EAGLE

EASILY APPLICABLE GRAPHICAL LAYOUT EDITOR

Manual

Version 5

7th Edition

005175100

Copyright © 2010 CadSoft Computer *All Rights Reserved*

How to reach us

Phone:	+1 954-237-0932
Fax:	+1 954-237-0968
E-mail:	support@cadsoftusa.com
Web:	http://www.cadsoftusa.com

Office Hours are:

Mon – Thu:	9 am to 5 pm EST
Fri:	9 am to 4 pm EST

CadSoft Computer
19620 Pines Blvd. Suite 217
Pembroke Pines, Florida 33029
U.S.A.

This software and documentation are copyrighted by CadSoft Computer, doing business under the trade name EAGLE. The software and documentation are licensed, not sold, and may be used or copied only in accordance with the EAGLE License Agreement accompanying the software and/or reprinted in this document. This software embodies valuable trade secrets proprietary to CadSoft Computer.

Specifications subject to change without notice.

© Copyright 1988-2010 CadSoft Computer. All rights reserved worldwide.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, scanning, digitizing, or otherwise, without the prior consent of CadSoft Computer.

Printing this manual for your personal use is allowed.

CadSoft Computer is a trading division of Newark Corporation.

Windows is a registered trademark of Microsoft Corporation.
Linux is a registered trademark of Linus Torvalds.
Mac is a registered trademark of Apple Computer, Inc.

Table of Contents

Chapter 1
Introduction.....19
1.1 What is in this Manual?.....19
1.2 General Comments About EAGLE Component Libraries.....20
1.3 Technical Terms.....20

Chapter 2
Installation.....23
2.1 What You Need.....23
2.2 New Installations.....23
Windows.....24
Linux.....24
Mac OS X.....25
2.3 Updating an Older Version.....26
Notes on Library Files.....26
In Case of Changes in the File Data Structure.....27
2.4 Changing or Extending the License.....27
2.5 Multiple User Licenses.....28
Notes for Installing in a Network.....28
Special Instructions Under Windows.....29
Path Information.....29
Different Operating Systems at the Working Computers.....29
2.6 Language Settings.....29
Windows.....30
Linux and Mac OS-X.....30

Chapter 3
EAGLE Modules and Editions.....31
3.1 EAGLE Modules.....31
The Layout Editor.....31
Schematic Editor.....31
Autorouter.....32
3.2 Different Editions.....32
Professional Edition.....32
General.....32
Layout Editor.....33
Schematic Editor.....33
Autorouter Module.....34

Standard Edition....34
Freemium Edition....34
Light Edition....34

Chapter 4
A First Look at EAGLE....37

4.1 The Control Panel....37
Library Summary....38
Design Rules....40
User Language Programs, Scripts, CAM Jobs....40
Projects....40
Menu Bar....41
File Menu....41
View Menu....42
Options Menu....43
Window Menu....46
Help Menu....46
4.2 The Schematic Editor Window....47
How You Obtain Detailed Information About a Command....48
Bubble Help And Tool Tips....48
Help Function....49
Command Parameters....49
GRID....50
The Action Toolbar....50
USE....50
SCRIPT....51
RUN....51
WINDOW....51
UNDO/REDO....51
Stop Icon....51
Go Icon....51
The Command Toolbar of The Schematic Editor....51
INFO....51
SHOW....52
DISPLAY....52
MARK....52
MOVE....52
COPY....53
MIRROR....53
ROTATE....53
GROUP....53
CHANGE....54

CUT....54
PASTE....54
DELETE....54
ADD....55
PINSWAP....55
GATESWAP....55
REPLACE....55
NAME....55
VALUE....55
SMASH....55
MITER....56
SPLIT....56
INVOKE....56
WIRE....56
TEXT....57
CIRCLE....57
ARC....57
RECT....57
POLYGON....57
BUS....57
NET....57
JUNCTION....57
LABEL....58
ATTRIBUTE....58
ERC....58
Commands Not Available in the Command Toolbar....59
ASSIGN....59
CLASS....59
CLOSE....59
EDIT....59
FRAME....59
EXPORT....59
LAYER....59
MENU....60
OPEN....60
PACKAGE....60
PRINT....60
QUIT....60
REMOVE....60
SET....60
TECHNOLOGY....61
UPDATE....61

WRITE....61
Mouse Keys....61
Selecting Neighbouring Objects....61
4.3 The Layout Editor Window....62
The Commands on the Layout Command Toolbar....63
INFO....63
SHOW....63
DISPLAY....63
MARK....63
MOVE....63
COPY....64
MIRROR....64
ROTATE....64
GROUP....64
CHANGE....65
CUT....65
PASTE....65
DELETE....65
ADD....66
PINSWAP....66
REPLACE....66
LOCK....66
NAME....66
VALUE....66
SMASH....67
MITER....67
SPLIT....67
OPTIMIZE....67
ROUTE....67
RIPUP....68
WIRE....68
TEXT....68
CIRCLE....69
ARC....69
RECT....69
POLYGON....69
VIA....70
SIGNAL....70
HOLE....70
ATTRIBUTE....70
RATSNEST....70
AUTO....70

ERC....71
DRC....71
ERRORS....71
4.4 The Library Editor Window....71
Load or Rename Package, Symbol, or Device....72
EDIT....72
REMOVE....72
RENAME....73
The Package Editing Mode....73
Design New Package....73
PAD....73
SMD....73
The Symbol Editing Mode....74
Design a New Symbol....74
PIN....74
The Device Editing mode....74
Create Actual Components from Symbols and Packages....75
ADD....76
NAME....76
CHANGE....76
PACKAGE....76
CONNECT....76
PREFIX....76
VALUE....76
TECHNOLOGY....77
ATTRIBUTE....77
DESCRIPTION....77
4.5 The CAM Processor....77
Generate Data....78
Starting the CAM Processor....78
Load Job File....79
Load Board....79
Set Output Parameters....79
Start Output....79
Define New Job....79
4.6 The Text Editor Window....79

Chapter 5
Principles for Working with EAGLE....81
5.1 Command Input Possibilities....81

Activate Command and Select Object....81
Command Line....81
History Function....82
The Context Menu....82
Function Keys....83
Script Files....85
Mixed Input....85
5.2 The EAGLE Command Language....85
Typographical Conventions....86
Enter key and Semicolon....86
Bold Type or Upper Case....86
Lower Case....86
Underscore....86
Spaces....87
Alternative Parameters....87
Repetition Points....87
Mouse Click....87
Entering Coordinates as Text....88
Relative values:....88
Polar values:....88
Right Mouse Click:....89
Modifier:....89
5.3 Grids and the Current Units....90
5.4 Aliases for DISPLAY, GRID, and WINDOW....91
Example: DISPLAY Alias....92
Example: GRID Alias....92
Example: WINDOW Alias....92
Editing, Renaming, Deleting of an Alias....93
5.5 Names and Automatic Naming....93
Length....93
Forbidden and Special Characters....94
Automatic Naming....94
5.6 Import and Export of Data....94
Script Files and Data Import....95
File Export Using the EXPORT Command....95
DIRECTORY....95
NETLIST....95
NETSCRIPT....96
PARTLIST....96
PINLIST....96
SCRIPT....96

IMAGE....96
5.7 The EAGLE User Language....97
5.8 Forward&Back Annotation....98
5.9 Configuring EAGLE Individually....99
Configuration Commands....99
The Menu Options/Set (SET Command)....100
Display Certain Layers Only....100
Contents of The Parameter Menus....100
Color Settings....100
Miscellaneous SET Options....103
The eagle.scr File....105
The eaglerc File....106
EAGLE Project File....107

Chapter 6
From Schematic to Finished Board....109

6.1 Creating the Schematic Diagram....109
Open the Schematic Diagram....110
Set the Grid....110
Place Symbols....110
Load Drawing Frame....110
Place Circuit Symbols (Gates)....112
Hidden Supply Gates....112
Devices with Several Gates....113
Designlink.ulp – Access to Farnell's Online Product Database..114
Wiring the Schematic Diagram....114
Draw Nets (NET)....114
Defining Cross-References for Nets....115
Cross-References for Contacts....116
Specifying Net Classes....117
Drawing a bus (BUS)....118
Pinswap and Gateswap....119
Power Supply....120
Define Attributes....121
Global Attributes....121
Attributes for Elements....122
ERC – Check and Correct Schematic....124
Organize Schematic Sheets....126
Points to Note for the Schematic Editor....127
Superimposed Pins....127
Open Pins when MOVEing....127
Merge Different Schematics....127

