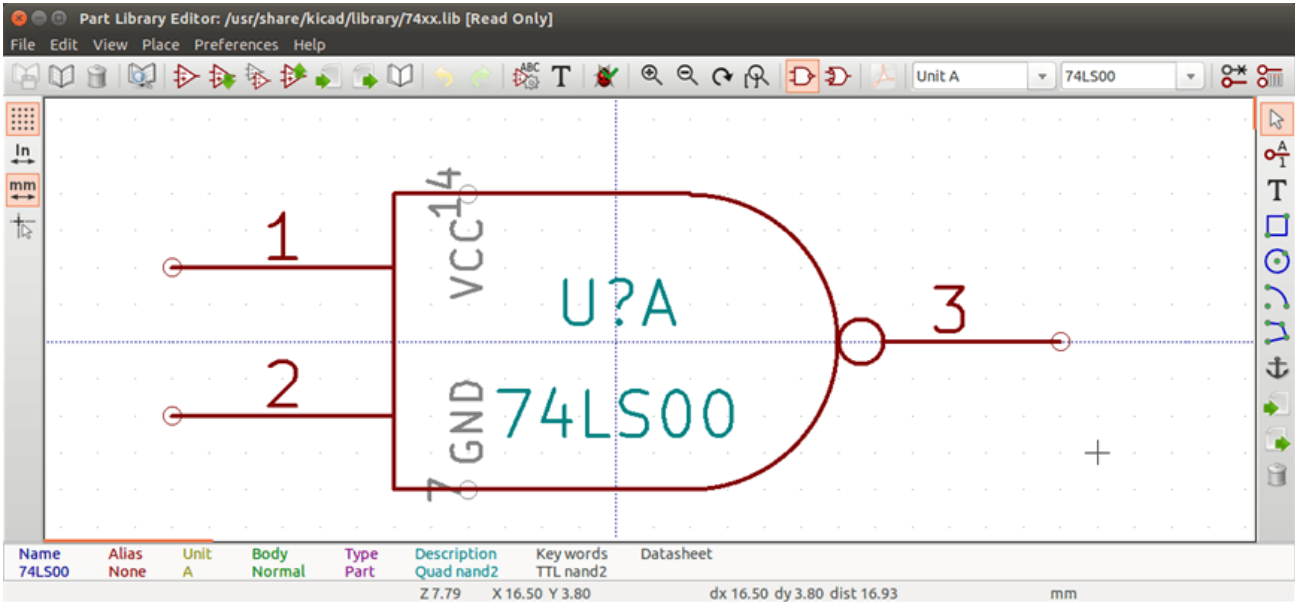
- Item-item grafis (garis, lingkaran, busur, teks, dsb) yang menyediakan definisi simbolis.
- Pin-pin yang memiliki properti grafis (*line, clock, inverted, low level active*, dsb) dan properti elektrik (*input,output,bidirectional*, dsb) yang digunakan oleh pengujian *Electrical Rules Check* (ERC).
- Atribut-atribut seperti referensi, nilai, nama *footprint* yang terkait untuk desain PCB, dsb.
- Alias yang digunakan untuk mengasosiasikan suatu simbol umum seperti 7400 beserta derivatifnya misalnya 74LS00, 74HC00, dan 7437. Keseluruhan alias tersebut berbagi pustaka simbol yang sama.

Desain simbol yang baik memerlukan hal-hal sebagai berikut:

- Menentukan apakah simbol perlu dibuat dalam bentuk lebih dari satu bagian.
  - Menentukan apakah simbol memiliki gaya bentuk alternatif yang disebut representasi De Morgan.
  - Membuat desain representasi simbolis menggunakan garis, persegi, lingkaran, poligon, dan teks.
  - Menambahkan pin dengan mendefinisikan secara teliti setiap elemen grafis pin, nama, penomoran, dan properti elektrik (*input, output, tri-state, power port*, dsb).
  - Menambahkan alias jika simbol lain memiliki desain dan susunan pin yang sama, atau menghapus salah satu jika simbol yang diperlukan telah tersedia dari simbol yang lain.
  - Menambahkan atribut opsional seperti nama *footprint* yang digunakan oleh perangkat lunak desain PCB, dan/atau menentukan bagian-bagian yang perlu ditampilkan.
  - Mendokumentasikan simbol dengan menambahkan deskripsi dan tautan ke lembar data, dsb.
  - Menyimpan ke dalam pustaka yang diinginkan.
-

## 12.3 Tentang Editor Pustaka Simbol










Tampilan jendela utama editor pustaka simbol ditunjukkan pada gambar di bawah ini. Editor ini terdiri dari tiga bilah alat yang digunakan untuk mengakses secara cepat fitur-fitur umum, dan satu area untuk melihat/mengedit simbol. Tidak semua perintah tersedia pada bilah alat, namun dapat diakses melalui menu.
















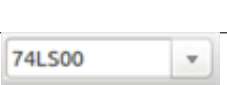




### 12.3.1 Bilah Alat Utama

Bilah alat utama terletak di bagian atas jendela utama seperti ditunjukkan pada gambar di bawah. Bilah alat utama terdiri dari peralatan manajemen pustaka, perintah batalkan/ulangi (*undo/redo*), perintah perbesar/perkecil tampilan (*zoom*), dan dialog properti simbol.














	Menyimpan pustaka yang sedang dipilih. Tombol akan dinonaktifkan jika tidak ada pustaka yang sedang dipilih, atau tidak ada perubahan pada pustaka yang sedang dipilih.
	Memilih pustaka yang akan diedit.
	Menghapus sebuah simbol dari pustaka yang sedang dipilih, atau pustaka yang digunakan oleh proyek jika tidak ada pustaka yang sedang dipilih.
	Membuka peramban pustaka simbol untuk memilih pustaka dan simbol yang akan diedit.
	Membuat simbol baru.
	Memuat simbol untuk diedit dari pustaka yang sedang dipilih.
	Membuat simbol baru dari simbol yang sedang dimuat.
	Menyimpan perubahan simbol ke dalam memori. Berkas pustaka tidak akan diubah.
	Mengimpor suatu simbol dari sebuah berkas.

	Mengekspor simbol ke sebuah berkas.
	Membuat pustaka baru yang berisi simbol yang sedang dibuka. Catatan: pustaka yang baru tidak ditambahkan ke proyek secara otomatis.
	Membatalkan perubahan terakhir.
	Mengulangi perubahan terakhir yang dibatalkan.
	Mengedit properti simbol yang sedang dibuka.
	Mengedit atribut simbol yang sedang dibuka.
	Menguji simbol dari kesalahan desain.
	Perbesar tampilan.
	Perkecil tampilan.
	Memuat ulang tampilan.
	Mengatur tampilan simbol agar muat dalam satu layar.
	Memilih gaya bentuk normal. Tombol akan dinonaktifkan jika simbol yang sedang dibuka tidak memiliki gaya bentuk alternatif.
	Memilih gaya bentuk alternatif. Tombol akan dinonaktifkan jika simbol yang sedang dibuka tidak memiliki gaya bentuk alternatif.
	Menampilkan dokumen yang terkait. Tombol akan dinonaktifkan jika tidak ada dokumentasi yang tersedia untuk simbol yang sedang dibuka.
	Memilih bagian yang akan ditampilkan. Menu ini akan dinonaktifkan jika simbol yang sedang dibuka tidak terdiri dari beberapa bagian.
	Memilih alias. Menu ini akan dinonaktifkan jika simbol yang sedang dibuka tidak memiliki alias.
	Mengedit pin: mengedit bentuk dan posisi pin secara independen untuk simbol yang terdiri dari beberapa bagian dan simbol alternatif.
	Menampilkan tabel pin.

### 12.3.2 Bilah Alat Elemen





Bilah alat vertikal yang berada di sisi kanan jendela utama berguna untuk meletakkan semua elemen yang diperlukan untuk mendesain sebuah simbol. Tabel di bawah ini menjelaskan setiap tombol pada bilah alat.

	Alat pemilih. Klik-kanan dengan alat pemilih akan membuka sebuah menu konteks untuk obyek di bawah kursor. Klik-kiri dengan alat pemilih akan menampilkan atribut dari obyek yang berada di bawah kursor. Atribut akan muncul di panel pesan pada bagian bawah jendela utama. Klik-ganda dengan alat pemilih akan membuka kotak dialog properti obyek yang berada di bawah kursor.
---	--


	Alat pin. Klik-kiri untuk menambahkan pin.
	Alat teks grafis. Klik-kiri untuk menambahkan teks grafis baru.
	Alat pembuat persegi. Klik-kiri untuk mulai menggambar sudut pertama persegi. Klik-kiri sekali lagi untuk meletakkan sudut persegi yang berlawanan.
	Alat pembuat lingkaran. Klik-kiri untuk mulai menggambar lingkaran dari titik tengah. Klik-kiri sekali lagi pada radius lingkaran yang diinginkan untuk menyelesaikan gambar.
	Alat pembuat busur. Klik-kiri untuk mulai menggambar busur dari titik tengah. Klik-kiri sekali lagi pada titik akhir busur untuk menyelesaikan gambar.
	Alat pembuat poligon. Klik-kiri untuk mulai menggambar poligon. Klik-kiri untuk setiap penambahan garis poligon. Klik-ganda untuk menyelesaikan gambar.
	Alat untuk mengatur titik jangkar. Klik-kiri untuk mengatur posisi jangkar pada simbol.
	Mengimpor simbol dari sebuah berkas.
	Mengekspor simbol ke sebuah berkas.
	Alat penghapus. Klik-kiri untuk menghapus suatu obyek dari simbol.

### 12.3.3 Bilah Alat Opsi

Bilah alat vertikal yang berada di sisi kiri jendela utama berguna untuk mengatur opsi-opsi gambar yang Anda buat. Tabel di bawah ini menjelaskan setiap tombol pada bilah alat.



	Mengaktifkan atau menonaktifkan tampilan <i>grid</i> .
	Gunakan satuan inci.
	Gunakan satuan milimeter.
	Mengaktifkan atau menonaktifkan kursor layar penuh.

## 12.4 Pemilihan dan Pemeliharaan Pustaka

Pemilihan pustaka dapat dilakukan melalui ikon  yang akan menampilkan semua pustaka yang tersedia yang dapat Anda pilih. Ketika suatu simbol dimuat atau disimpan, simbol tersebut akan diletakkan pada pustaka yang telah dipilih. Nama pustaka yang digunakan oleh simbol adalah isi dari atribut nilainya.

---


**Catatan**

- Untuk dapat mengakses isi pustaka, Anda harus memuat pustaka tersebut ke Eeschema.
  - Isi dari pustaka yang sedang dibuka dapat disimpan setelah dilakukan modifikasi, dengan menekan  pada bilah alat utama.
  - Simbol dapat dihapus dari pustaka dengan menekan .
- 

### 12.4.1 Memilih dan Menyimpan Simbol

Ketika Anda mengedit sebuah simbol, Anda sebenarnya tidak benar-benar bekerja pada simbol tersebut, akan tetapi yang sedang Anda kerjakan adalah salinan dari simbol yang disimpan di memori komputer. Setiap modifikasi dapat dengan mudah dibatalkan. Simbol dapat dimuat dari pustaka lokal atau dari simbol yang sudah ada.


#### 12.4.1.1 Pemilihan Simbol

Klik  pada bilah alat utama untuk menampilkan daftar simbol yang tersedia yang dapat Anda pilih dan muat dari pustaka yang sedang dipilih.

---

**Catatan**



Jika sebuah simbol dipilih melalui alias, maka yang akan ditampilkan pada bilah judul jendela adalah nama dari simbol yang dimuat, dan bukan alias yang dipilih. Daftar alias simbol selalu dimuat bersama dengan tiap simbolnya, dan dapat diedit.

Anda bisa membuat sebuah simbol baru dengan memilih alias dari simbol menggunakan . Item yang pertama pada daftar alias adalah nama utama dari simbol.

---

---


**Catatan**


Klik pada  juga bisa memuat simbol yang sebelumnya telah disimpan melalui .


---

#### 12.4.1.2 Menyimpan Simbol

Setelah dilakukan modifikasi, kita bisa menyimpan simbol pada pustaka yang saat ini dipilih, pada pustaka baru, atau mengekspor ke suatu berkas cadangan.

Untuk menyimpan simbol yang sudah dimodifikasi ke pustaka yang sedang dipilih, klik . Ingat bahwa perintah ini hanya akan menyimpan perubahan simbol memori lokal. Dengan demikian, Anda masih bisa berubah pikiran sebelum menyimpan ke dalam pustaka.

Untuk menyimpan perubahan simbol secara permanen ke berkas pustaka, klik . Berkas pustaka yang ada akan ditimpa dengan perubahan simbol.

Jika Anda ingin membuat sebuah pustaka baru yang berisi simbol yang saat ini dibuka, klik . Anda harus memasukkan nama pustaka yang baru.

---



---


**Catatan**

Pustaka yang baru tidak secara otomatis ditambahkan ke proyek yang sedang dibuka.

Anda harus menambahkan pustaka baru yang ingin Anda gunakan di skematik ke daftar pustaka proyek di Eeschema, dengan menggunakan [kotak dialog Symbol Library Table](#).






---



Klik pada  untuk membuat sebuah berkas yang berisi hanya simbol yang saat ini dibuka. Berkas ini adalah berkas pustaka standar yang berisi hanya satu simbol saja. Berkas ini dapat digunakan untuk mengimpor simbol ke pustaka yang lain. Sebenarnya, perintah pembuatan pustaka baru dan perintah ekspor adalah sama.

#### 12.4.1.3 Mentransfer Simbol ke Pustaka Lain

Anda dapat dengan mudah menyalin sebuah simbol dari pustaka sumber ke pustaka tujuan dengan menggunakan langkah-langkah berikut:

- Pilih pustaka sumber dengan klik .
- Muat simbol yang akan ditransfer dengan klik . Simbol akan ditampilkan pada layar editor.
- Pilih pustaka tujuan dengan klik .
- Simpan simbol yang sedang dibuka ke pustaka yang baru pada memori lokal dengan klik .
- Simpan simbol ke berkas pustaka lokal dengan klik .


#### 12.4.1.4 Membatalkan Perubahan Simbol

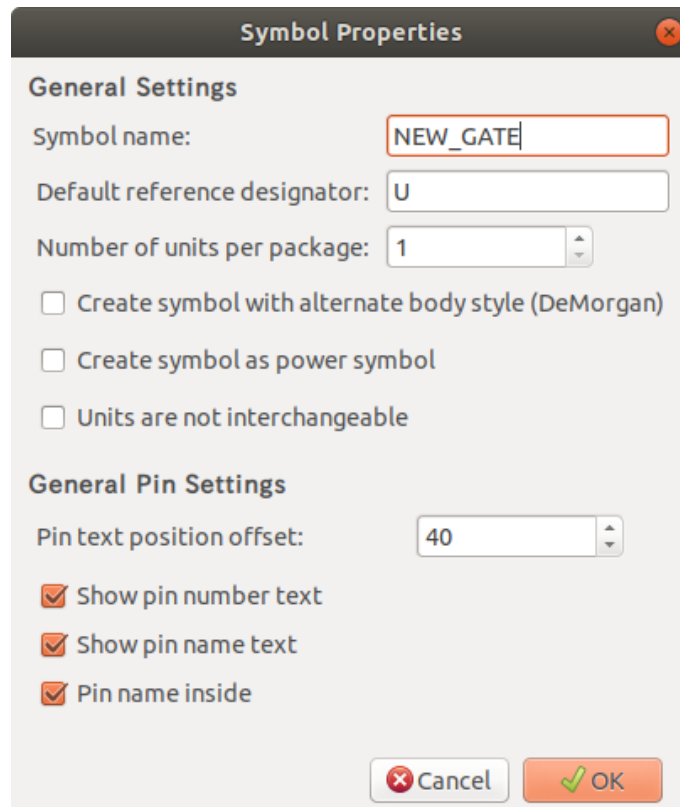
Ketika Anda bekerja pada sebuah simbol, simbol yang diedit hanya merupakan sebuah salinan dari simbol yang sebenarnya di dalam pustaka. Hal ini berarti bahwa selama Anda belum menyimpan simbol tersebut, Anda masih bisa memuat ulang simbol untuk membatalkan semua perubahan yang telah Anda buat. Jika Anda telah menyimpan simbol ke memori lokal dan belum menyimpannya ke dalam berkas pustaka, Anda bisa dengan mudah keluar dan memulai kembali. Eeschema akan membatalkan semua perubahan yang sudah Anda buat.

## 12.5 Membuat Simbol Pustaka

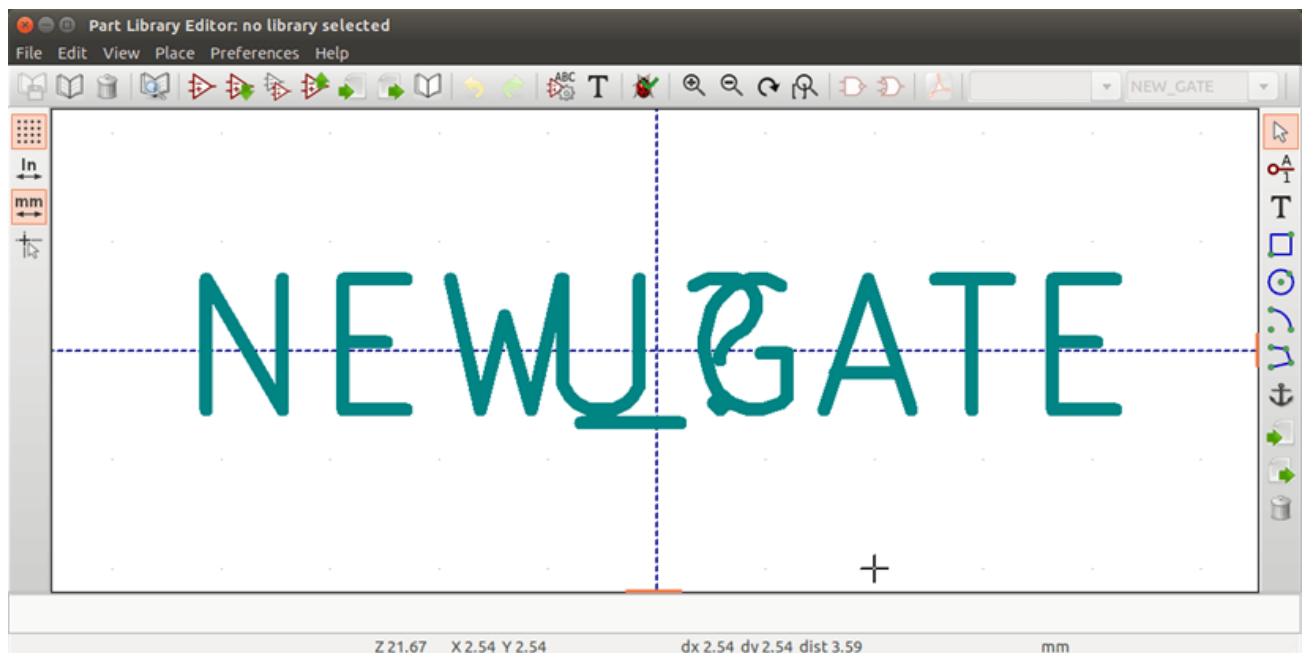
### 12.5.1 Membuat Simbol Baru



Simbol yang baru dapat dibuat dengan klik . Anda harus memasukkan nama simbol (nama ini digunakan sebagai nilai *default* untuk atribut *value* di editor skematik), penunjuk referensi (U, IC, R...), jumlah bagian per paket (sebagai contoh, 7400 terdiri dari 4 bagian per paket) dan apakah akan digunakan gaya bentuk alternatif (sering disebut DeMorgan). Jika atribut penunjuk referensi kosong, maka nilainya akan kembali ke *default* yaitu "U". Properti simbol dapat diubah nanti, namun sebaiknya Anda mengisi dengan sebaik-baiknya pada saat pembuatan simbol.








Simbol yang baru akan dibuat menggunakan properti di atas, dan akan muncul pada layar editor seperti gambar di bawah ini.



### 12.5.2 Membuat Simbol dari Simbol yang Lain


Terkadang, simbol yang ingin Anda buat berbentuk mirip dengan simbol yang sudah ada di pustaka. Anda dapat memuat dan memodifikasi simbol yang sudah ada dengan mudah.