6.2 Considerations Prior to Creating a Board....127
Checking the Component Libraries....127
Agreement with the Board Manufacturer....128
Specifying the Design Rules....128
General Principles....129
Layers....130
Minimum Clearance and Distance....131
Sizes....132
Restring (Pad and Via Diameter)....132
Shapes....134
Supply....136
Masks....137
Misc....138
6.3 Create Board....139
Without the Schematic Editor....140
Specify the Board Outline....140
Arrange Components....142
Attributes for Components and Global Attributes....144
Boards with Components on Both Sides....144
Exchanging Packages....144
PACKAGE Command....145
REPLACE command....146
Changing the Technology....146
Define Forbidden Areas....146
Routing – Placing Tracks Manually....147
Defining a Copper Plane with POLYGON....149
6.4 DRC – Checking the Layout and Correcting Errors....152
The DRC Errors Window....154
Error Messages and their Meaning....155
6.5 Multilayer Boards....158
Inner Layer....159
Signal Layers....159
Supply Layers with Polygons and More than One Signal....159
Power Supply Layer with One Signal....159
Supply Layers and Autorouter....161
Multilayer Boards with Through Vias....161
Layer Setup....161
Multilayer with Blind and Buried Vias....162
Disambiguation....162
Displaying Vias....163
Layer Setup....163

Hints For Working With Blind, Buried, and Micro Vias.....168
Micro Via – A Special Case of Blind Via.....169
6.6 Updating Components (Library Update).....170
6.7 Print Out Schematic and Layout.....171
Settings of the Print Dialog.....171
6.8 Combining Small Circuit Boards on a Common Panel.....174
6.9 Consistency Lost between Schematic and Layout 175

Chapter 7
The Autorouter.....177
7.1 Basic Features.....177
7.2 What Can be Expected from the Autorouter.....177
7.3 Controlling the Autorouter.....178
Bus Router.....178
Routing Pass.....179
Optimization.....179
7.4 What Has to be Defined Before Autorouting.....179
Design Rules.....179
Track Width and Net Classes.....179
Grid.....180
Placement Grid.....180
Routing Grid.....180
Memory Requirement.....181
Layer.....182
Preferred Directions.....182
Restricted Areas for the Autorouter.....183
Cost Factors and Other Control Parameters.....183
7.5 The Autorouter Menu.....183
7.6 How the Cost Factors Influence the Routing Process.....186
Layer Costs.....187
cfBase.xx: 0..20.....187
Costs.....187
cfVia: 0..99.....187
cfNonPref: 0..10.....187
cfChangeDir: 0..25.....188
cfOrthStep, cfDiagStep.....188
cfExtdStep: 0..30.....188
cfBonusStep, cfMalusStep: 1..3.....188
cfPadImpact, cfSmdImpact: 0..10.....188
cfBusImpact: 0..10.....189
cfHugging: 0..5.....189

cfAvoid 0..10....189
cfPolygon 0..30....189
Maximum....189
mnVia 0..30....189
mnSegments 0..9999....189
mnExtdSteps 0..9999....189
7.7 Number of Ripup/Retry Attempts....190
7.8 Routing Multi-Layer Boards....190
Supply Layers....190
Polygons as Supply Layers....191
7.9 Backup and Interruption of Routing....191
7.10 Information for the User....192
Status Display....192
Log file....193
7.11 Parameters of a Control File....193
7.12 Practical Hints....195
General....195
Single-Sided Boards....195
SMD Boards With Supply Layers....195
What can be done if not all signals are routed?....196
7.13 The Follow-me Router....196
Partial and Full Mode....196
Configuration....197
Routing Parameters....198
Notes....199

Chapter 8
Component Design Explained through Examples....201

8.1 Definition of a Simple Resistor....201
Resistor Package....202
Define a New Package....202
Set the Grid....202
Solder Pads....202
Pad Name....203
Silkscreen and Documentation Print....203
Labeling....204
Restricted area for components....205
Description....205
Note....205
Resistor Symbol....205
Define a New Symbol....205

Set the Grid....206
Place the Pins....206
Pin Names....208
Schematic Symbol....208
Resistor Device....209
Define a New Device....209
Selecting, Naming and Configuring Symbols....209
Selecting the Package....210
Connections Between Pins and Pads....210
Define Prefix....211
Value....211
Description....212
Save....212
Library Description....213
Use Library....213
8.2 Defining a Complex Device....213
Creating a New Library....215
Drawing the Pin-Leaded Package....215
Set the Grid....216
Place Pads....216
Pad Name....217
Draw the Silk Screen Symbol....217
Package Name and Package Value....217
Areas Forbidden to Components....218
Description....218
Save....219
Defining the SMD Package....219
Set the Grid....220
Placing SMD Solder Pads....221
SMD Names....222
Draw the Silk Screen....223
Package Name and Package Value....223
Area Forbidden to Components....224
Locating Point (Origin)....224
Description....224
Save....224
Defining the Logic Symbol for the Schematic Diagram....225
Check the Grid....226
Place the Pins....226
Pin Name....227
Draw the Symbol....227
Placeholders for NAME and VALUE....227

Save....227
Defining a Power Supply Symbol....228
Check the Grid....228
Place the Pins....228
Pin Name....229
Placeholders for NAME and VALUE....229
Associating the Packages and Symbols to Form a Device Set....229
Select Symbols....230
Naming the Gates....230
Specify Addlevel and Swaplevel....231
Choosing the Package Variants....231
The Connect Command....232
Defining Technologies....233
Specifying the Prefix....234
Value....234
Description....235
Save....235
8.3 Supply Voltages....236
Component Power Supply Pins....236
Invisible Supply Pins....236
Pins with the Same Names....237
8.4 Supply Symbols....238
8.5 Attributes....240
Define Attributes....240
Display Attributes....242
Placeholders in Symbol and Package....243
8.6 Labeling of Schematic Symbols....243
8.7 More about the Addlevel Parameter....244
Summary....244
Relay: Coil and First Contact must be Placed....245
Connector: Some Connection Pins can be Omitted....245
Connector with Fixing Hole and Restricted Area....246
8.8 Defining Components with Contact Cross-References....247
Define Symbol....247
Define Device....248
Define Package....248
8.9 Drawing Frames....249
8.10 Components on the Solder Side....250
8.11 Components with Oblong Holes....251
8.12 Creating New Package Variants....251

Package from Another Library....251
Defining the Package Variant....252
Connect Command....253
Defining Technologies....253
Save....254
Using a Modified Package from Another Library....254
Copying the Package....254
Defining the Variant....255
8.13 Defining Packages in Any Rotation....255
Rotating a Package as a Whole....255
Packages with Radial Pad Arrangement....256
8.14 Library and Part Management....256
Copying of Library Elements....256
Within a Library....256
From One Library into Another....257
Composition of Your own Libraries....259
Removing and Renaming Library Elements....260
Update Packages in Libraries....260

Chapter 9
Preparing Manufacturing Data....263
9.1 Which Data do we Need for Board Manufacture?....263
Gerber Plot Data....264
GERBER_RS274X....264
GERBERAUTO and GERBER....264
Drill Data....265
EXCELLON....265
EXCELLON_RACK....265
SM1000 and SM3000....265
Further Drill Data Devices....266
Prototype Manufacture With a Milling Machine....266
outlines.ulp....266
mill-outlines.ulp....266
Film Generation Using PostScript Files....266
Printing on a Film....267
Data for Pick-and-place Machines and In-circuit Testers....267
Documentation....268
Parts List....268
Drill Plan....269
Drill Legend....270
9.2 Rules that Save Time and Money....271
9.3 Quick Guide for Data Output....272

Job gerb274x.cam....272
Job excellon.cam....273
Job gerber.cam....274
9.4 Which Files do I Need for my Board?....275
Files List....276
Placeholders for Output File Name Generation....278
Hints Concerning File Extensions:....279
9.5 Peculiarities of Multilayer Boards....279
Simple Inner Layers....279
Supply Layer....279
Drill Data for Multilayer Boards With Blind and Buried Vias....280
9.6 Set Output Parameters....280
9.7 Automating the Output with CAM Processor Jobs....284
Defining a New CAM Job....284
Extending gerber.cam Job for Multilayer Boards....285
Error Message: Apertures Missing....286
Gerber Info Files....286
Drill Data Generation with Separate Rack File....287
Define a Drill Configuration (Rack) File....287
Define Job for Drill Data Output....287
Drill Info File....288
9.8 Device Driver Definition in eagle.def....288
Creating Your Own Device Driver....288
Example 1: Gerber(auto) device, Millimetre....288
Example 2: EXCELLON Device, Output with Leading Zeros....289
Units in the Aperture and Drill Table....289
9.9 Gerber Files for Photoplotters with Fixed Aperture Wheels....290
Defining the Aperture Table (Wheel)....290
Aperture Emulation....292

Chapter 10
Appendix....293
10.1 Layers and their Usage....293
In Layout and Package Editor....293
In Schematic, Symbol, and Device Editor....294
10.2 EAGLE Files....294
10.3 EAGLE Options at a Glance....295
10.4 Configuration of the Text Menu....299
10.5 Text Variables....300
10.6 Options for Experts in eaglerc....300
Change Component Value Warning 300

Consistency Check....300
Delete Wire Joints....301
Device Name as Value for all Components....301
Disable Ctrl for Radius Mode....301
Group Selection....301
Open Project....301
Panning Drawing Window....301
Polygon Edges as Continuous Lines....302
Reposition of the Mouse Cursor....302
Units in Dialogs....302
10.7 Error Message....303
When Loading a File....303
Restring smaller than in older version....303
Library objects with the same names....303
Pad, Via Replaced with a Hole....304
Skipped unsuitable objects....305
Can't Update File....305
In a Library....306
Package/Symbol is in use....306
In the CAM Processor....307
Polygon may cause extremely large plot data....307
In the Light, Freemium or Standard Edition....307
Can't perform the requested action....307
10.8 EAGLE License Agreement....307

This
page
has been
left free
intentionally.