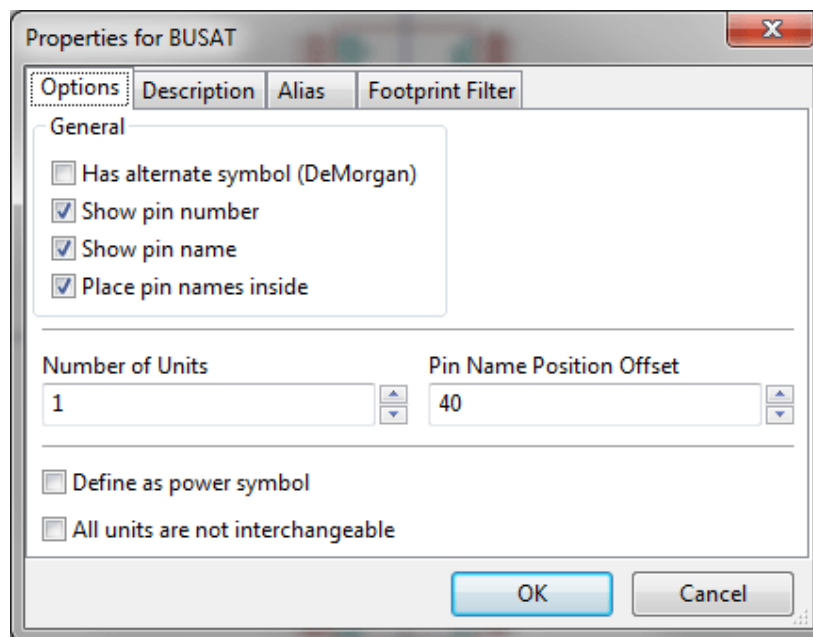
- Muat simbol yang akan digunakan sebagai titik mulai.

- Klik  atau ubah namanya dengan klik-kanan pada atribut nilai dan ubah isinya. Jika Anda memilih untuk menduplikasi simbol yang ada, Anda akan diminta untuk memasukkan nama simbol yang baru.
- Jika model simbol memiliki alias, Anda akan diminta untuk menghapus alias dari simbol yang baru yang bisa menyebabkan konflik dengan pustaka yang ada. Jika Anda menjawab "No" maka pembuatan simbol baru akan dibatalkan. Pustaka simbol tidak bisa menggunakan nama atau alias yang sama dengan yang lain.
- Edit simbol baru sesuai keinginan Anda.
- Perbarui simbol baru pada pustaka dengan klik  atau simpan ke pustaka baru dengan klik , atau jika Anda ingin menyimpan simbol baru ini ke pustaka lain yang sudah ada, pilih pustaka lain dengan klik  dan simpan simbol yang baru.
- Simpan berkas pustaka dengan klik .

### 12.5.3 Properti Simbol

Properti simbol harus diisi secara teliti pada saat pembuatan simbol, atau kita juga bisa mengambilnya dari simbol yang disalin.

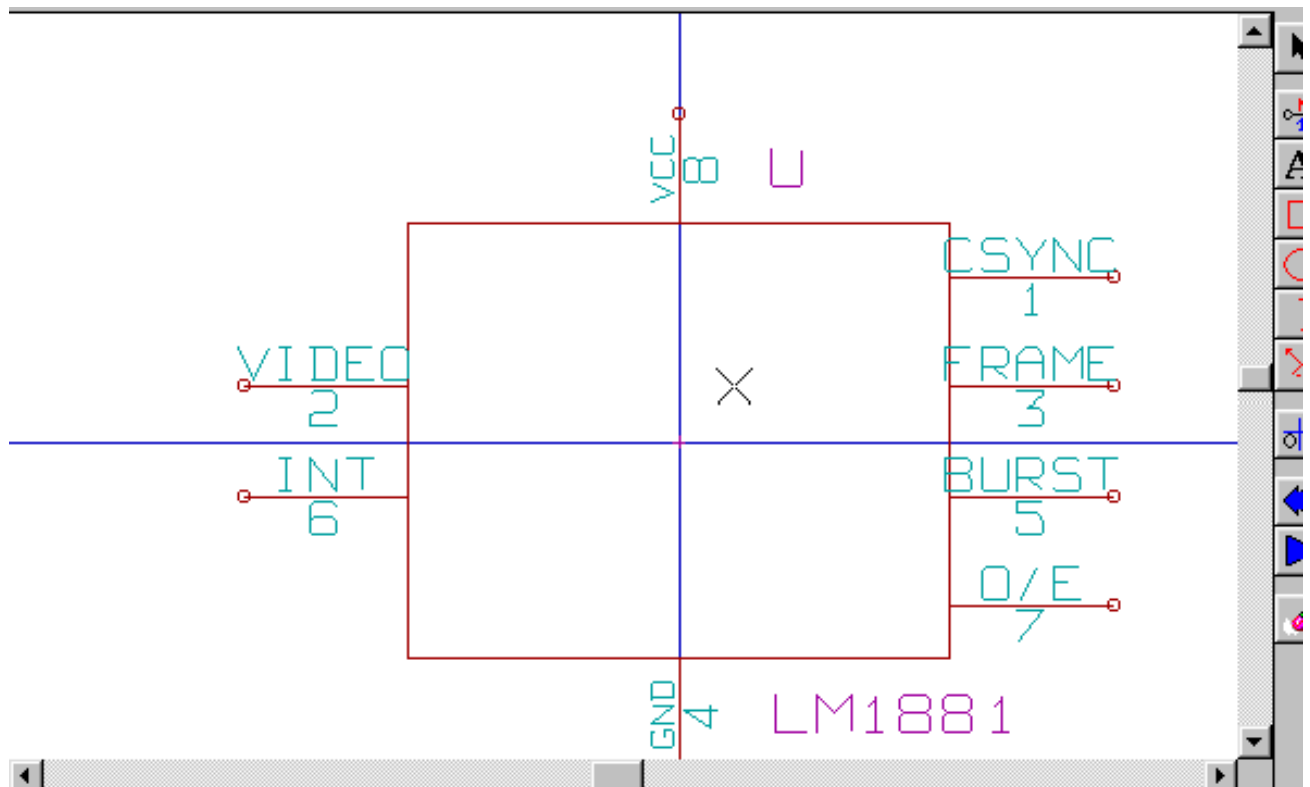
Untuk mengubah properti simbol, klik  untuk menampilkan kotak dialog seperti gambar berikut.




Sangat penting untuk mengisi jumlah bagian per paket dan representasi simbolis alternatif secara benar, jika diaktifkan, karena saat pin diedit atau dibuat, pin-pin yang terkait pada setiap bagian akan terkena efeknya. Jika Anda mengubah jumlah bagian per paket setelah membuat dan mengedit pin, maka akan ada pekerjaan tambahan untuk menambahkan pin-pin bagian dan simbol yang baru. Meskipun demikian, kita bisa mengubah nilai properti tersebut kapanpun.



Opsi "*Show pin number*" dan "*Show pin name*" akan menampilkan atau menyembunyikan teks nomor pin dan nama pin. Teks ini akan tampak jika opsi yang sesuai dicentang. Opsi "*Place pin names inside*" akan menentukan posisi nama pin relatif terhadap badan pin. Teks ini akan ditampilkan di dalam garis bentuk simbol jika opsi ini dicentang. Dalam kasus ini, properti "*Pin Name Position Offset*" akan menentukan pergeseran teks dari badan luar pin. Anda bisa menggunakan nilai dari 30 ke 40 (dalam 1/1000 inci).

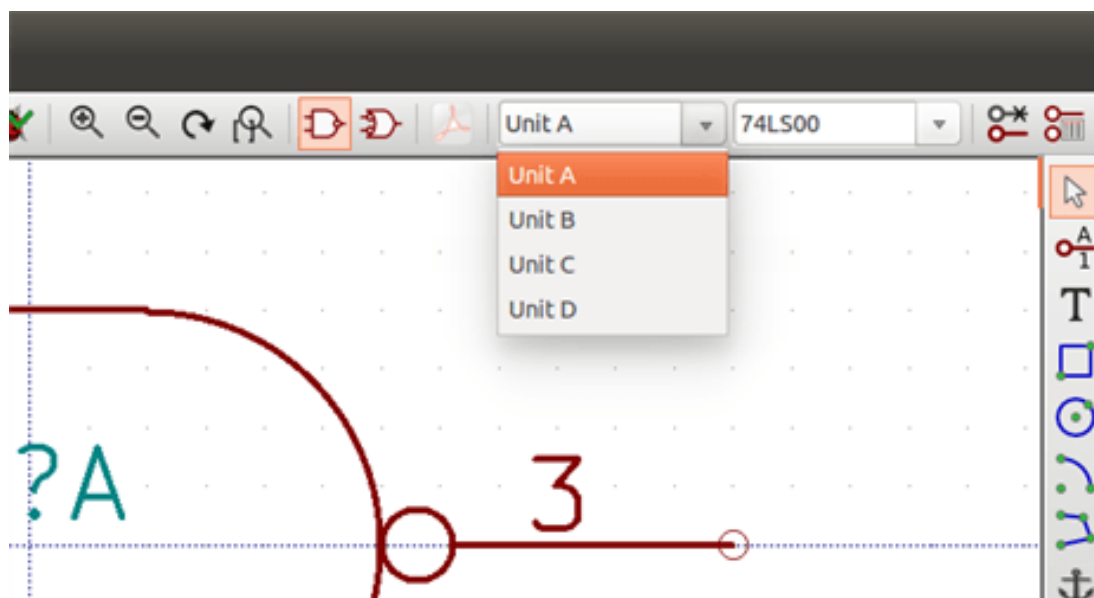
Contoh di bawah ini menampilkan sebuah simbol dengan opsi "*Place pin name inside*" dicentang. Perhatikan posisi dari nama dan nomor pin.



#### 12.5.4 Simbol dengan Representasi Simbolis Alternatif

Jika simbol memiliki lebih dari satu representasi simbolis, Anda harus memilih salah satu representasi yang akan diedit. Untuk mengedit representasi normal, klik .

Untuk mengedit representasi alternatif, klik . Gunakan  seperti gambar di bawah untuk memilih bagian yang akan diedit.



## 12.6 Elemen Grafis

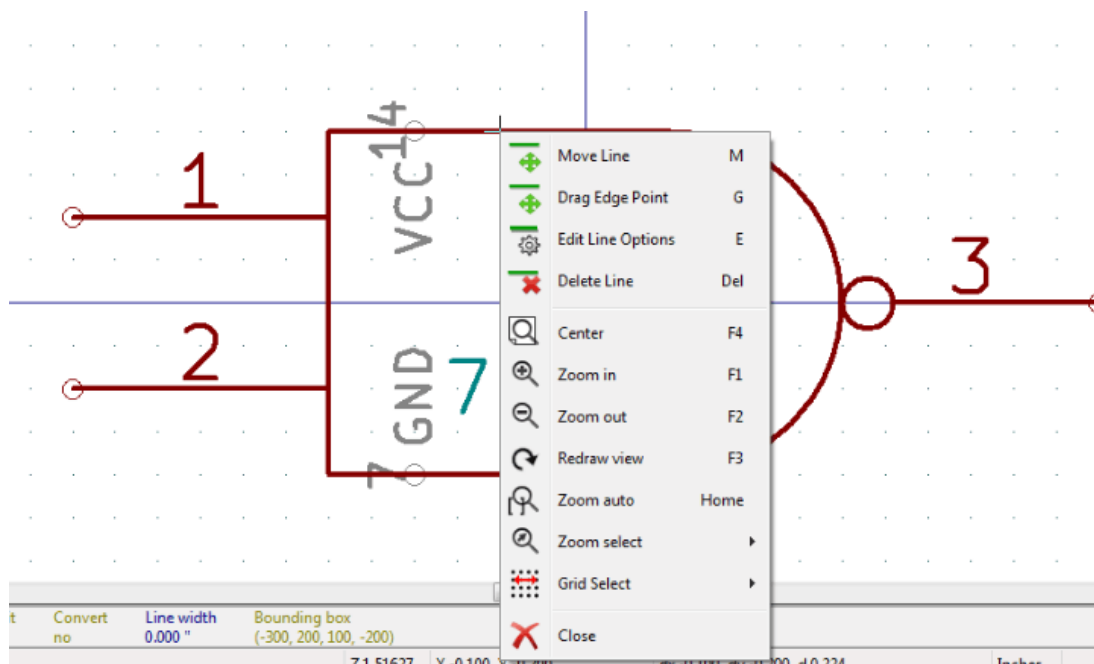
Elemen grafis hanya digunakan untuk menciptakan representasi simbol dan tidak memiliki informasi koneksi elektrik. Desain elemen grafis bisa dilakukan dengan menggunakan peralatan berikut:

- Garis dan poligon, didefinisikan oleh titik awal dan titik akhir.
- Persegi, didefinisikan oleh dua sudut diagonal.
- Lingkaran, didefinisikan oleh titik tengah dan radius.
- Busur, didefinisikan oleh titik mulai dan titik akhir busur dan titik tengahnya. Busur bisa memiliki sudut  $0^\circ$  hingga  $180^\circ$ .

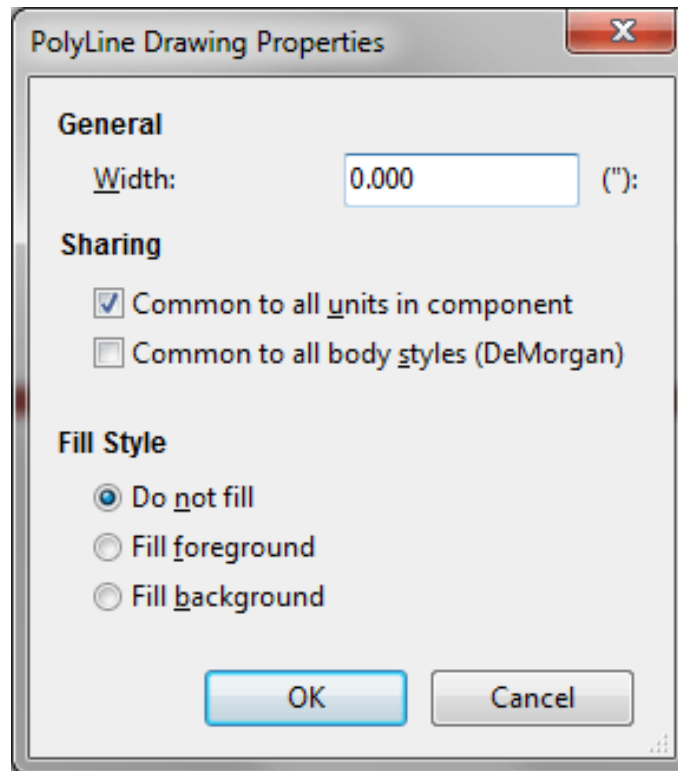
Bilah alat vertikal di sebelah kanan jendela utama berguna untuk meletakkan semua elemen grafis yang diperlukan untuk mendesain tampilan sebuah simbol.

### 12.6.1 Keanggotaan Elemen Grafis

Setiap elemen grafis (garis, busur, lingkaran, dsb) dapat didefinisikan sebagai bentuk umum untuk keseluruhan bagian dan/atau gaya bentuk atau spesifik untuk bagian dan/atau gaya bentuk tertentu. Opsi-opsi elemen dapat diakses dengan klik-kanan pada elemen untuk menampilkan menu konteks elemen yang dipilih. Gambar di bawah menampilkan menu konteks untuk elemen garis.



Anda juga bisa melakukan klik-ganda pada sebuah elemen untuk mengubah propertinya. Gambar di bawah menampilkan dialog properti untuk elemen poligon.



Properti dari suatu elemen grafis antara lain:

- Lebar garis, mendefinisikan lebar dari garis elemen pada gambar yang sedang dikerjakan.
- Pengaturan "*Common to all units in symbol*" menentukan apakah elemen grafis digambar untuk setiap bagian untuk simbol yang terdiri dari beberapa bagian per paket, ataukah elemen grafis hanya digambar untuk satu bagian saja.
- Pengaturan "*Common by all body styles (DeMorgan)*" menentukan apakah elemen grafis digambar untuk setiap representasi simbolis pada simbol dengan gaya bentuk alternatif, ataukah elemen grafis hanya digambar untuk gaya bentuk saat ini saja.
- Pengaturan "*Fill Style*" menentukan apakah simbol yang didefinisikan oleh elemen grafis akan digambar berupa garis saja tanpa diwarnai, ataukah diisi menggunakan warna latar depan atau latar belakang.

### 12.6.2 Elemen Teks Grafis

**T** Ikon digunakan untuk membuat teks grafis. Teks grafis akan selalu terbaca, bahkan jika simbol ditampilkan terbalik. Ingat bahwa teks grafis bukanlah atribut.

## 12.7 Simbol dengan Beberapa Bagian dan Gaya Bentuk Alternatif

Simbol bisa memiliki dua representasi simbolis (simbol standar dan simbol alternatif yang biasa disebut "DeMorgan") dan/atau memiliki lebih dari satu bagian per paket (misalnya gerbang logika). Beberapa simbol bisa memiliki lebih dari satu bagian per paket, dan masing-masing bagian memiliki konfigurasi pin dan simbol yang berbeda.

Sebagai contoh, sebuah relay dengan dua saklar bisa didesain menjadi sebuah simbol dengan tiga bagian yang berbeda: satu buah koil, saklar 1, dan saklar 2. Mendesain sebuah simbol dengan beberapa bagian per paket dan/atau gaya bentuk alternatif sangatlah fleksibel. Suatu pin atau item simbol bentuk bisa bersifat umum untuk semua bagian atau spesifik untuk satu bagian tertentu, juga bisa bersifat umum untuk kedua jenis representasi simbolis maupun spesifik untuk salah satu representasi simbolis.

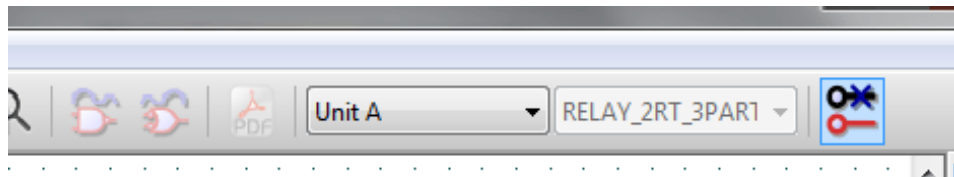
Secara *default*, pin bersifat spesifik untuk setiap representasi simbolis dari tiap bagian, karena nomor pin adalah spesifik ke satu bagian, dan bentuknya bergantung kepada representasi simbolis. Ketika pin bersifat umum untuk setiap bagian atau setiap

representasi simbolis, Anda hanya perlu membuatnya sekali saja untuk keseluruhan unit dan keseluruhan representasi simbolis (hal ini biasanya digunakan untuk pin *power*). Hal ini juga berlaku untuk bentuk tampilan grafis dan teks, yang bisa bersifat umum untuk setiap bagian (namun biasanya spesifik untuk setiap representasi simbolis).

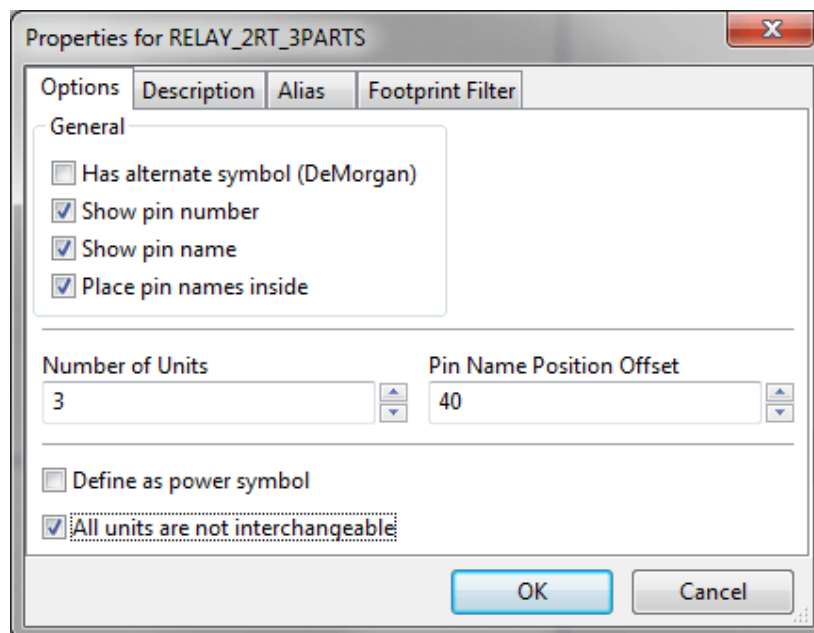
### 12.7.1 Contoh Simbol dengan Beberapa Bagian dengan Simbol Berbeda:

Berikut adalah contoh sebuah relay yang memiliki tiga bagian per paket, yaitu saklar 1, saklar 2, dan koil:

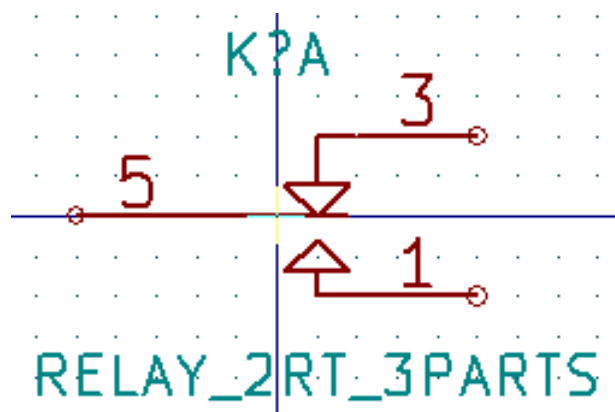
Opsi: pin-pin tidak terkait. Kita bisa menambahkan atau mengedit pin untuk tiap bagian tanpa ada pengaruh ke pin di bagian yang lain.



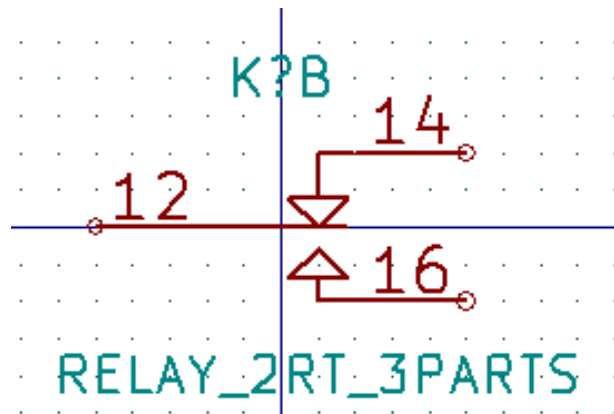
Opsi "All units are not interchangeable" harus dicentang.



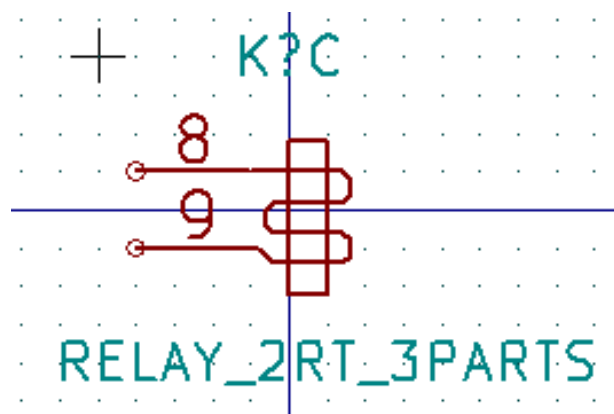
Bagian 1



## Bagian 2



## Bagian 3

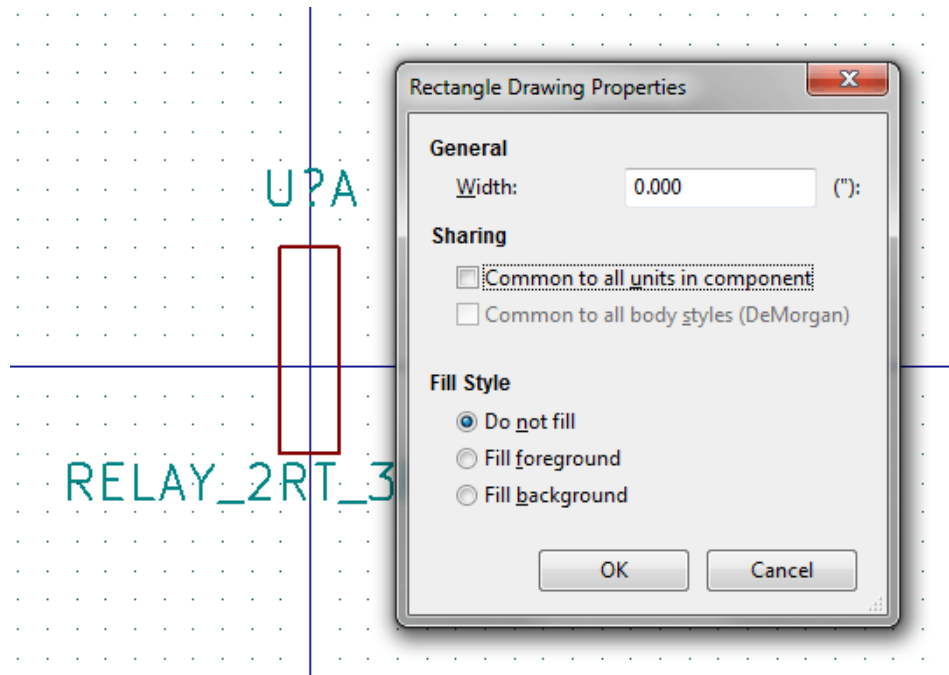


Bagian ini tidak memiliki tata letak pin dan simbol yang sama, maka tidak dapat dipertukarkan dengan bagian 1 dan 2.

### 12.7.1.1 Elemen Simbolis Grafis


Gambar di bawah ini menunjukkan properti untuk sebuah elemen bentuk grafis. Dari contoh relay di atas, ketiga bagian memiliki representasi simbolis yang berbeda. Maka, setiap bagian dibuat secara terpisah dan pada tiap elemen bentuk grafis, opsi "*Common to all units in symbol*" harus dinonaktifkan.





## 12.8 Membuat dan Mengedit Pin



Klik  untuk membuat dan memasukkan pin. Untuk mengedit properti pin, klik-ganda pada pin atau klik-kanan pada pin untuk membuka menu konteks. Pin harus dibuat dengan hati-hati, karena setiap kesalahan akan memiliki konsekuensi pada desain PCB. Setiap pin yang telah diletakkan dapat diedit, dihapus, dan/atau dipindahkan.

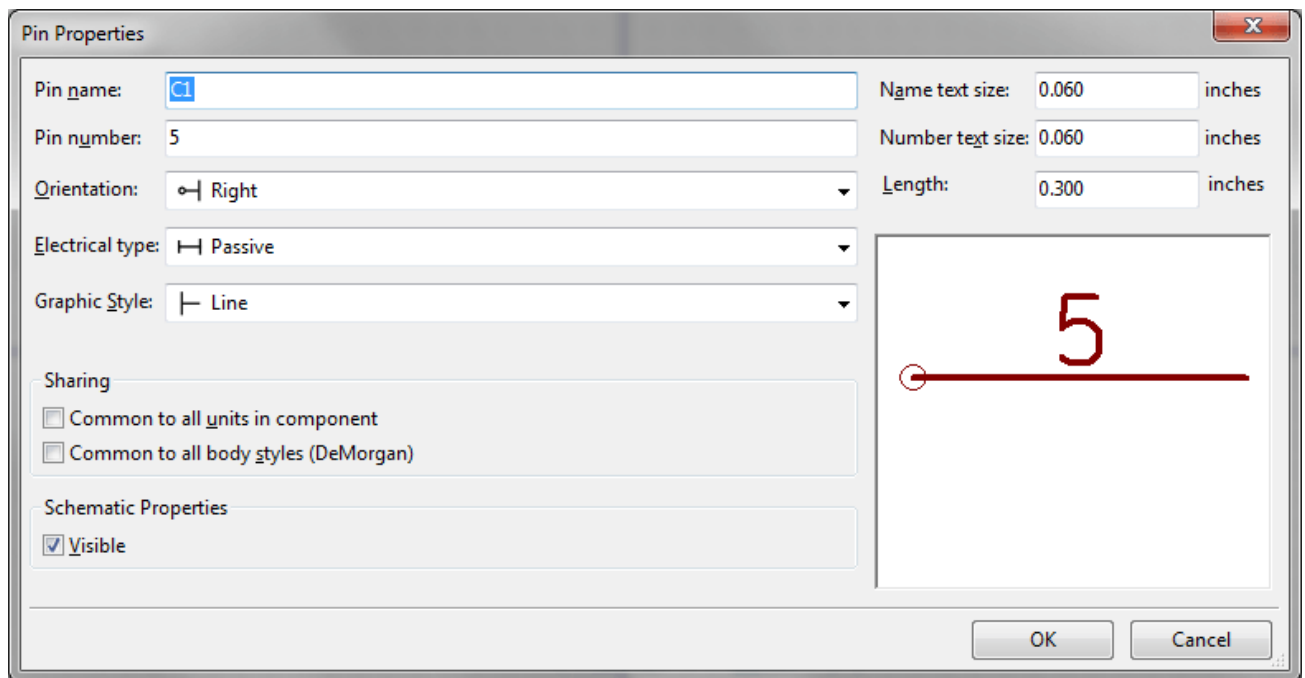
### 12.8.1 Tentang Pin

Sebuah pin didefinisikan oleh representasi grafis, nama, dan "nomor". Penomoran pin didefinisikan oleh satu set 4 huruf dan/atau angka. Supaya dapat diuji oleh alat *Electrical Rules Check* (ERC), tipe elektrik dari pin harus didefinisikan secara benar (*input*, *output*, *tri-state*, ...). Jika tipe ini tidak didefinisikan secara benar, maka hasil pengujian ERC akan menjadi tidak akurat.

Catatan penting:

- Jangan menggunakan tanda spasi pada nama dan nomor pin.
- Untuk mendefinisikan nama pin dengan sinyal yang terbalik (*inverted*), yang ditunjukkan dengan garis-atas (*overline*), gunakan karakter ~ (tilde). Karakter ~ berikutnya akan menonaktifkan garis-atas. Sebagai contoh, \~FO~O akan menampilkan FO O.
- Jika nama pin dikurangi menjadi satu simbol saja, maka pin akan dianggap tidak memiliki nama.
- Nama pin yang dimulai dengan karakter # digunakan untuk simbol *port power*.
- Nomor pin terdiri dari 1 hingga 4 huruf dan/atau angka. Contoh nomor yang valid adalah 1,2,...9999. A1, B3, Anod, Gnd, Wire, dsb juga dianggap valid.
- Tidak boleh ada nomor pin yang sama pada satu simbol.

### 12.8.2 Properti Pin

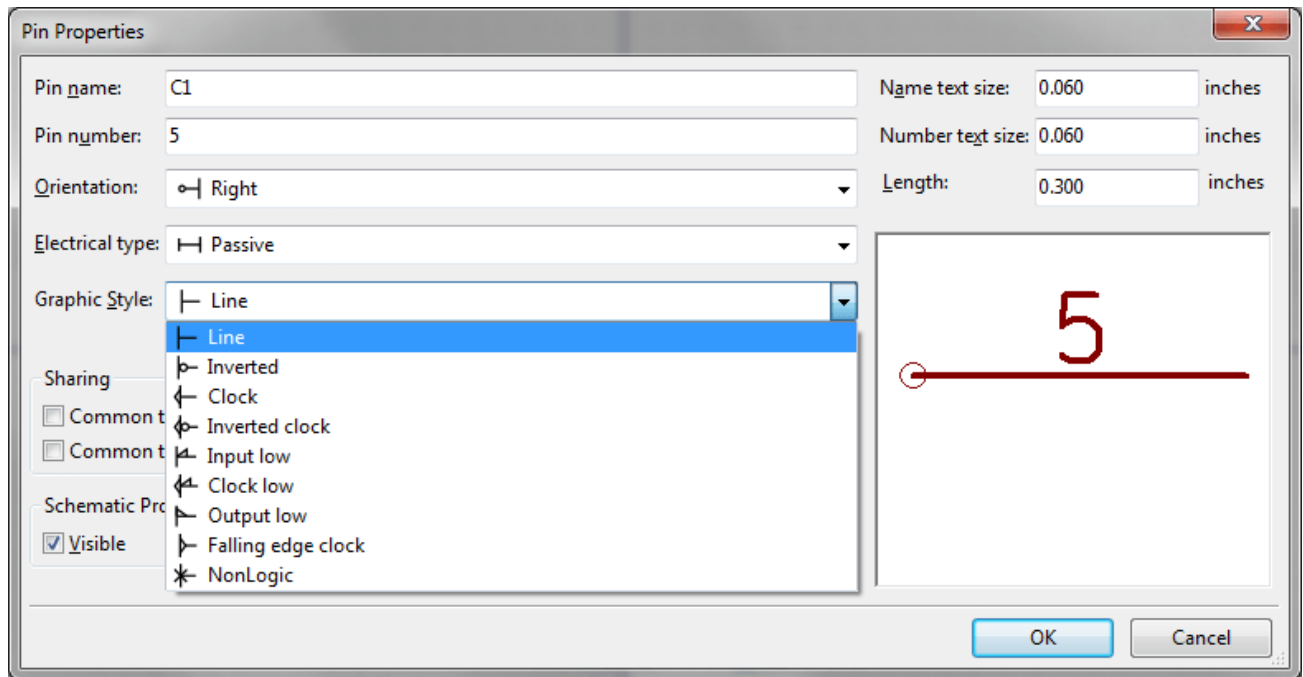


Kotak dialog properti pin digunakan untuk mengedit karakteristik dari sebuah pin. Kotak dialog ini akan muncul secara otomatis ketika Anda membuat sebuah pin atau melakukan klik-ganda pada pin yang sudah ada. Kotak dialog ini dapat digunakan untuk mengubah:

- Nama dan ukuran teksnya.
- Nomor dan ukuran teksnya.
- Panjang.
- Tipe elektrikl dan grafis.
- Keanggotaan pada bagian dan representasi alternatif.
- Tampilan.

### 12.8.3 Gaya Grafis Pin

Gambar di bawah ini menampilkan beberapa gaya grafis suatu pin. Pilihan gaya grafis tidak akan mempengaruhi tipe elektrikl suatu pin.



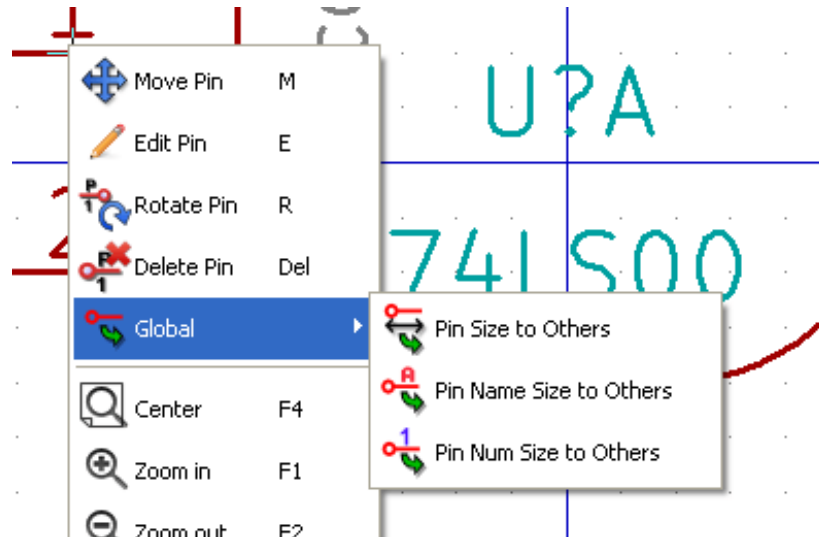
#### 12.8.4 Tipe Elektrikal Pin

Penting untuk memilih tipe elektrikal yang benar untuk peralatan ERC. Tipe elektrikal yang bisa dipilih antara lain:

- *Bidirectional*, mengindikasikan pin-pin dua-arah yang dapat digunakan sebagai masukan dan keluaran (misalnya *bus* data mikroprosesor).
- *Tri-state*, digunakan untuk keluaran dengan tiga kondisi.
- *Passive*, digunakan untuk pin simbol pasif, resistor, konektor, dsb.
- *Unspecified*, dapat digunakan jika tidak diperlukan pengujian ERC.
- *Power input*, digunakan untuk pin *power* simbol. Pin *power* akan terhubung secara otomatis ke pin *power* yang lain dengan nama yang sama.
- *Power output*, digunakan untuk keluaran regulator.
- Tipe *Open emitter* dan *Open collector* dapat digunakan untuk keluaran logika.
- *Not connected*, digunakan ketika sebuah simbol memiliki pin yang tidak memiliki koneksi internal.


#### 12.8.5 Properti Global Pin

Anda bisa mengubah panjang atau ukuran teks nama dan/atau nomor semua pin dengan menggunakan perintah *Global* pada menu konteks pin. Klik pada parameter yang ingin diubah, dan ketik nilai yang baru, yang akan diterapkan ke semua pin simbol yang sedang dibuka.



### 12.8.6 Mendefinisikan Pin untuk Banyak Bagian dan Representasi Simbolis Alternatif

Simbol yang terdiri dari beberapa bagian dan/atau representasi grafis akan menjadi problematis ketika kita membuat dan mengedit pin. Kebanyakan pin bersifat spesifik untuk setiap bagian (karena nomor pin bersifat spesifik untuk setiap bagian) dan untuk setiap representasi simbolis (karena bentuk dan posisinya spesifik untuk setiap representasi simbolis). Pembuatan dan pengeditan pin bisa menjadi problematis untuk simbol dengan beberapa bagian per paket dan representasi simbolis alternatif. Editor pustaka simbol bisa melakukan pembuatan banyak pin secara simultan. Secara *default*, perubahan yang dilakukan pada suatu pin akan diterapkan untuk keseluruhan bagian untuk simbol dengan beberapa bagian dan keseluruhan representasi untuk simbol dengan representasi simbolis alternatif.


Satu-satunya pengecualian adalah tipe grafis dan nama pin. Dependensi ini dibuat agar pembuatan dan pengeditan pin menjadi lebih mudah pada kebanyakan kasus. Dependensi ini bisa dinonaktifkan dengan mengubah opsi  pada bilah alat utama. Anda akan bisa membuat pin untuk setiap bagian dan representasinya secara benar-benar terpisah.

Sebuah simbol bisa memiliki dua buah representasi simbolis (representasi yang dikenal sebagai "DeMorgan") dan bisa dibuat menjadi lebih dari satu bagian seperti halnya pada simbol dengan gerbang logika. Untuk beberapa simbol tertentu, Anda mungkin menginginkan beberapa elemen grafis dan pin yang berbeda. Seperti contoh relay yang ditunjukkan pada [bagian sebelumnya](#), sebuah relay dapat ditampilkan menjadi tiga bagian yang berbeda: koil, kontak saklar 1, dan kontak saklar 2.

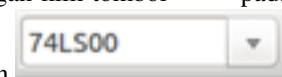
Manajemen simbol dengan beberapa bagian dan simbol dengan representasi simbolis alternatif bersifat fleksibel. Sebuah pin bisa bersifat umum atau spesifik untuk suatu bagian tertentu. Sebuah pin juga bisa bersifat umum untuk kedua representasi simbolis atau spesifik untuk salah satu representasi simbolis saja.

Secara *default*, pin bersifat spesifik untuk setiap representasi untuk setiap bagian, karena penomorannya berbeda untuk setiap bagian, dan desainnya berbeda untuk setiap representasi simbolis. Jika sebuah pin bersifat umum untuk semua bagian, pin tersebut hanya perlu digambar sekali saja, seperti pada kasus pin *power*.