Chapter 1
Introduction

This manual describes the use of the EAGLE software and its basic principles. The order of chapters follows the typical process from drawing a schematic to a ready-to-use layout.

1.1 What is in this Manual?

A chapter's main heading is intended to tell you briefly what the contents of that chapter are. Here in the first chapter we want to give a quick overview what you can expect from this manual.

Chapter 1 – Introduction
Contains a preview of the manual.

Chapter 2 – Installation
Deals with the program's installation.

Chapter 3 – EAGLE Modules and Editions
Explains the various program variants.

Chapter 4 – A First Look at EAGLE
Gives a preview of the program's structure and describes the editor windows and their commands.

Chapter 5 – Principles for Working with EAGLE
Examines the basic ways of using and configuring EAGLE.

Chapter 6 – From Schematic to Finished Layout
Follows the route from schematic to layout.

Chapter 7 – The Autorouter
Dedicated to the Autorouter module and its configuration.

Chapter 8 – Component Design Explained through Examples
Explains the definition of library components through examples and informs about library and component management.

Chapter 9 – Preparing the Manufacturing Data
Everything you need to know about generating manufacturing data.

Appendix
Lists useful additional information and explains some error messages EAGLE prompts in certain situations.

For a quick, hands-on introduction, refer to the EAGLE Tutorial. Please read the tutorial for a better understanding before working with the manual.

Anybody who has already been working with a prior version of EAGLE is advised to read the file ***UPDATE*** under Linux and Mac or ***UPDATE.TXT*** under Windows. It contains a description of all the differences from earlier versions. This file is located in the *eagle/doc* directory. Please read it before you start working with the new EAGLE.

Information that was not available or that has been changed since finishing this manual is also described in ***UPDATE*** or ***UPDATE.TXT***, or, if existing, in a *README* file.

Detailed information, especially about the EAGLE command language and the EAGLE User Language, is available on the help pages. You can reach a basic understanding very quickly by using this manual, and you can use the convenient search features of the help function to quickly locate the answers to particular questions.

1.2 General Comments About EAGLE Component Libraries

The component libraries supplied with EAGLE have been compiled with great care as an additional service to you, our customer. However, the large number of available components and suppliers of these components means that the occasional discrepancy is unavoidable. Please note, therefore, that CadSoft takes no responsibility for the complete accuracy of information included in library files.

Please note that libraries are not necessarily identical to former libraries with the same name. Therefore, it is advisable to back up your old libraries before installing the new ones.

Additional new libraries, that have not yet been officially released, can be found on CadSoft's internet site in the download section of www.cadsoftusa.com.

1.3 Technical Terms

In this manual, in the help function, and in EAGLE itself we frequently use some technical terms that should be explained here in a few words.

Airwire:
Unrouted connection on a board, displayed in the unrouted layer (= rubber band).

Annulus Symbol:
An isolation ring that will be drawn in a supply layer or in a copper-filled area.

Blind Via:
A plated-through hole for changing the layer of a track which has not been drilled through all layers in the production process of a multilayer board.

Buried Via:
A plated-trough hole, which has been drilled through the current layer stack in the production process like a normal (through) via, but does not connect all layers of the whole board.

Core:
Two copper layers applied to a solid substrate.

Design Rule Check (DRC):
EAGLE can identify the violation of certain Design Rules (e.g. if two different tracks overlap or are too close) with the DRC.

Device:
A fully defined element in a library. Consists of at least one Package and one Symbol.

Device Set:
Consists of Devices that use the same Symbols for the Schematic but have different Package variants or technologies.

Drill:
Plated-through drilling in the layout (in pads and vias)

Electrical Rule Check (ERC):
EAGLE can identify the violation of certain electrical rules (e.g. if two outputs are connected) with the ERC. It also checks the consistency of the schematic and the layout.

Follow-me Router:
The manual ROUTE command offers an operating mode that calculates and displays the connection of a selected signal automatically. The current position of the mouse cursor determines the trace of the connection. Only available with the Autorouter module.

Forward&Back Annotation:
Transforms all the actions one makes in a schematic online into the layout (and with limitations from layout into schematic). Both files are consistent all the time.

Gate:
The term *Gate* is used in this manual for a part of a component which can be individually placed on a schematic. This can be one Gate of a TTL component, one contact pair in a relay, or an individual resistor from a resistor array.

Hole:
Non plated-through drilling in the layout (e.g. a mounting hole).

Layer Stack:
Current number and order of copper and isolation layers which are used to build up a printed circuit board.

Micro via:
A plated-through hole (like Blind via) with a relatively small drill diameter which connects an outer layer with the next reachable inner layer.

Net:
Electrical connection in a schematic.

Package:
Component footprint stored in a library.

Pad:
Through-hole pad associated with a Package.

Pin:
Connection point on a Schematic Symbol.

Prepreg:
Used in a compound of inner and outer layers for multilayer boards.

Rack:
Configuration table for a drilling machine. Needed for generating drill data.

Ratsnest:
Command for calculating the shortest airwires and for hiding or displaying certain airwires for a better overview.

Restring:
Pronunciation: *rest-ring.* Setting that determines the width of the copper ring around a plated-through hole of a pad or via.

Signal:
Electrical connection in a board.

Supply Symbol:
Represents a supply signal in the schematic. Causes the ERC to run special checks.

Symbol:
Schematic representation of a component, stored in a Library.

User Language:
Freely programmable, C-like language for data import and export.

Via:
Plated-through hole for changing the layer of a track. See also Micro via, Blind via, and Buried via.

Wheel:
Aperture configuration file. Generated with Gerber data for board manufacturing.

Wire:
Electrical connection in a board, or a line (since lines are drawn with the WIRE command).

Chapter 2
Installation

2.1 What You Need

For installing EAGLE you need to have an EAGLE installation file, your personal installation code, and the appropriate license file.

If you have asked for an upgrade for an existing current installation or for an extension of your license, you don't have to install EAGLE anew. Your new installationcode and its appropriate license file will update your existing license.

If you intend to install the EAGLE Freeware, the EAGLE installation file is all you need.

> *Keep the your personal installation code in a safe place, inaccessible to unauthorized persons! Never give the license file or installation code to others! Never send your installation code by e-mail to CadSoft or to anyone else.*

2.2 New Installations

On the CadSoft website you will always find the newest installation files. Keep ready your personal license data, consisting of your license file *license.key* and the installation code.
The installation routine determines depending on the operating systems' language used whether EAGLE will be installed in English or in German language.
Any recent information about installation may be found in the *README* files.

Windows

EAGLE for Windows is available as a self-extracting archive, which is named, for example, *eagle-win-5.7.0.exe*. Double-click this file and the *WinZip Self-Extractor window* will appear. The installation routine starts with extracting the files from the archive. Click the *Setup* button to begin.
The version number in the file name may, depending on the current version, differ.

Now the actual installation starts. Follow the instructions step by step.

At the end of this process you will be asked how you want to license EAGLE:

Use License file
This implies that you have already bought an EAGLE license. In this case the following dialog asks you for the path to your license file *license.key* and for your personal installation code which you have got from CadSoft.

Use Freemium Code
If you registered at *http://www.element-14.com/eagle-freemium* and got a Freemium code, use this option.

Run as Freeware
EAGLE will be licensed with the limits of the Light Edition. It's possible to upgrade to a purchased license without reinstallation at any time.

Don't license now
postpones the licensing on the first EAGLE start.

The Windows Start menu now contains in the *Programs* section the new entry *EAGLE Layout Editor 5.7.0*. Tutorial and manual in pdf file format are also part of the installation.

Uninstalling EAGLE can be done by usual means of your operating system using Windows uninstall tools.

Linux

EAGLE for Linux is available as self-extracting shell script with set-up dialog. You may install the program as *Root/Administrator* or as a normal *User*.