Sebagai contoh adalah pin keluaran 7400, *quad dual input NAND gate*. Karena ada empat bagian dan dua representasi simbolis, maka terdapat delapan pin keluaran yang terpisah yang didefinisikan pada definisi simbol. Saat membuat sebuah simbol 7400 yang baru, bagian A dari representasi simbolis normal akan ditampilkan pada editor pustaka. Untuk mengedit gaya pin pada

representasi simbolis alternatif, kita harus terlebih dahulu mengaktifkannya dengan klik tombol  pada bilah alat. Untuk

mengedit nomor pin untuk tiap bagian, pilih bagian yang diinginkan menggunakan



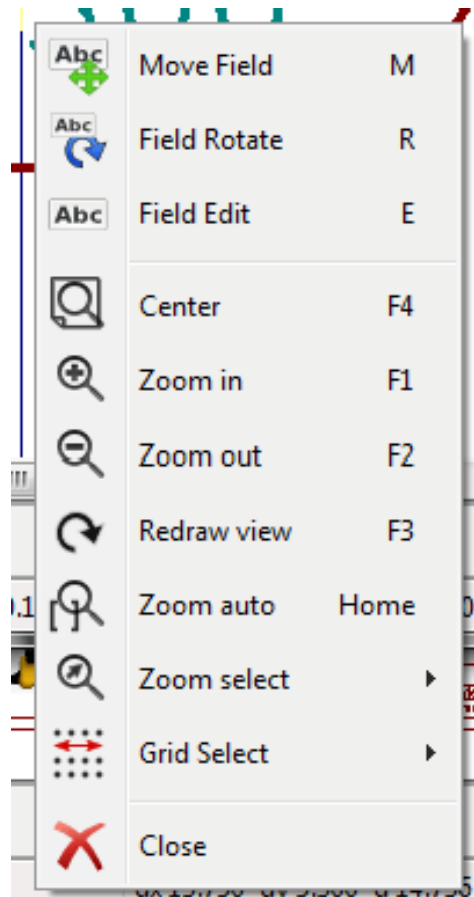
## 12.9 Atribut-atribut Simbol

Semua simbol pustaka didefinisikan oleh empat atribut *default*. Atribut penunjuk referensi, nilai, *footprint*, dan tautan berkas dokumentasi akan dibuat setiap kali sebuah simbol dibuat atau disalin. Atribut yang dibutuhkan hanya penunjuk referensi dan

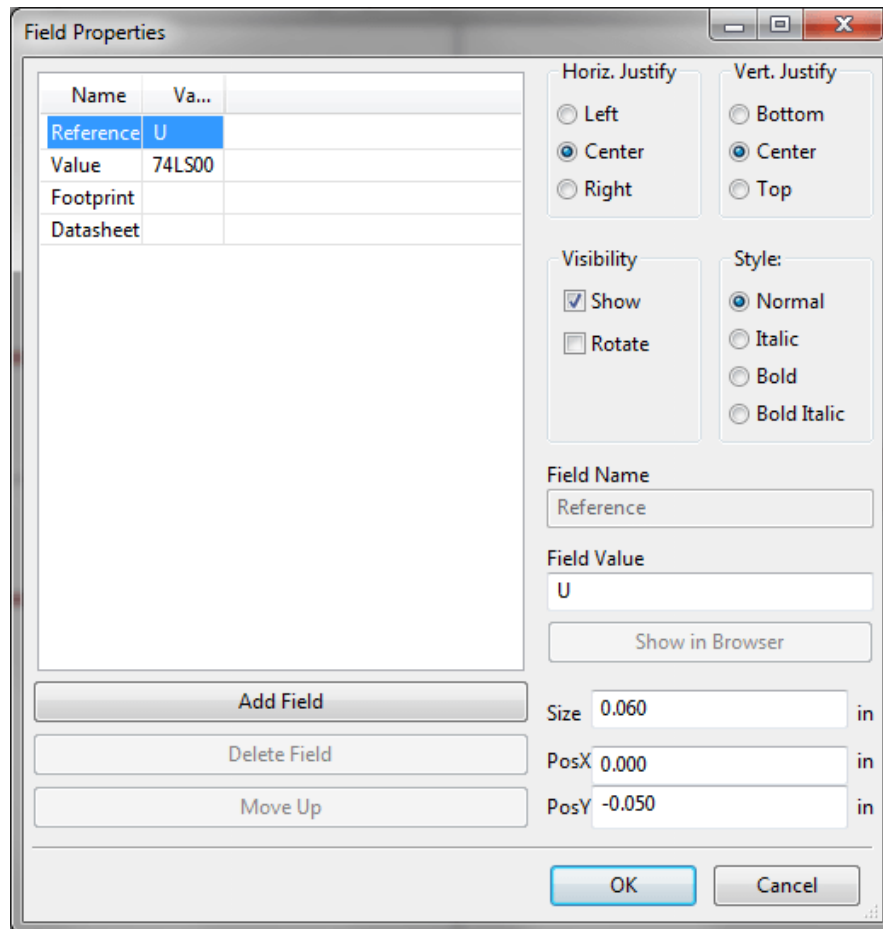
nilai. Untuk atribut yang sudah ada, Anda bisa menggunakan perintah menu konteks dengan klik-kanan pada pin. Simbol yang ada di pustaka umumnya didefinisikan dengan empat atribut *default* tersebut. Atribut tambahan seperti vendor, nomor *part*, harga dsb dapat ditambahkan ke simbol namun umumnya hal ini dilakukan di editor skematik sehingga atribut-atribut tambahan tersebut dapat diaplikasikan ke semua simbol pada skematik.

### 12.9.1 Mengedit Atribut Simbol

Untuk mengedit atribut simbol yang sudah ada, klik-kanan pada teks atribut untuk menampilkan menu konteks atribut seperti ditunjukkan pada gambar di bawah ini.



Untuk mengedit atribut yang masih kosong, menambahkan atribut baru, atau menghapus atribut opsional, gunakan **T** pada bilah alat utama untuk membuka kotak dialog properti atribut seperti ditunjukkan pada gambar di bawah ini.



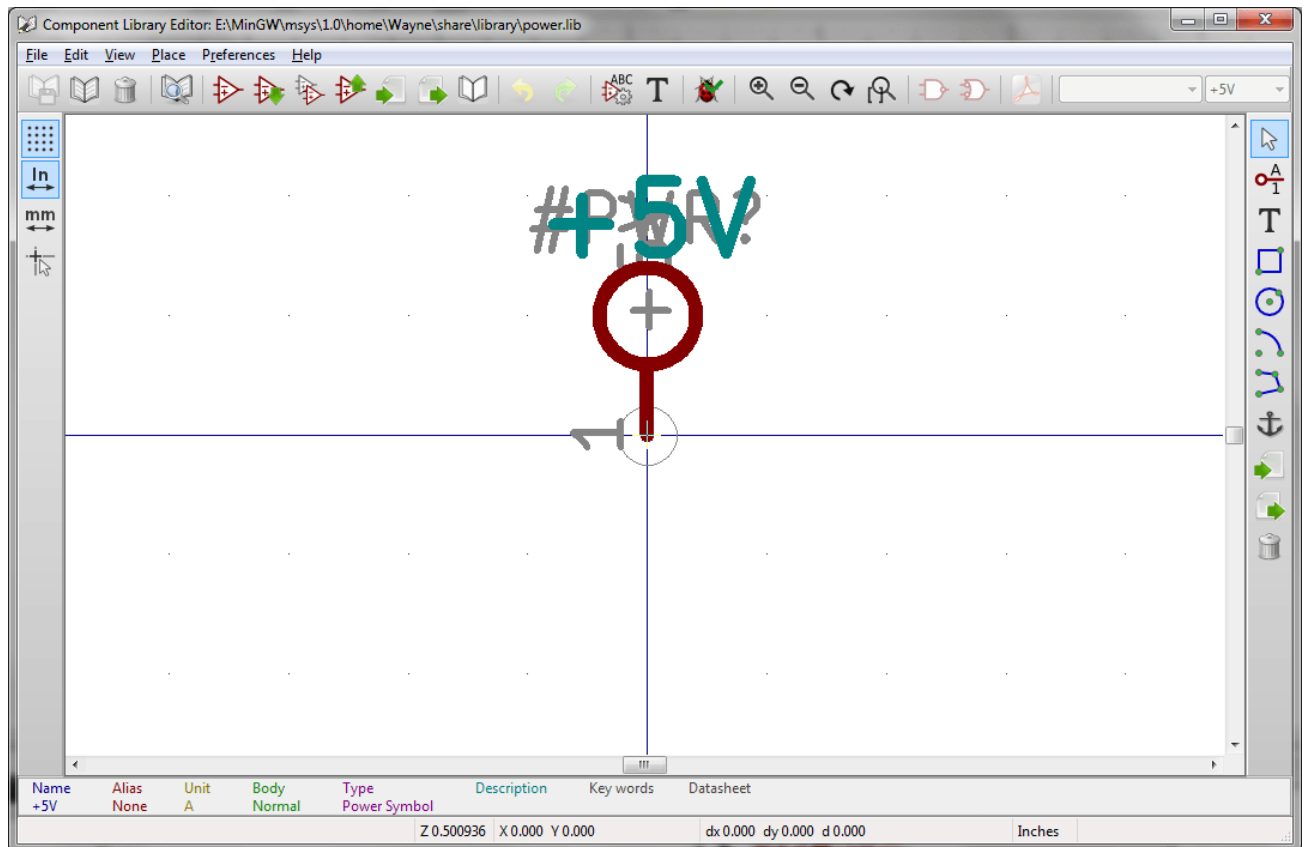
Atribut merupakan seksi teks yang berasosiasi dengan simbol. Hal ini berbeda dengan teks yang ada pada representasi grafis suatu simbol.

Catatan penting:

- Memodifikasi atribut nilai, secara efektif akan membuat sebuah simbol baru menggunakan simbol yang sedang dibuka sebagai titik awal untuk simbol yang baru. Simbol baru ini akan menggunakan nama dari atribut nilai saat Anda menyimpannya ke pustaka yang sedang dipilih.
- Kotak dialog editor atribut di atas harus digunakan untuk mengedit atribut yang masih kosong atau jika memiliki atribut yang tak-tampak.
- *Footprint* didefinisikan sebagai *footprint* absolut menggunakan format LIBNAME:FPNAME di mana LIBNAME adalah nama dari pustaka *footprint* yang didefinisikan pada tabel pustaka *footprint* (lihat bagian "*Footprint Library Table*" pada dokumen "*Pcbnew Reference Manual*"), dan FPNAME adalah nama *footprint* di pustaka LIBNAME.

## 12.10 Simbol *Power*

Simbol *power* dibuat dengan cara yang sama seperti simbol biasa. Mungkin akan lebih baik jika simbol ini diletakkan pada pustaka khusus seperti `power.lib`. Simbol *power* terdiri dari sebuah simbol grafis dan sebuah pin dengan tipe "*Power Invisible*". Simbol *port power* diperlakukan sama seperti simbol lainnya oleh perangkat lunak pembuat skematik. Ada beberapa hal penting yang perlu diperhatikan. Berikut adalah contoh dari simbol *power* +5V.



Untuk membuat sebuah simbol *power*, lakukan langkah-langkah berikut:

- Tambahkan sebuah pin dengan tipe "*Power input*" dengan nama +5V (penting karena nama ini akan membuat hubungan dengan *net* +5V), dengan nomor pin 1, panjang 0, dan gaya grafis "*Line*".
- Letakkan sebuah lingkaran kecil dan sebuah segmen dari pin untuk membuat lingkaran seperti gambar.
- Titik jangkar simbol ini berada di pin.
- Nilai simbol adalah +5V.
- Referensi simbol adalah \#+5V. Teks referensi tidak penting, kecuali bahwa karakter pertama harus dimulai dengan # untuk mengindikasikan bahwa simbol ini adalah simbol *power*. Sesuai kesepakatan, setiap simbol yang atribut referensinya dimulai dengan # tidak akan muncul di daftar simbol atau di Netlist dan referensi dideklarasikan sebagai tak-tampak.

Metode yang lebih mudah untuk membuat simbol *port power* baru adalah dengan menggunakan simbol lain sebagai model:

- Muat simbol *power* yang sudah ada.
- Edit nama pin dengan nama simbol *power* yang baru.
- Edit atribut nilai ke nama yang sama dengan nama pin, jika Anda ingin menampilkan nilai *port power*.
- Simpan simbol yang baru.

## Bab 13

# LibEdit - Simbol

### 13.1 Gambaran Umum

Sebuah simbol terdiri dari elemen-elemen berikut ini:

- Representasi grafis (bentuk geometris, teks).
- Pin.
- Atribut atau teks yang digunakan oleh *post processor*: Netlist, daftar simbol.

Ada dua atribut yang akan diinisialisasi: referensi dan nilai. Nama desain terkait dengan simbol, dan nama *footprint* yang terkait, atribut yang lain adalah atribut bebas, atribut tersebut bisa dibiarkan kosong, dan bisa diisi pada saat pembuatan skematik.

Bagaimanapun, manajemen dokumentasi yang terkait dengan simbol akan memudahkan riset, penggunaan, dan pemeliharaan pustaka. Dokumentasi terdiri dari:

- Sebaris komentar.
- Sebaris kata kunci seperti TTL CMOS NAND2, dipisahkan oleh spasi.
- Nama berkas terlampir (contohnya catatan aplikasi atau berkas PDF).

Direktori *default* untuk berkas terlampir:

```
kicad/share/library/doc
```

Jika tidak ditemukan:

```
kicad/library/doc
```

Di Linux:

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

```
/usr/share/kicad/library/doc
```

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

Penggunaan kata kunci akan memudahkan kita dalam melakukan pencarian secara selektif untuk suatu simbol berdasarkan berbagai kriteria seleksi. Komentar dan kata kunci ditampilkan di berbagai menu, dan terutama ketika Anda memilih sebuah simbol dari pustaka.

Simbol juga memiliki sebuah titik jangkar. Rotasi atau pencerminan dilakukan relatif terhadap titik jangkar, dan pada saat peletakan, titik ini digunakan sebagai posisi referensi. Oleh karena itu, kita perlu memposisikan titik jangkar secara akurat.

Sebuah simbol bisa memiliki alias, yaitu suatu nama ekuivalen. Dengan demikian, Anda bisa banyak mengurangi jumlah simbol yang harus dibuat (sebagai contoh, 74LS00 bisa memiliki alias seperti 7400, 74HC00, 74HCT00...).

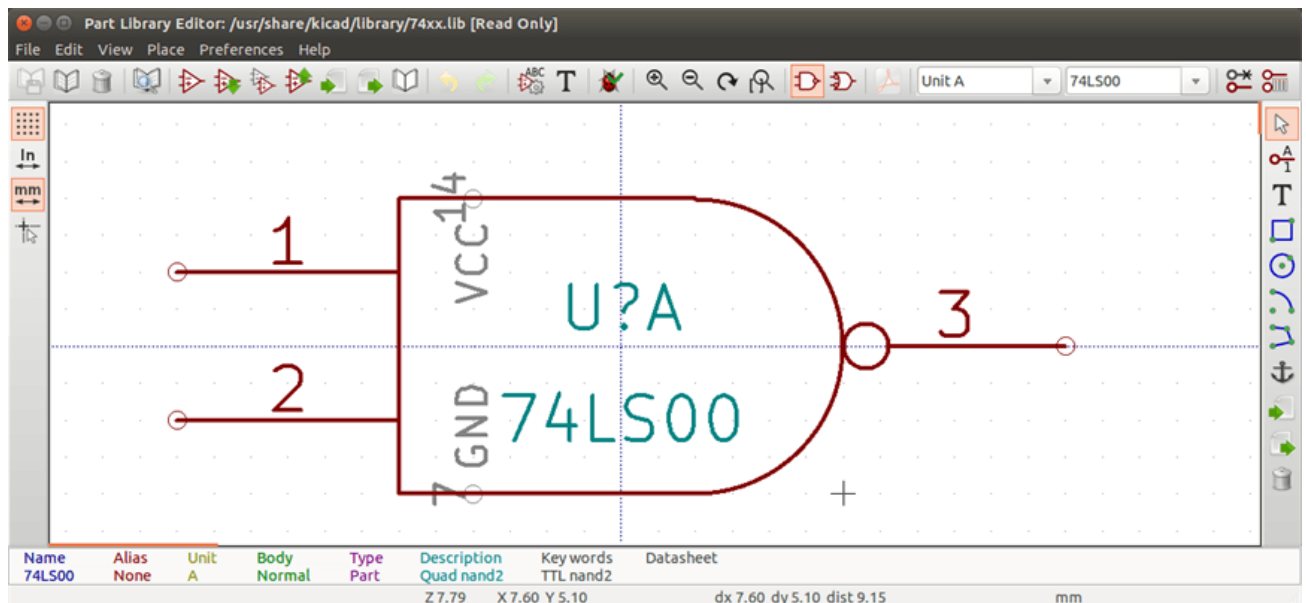
Simbol didistribusikan dalam bentuk pustaka (diklasifikasikan berdasarkan topik, atau manufaktur) untuk memfasilitasi manajemen simbol.

---



## 13.2 Memposisikan Jangkar Simbol

Jangkar berada pada koordinat (0,0) dan ditampilkan sebagai sumbu berwarna biru pada layar Anda.



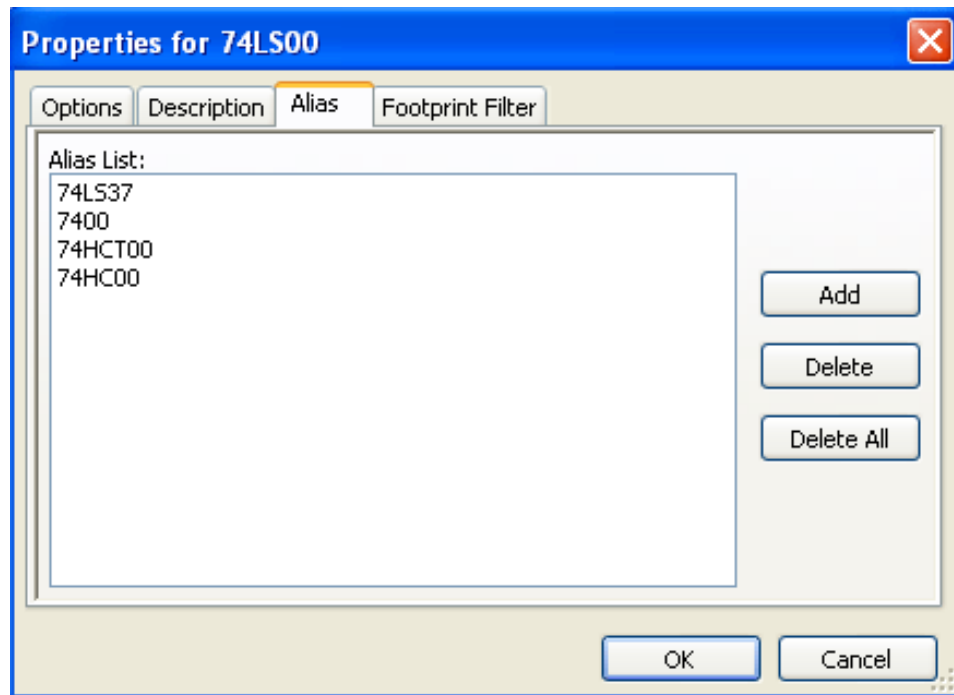
Jangkar dapat diposisikan ulang dengan menggunakan ikon  dan klik pada posisi jangkar yang diinginkan. Gambar akan secara otomatis ditengahkan pada titik jangkar yang baru.

## 13.3 Alias Simbol

Alias adalah nama lain yang terkait dengan simbol yang sama pada pustaka. Simbol dengan susunan pin dan representasi yang sama dapat diwakili oleh hanya satu simbol, dengan beberapa alias (misalnya 7400 dengan alias 74LS00, 74HC00, 74LS37).

Dengan menggunakan alias, kita bisa membangun pustaka yang lengkap dengan cepat. Selain itu, karena pustaka menjadi lebih ringkas, maka akan lebih cepat dimuat oleh KiCad.

Untuk mengubah daftar alias, Anda harus memilih jendela editor utama melalui ikon  dan pilih tab *Alias*.



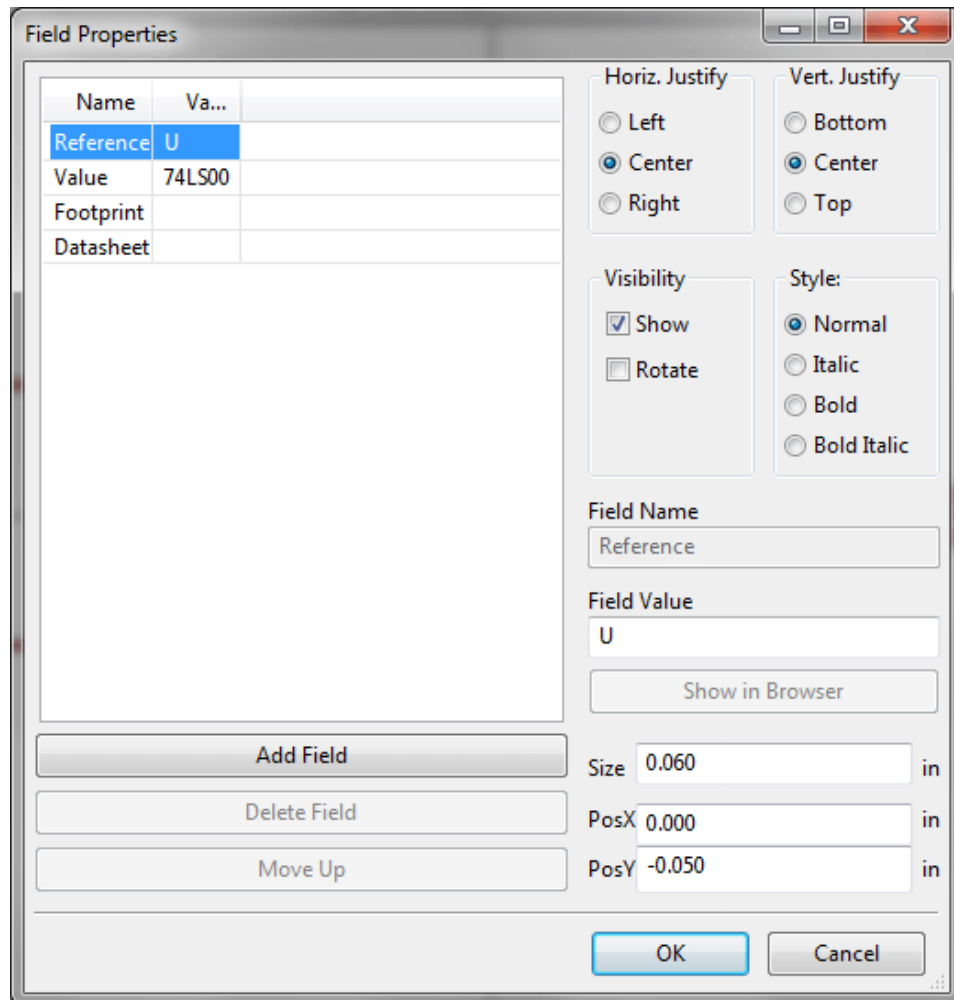
Anda dapat menambahkan atau menghapus alias. Alias saat ini tidak dapat dihapus karena sudah diedit.

Untuk menghapus semua alias, Anda harus memilih simbol utama. Simbol pertama dalam daftar alias di jendela pemilihan bilah alat utama.

## 13.4 Atribut-atribut Simbol

Editor atribut dapat diakses melalui **T**.


Ada empat atribut khusus (teks yang terlampir pada simbol), dan beberapa atribut pengguna yang dapat dikonfigurasi.

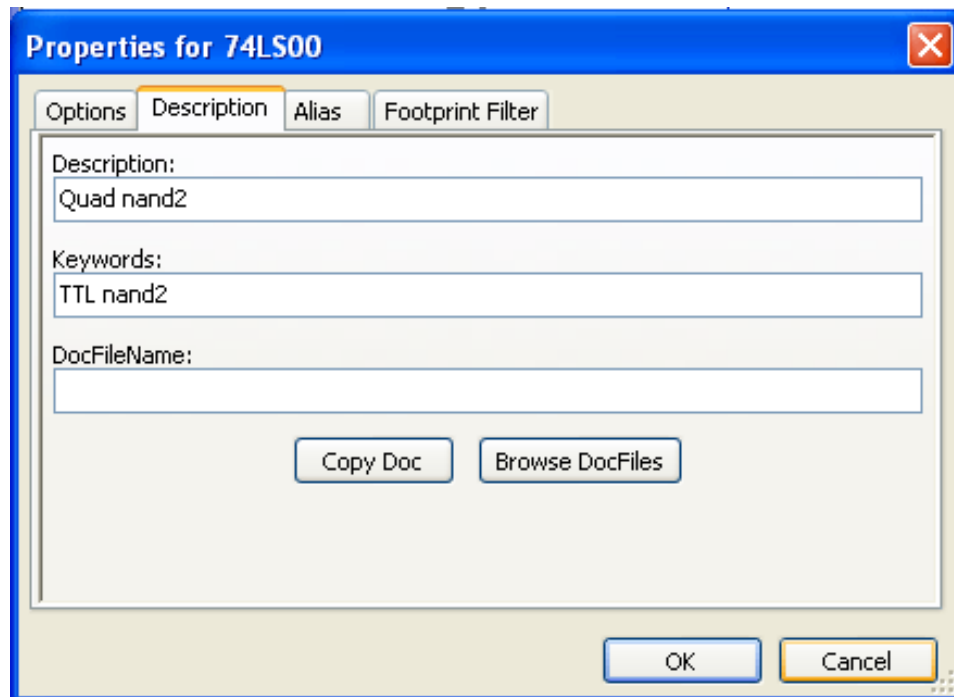


Atribut khusus:

- Referensi (*reference*).
- Nilai (*value*). Yaitu nama simbol di pustaka dan atribut nilai *default* di skematik.
- *Footprint*. Yaitu nama *footprint* yang digunakan untuk pembuatan PCB. Atribut ini tidak begitu berguna jika kita menggunakan CvPcb untuk membuat daftar *footprint*, namun bersifat wajib jika kita tidak menggunakan CvPcb.
- Lembar kerja (*sheet*). Ini adalah atribut cadangan, namun pada saat tulisan ini dibuat, atribut ini belum digunakan.

## 13.5 Dokumentasi Simbol

Untuk mengedit informasi dokumentasi, kita harus membuka jendela editor utama melalui ikon , kemudian pilih tab *Description*.



Pastikan untuk memilih alias yang benar, atau simbol utama, karena dokumentasi inilah yang akan membedakan antar alias. Tombol *Copy Doc* berguna untuk menyalin informasi dokumentasi dari simbol utama ke alias yang saat ini sedang diedit.

### 13.5.1 Kata Kunci Simbol

Kata kunci berguna bagi kita untuk melakukan pencarian secara selektif untuk suatu simbol berdasarkan kriteria pemilihan yang spesifik (fungsi, teknologi, dsb.).

Peralatan pencari pada Eeschema tidak membedakan huruf besar dan kecil. Kata kunci yang sering digunakan pada pustaka antara lain:

- CMOS TTL untuk keluarga logika
- AND2 NOR3 XOR2 INV... untuk gerbang (AND2 = 2 masukan gerbang AND, NOR3 = 3 masukan gerbang NOR).
- JKFF DFF... untuk fli-flop JK atau D.
- ADC, DAC, MUX...
- OpenCol untuk gerbang dengan keluaran *open-collector*. Jika Anda melakukan pencarian dengan kata kunci *NAND2 OpenCol*, Eeschema akan menampilkan daftar simbol yang memiliki kedua kata kunci tersebut.

### 13.5.2 Dokumentasi Simbol (Doc)

Baris komentar (dan kata kunci) ditampilkan di berbagai menu, terutama ketika Anda memilih sebuah simbol pada daftar simbol di pustaka dan di menu ViewLib.

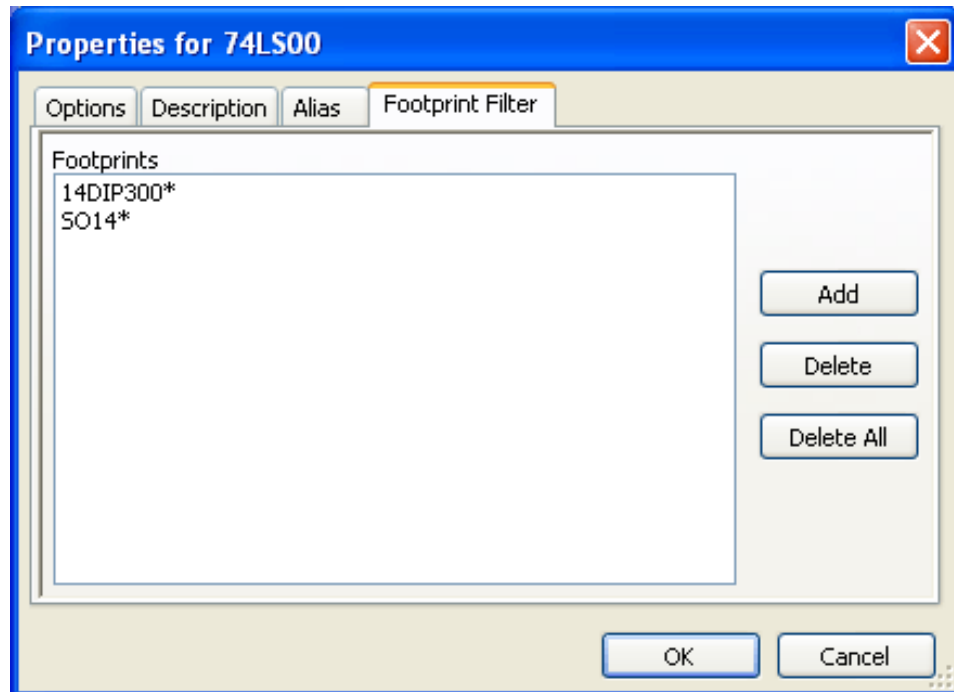
jika berkas Doc. ini tersedia, kita bisa mengaksesnya pada perangkat lunak pembuat skematik, melalui menu *pop-up* yang akan tampil jika kita melakukan klik-kanan pada simbol.

### 13.5.3 Berkas Dokumen Terkait (DocFileName)

Bagian ini adalah berkas terlampir yang tersedia (dokumentasi, skematik aplikasi), dalam bentuk berkas PDF, diagram skematik, dsb.

### 13.5.4 Penyaring *Footprint* untuk CvPcb

Anda bisa memasukkan sebuah daftar yang berisi *footprint* yang diperbolehkan untuk simbol. Daftar ini berfungsi sebagai penyaring yang akan digunakan oleh CvPcb untuk menampilkan hanya *footprint* yang diperbolehkan. Daftar yang kosong akan diabaikan.



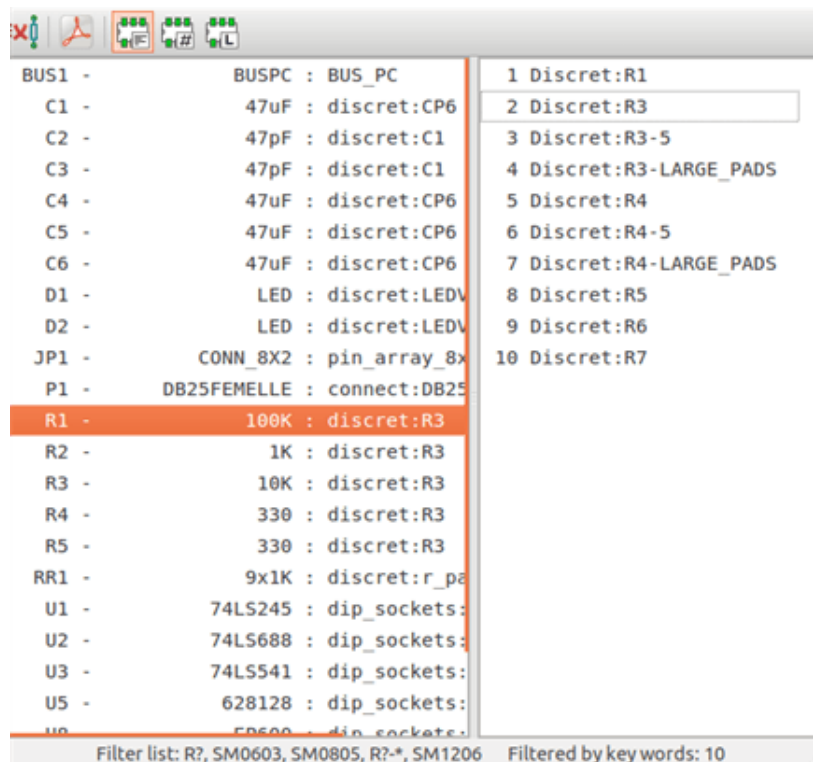
Karakter *wild-card* boleh digunakan.

Kode SO14\* artinya CvPcb akan menampilkan semua *footprint* dengan nama yang dimulai dengan SO14.

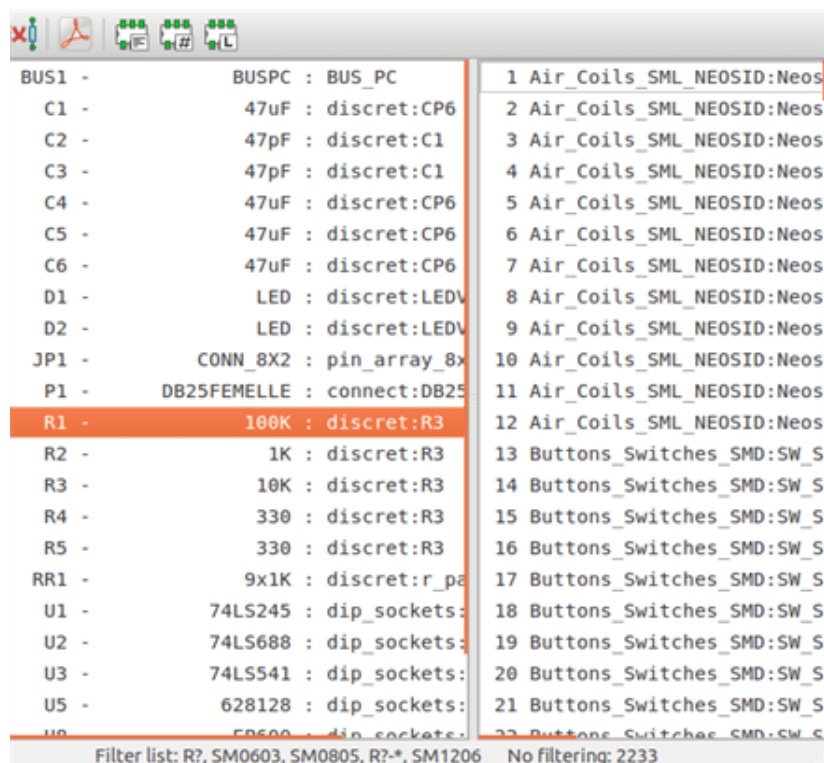
Untuk resistor, R? akan menampilkan semua *footprint* dengan nama 2 huruf yang dimulai dengan huruf R.

Gambar di bawah ini menampilkan contoh penggunaan penyaring.

Menggunakan penyaring



Tanpa penyaring




## 13.6 Pustaka Simbol


Anda bisa dengan mudah membuat sebuah berkas pustaka simbol grafis yang berisi simbol-simbol yang sering digunakan. Berkas ini bisa digunakan untuk pembuatan simbol (segitiga, bentuk gerbang AND, OR, Exclusive OR, dsb.).

Berkas tersebut disimpan secara *default* di direktori pustaka dan memiliki ekstensi `.sym`. Simbol-simbol tersebut tidak dikumpulkan di pustaka seperti simbol pada umumnya, karena simbol-simbol tersebut jumlahnya tidak terlalu banyak.

### 13.6.1 Mengekspor atau Membuat Simbol

Suatu simbol bisa diekspor menggunakan tombol . Secara umum, Anda bisa membuat hanya satu gambar saja, dan sebaiknya Anda menghapus semua pin, jika ada.


### 13.6.2 Mengimpor Simbol

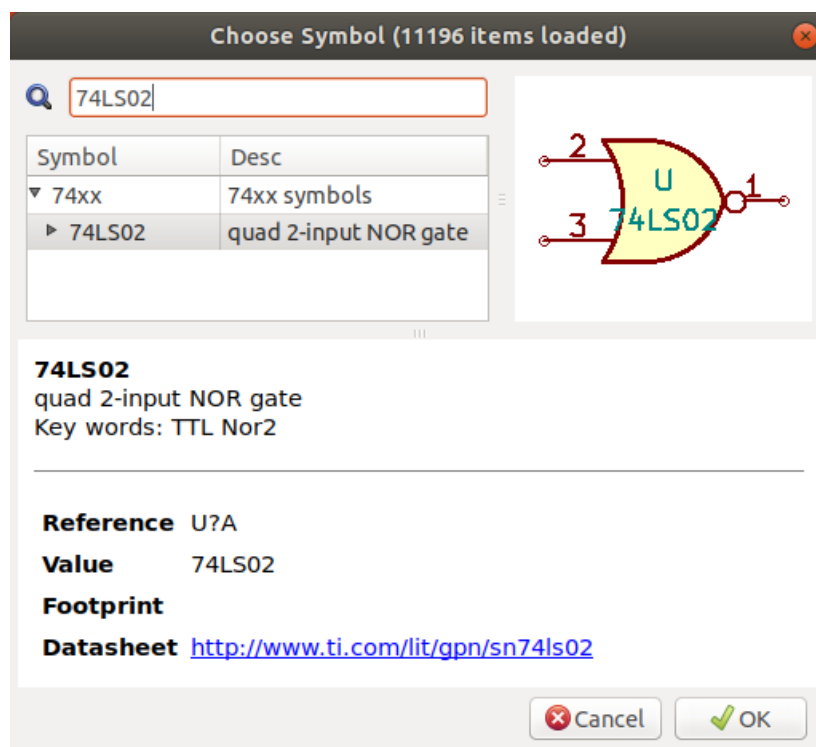
Mengimpor berarti menambahkan gambar ke simbol yang sedang diedit. Kita bisa mengimpor simbol menggunakan tombol . Gambar yang berhasil diimpor akan ditambahkan ke gambar yang sedang dibuka.

## Bab 14

# Peramban Pustaka Simbol

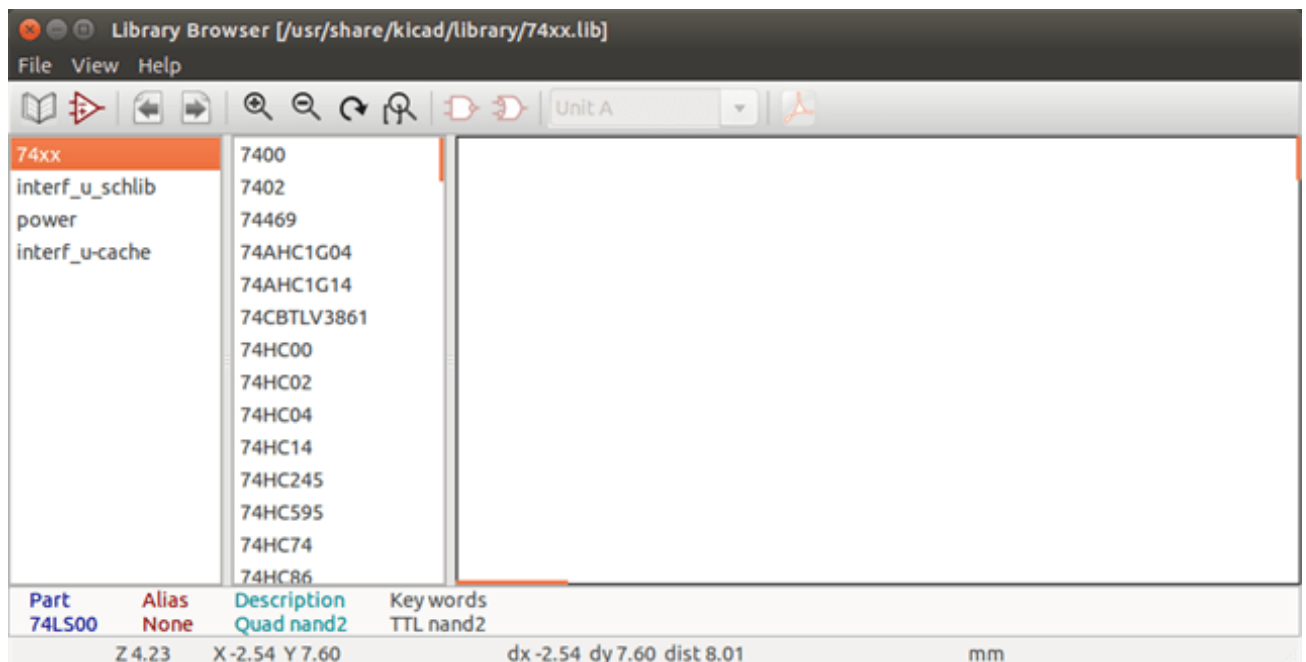
### 14.1 Pengenalan

Peramban pustaka simbol (*Symbol Library Browser*) berguna untuk memeriksa isi pustaka simbol secara cepat. Alat ini bisa diakses melalui ikon  pada bilah alat utama, dan memilih "Library Browser" pada menu "View" atau klik-ganda gambar simbol pada jendela "Choose Symbol".

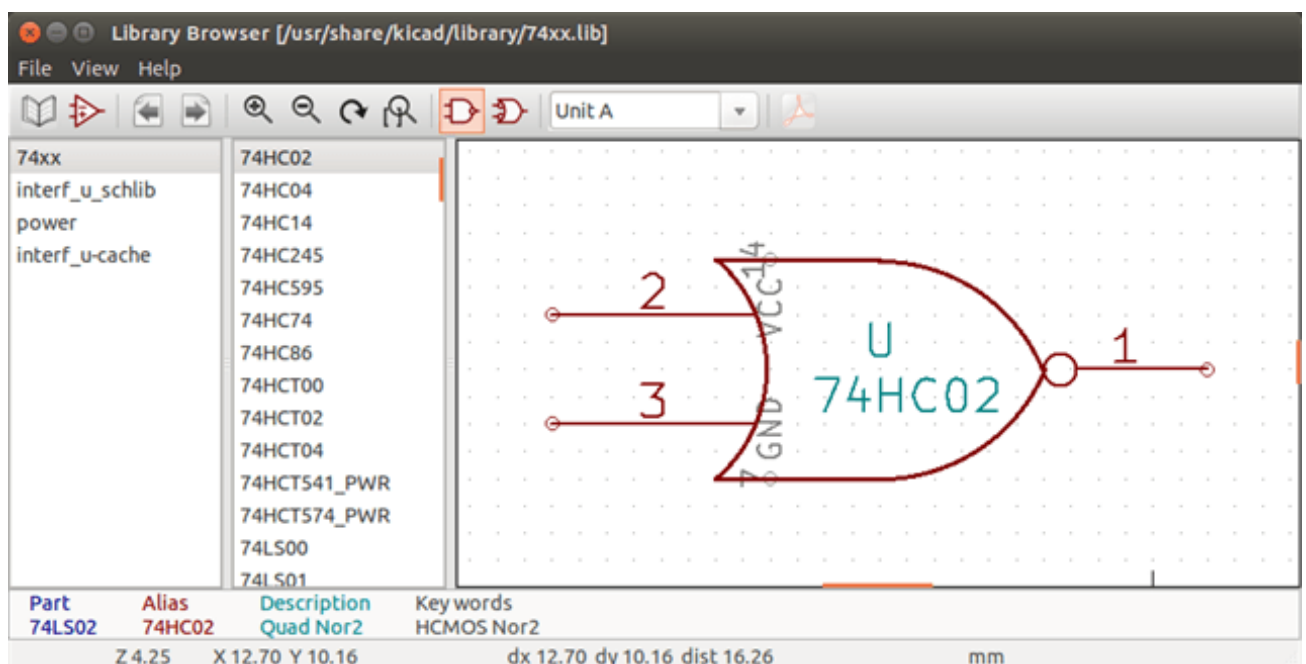




## 14.2 Viewlib - Tampilan Utama



Untuk memeriksa isi pustaka, pilih satu pustaka dari daftar pada panel sebelah kiri. Semua simbol pada pustaka yang dipilih akan ditampilkan pada panel kedua. Pilih nama simbol untuk melihat simbol tersebut.