Initiate the installation process in a file manager with a double-click on the shell script file *eagle-lin-5.7.0.run.*

Please check, whether the file is marked as *executable* before. If this is not the case, you could, for example, use the *chmod* command in the console to change its attributes.

You can start the shell script form the console, as well, by typing:

```
/path_to_file/eagle-lin-5.7.0.run
```

The version number in the file name may differ, depending on the currently available EAGLE version.

The installation routine starts with extracting the files from the archive. Then the actual installation process starts. Follow the instructions step by step.

At the end you will be asked how you want to license EAGLE:

Use License file
This implies that you have already bought an EAGLE license. In this case the following dialog asks you for the path to your license file *license.key* and for your personal installation code which you have got from CadSoft.

Use Freemium Code
If you registered at *http://www.element-14.com/eagle-freemium* and got a Freemium code, use this option.

Run as Freeware
EAGLE will be licensed with the limits of the Light Edition. It's possible to upgrade to a purchased license without reinstallation at any time.

Don't license now
postpones the licensing on the first EAGLE start.

To start EAGLE, execute *bin/eagle* in the installation directory.

Tutorial and manual in pdf file format are also part of the installation. They can be found in the *doc* directory of the EAGLE installation.

Delete the entire EAGLE directory, if you want to uninstall EAGLE.

Mac OS X

EAGLE for Mac OS X is available as Universal Binary and works on Power-PC and on Intel based computers. In order to start the installation process double-click onto the icon that represents the EAGLE archive which is named, for example, *eagle-mac-5.7.0.zip*. The archive will be extracted automatically. A new folder icon named *eagle-mac-5.7.0.pkg* will appear on the Desktop now. Click it with the mouse and the installation process starts.
The version number in the file name may differ, depending on the current EAGLE version.

Follow the instructions step by step.

Confirm the EAGLE Software License Agreement by clicking *Continue*, if you agree with it. If not, you can abort the installation by clicking the *Go back* button.
The next step sets up the EAGLE destination folder and possibly a destination volume. By clicking *Continue* you accept the suggested directory. Click the *Choose* button and you may select any other location for the EAGLE folder. After confirming the installation path the files are copied into the EAGLE folder.

After finishing the installation you are able to start EAGLE by clicking the *EAGLE* icon in the *Finder's Applications/EAGLE* entry.

While starting EAGLE for the first time, you will be asked how to license EAGLE:

Use License file
This implies that you have already bought an EAGLE license. In this case the following dialog asks for the path to your personal license file *license.key* and for your personal installation code you have got from CadSoft.

Use Freemium Code
If you registered at *http://www.element-14.com/eagle-freemium* and got a Freemium code, use this option.

Run as Freeware
EAGLE will be licensed with the limits of the Light Edition. It's possible to upgrade to a purchased license without reinstallation at any time.

Tutorial and manual in pdf file format are part of the installation and are located in the *doc* folder of the EAGLE installation.

In case you want to uninstall EAGLE from your computer, simply delete the *EAGLE* entry in the the *Finder*'s *Applications* folder.

2.3 Updating an Older Version

For reasons of safety it is good practice to create a backup of your previous data before proceeding!

Please read the file *update.txt* in the *EAGLE/doc* directory, in order to familiarize yourself with the changes in the new version of the program. Additional notes on installing an update may be found in the latest *README* files.

You should follow the same procedure described in the section on *New Installations* on page 23. The new EAGLE release will be installed into a new folder with its sub-folders for Libraries, Script files, User Language Programs and so on.

After starting EAGLE for the first time, please check the path settings in the Control Panel's *Options/Directories..* menu.
The path settings are taken from the EAGLE configuration file *eaglerc(.usr)*, if existing, from a previous EAGLE version installed. Modify the settings if necessary. The variable *$EAGLEDIR* stands for the current EAGLE installation directory.

Notes on Library Files

All files from previous versions can be used with the new EAGLE version. Please check which library files are *in use,* and available for the ADD command. To make sure that you are working with those of the new EAGLE

version you should, for example in the Schematic Editor, type the following command in the command line

```
USE -*
```

This removes all libraries from the buffer. Then type

```
USE *
```

to load all libraries of the currently given directory or other directories.
The information about libraries *in use* is stored in the *eagle.epf* file of the currently active project.

If you have added new Devices to existing libraries, you should re-name and copy these libraries to a separate folder to avoid over writing, muddling up files with newer or older ones, and in the worst case loosing your work prior to updating. This same rule holds true for ULPs and Script files as well.

In Case of Changes in the File Data Structure

In case of an update where it was necessary to change the file data structure, it may be wise to save your own library files from the earlier version in the new EAGLE. Expanding the tree view's library preview or showing all libraries by the first ADD command cause additional time in screen update viewing, depending on your computer speed. EAGLE has to update the files temporarily to the new file format before showing the libraries' contents.

In case you have a lot of files, there is a quick and comfortable way to solve this issue. You need two tools to achieve this:
The User Language Program *run-loop-all-lbr-script.ulp* and a Script file that contains one line:

```
WRITE;
```

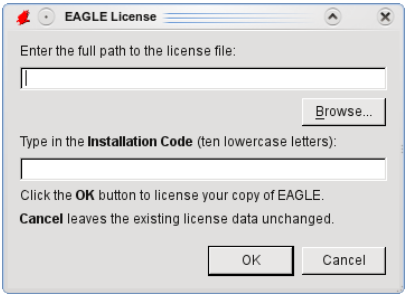
Edit one of the library files that shall be updated and start the ULP. You will be asked for the Script file to be executed, then all libraries which are in the same directory will be updated.

> *Files from earlier versions can be used directly with the current one. In case the files were made with a version prior to EAGLE 2.60 you have to convert them with a program named Update26.exe. Detailed information about this can be found on page 305.*

2.4 Changing or Extending the License

If you are changing your license you will receive new installation code together with a new license file l*icense.key*.

Run EAGLE, and select the item *Product Registration* in the Control Panel under the *Help* menu. You will now be asked for the path to the *license.key* file. Choose it and then type, in the second line, your personal installation code. Then click *OK*.



➢ ***The License Dialog***

The program has now been re-licensed. You can call up the license data at any time in the EAGLE Control Panel by means of the *Help/Product Information* menu.

So you can upgrade an already installed Light Edition with a Professional Edition, for example.

2.5 Multiple User Licenses

Multiple-user licenses may be installed separately on different computers, or may be used in a network within the scope of the license conditions. The installation procedure in a network is generally the same as that on a stand-alone computer, and is described in the section on *New Installations* on page 23.

Notes for Installing in a Network

The user rights of multiple-user licenses differ from those of the single-user version. The software itself does not contain any special network mechanism, and can therefore be used on most networks.

A typical installation may be performed as the following:

The EAGLE program is installed on a server.

Library, design, ULP, project and other directories can be freely chosen.

After installation, EAGLE is started and licensed from one of the workstations. This requires write access in the subdirectory *bin* of the EAGLE directory. The license file *eagle.key* that has been created is not changed again after the installation. Write access is no longer required.

EAGLE can now be called from all the other workstations.
Please ensure that all the workstations call EAGLE in the same way as was used when licensing.

A private working directory (local or on the network) can now be set up for each computer. A user-specific configuration file (*eaglerc.usr* under Windows, or *.eaglerc* under Linux or Mac OS X) is located in this directory. Further subdirectories can then, for instance, contain individual projects.

Alternatively it is possible for each computer on which EAGLE is to be available to have a separate license. In this case, copy the *eagle.key* file that was created in the EAGLE program directory (*eagle/bin*) into the computer's private working directory. When first starting the program, enter the installation code and the path to the license file *license.key*.
This procedure is recommended, for instance, for multiple-user licenses for 3 to 5 users who will only work at specific computers.

In network situations it is suggested that all users access one location for Library, ULP, Script, etc. This should be a single network location folder. This precaution is recommended because any custom Device, ULP, Script created by one user can now be utilized by the entire team.

Special Instructions Under Windows

Path Information

It has been found to be helpful to use the server names in UNC notation when giving the path for calling EAGLE, rather than the drive letters. For example:

```
\\netservername\eagle\bin\eagle.exe
```

Different Operating Systems at the Working Computers

If network computers having different Windows systems are in use, it is first necessary to perform an installation as described above.

The following situation can arise:
All the computers that are using, for instance, Windows XP can run EAGLE. Computers with Windows 2000, however, cannot. In order to be able to operate the computers running Windows 2000 as well, copy the files *eagle.exe* and *eagle.key*, located in the *eagle\bin* directory on the server, to, for instance, *eaglexp.exe* and *eaglexp.key*. Now start all the computers that run Windows XP, using the command

```
EAGLEXP
```

Then start EAGLE from a computer with Windows 2000, and license it again with the installation code and the license file. The *eagle.key* file is then valid for all the Windows 2000 computers. The program is run with

```
EAGLE
```

You need write authorization for the *eagle\bin* directory to perform the copying and licensing!