## 14.3 Bilah Alat Atas pada Peramban Pustaka Simbol

Bilah alat atas pada Peramban Pustaka Simbol ditampilkan pada gambar di bawah.



The available commands are:

	Pemilihan pustaka yang juga bisa dipilih pada daftar yang ditampilkan.
	Pemilihan simbol yang juga bisa dipilih pada daftar yang ditampilkan.
	Menampilkan simbol sebelumnya.
	Menampilkan simbol berikutnya.
	Peralatan perbesar/perkecil tampilan.
	Pemilihan representasi (normal atau terkonversi) jika tersedia.
	Pemilihan bagian untuk simbol yang terdiri dari beberapa bagian.
	Jika tersedia, tampilkan dokumen yang terkait. Tombol akan tersedia hanya ketika dibuka melalui kotak dialog peletakan simbol dari Eeschema.
	Menutup peramban dan meletakkan simbol yang terpilih ke Eeschema. Ikon ini tersedia hanya jika peramban dibuka melalui Eeschema (klik-ganda pada simbol di pemilih komponen).

## Bab 15

## Membuat Netlist dan Berkas BOM Kustom

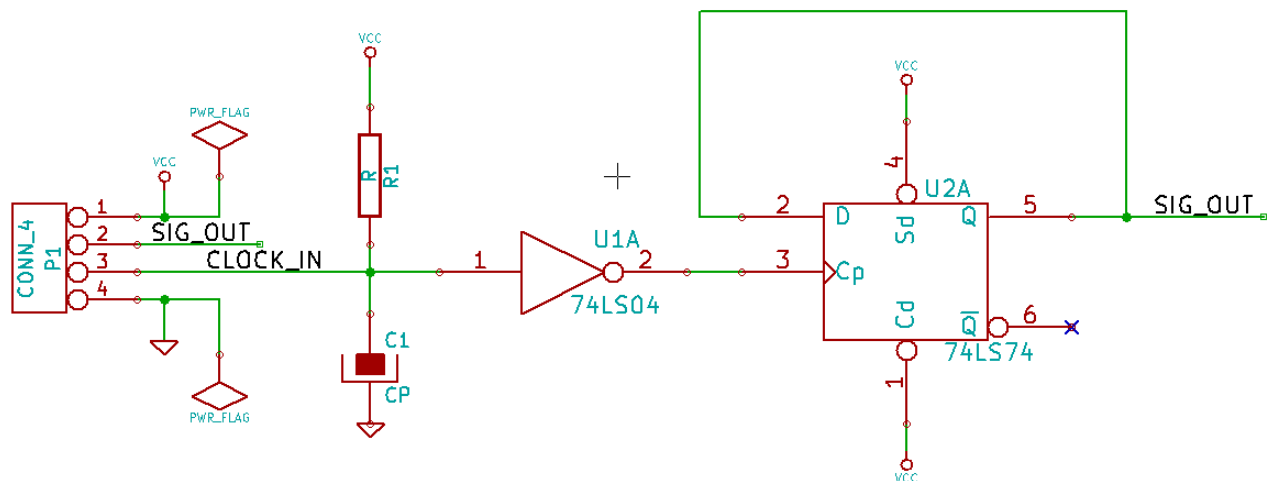
## 15.1 Berkas Netlist Antara

Berkas BOM dan Netlist bisa didapatkan dengan mengkonversi berkas Netlist antara yang dibuat oleh Eeschema.

Berkas ini menggunakan sintaks XML dan disebut sebagai Netlist antara. Netlist antara berisi sejumlah besar data mengenai papan sirkuit Anda. Maka dari itu, berkas ini bisa digunakan pasca-pemrosesan untuk membuat BOM atau berkas laporan lainnya.

Tergantung dari keluarannya (BOM atau Netlist), sejumlah berkas turunan dari Netlist antara akan digunakan pasca-pemrosesan.

### 15.1.1 Contoh Skematik



### 15.1.2 Contoh Berkas Netlist Antara

Netlist antara (menggunakan sintaks XML) yang terkait dengan sirkuit di atas ditunjukkan seperti di bawah ini.

```
<?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="/" tstamp="/" />
      <tstamp>4C6E2141</tstamp>
    </comp>
    <comp ref="U2">
      <value>74LS74</value>
      <libsource lib="74xx" part="74LS74"/>
      <sheetpath names="/" tstamp="/" />
      <tstamp>4C6E20BA</tstamp>
    </comp>
    <comp ref="U1">
      <value>74LS04</value>
      <libsource lib="74xx" part="74LS04"/>
      <sheetpath names="/" tstamp="/" />
      <tstamp>4C6E20A6</tstamp>
    </comp>
    <comp ref="C1">
      <value>CP</value>
      <libsource lib="device" part="CP"/>
      <sheetpath names="/" tstamp="/" />
      <tstamp>4C6E2094</tstamp>
    </comp>
    <comp ref="R1">
      <value>R</value>
      <libsource lib="device" part="R"/>
      <sheetpath names="/" tstamp="/" />
      <tstamp>4C6E208A</tstamp>
    </comp>
  </components>
  <libparts>
    <libpart lib="device" part="C">
      <description>Condensateur non polarise</description>
      <footprints>
        <fp>SM*</fp>
        <fp>C?</fp>
        <fp>C1-1</fp>
      </footprints>
      <fields>
        <field name="Reference">C</field>
        <field name="Value">C</field>
      </fields>
      <pins>
        <pin num="1" name="~" type="passive"/>
        <pin num="2" name="~" type="passive"/>
      </pins>
    </libpart>
    <libpart lib="device" part="R">
      <description>Resistance</description>
      <footprints>
        <fp>R?</fp>
        <fp>SM0603</fp>
      </footprints>
    </libpart>
  </libparts>
</export>
```

```
<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 Konversi ke Format Netlist yang Baru

Dengan menerapkan sebuah penyaring pasca-pemrosesan ke berkas Netlist antara, Anda bisa membuat berkas-berkas Netlist asing, demikian pula berkas-berkas BOM. Karena proses konversi ini adalah transformasi dari teks ke teks, penyaring pasca-pemrosesan dapat dibuat menggunakan Python, XSLT, atau peralatan lainnya yang mampu mengolah masukan XML.

XSLT adalah bahasa XML yang sangat sesuai untuk transformasi XML. Ada sebuah program yang disebut *xsltproc* yang bisa diunduh dan diinstal secara bebas. Program *xsltproc* dapat digunakan untuk membaca berkas masukan Netlist antara dalam format XML, menerapkan sebuah *stylesheet* pada berkas masukan tersebut, dan menyimpan hasil transformasinya ke sebuah berkas keluaran. Penggunaan *xsltproc* memerlukan sebuah berkas *stylesheet* yang menggunakan konvensi XSLT. Proses konversi yang lengkap ditangani oleh Eeschema, setelah dikonfigurasi untuk menjalankan *xsltproc* dengan cara yang spesifik.

## 15.3 Pendekatan XSLT

Dokumen yang menjelaskan Transformasi XSL (XSLT) tersedia di alamat berikut:

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

### 15.3.1 Membuat Berkas Netlist Pads-Pcb

Format pads-pcb terdiri dari dua bagian.

- Daftar *footprint*.
- Daftar *net* : pengelompokan referensi *pad* berdasarkan *net*.

Berikut adalah sebuah *stylesheet* untuk mengkonversi berkas Netlist antara ke format Netlist pads-pcb:

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

  How to use:
    https://lists.launchpad.net/kicad-developers/msg05157.html
-->

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

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

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

<!-- for each component -->
<xsl:template match="comp">
  <xsl:text> </xsl:text>
  <xsl:value-of select="@ref"/>
  <xsl:text> </xsl:text>
  <xsl:choose>
```

```

        <xsl:when test = "footprint != '' ">
            <xsl:apply-templates select="footprint"/>
        </xsl:when>
        <xsl:otherwise>
            <xsl:text>unknown</xsl:text>
        </xsl:otherwise>
    </xsl:choose>
    <xsl:text>&nl;</xsl:text>
</xsl:template>

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

<!-- for each node -->
<xsl:template match="node">
    <xsl:text> </xsl:text>
    <xsl:value-of select="@ref"/>
    <xsl:text>.</xsl:text>
    <xsl:value-of select="@pin"/>
    <xsl:text>&nl;</xsl:text>
</xsl:template>

</xsl:stylesheet>

```

Dan berikut adalah berkas keluaran pads-pcb setelah menjalankan 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*
```

Perintah untuk menjalankan konversi ini adalah:

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

### 15.3.2 Membuat Berkas Netlist Cadstar

Format Cadstar terdiri dari dua bagian.

- Daftar *footprint*.
- Daftar *net* : pengelompokan referensi *pad* berdasarkan *net*.

Berikut adalah berkas *stylesheet* untuk menjalankan konversi ini:

```
<?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>

```

Berikut adalah berkas keluaran 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 Membuat Berkas Netlist OrcadPCB2

Format ini hanya memiliki satu bagian saja, yaitu daftar *footprint*. Setiap *footprint* berisi daftar *pad* dengan referensi ke sebuah *net*.

Berikut adalah *stylesheet* untuk konversi ini:

```

<?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 number 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" 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:template>

</xsl:stylesheet>

```

Berikut adalah berkas keluaran 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 )
)
)
*

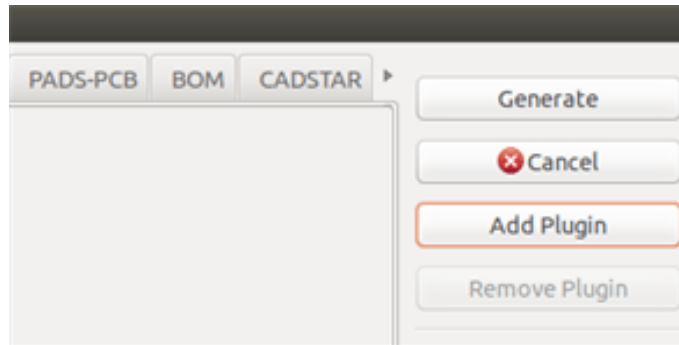
```

### 15.3.4 Antarmuka *Plugin* Eeschema

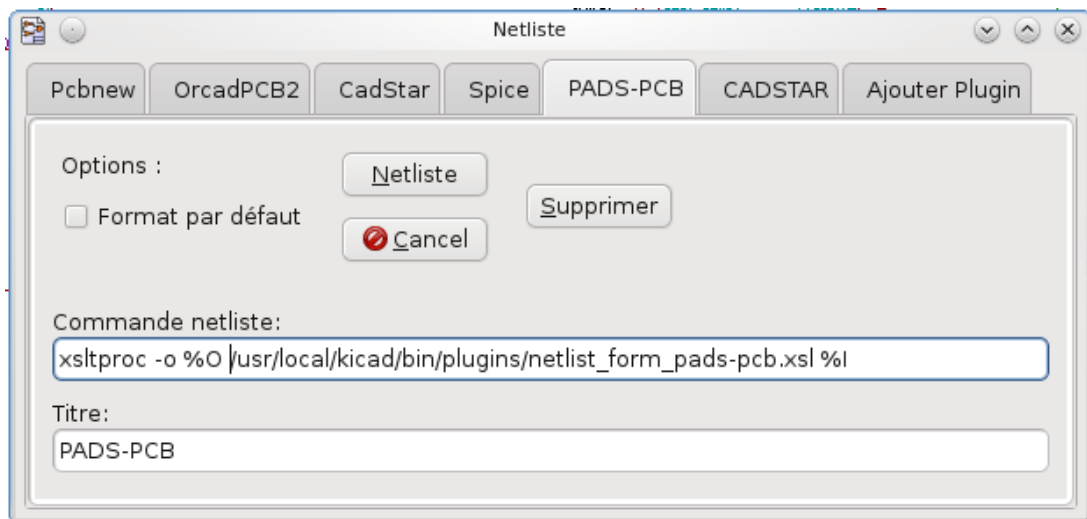
Konverter Netlist antara bisa dijalankan secara otomatis di Eeschema.

#### 15.3.4.1 Membuka Kotak Dialog *Plugin*

Kita bisa menambahkan sebuah tab antarmuka *plugin* Netlist yang baru dengan klik pada tombol *Add Plugin*



Berikut adalah bentuk data konfigurasi untuk tab PadsPcb:



#### 15.3.4.2 Parameter Konfigurasi *Plugin*

Kotak dialog konfigurasi *plugin* Eeschema memerlukan informasi berikut:

- Judul: misalnya nama dari format Netlist.
- Perintah untuk menjalankan konverter.

Setelah Anda menekan tombol Netlist, akan terjadi hal berikut ini:

1. Eeschema membuat sebuah berkas Netlist antara dengan ekstensi \*.xml, misalnya test.xml.
2. Eeschema menjalankan *plugin* dengan membaca test.xml dan membuat test.net.



### 15.3.4.3 Membuat Berkas Netlist dengan Baris Perintah

Diasumsikan kita menggunakan program *xsltproc.exe* untuk menerapkan *stylesheet* ke berkas antara, maka *xsltproc.exe* dieksekusi dengan perintah berikut:

```
xsltproc.exe -o <nama berkas keluaran> <nama berkas style-sheet> <berkas XML yang akan dikonversi>
```

Berikut baris perintah yang digunakan KiCad di sistem operasi Windows:

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

Di sistem operasi Linux, baris perintahnya adalah:

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

Di mana *netlist\_form\_pads-pcb.xml* adalah *stylesheet* yang digunakan. Jangan lupa untuk menggunakan tanda petik ganda pada nama berkas, supaya Anda bisa menggunakan spasi setelah dilakukan konversi oleh Eeschema.

Baris perintah menerima parameter berikut untuk nama berkas:

Parameter yang didukung adalah:

- %B ⇒ nama berkas dan *path* untuk berkas keluaran yang dipilih, tanpa *path* dan ekstensi.
- %I ⇒ nama berkas lengkap dan *path* dari berkas masukan sementara (berkas Netlist antara).
- %O ⇒ nama berkas lengkap dan *path* untuk berkas keluaran yang dipilih.

%I akan diganti dengan nama berkas antara yang sesungguhnya.

%O akan diganti dengan nama berkas keluaran yang sesungguhnya.

### 15.3.4.4 Format Baris Perintah: Contoh xsltproc

Format baris perintah untuk *xsltproc* adalah sebagai berikut:

```
<path ke xsltproc> xsltproc <parameter xsltproc>
```

Penggunaan di Windows:

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

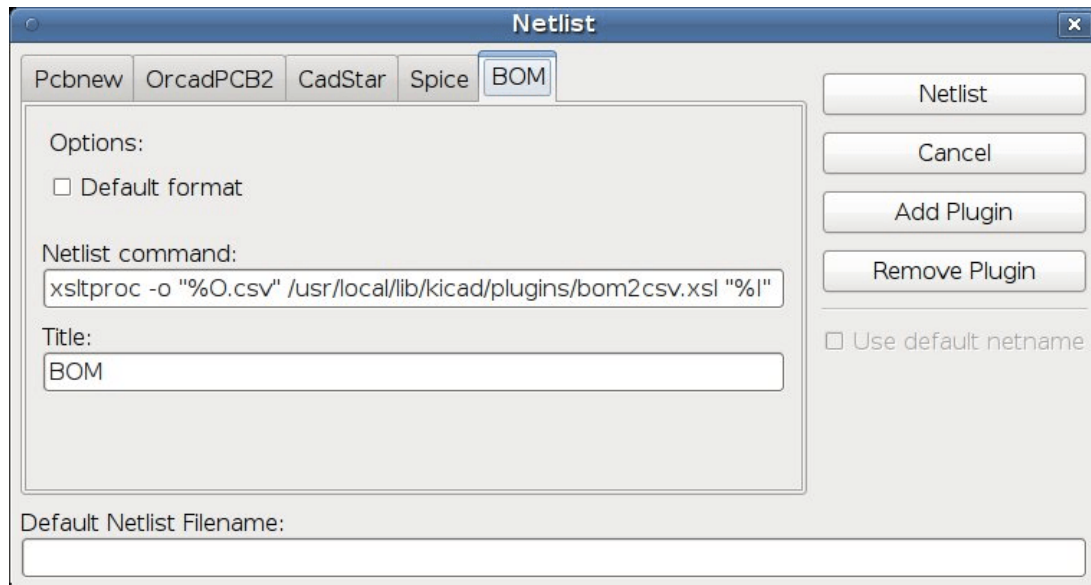
Penggunaan di Linux:

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

Contoh di atas mengasumsikan *xsltproc* terinstal pada komputer Windows dan semua berkas berada di direktori *kicad/bin*.

### 15.3.5 Pembuatan *Bill of Materials*

Karena berkas Netlist antara berisi semua informasi mengenai komponen-komponen yang digunakan, maka kita bisa membuat BOM berdasarkan berkas tersebut. Berikut adalah tampilan jendela pengaturan *plugin* (di Linux) untuk membuat berkas *Bill of Materials* (BOM) kustom:



Path menuju berkas *stylesheet* `bom2csv.xsl` tergantung pada sistem yang kita gunakan. *Stylesheet* XSLT terbaik yang ada saat ini untuk pembuatan BOM adalah `bom2csv.xsl`. Anda bebas melakukan modifikasi sesuai kebutuhan, dan jika Anda merasa hasil modifikasi tersebut akan bermanfaat untuk orang lain, Anda bisa memasukkan hasil modifikasi tersebut sebagai bagian dari proyek KiCad.

## 15.4 Format Baris Perintah: Contoh Skrip Python

Format baris perintah untuk Python adalah sebagai berikut:

```
python <nama berkas skrip> <nama berkas masukan> <nama berkas keluaran>
```

Penggunaan di Windows:

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

Penggunaan di Linux:

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

Dengan asumsi Python telah terinstal di komputer Anda.

## 15.5 Struktur Netlist Antara

Contoh berikut ini menunjukkan format berkas Netlist.

```
<?xml version="1.0" encoding="utf-8"?>
<export version="D">
  <design>
    <source>F:\kicad_aux\netlist_test\netlist_test.sch</source>
    <date>29/08/2010 21:07:51</date>
    <tool>eeschema (2010-08-28 BZR 2458)-unstable</tool>
  </design>
  <components>
    <comp ref="P1">
      <value>CONN_4</value>
      <libsource lib="conn" part="CONN_4"/>
      <sheetpath names="/" tstamps="/" />
      <tstamp>4C6E2141</tstamp>
    </comp>
```

```
<comp ref="U2">
  <value>74LS74</value>
  <libsource lib="74xx" part="74LS74"/>
  <sheetpath names="/" tstamps="/">
  <tstamp>4C6E20BA</tstamp>
</comp>
<comp ref="U1">
  <value>74LS04</value>
  <libsource lib="74xx" part="74LS04"/>
  <sheetpath names="/" tstamps="/">
  <tstamp>4C6E20A6</tstamp>
</comp>
<comp ref="C1">
  <value>CP</value>
  <libsource lib="device" part="CP"/>
  <sheetpath names="/" tstamps="/">
  <tstamp>4C6E2094</tstamp>
<comp ref="R1">
  <value>R</value>
  <libsource lib="device" part="R"/>
  <sheetpath names="/" tstamps="/">
  <tstamp>4C6E208A</tstamp>
</comp>
</components>
<libparts/>
<libraries/>
<nets>
  <net code="1" name="GND">
    <node ref="U1" pin="7"/>
    <node ref="C1" pin="2"/>
    <node ref="U2" pin="7"/>
    <node ref="P1" pin="4"/>
  </net>
  <net code="2" name="VCC">
    <node ref="R1" pin="1"/>
    <node ref="U1" pin="14"/>
    <node ref="U2" pin="4"/>
    <node ref="U2" pin="1"/>
    <node ref="U2" pin="14"/>
    <node ref="P1" pin="1"/>
  </net>
  <net code="3" name="">
    <node ref="U2" pin="6"/>
  </net>
  <net code="4" name="">
    <node ref="U1" pin="2"/>
    <node ref="U2" pin="3"/>
  </net>
  <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 Struktur Berkas Netlist Umum

Berkas Netlist antara terdiri dari lima bagian.

- Bagian *header*.
- Bagian komponen.
- Bagian *lib parts*.
- Bagian pustaka.
- Bagian *net*.

Isi berkas ini memiliki pembatas `<export>`

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

### 15.5.2 Bagian Header

Bagian *header* memiliki pembatas `<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>
```

Bagian ini bisa dianggap sebagai bagian komentar.

### 15.5.3 Bagian Komponen

Bagian komponen memiliki pembatas `<components>`

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

Bagian ini berisi daftar komponen yang ada pada skematik Anda. Setiap komponen dijelaskan seperti ini:

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

<b>libsource</b>	nama pustaka di mana komponen ini berada.
<b>part</b>	nama komponen di dalam pustaka.
<b>sheetpath</b>	<i>Path</i> lembar kerja di dalam hirarki: mengidentifikasi lembar kerja di dalam keseluruhan hirarki skematik.
<b>tstamps (time stamps)</b>	tanda waktu berkas skematik.
<b>tstamp (time stamp)</b>	tanda waktu komponen.

### 15.5.3.1 Catatan Mengenai Tanda Waktu Komponen

Untuk mengidentifikasi sebuah komponen di dalam Netlist dan kemudian di papan sirkuit, digunakan referensi tanda waktu yang bersifat unik untuk setiap komponen. KiCad juga menyediakan cara lain untuk mengidentifikasi sebuah komponen dengan *footprint* yang terkait di papan sirkuit. Dengan demikian, bisa dilakukan anotasi ulang komponen-komponen pada proyek skematik, dan tautan antara komponen dan *footprint* akan tetap terjaga.

Tanda waktu menjadi pengidentifikasi yang bersifat unik untuk setiap komponen atau lembar kerja pada suatu proyek skematik. Namun di hirarki kompleks, satu lembar kerja yang sama bisa digunakan lebih dari satu kali, sehingga lembar kerja ini akan berisi komponen-komponen dengan tanda waktu yang sama.

Lembar kerja tersebut memiliki sebuah pengidentifikasi unik di hirarki kompleks: *path* lembar kerja. Maka, komponen (di dalam hirarki kompleks) memiliki pengidentifikasi unik berupa: *path* lembar kerja + tanda waktu.

### 15.5.4 Bagian *Libparts*

Bagian *libparts* memiliki pembatas `<libparts>`, dan isi dari bagian ini didefinisikan di pustaka skematik. Bagian ini berisi:

- The allowed footprints names (names use wildcards) delimiter `<fp>`.
- Pembatas `<fields>`, atribut-atribut yang didefinisikan di dalam pustaka.
- Pembatas `<pins>`, daftar pin.

```
<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>
```

Baris seperti `<pin num="1" type="passive"/>` juga menampilkan tipe pin elektrik. Tipe pin elektrik yang bisa digunakan antara lain:

Input	Pin masukan biasa
Output	Pin keluaran biasa
Bidirectional	Masukan atau keluaran
Tri-state	<i>Bus</i> masukan/keluaran
Passive	Komponen pasif biasa
Unspecified	Tipe elektrik yang tidak diketahui
Power input	Masukan <i>power</i> suatu komponen
Power output	Keluaran <i>power</i> , seperti keluaran regulator
Open collector	<i>Open collector</i> yang biasanya ada di komparator analog
Open emitter	<i>Open emitter</i> yang terkadang ada di logika
Not connected	Harus dibiarkan terbuka di skematik

### 15.5.5 Bagian Pustaka

Bagian pustaka memiliki pembatas `<libraries>`. Bagian ini berisi daftar pustaka skematik yang digunakan pada proyek.

```
<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 Bagian Net

Bagian *net* memiliki pembatas `<nets>`. Bagian ini berisi "konektivitas" skematik.

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

Bagian ini menampilkan daftar semua *net* pada skematik.

Sebuah *net* bisa memiliki daftar seperti berikut ini.

```
<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	Pengidentifikasi internal untuk <i>net</i> ini
name	Nama <i>net</i>
node	Referensi pin yang terhubung ke <i>net</i> ini

## 15.6 Lebih Lanjut Tentang xsltproc

Silakan mengacu pada halaman: <http://xmlsoft.org/XSLT/xsltproc.html>

### 15.6.1 Pengenalan

xsltproc adalah sebuah alat baris perintah untuk menerapkan *stylesheet* XSLT ke dokumen XML. Alat ini dikembangkan sebagai bagian dari proyek GNOME, namun dapat digunakan secara independen di luar desktop GNOME.

xsltproc dijalankan melalui baris perintah dengan nama *stylesheet* yang akan digunakan diikuti dengan nama berkas yang mana *stylesheet* akan diterapkan. Alat ini akan menggunakan masukan standar jika nama berkas yang diberikan berupa -.

Jika tersedia sebuah *stylesheet* di dalam dokumen XML dengan *Style-sheet Processing Instruction*, kita tidak perlu lagi menyebutkan sebuah *stylesheet* pada baris perintah. xsltproc akan secara otomatis mendeteksi *stylesheet* yang tersedia dan menggunakannya. Secara *default*, keluaran dari perintah ini adalah ke *stdout*. Keluaran bisa disimpan ke sebuah berkas dengan menggunakan opsi -o.

### 15.6.2 Sinopsis

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

### 15.6.3 Opsi-opsi Baris Perintah

-V atau --version

Menampilkan versi libxml dan libxslt yang digunakan.

-v atau --verbose

Menampilkan setiap langkah yang diambil oleh xsltproc dalam memproses *stylesheet* dan dokumen.

-o atau --output file

Mengarahkan keluaran ke berkas dengan nama *file*. Untuk keluaran lebih dari satu, juga dikenal sebagai "*chunking*", -o direktori/ akan mengarahkan berkas keluaran ke direktori yang ditentukan. Direktori tersebut harus sudah ada sebelumnya.

--timing

Menampilkan waktu yang digunakan untuk menguraikan *stylesheet*, menguraikan dokumen, dan menerapkan *stylesheet* dan menyimpan hasilnya. Ditampilkan dalam satuan milidetik.

--repeat

Menjalankan transformasi sebanyak 20 kali. Digunakan untuk pengujian pewaktuan.

--debug

Menghasilkan sebuah pohon XML dari dokumen yang ditransformasikan, untuk keperluan *debugging*.

--novalid

Lewati memuat DTD dokumen.

--noout

Jangan mengeluarkan hasilnya.

--maxdepth value

Atur kedalaman maksimum dari tumpukan templat sebelum libxslt menyimpulkan bahwa ini adalah putaran tak terbatas. Nilai *default* adalah 500.

--html

Dokumen masukan berupa berkas HTML.

*--param name value*

Memberikan parameter nama *name* dan nilai *value* ke *stylesheet*. Anda bisa memberikan beberapa pasang nama/nilai hingga jumlah maksimum 32. Jika nilai yang diberikan berupa *string* dan bukan *node identifier*, gunakan *--stringparam*.

*--stringparam name value*

Memberikan parameter berupa nama *name* dan nilai *value* di mana *value* adalah *string* dan bukan *node identifier*. (Catatan: *string* harus utf-8.)

*--nonet*

Jangan menggunakan Internet untuk mengambil DTD, entitas atau dokumen.

*--path paths*

Menggunakan daftar (dipisahkan oleh spasi atau kolom) *path* sistem berkas untuk memuat DTD entitas atau dokumen.

*--load-trace*

Menampilkan ke stderr semua dokumen yang dimuat selama pemrosesan.

*--catalogs*

Gunakan katalog SGML yang disebutkan di SGML\_CATALOG\_FILES untuk menyelesaikan lokasi entitas eksternal. Secara *default*, xsltproc akan melihat katalog yang disebutkan di XML\_CATALOG\_FILES. Jika tidak disebutkan di situ, maka digunakan /etc/xml/catalog.

*--xinclude*

Memproses dokumen masukan menggunakan spesifikasi Xinclude. Lebih lanjut mengenai hal ini bisa ditemukan pada spesifikasi Xinclude: <http://www.w3.org/TR/xinclude/>

*--profile --norman*

Menghasilkan keluaran berupa informasi *profiling* yang menjelaskan jumlah waktu yang dihabiskan dalam setiap bagian *stylesheet*. Hal ini berguna untuk mengoptimasi performa *stylesheet*.

*--dumpextensions*

Mengeluarkan daftar semua ekstensi terdaftar ke stdout.

*--nowrite*

Jangan menulis ke berkas atau sumber apapun.

*--nomkdir*

Jangan membuat direktori.

*--writesubtree path*

Diperbolehkan menulis berkas hanya di dalam *path*.

*--nodtdattr*

Jangan menerapkan atribut *default* dari DTD dokumen.

### 15.6.4 Nilai Kembali Xsltproc

xsltproc akan mengembalikan sebuah nomor status yang akan berguna saat kita menjalankannya di dalam sebuah skrip.

0: normal

1: tidak ada argumen

2: terlalu banyak parameter

3: opsi tidak diketahui



- 4: gagal mengurai *stylesheet*
- 5: terdapat kesalahan pada *stylesheet*
- 6: terdapat kesalahan pada salah satu dokumen
- 7: metode `xsl:output` tidak didukung
- 8: parameter *string* berisi tanda petik dan tanda petik ganda
- 9: kesalahan pemrosesan internal
- 10: pemrosesan terhenti oleh suatu pesan penghentian
- 13: tidak dapat menulis hasil ke berkas keluaran

### 15.6.5 Informasi Lebih Lanjut Tentang `xsltproc`

Halaman web libxml: <http://www.xmlsoft.org/>

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

## Bab 16

# Simulator

Eeschema menyediakan simulator sirkuit elektrik dengan menggunakan **ngspice** sebagai mesin simulasi.

Ketika bekerja dengan simulator, Anda akan membutuhkan pustaka resmi *pspice*. Pustaka ini memiliki simbol-simbol umum yang digunakan untuk simulasi seperti sumber tegangan/arus atau transistor dengan pin yang penomorannya sesuai dengan spesifikasi urutan *node* ngspice.

Disediakan pula beberapa proyek demo untuk menunjukkan kemampuan simulator. Anda bisa membukanya di direktori *demo/simulation*.

### 16.1 Menerapkan Model

Sebelum melakukan simulasi, kita harus menentukan model Spice untuk masing-masing komponen.

Setiap komponen hanya bisa mendapatkan satu model, meskipun komponen tersebut terdiri dari beberapa bagian. Dalam hal ini, model Spice diterapkan pada bagian yang pertama.

Komponen-komponen pasif dengan referensi yang cocok dengan suatu tipe divais dalam notasi Spice ( $R^*$  untuk resistor,  $C^*$  untuk kapasitor,  $L^*$  untuk induktor) akan memiliki model yang diterapkan secara implisit dan menggunakan atribut nilai untuk menentukan propertinya.

---

#### Catatan

Ingat bahwa di Spice, penggunaan *M* berarti milli dan *Meg* berarti mega. Jika Anda lebih suka menggunakan *M* untuk mengindikasikan mega, Anda bisa melakukan pengaturan di [kotak dialog pengaturan simulasi](#).

---

Informasi model Spice disimpan sebagai teks pada atribut-atribut simbol. Anda bisa mendefinisikannya di *Symbol Editor* atau di *Schematics Editor*. Buka kotak dialog *Symbol Properties* dan klik pada tombol *Edit Spice Model* untuk membuka dialog *Spice Model Editor*.

Kotak dialog *Spice Model Editor* mempunyai tiga tab sesuai dengan tipe modelnya. Ada dua opsi umum untuk semua tipe model:

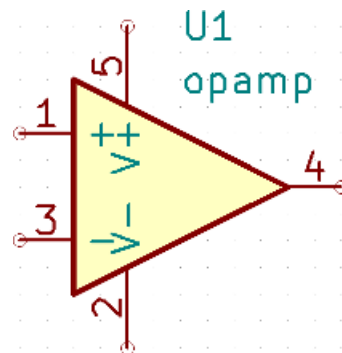
Disable symbol for simulation	Jika dicentang, komponen akan dikecualikan dari simulasi.
-------------------------------	---

## Alternate node sequence

Kita bisa mengesampingkan pin simbol untuk pemetaan *node* model. Untuk menetapkan pemetaan yang berbeda, tentukan nomor pin dalam urutan yang diinginkan oleh model.

*Contoh:*

```
* 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
```



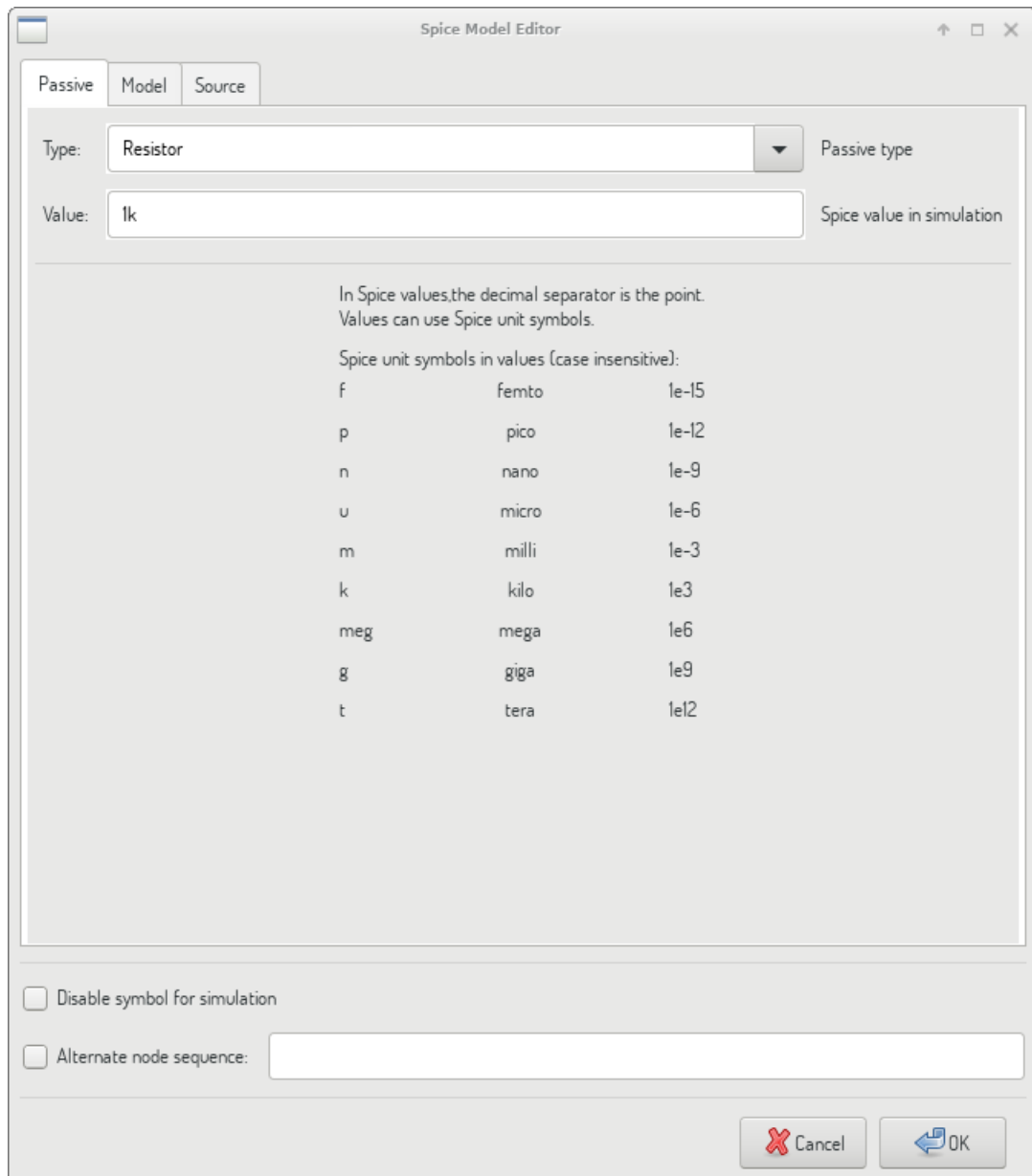
Untuk mencocokkan pin-pin simbol ke *node* model Spice di atas, kita perlu menggunakan opsi sekuens *node* alternatif dengan nilai: "1 3 5 2 4". Ini adalah daftar nomor pin sesuai dengan urutan model *node* Spice.

### 16.1.1 Tab *Passive*

Di tab *Passive*, kita bisa menerapkan sebuah model divais pasif (resistor, kapasitor, atau induktor) ke suatu komponen. Namun hal ini jarang digunakan, karena umumnya komponen pasif telah memiliki model yang diterapkan [secara implisit](#), kecuali jika referensi komponen tidak cocok dengan tipe divais aktual.

#### Catatan

Model divais pasif yang ditetapkan secara eksplisit memiliki prioritas lebih tinggi daripada model divais yang ditetapkan secara implisit. Artinya, setelah model divais pasif telah ditetapkan, atribut referensi dan nilai akan diabaikan selama simulasi. Hal ini mungkin akan membingungkan ketika nilai model yang ditetapkan tidak sesuai dengan nilai yang ditampilkan pada lembar kerja skematik.

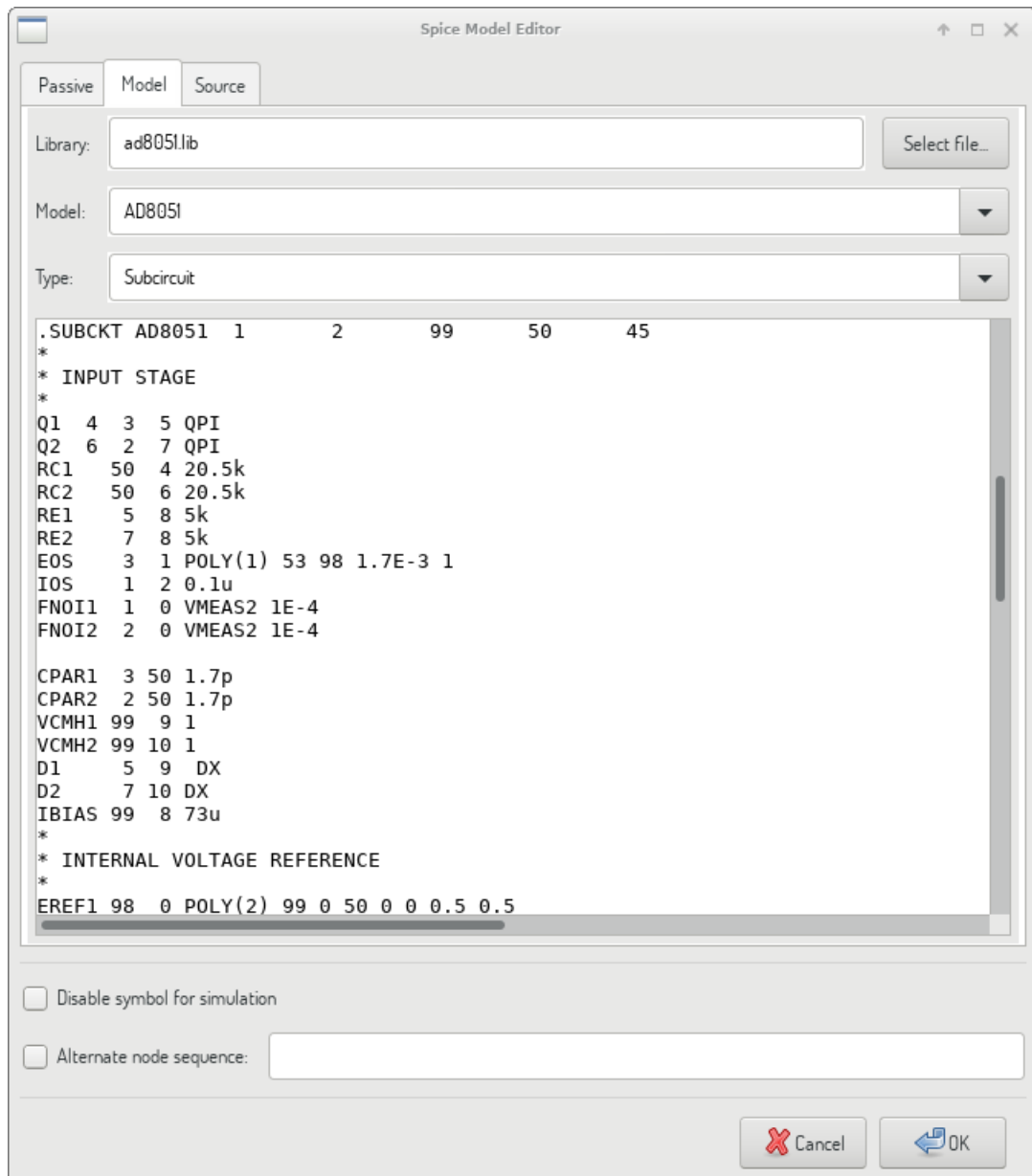


Type	Memilih tipe divais (resistor, kapasitor, atau induktor).
Value	Menentukan properti divais (resistansi, kapasitansi, atau induktansi). Nilainya bisa menggunakan prefiks satuan Spice (seperti ditampilkan di bawah kotak masukan teks) dan harus menggunakan titik sebagai pemisah desimal. Ingat bahwa Spice tidak bisa menginterpretasikan secara tepat prefiks yang disilangkan dengan nilai (misalnya 1k5).