2.6 Language Settings

EAGLE decides due to the operating systems' language which language to use. If the systems' language, for example, is set to german, EAGLE will use german language. In case you don't like the automatically selected language, you have the following possibilities to change it.

Windows

EAGLE takes care on a variable named *LANG*. For changing it go to the Windows Control Panel where you can define environment variables. Set up a variable named LANG. For english language the value is typically set to *en_US* or *en_GB*. *F*or german language the value should be set to *de_DE*, *de_CH*, or *de_AT*. In case you would like to use a batch file to start EAGLE, it could look like this:

```
SET LANG=en_GB
C:\Program files\eagle-5.x.x
start \bin\eagle.exe
```

This is of use, if there are other applications that react on the LANG variable. The batch affects EAGLE, only.

Linux and Mac OS-X

The same as described for the Windows EAGLE can be done for Linux and Mac OS-X. There you have to define the variable with the systems' EXPORT command.

You could also use a script file in order to start EAGLE:

```
LANG=en_US
/home/user/eagle-5.x.x/bin/eagle
```

EAGLE is planned to support more and more languages. Please refer to the file *UPDATE.txt* for information about currently supported languages.

Chapter 3
EAGLE Modules and Editions

3.1 EAGLE Modules

A number of EAGLE editions are offered. You can add an Autorouter Module and/or a Schematic Editor to the Layout Editor.
A stand-alone Schematic Editor can be used for drawing wiring diagrams. In this case you won't need the Layout Editor. The user interface is identical for all parts of the program.

The Layout Editor

The Layout Editor, which allows you to design Printed Circuit Boards (PCBs) comes with the Library Editor, the Computer Aided Manufacturing (CAM) Processor, and the Text Editor. With the Library Editor you can already design Packages (footprints), Symbols and Devices (for a schematic). The CAM Processor is the program which generates the output data for the production of the PCB (e.g. Gerber or drill files). It is also possible to use User Language programs and Script files.

Schematic Editor

The Schematic Editor without Layout Editor is applicable for drawing electrical wiring diagrams (connection scheme, contact plans...). The Schematic Editor comes, as well as the Layout Editor, with the full Library Editor for designing Symbols for the Schematic and Packages for the Layout, with the CAM Processor, and the Text Editor. You can also use User Language programs and Script files.

If you want to draw Schematic diagrams for electronic systems you should have Schematic and Layout Editor. You can generate the associated circuit board at any time with a mouse-click. EAGLE then changes to the Layout Editor, where the packages are placed next to an empty board - connected via airwires (rubber bands). From here you can go on designing with the Layout Editor as usual. Schematic and layout are automatically kept consistent by EAGLE (Forward&Back Annotation). Schematic diagrams can consist of a maximum of 999 sheets in the Professional Edition (99 sheets in the Standard Edition). On the left side of the Schematic Editor window the preview of the sheets is displayed.

Autorouter

You can route the airwires automatically if you own the Autorouter module. You can choose single nets, groups of nets or all nets for the automatic routing pass. The program will handle various network classes having different track widths and minimum clearances.

The Autorouter also serves as basic engine of the Follow-me router. This is an advanced operating mode of the ROUTE command for manual routing, which calculates and displays the trace of a selected signal automatically.

3.2 Different Editions

EAGLE offers various performance/price categories (editions) called Light, Freemium, Standard, and Professional. The facilities mentioned in this manual always refer to the Professional edition.

Professional Edition

General

- maximum drawing area 64 x 64 inches
- resolution 1/10,000 mm (0.1 microns)
- mm or inch grid
- up to 255 drawing layers
- command (Script) files
- C-like User Language for data export and import and the realization of self-defined commands
- easy library editing
- composition of self-defined libraries with already existing elements by Drag&Drop
- easy generation of new Package variants from other libraries by Drag&Drop
- free rotation of package variants (0.1-degree steps)
- library browser and powerful component search function
- technology support (e. g. 74L00, 74LS00..)
- Easy definition of labelled drawing frames
- free definable attributes, applicable for Devices in the Library and in Schematic or Layout
- integrated PDF data export function
- export function for graphic files (BMP, TIF, PNG...)
- printouts via the OS's printer drivers with print preview
- partlist generation with database support (*bom.ulp*)
- Drag&Drop in the Control Panel

- context menu with object-specific commands for all objects, available through a right mouse click
- properties of objects can be accessed and edited via context menu
- automatic backup function

Layout Editor

- full SMD support
- support of Blind and Buried vias
- rotation of objects in arbitrary angles (0.1-degree steps)
- components can be locked against moving
- texts can be placed in any orientation
- dynamic calculation of signal lines while routing the layout
- magnetic-pads function
- tracks can be layed out with rounded corners in any radius
- mitering to smooth wire joints
- Design Rule Check for board layouts (checks e.g. overlaps, measures of pads or tracks)
- copper pouring (ground plains)
- Package variants support
- user-definable, free programmable User Language to generate data for mounting machines, test equipments, milling machines or any other data format
- output of manufacturing data for pen plotters, photo plotters and drilling machines with the CAM Processor

Schematic Editor

- up to 999 sheets per schematic
- icon preview for sheets
- sorting schematic sheets by Drag&Drop
- cross references for nets
- automatic generation of contact cross references
- simple copying of parts
- replace function for parts without loss of consistency between schematic and layout
- Online-Forward&Back Annotation between schematic and board
- automatic board generation
- automatic generation of supply signals
- Electrical Rule Check (error check in the Schematic and consistency check between Schematic and Layout)

Autorouter Module

- fully integrated into basic program
- uses the set of Design Rules you defined for the layout
- change between manual and automatic routing at any time
- basic engine for the Follow-me router, a tool that supports you in manual routing; the trace of a selected signal will be calculated automatically
- ripup&retry algorithm
- user-definable strategy (by cost factors)
- routing grid down to 0.8 mil (0.02 mm)
- no placement restrictions
- up to 16 signal layers (with user definable preferred directions)
- up to 14 supply layers
- full support of Blind and Buried vias
- takes into consideration various net classes

Standard Edition

Compared to the Professional Edition the following restrictions apply to the Standard Edition in the Layout Editor:

- The layout area is restricted to a maximum of 160 x 100 mm (about 6.3 x 3.9 inches). Outside this area it is not possible to place Packages and draw signals.
- A maximum number of 6 signal layers are allowed (Top, Route2, Route3, Route14, Route15, Bottom).
- A schematic can consist of a maximum of 99 sheets.

Freemium Edition

The Freemium Edition is a *Free Premium*, which is available only after registration on *http://www.element-14.com/eagle-freemium* and has the following limitations:

- The board area is restricted to 100 x 80 mm (about 3.9 x 3.2 inches), which corresponds to half of a Eurocard.
- Only 4 signal layers can be used (Top, Route2, Route15, Bottom).
- A schematic can consist of a maximum number of 4 sheets.
- The Freemium license is limited to one single user and computer, and requires an active connection to the Internet in order to work.
- The license expires 60 days after installation.

Light Edition

The following restrictions apply to the EAGLE Light Version:

- The board area is restricted to 100 x 80 mm (about 3.9 x 3.2 inches). Outside this area it is not possible to place Packages and draw signals.
- Only two signal layers can be used (no inner layers).
- A schematic can consist of only one single sheet.

Larger layouts and schematics can be printed with the *smaller* editions. The CAM processor can generate manufacturing data as well.

This
page
has been
left free
intentionally.

Chapter 4
A First Look at EAGLE

4.1 The Control Panel

The Control Panel normally appears after starting EAGLE, and this is the program's control center. All the files specific to EAGLE are managed here, and some basic settings can be made. It is similar to the familiar file managers used by a wide variety of applications and operating systems. Each EAGLE file is displayed in the tree view by means of a small symbol.

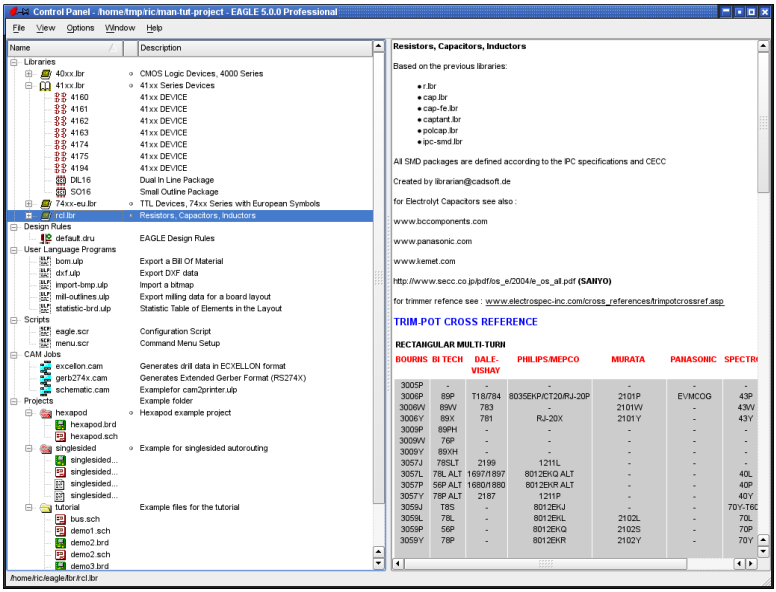
A context menu is opened by clicking with the mouse on an entry in the tree view. This allows you, depending on the object, to carry out a variety of actions (rename, copy, print, open, create new etc.).

The Control Panel supports Drag&Drop. This can also be done between different programs. You can, for instance, copy files, move them, or create links on the desktop. User Language programs or script files that are pulled with the aid of the mouse out of the Control Panel and into an editor window are started automatically. If, for instance, you pull a board file with the mouse into the Layout Editor, the file is opened.

The tree structure provides a quick overview of the libraries, Design Rules, User Language programs, script files, CAM jobs and projects. Special libraries, text, manufacturing and documentation files can belong to a project as well as schematic diagrams and layouts.

The first time it is called, the Control Panel will appear very much as shown in the following diagram. If an object is selected in the tree view, further relevant information is displayed in the right hand part of the window.

Simply click onto various folders and files in order to experiment with the Control Panel's facilities.



➢ ***Control Panel: On the right, the description of the RCL library***

Library Summary

The possibility of displaying the contents of the libraries is particularly interesting. It provides a very rapid overview of the available Devices.

Expand the *Libraries* entry, and you can see the available libraries. In the *Description* field you can see a brief description of the contents. If a library is selected, you will see more extensive information about the library in the right hand part of the Control Panel. If you then expand a library entry, the contents will be displayed together with a short description of each element. Devices and Packages are marked with a small icon.

Now select, for example, a Device:
The description of the Device and a graphical representation of it appear on the right. The available Package and technology variants are listed. If you click onto one of the Package versions, the Package preview shown above will change.

If a Schematic Editor window is open, the entry *ADD* will be shown right of the variant name. Click it and the Device is attached to the mouse cursor as soon as it is over the Schematic Editor window. Now you can drop it in the schematic.

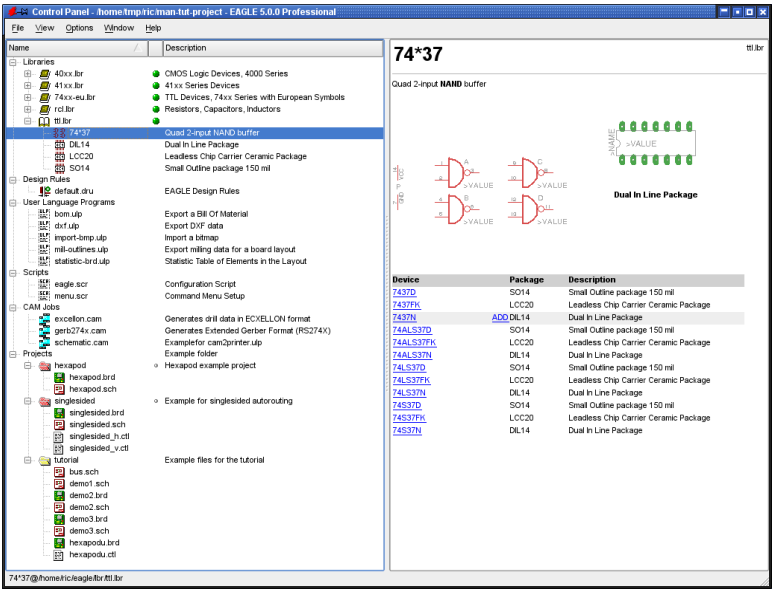
If you are only working with the Layout Editor, this will of course also operate with Packages. It is, additionally, possible to drag a Device from the tree view into a schematic diagram and to place it there by means of

Drag&Drop. If it has more than one Package version, the ADD dialog opens automatically, so that the desired Package can be selected.

The green marker behind the library entry indicates that this library is *in use*. This means that it can be used in the current project. Devices in this library will be examined by the search function in the ADD dialog of the schematic diagram or of the layout. This makes them available for the project. The library will not be examined if the marking is gray.

If starting EAGLE without a project (no *eagle.epf* file is read, the project has been closed before exiting EAGLE last time) and creating a new project (⇒ *File/New/Project*) all libraries will be *in use* automatically. However, opening an already existing project, where only certain libraries are *in use* before creating the new project, will adopt this selection.

If the Library Editor window is open you can Drag&Drop a complete Device set or Package definition from the Control Panel into the library window. This way you can copy it from one library into another.



➢ ***Control Panel: Library summary with Device view***

Design Rules

Special Design Rules can be specified in EAGLE to govern the board design. These can be saved as data sets in special files (**.dru*).

The parameter set that is to govern the current project is specified in the *Design Rules* branch of the tree view. If no data has been provided for the Design Rules (DRC command), EAGLE will itself provide parameters. The marking to the right of the file entry specifies the default parameter set for the current project. The layout will be checked by the DRC in accordance with these criteria. Further information about the DRC and the Design Rules is found starting on page 128.

User Language Programs, Scripts, CAM Jobs

These entries show the contents of the *ulp*, *scr* and *cam* directories. They contain various User Language programs (**.ulp*), script files (**.scr*) and CAM jobs (**.cam*) for the output of data using the CAM Processor. If one of these files is selected in the Control Panel, you will see a full description of the file.

The paths can be set by means of the *Options/Directories* menu. This is discussed in more detail later in this chapter.

Projects

The various projects are managed from the Control Panel. A click onto the *Projects* entry displays various folders. These are located under the path set under *Options/Directories/Projects*. It is allowed to define more than one path there.

A project usually consists of a folder which represents the project by its name and the project's configuration file *eagle.epf*. The folder usually contains all files that belong to your project, for example, schematic and board file, special library files, script files and so on.

Project directories that contain the project file *eagle.epf* will be marked with a red folder icon. 'Normal' folders will be marked with a yellow icon.

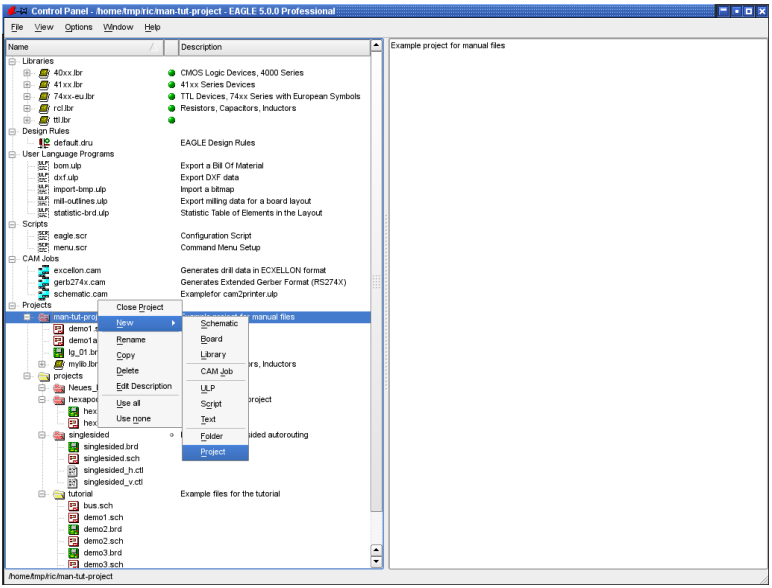
The project to be edited is selected in the *Projects* branch. On the right of the project's name you will find a marker which is either gray or green. With the help of this marker one can open or close projects. Clicking onto a gray marker, loads the project. The marker appears green now. Clicking onto the green marker again or clicking onto another gray marker closes the current project respectively opens another project after closing the current one. This way one can switch easily from one project to another.
As an alternative you can open or close a project by double-clicking onto the entry in the tree view or by pressing the S*pace* or E*nter* key.

While closing a project the settings of the currently opened Editor windows will be stored in the corresponding project file *eagle.epf*, provided that the option *Automatically save project file* is set in the *Options/Backup* menu.

If the project file was generated by another EAGLE version than currently used, you will be asked, if it is allowed to overwrite the file.

New projects are created by clicking the right mouse button onto a folder entry in this branch. A context menu opens which permits new files and directories to be created and the individual projects to be managed.
Selecting the option *New/Project* invokes a new folder which has to be given the project's title. The project file *eagle.epf* will be created automatically.