### 16.1.2 Tab Model

Tab *Model* digunakan untuk menetapkan suatu semikonduktor atau model kompleks yang didefinisikan pada sebuah berkas pustaka eksternal. Pustaka model Spice terkadang disediakan oleh manufaktur.

Kotak teks utama menampilkan isi berkas pustaka yang dipilih. Sudah menjadi praktik yang umum untuk memasukkan penjelasan model ke dalam berkas pustaka, termasuk urutan *node*.

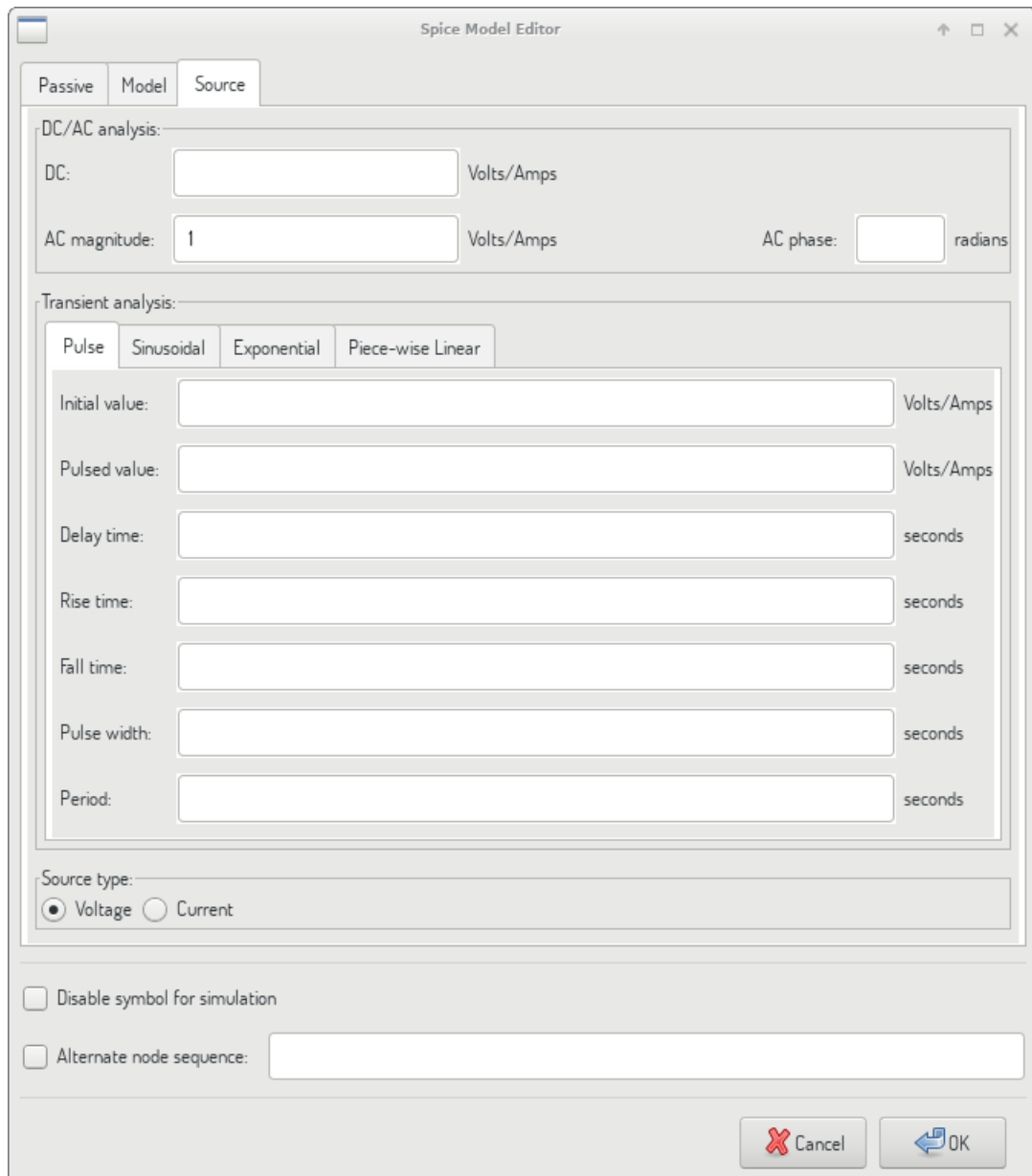


File	<i>Path</i> menuju ke berkas pustaka Spice. Berkas ini akan digunakan oleh simulator, dan ditambahkan dengan direktif <i>.include</i> .
Model	Model divais yang dipilih. Ketika ada sebuah berkas yang dipilih, daftar ini akan diisi dengan model yang tersedia untuk dipilih.
Type	Pilihan tipe model (subsirkuit, BJT, MOSFET atau dioda). Normalnya, pilihan ini ditentukan secara otomatis ketika sebuah model dipilih.

### 16.1.3 Tab Source

Tab *Source* digunakan untuk menetapkan model sumber sinyal atau *power*. Ada dua bagian: *DC/AC analysis* dan *Transient analysis*. Masing-masing mendefinisikan parameter sumber untuk tipe simulasi yang terkait.

Opsi *Source type* diaplikasikan untuk semua tipe simulasi.



The image shows the 'Spice Model Editor' dialog box with the 'Source' tab selected. It is divided into two main sections: 'DC/AC analysis' and 'Transient analysis'.

**DC/AC analysis:**

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

**Transient analysis:**

Sub-tabs: Pulse, Sinusoidal, Exponential, Piece-wise Linear (Pulse is selected).

- 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

**Options:**

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

Buttons: Cancel, OK

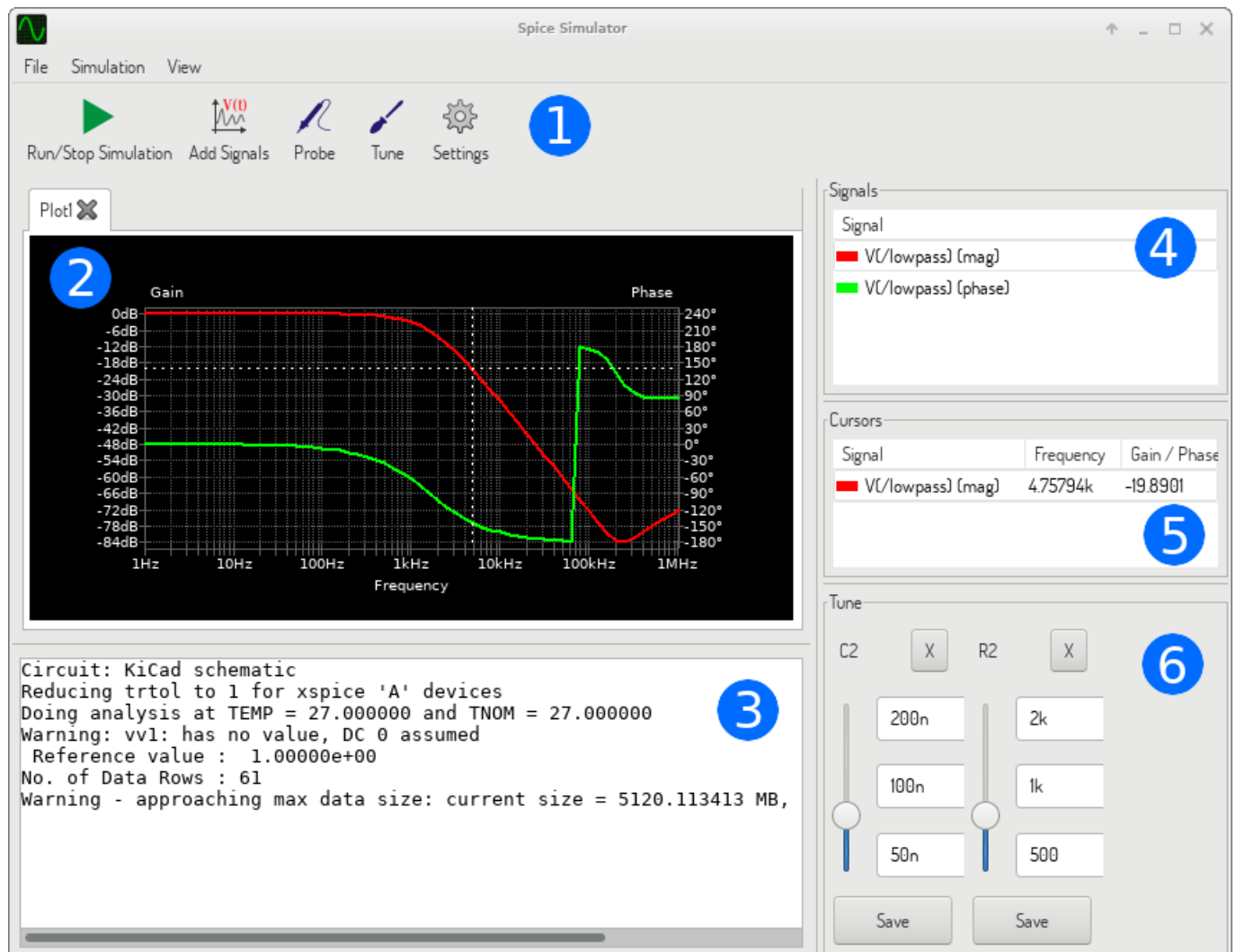
Silakan mengacu ke [dokumentasi ngspice](#), Bab 4 (*Voltage and Current Sources*) untuk informasi lebih lanjut.

## 16.2 Direktif Spice

Dimungkinkan untuk menambahkan direktif-direktif Spice dengan meletakkannya pada atribut teks di lembar kerja skematik. Pendekatan ini mudah dilakukan untuk mendefinisikan tipe simulasi *default*. Fungsi ini terbatas pada direktif Spice yang dimulai dengan tanda titik (misalnya ".tran 10n 1m"). Tidak dimungkinkan untuk meletakkan komponen tambahan menggunakan atribut teks.

## 16.3 Simulasi

Untuk menjalankan simulasi, buka kotak dialog *Spice Simulator* dengan memilih menu *Tools* → *Simulator* pada jendela editor skematik.



Kotak dialog ini dibagi menjadi beberapa bagian:

- Bilah alat
- Panel Plot
- Konsol keluaran
- Daftar sinyal
- Daftar kursor
- Panel Tune

### 16.3.1 Menu

#### 16.3.1.1 Menu *File*

New Plot	Membuat tab baru pada panel <i>plot</i> .
Open Workbook	Membuka daftar sinyal yang di-plot.
Save Workbook	Menyimpan daftar sinyal yang di-plot.
Save as image	Mengekspor <i>plot</i> yang aktif ke berkas <code>.png</code> .
Save as .csv file	Mengekspor data mentah titik-titik <i>plot</i> ke berkas <code>.csv</code> .
Exit Simulation	Menutup kotak dialog.

### 16.3.1.2 Simulasi

Run Simulation	Menjalankan simulasi menggunakan pengaturan saat ini.
Add signals. . .	Membuka kotak dialog untuk memilih sinyal yang akan diplot.
Probe from schematics	Menjalankan alat <a href="#">Probe</a> skematik.
Tune component value	Menjalankan alat <a href="#">Tuner</a> .
Show SPICE Netlist. . .	Membuka kotak dialog yang menampilkan Netlist yang telah dibuat untuk sirkuit yang disimulasikan.
Settings. . .	Membuka <a href="#">kotak dialog pengaturan simulasi</a> .

### 16.3.1.3 Menu View

Zoom In	Perbesar tampilan pada <i>plot</i> yang aktif.
Zoom Out	Perkecil tampilan pada <i>plot</i> yang aktif.
Fit on Screen	Mengatur tampilan untuk menampilkan semua <i>plot</i> .
Show grid	Tampilkan/sembunyikan grid.
Show legend	Tampilkan/sembunyikan legenda <i>plot</i> .

## 16.3.2 Bilah Alat



Bilah alat bagian atas digunakan untuk mengakses perintah-perintah yang sering dijalankan.

Run/Stop Simulation	Jalankan atau hentikan simulasi.
Add Signals	Membuka kotak dialog untuk memilih sinyal yang akan di-plot.
Probe	Menjalankan alat <a href="#">Probe</a> skematik.
Tune	Menjalankan alat <a href="#">Tuner</a> .
Settings	Membuka <a href="#">kotak dialog pengaturan simulasi</a> .

### 16.3.3 Panel Plot

Memvisualisasikan hasil simulasi dalam bentuk *plot*. Kita bisa membuka beberapa *plot* pada tab yang berbeda, namun ketika simulasi dijalankan, hanya tab yang aktif saja yang akan diperbarui. Dengan demikian dimungkinkan untuk membandingkan hasil simulasi dengan parameter yang berbeda.

*Plot* dapat diatur dengan menampilkan/menyembunyikan grid dan legenda menggunakan menu [View](#). Ketika legenda ditampilkan, kita bisa menggeser untuk mengubah posisinya.

Interaksi panel *plot*:

- Putar roda tetikus untuk memperbesar/memperkecil tampilan.
- Klik-kanan untuk membuka menu konteks untuk mengatur tampilan.
- Gambar sebuah kotak pemilih untuk memperbesar area yang dipilih.
- Geser kursor untuk mengubah koordinatnya.

### 16.3.4 Konsol Keluaran

Konsol keluaran menampilkan pesan-pesan dari simulator. Kita perlu memeriksa konsol untuk memastikan tidak ada pesan kesalahan dan peringatan.



### 16.3.5 Daftar Sinyal

Menampilkan daftar sinyal yang ditampilkan pada *plot* yang aktif.

Interaksi daftar sinyal:

- Klik-kanan untuk membuka menu konteks untuk menyembunyikan sinyal atau mengubah kursor.
- Klik-ganda untuk menyembunyikan sinyal.

### 16.3.6 Daftar Kursor

Menampilkan daftar kursor dan koordinatnya. Setiap sinyal bisa memiliki satu kursor yang ditampilkan. Konfigurasi visibilitas kursor diatur menggunakan daftar [Signals](#).

### 16.3.7 Panel Tune

Menampilkan komponen-komponen yang diambil menggunakan alat [Tuner](#). Panel ini digunakan untuk memodifikasi nilai komponen secara cepat dan mengamati pengaruhnya pada hasil simulasi - setiap kali nilai sebuah komponen diubah, simulasi akan dijalankan ulang dan *plot* akan diperbarui.

Untuk setiap komponen, ada beberapa kendali yang terkait:

- Atribut teks *top* mengatur nilai maksimum komponen.
- Atribut teks *middle* mengatur nilai aktual komponen.
- Atribut teks *bottom* mengatur nilai minimum komponen.
- *Slider* digunakan untuk mengubah nilai komponen secara halus.
- Tombol *Save* memodifikasi nilai komponen pada skematik ke nilai yang dipilih dengan menggunakan *slider*.
- Tombol *X* menghapus komponen dari panel *Tune* dan mengembalikan ke nilai aslinya.

Ketiga atribut teks di atas mengenali prefiks satuan Spice.

### 16.3.8 Alat Tuner

Alat *Tuner* digunakan untuk memilih komponen untuk keperluan *tuning*.

Untuk memilih komponen untuk *tuning*, klik pada satu komponen di editor skematik ketika alat ini aktif. Komponen yang dipilih akan muncul di panel [Tune](#). Kita hanya bisa melakukan *tuning* komponen pasif.

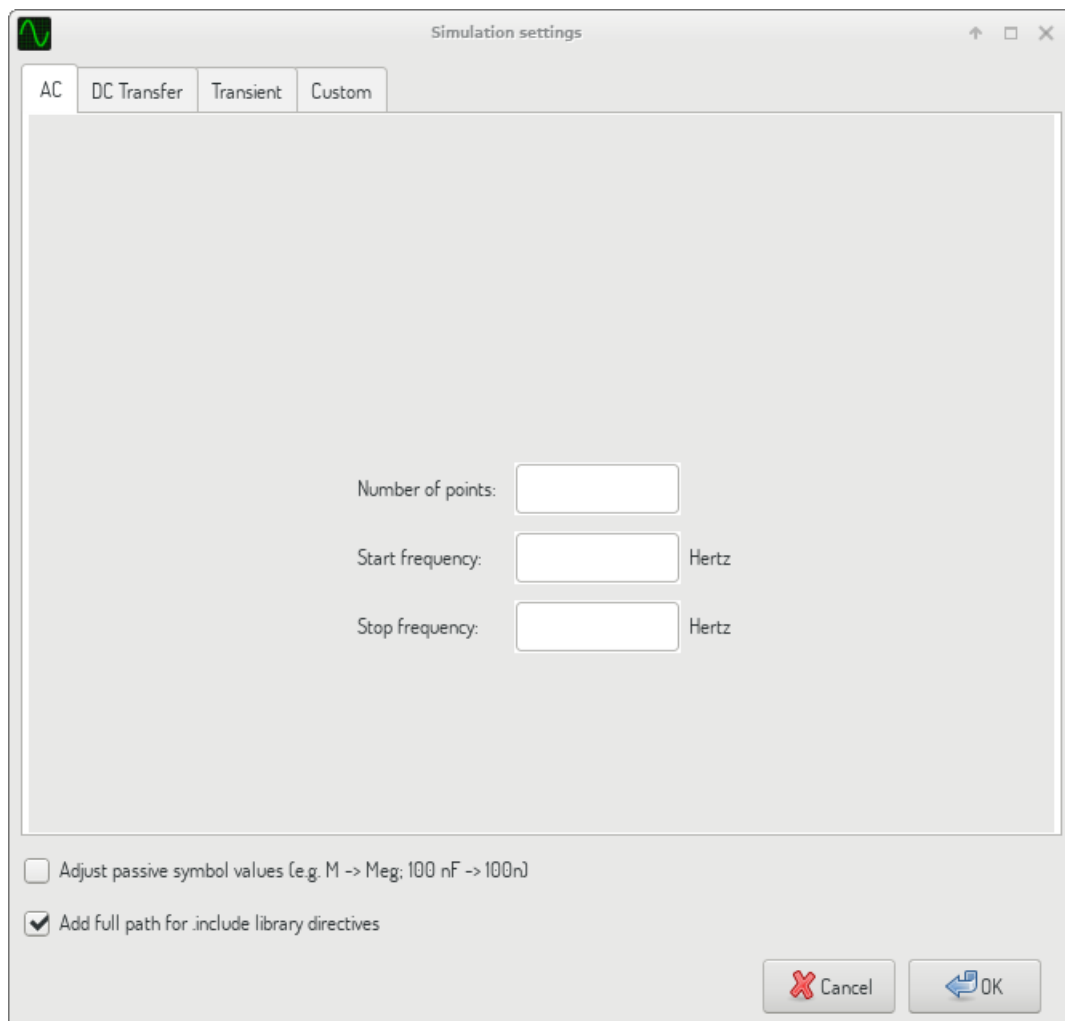
### 16.3.9 Alat Probe

Alat *Probe* digunakan untuk memilih sinyal untuk dilakukan *plotting*.

Untuk menambahkan sinyal yang akan diplot, klik pada *wire* yang diinginkan di editor skematik ketika alat ini aktif.

---

### 16.3.10 Pengaturan Simulasi



Kotak dialog pengaturan simulasi digunakan untuk mengatur tipe dan parameter-parameter simulasi. Ada empat tab pada kotak dialog ini:

- Tab *AC*
- Tab *DC Transfer*
- Tab *Transient*
- Tab *Custom*

Tiga tab pertama digunakan untuk memasukkan parameter-parameter simulasi. Tab yang terakhir digunakan untuk menuliskan direktif kustom Spice untuk membuat sebuah simulasi. Untuk informasi lebih lanjut mengenai tipe dan parameter-parameter simulasi, lihat [dokumentasi ngspice](#), Bab 1.2.

Cara lain untuk mengkonfigurasi sebuah simulasi adalah dengan menuliskan [direktif Spice](#) ke atribut teks pada skematik. Setiap direktif atribut teks yang berhubungan dengan tipe simulasi akan ditimpa dengan pengaturan yang dipilih pada kotak dialog. Hal ini berarti bahwa setelah Anda menggunakan kotak dialog simulasi, kotak dialog akan menerima direktif-direktif skematik hingga simulator dibuka kembali.

Ada dua opsi yang umum untuk semua tipe simulasi:

---

Adjust passive symbol values	Mengganti nilai simbol pasif untuk mengkonversi notasi nilai komponen umum ke notasi Spice.
Add full path for .include library directives	Menambahkan nama berkas pustaka model Spice dengan <i>path</i> lengkap. Normalnya, ngspice memerlukan <i>path</i> lengkap untuk mengakses sebuah berkas pustaka.