You can also use the *File/Open/Project* or the *File/New/Project* menu to open or create a new project.



➢ ***Context menu for project management***

The context menu contains the *Edit Description* item. A description of the project can be entered here, and this is then displayed in the *Description* box.

Menu Bar

The Control Panel allows various actions to be executed and settings made through pull-down menus that are explained below.

File Menu

The *File* menu contains the following items:

New

Creates a new layout (board), schematic, library, CAM job, ULP, script or text file. The *Project* option creates a new project. This initially consists simply of a new directory in which the files for a new project are handled. These will

consist as a rule of the schematic diagram and layout, possibly of special libraries, script files, User Language programs, documentation files etc. and of the file *eagle.epf*, in which project-specific settings are stored.

The default directories for the various file types are defined in the *Options/Directories* menu.

CAM jobs are definitions for generating output data with the CAM Processor.

Script and ULP files are text files containing command sequences in the EAGLE command language or the EAGLE User Language. They can be created and edited with the EAGLE Text Editor or with an external text editor.

Open

Opens an existing file of the types mentioned above.

Open recent projects

Lists recently used projects.

Save all

All changed files are saved. The current settings for the project are saved in the file *eagle.epf*, even if the option *Automatically save project file* in the menu *Options/Backup...* is switched off. User-specific settings are stored in the file *eaglerc.usr* (Windows) or *.eaglerc* (Linux/Mac).

Close project

The project will be closed. Project-specific settings are saved in the *eagle.epf* file of the current project directory.

Exit

The program is terminated. When EAGLE is started again, the last program status is restored, i.e. the windows and other working environment parameters appear unchanged. If there was no project loaded only the Control Panel will be opened next time.

The current status is also saved when you leave EAGLE with *Alt-X* from any program part.

> *If you have deactivated the Pull-down menu of the Editor windows with the Options/User interface menu, Alt+X won't work. Use the QUIT command instead. You could even assign the QUIT command to Alt+X with the help of the ASSIGN command.*

View Menu

Refresh

The contents of the tree view are updated.

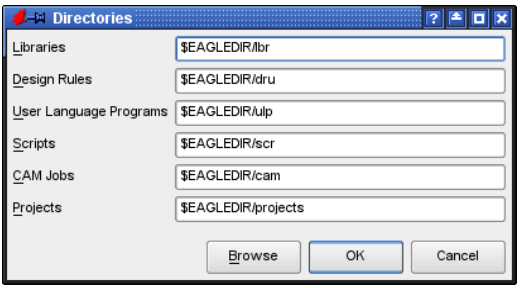
Sort

The contents of the tree view will be sorted *by name* or *by type*.

Options Menu

Directories

The default directories for particular EAGLE files are entered in the directories dialog box.
More than one path may be entered for each of these. In the Windows version the entries are separated by semicolons, while a colon is used in the Linux and Mac version. The *Projects* directory is the default directory for the Text Editor.
The *Projects* directory contains subdirectories, each of which represents a particular project. Each of the project directories contains an EAGLE project file (*eagle.epf*). A project directory and its subdirectories usually contain all the files that are associated with one particular project, such as the schematic diagram and the layout, text files, manufacturing data, documentation files and so on.



➢ ***The directories dialog in the Options menu***

Type the path directly into the corresponding box, or select the desired directory by clicking the *Browse* button.

The default settings can be seen in the diagram above. *$EAGLEDIR* stands for the installation's EAGLE directory.

You may also use *$HOME* for your home directory under Linux. Under Windows it is possible to define this environment variable wit the SET command.

If a HOME variable has not been set within the Environment variable, then under Windows EAGLE will suggest the directory *Application Data*.

This directory is defined in the Windows registry in:

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVers
ion\Explorer\Shell Folders\AppData
```

In this folder you can also find the user-specific configuration file *eaglerc.usr*.

It is of course also possible to specify paths with an absolute format.

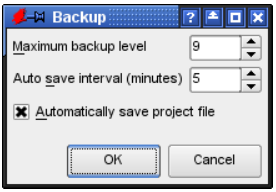
The HOME variable must not point to the root directory of a drive!

Backup

When files are saved, EAGLE creates backup copies of the previous files. The *maximum backup level* field allows you to enter the maximum number of backup copies (default: 9). Backup files have different file extensions, enumerated sequentially. Schematic files receive the ending *s#x*, board files *b#x*, and library files *l#x*. *x* can run from 1 to 9. The file with $x = 1$ is the newest one.

The automatic backup function also permits the backup to be scheduled. The time-interval can be between 1 and 60 minutes (default: 5 minutes). The backup files have the endings *b##*, *s##* and *l##* respectively.

All these backup files can be further processed in EAGLE if they are renamed and given the usual file endings (*brd*, *sch*, *lbr*).



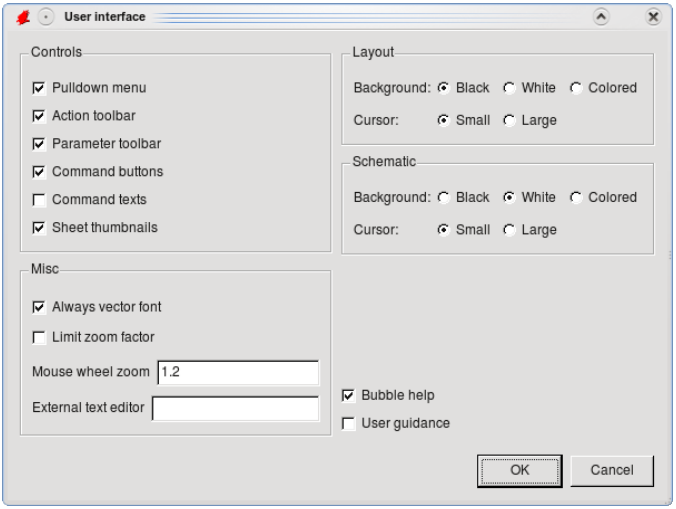
➢ ***Backup dialog***

If the option to *Automatically save project file* is chosen, your project is automatically saved when you close the current project or leave the program.

User Interface

The User Interface dialog allows the appearance of the editor windows for the layout, schematic diagram and library to be adjusted to your preferences. You can also access this menu from the Editor windows.

In the *Controls* box you specify which objects are to be displayed in the editor window. If you deactivate all the *Controls*, only the command line will remain for entry. This maximizes the free area available for the drawing.



➢ ***Settings for the User Interface***

The option *Always vector font* shows and prints texts with the built-in vector font, independently from the originally used font. Using the Vector font guarantees that the output with a printer or the CAM Processor is exactly the same as shown in the editor window. Fonts other than vector font depend on the systems' settings and cannot be controlled by EAGLE. The output of non-vector fonts may differ from the editor's view.
Opening the *User Interface* dialog from one of the Editor windows (for example, the Layout Editor) the *Always vector font* option offers an additional item *Persistent in this drawing*. Setting this option causes EAGLE to save the *Always vector font* setting in the current drawing file. So you can be quite sure that the layout will be shown with vector font at another's person computer (for example, at a board house).
Please see the help function for details (TEXT command).

Limit zoom factor limits the maximum zoom factor in an editor window. At maximum zoom level the width of the drawing is about one Millimetre (approx. 40 mil). Switching off this option allows you to zoom until the 0.1 Micron (0.004 mil) grid will become visible.

If you are working with a wheel mouse, you can zoom in and out by turning the mouse wheel. *Mouse wheel* zoom determines the zoom factor. The value 0 switches this function off. The wheel is used for scrolling then.

The field *External text editor* allows you to specify an alternative for the built-in EAGLE text editor. Further details on this can be found in the help function in the section *Editor windows/Text editor*.

The background color and the appearance of the drawing cursor can be separately adjusted for the layout and the schematic diagram editors. The *background* may be black, white or shown in any other color (*Colored*). The background color definition is described on page 100.

The cursor can be displayed optionally as *small* cross or as *large* cross-hairs.

The check boxes in the *Help* area allow to switch on or off the popup texts for the command icons (*Bubble help*) and the help texts for the commands in the status line (*User guidance*).
Selecting the *User guidance* check box displays, addition information about the selected object, like the net or signal name, the net class, or the part's name and value (with NET, MOVE, ROUTE, SHOW...), instructions about the possible mouse actions in the status bar of the editor window.

Window Positions

Here you can store the positions and the sizes of the currently open Editor windows. Each file that will be opened from now on appears in its Editor window at the given position and size parameters that were stored.
If you delete the stored positions again, EAGLE determines the location of an Editor window and uses a fixed size for it, which is the default setting.

Window Menu

From the *Window* menu you can choose the window (schematic, board, etc.) to be displayed in the foreground. The number on the left is the window number. It allows you to choose a window when combined with the *Alt* key (e.g. *Alt+1* selects window 1).

The combination *Alt+0* can be used anywhere in the program to bring the Control Panel into the foreground.

> *The functionality of Alt+window_number is supported in the Windows and in the Linux version only.*

Help Menu

The *Help* menu contains an item for calling the help function, as well as items for installing a new license (*Product registration*) and getting information about the program version etc. (*Product information)*.

Product registration

The registration dialog is called automatically when you start EAGLE the first time. If you want to install an upgrade you must start this dialog from the *Help* menu, and then enter the necessary information according to the *License/Product Registration* section of the help function. All editor windows have to be closed before. Read the notes in the chapter *Installation* for more information.

Check for Update

By default EAGLE checks for newer EAGLE versions on the CadSoft web server. In case there is a newer version available you will be informed by a message window. EAGLE informs you about a new software release, but won't download it automatically.

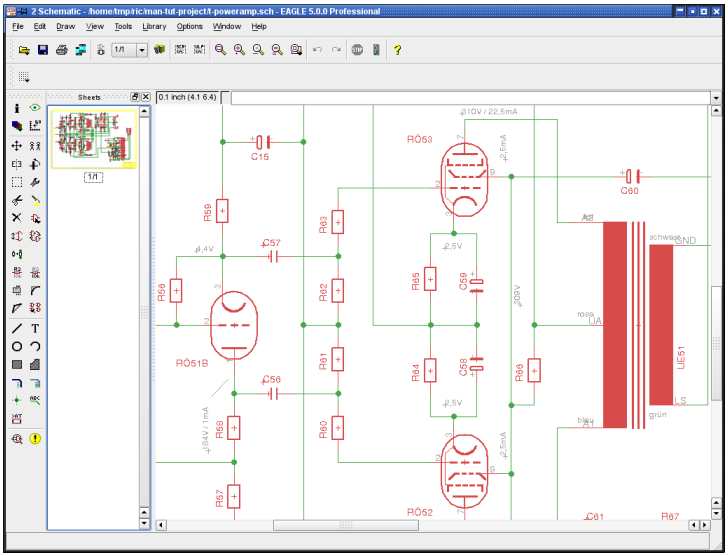
Click the *Configure* button to let EAGLE look for the software version *daily*, *weekly,* or *never*. If you are working with a Proxy server, type in it's name. In case you are interested in the newest beta version, you can activate this check, too.

Product information

If you contact our technical support you should provide the information you will find under this menu item.

4.2 The Schematic Editor Window

The Schematic Editor window opens when you load an existing schematic or create a new one. There are several ways of opening files in EAGLE.
You can, for instance, load a schematic diagram by means of the *File/Open/Schematic* menu in the Control Panel. Alternatively double-click onto a schematic diagram file in the tree view.
If you want to create a new schematic, select the menu *File/New/Schematic*. This will open a schematic with the name *untitled.sch* in the current project directory.



➢ *The Schematic Editor*

If you want to create a schematic diagram straight away in a new project, you may for example click with the right mouse button onto a project in the *Projects* entry of the tree view, and select the *New project* option from the context menu. The new project receives a name. Then click onto this entry with the right mouse button. Now select *New/Schematic* from the context menu.
A new schematic opens in this project directory.

On top you will see the **title bar**, which contains the file name, and then the **menu bar**, and the **action toolbar**.

Below the action toolbar there is the **parameter toolbar,** which contains different icons, depending on the active command.

Above the working area you will find the **coordinate display** on the left, with the **command line**, where commands can be entered in text format, to the right of it.

EAGLE accepts commands in different but equivalent ways: as mouse clicks, text via keyboard, or from command (script) files.

On the left of the work space you find the **command toolbar**, which contains most of the Schematic Editor's commands.

In the **status line**, at the bottom of the screen, instructions for the user appear, if a command is active.

On the left you you can see the preview of the schematic sheets. You can sort the sheets via Drag&Drop.

Each of the toolbars can be displayed or hidden using *Options/User Interface*. It is also possible to rearrange the toolbars within certain limits with the aid of the mouse. The command toolbar, for instance, can also be placed on the right, or the action and parameter toolbars can be placed together on one line.

How You Obtain Detailed Information About a Command

Bubble Help And Tool Tips

If the mouse cursor remains above an icon for longer than a certain time, the name of the EAGLE command appears. You also see a short explanation below in the status line.

For example, move the cursor over the WIRE icon. Bubble help with the word *Wire* appears directly by the cursor. The short description, *Draw lines*, appears in the status line.

If you select the command, a short note appears below in the status line, indicating what would normally be expected as the next action. For instance, if you click onto the WIRE icon, the status line will display the instruction: *Left-click to start wire* .

These functions can be activated or cancelled in the Control Panel by means of the *Options/User Interface* menu.

Help Function

If you want to learn more about a command, e.g. the WIRE command, click its icon in the command toolbar, then click the help icon.

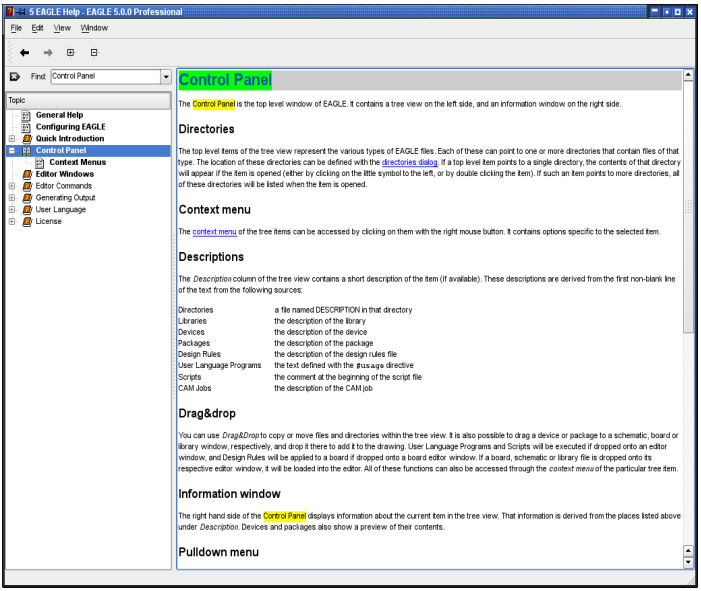
As an alternative you can type

```
HELP WIRE ←
```

in the command line. The ← character symbolizes the *Enter* key.

The contents of the EAGLE Help is stored in a single HTML file and can be viewed for example with a web browser, as well. It also offers a full-text search.

After typing in a search term in the *Find* line, EAGLE help no longer shows all pages but only the pages containing this expression. The keys F3 and *Shift+F3* allow you to go to the next or previous location. Each search term found will be marked. Green indicating the currently found term, yellow for all others.



➢ ***EAGLE Help window***

Command Parameters

A number of EAGLE commands need additional parameters. Refer to the help pages for a description of the textual entry of parameters (via command line or script file).

Most of the parameters can be entered by clicking the appropriate icons in the parameter toolbar, which changes according to the selected command. These icons also show bubble help explanations.

This is how the parameter toolbar appears when the NET command is activated.

➢ ***Parameter toolbar of the NET command***

On the left is the GRID icon for setting the grid pitch. To the right are buttons for the bend mode (SET WIRE_BEND) of the net line, followed by the miter radius for smoothing line joints with the options straight or rounded (see MITER command). Next to this is the *Style* menu where the type of line is defined. On the far right is a value menu for assigning a *Net class*.

GRID

This icon is available at any time. It is used to adjust the grid and to select the current unit. In EAGLE, any value relates to the current unit.

A right-click onto the icon opens a popup menu that contains the entry *Last*. So you can switch back to the previously chosen grid setting. The *New...* entry allows to define so-called Aliases. More about this in chapter 5.

The Action Toolbar

This toolbar is composed of the following icons:

From the left: Open file, save file, print file, call CAM Processor, open/create corresponding board file (BOARD command).

Load, remove, or create a new schematic sheet.

USE

Select libraries which will be taken into consideration by the ADD dialog. Can also be done with the *Library/Use* menu item or by clicking the markers in the *Libraries* branch of the Control Panel's tree view. The context menu of the entry *Libraries* or of its subfolders contains the entries *Use all* and *Use none* for a quick and simple selection/deselection of all libraries (of the folder).
This command has to be used in script files in order to choose the library you want to take parts from.

SCRIPT

Execute a script file. This enables you to execute any command sequence with a few mouse clicks.
A right-click onto the icon shows a list of recently executed script files.

RUN

Start a User Language program (ULP).
A right-click onto the icon shows a menu that contains a list of recently used User Language Programs.

WINDOW

These icons represent different modes of the WINDOW command:
Fit drawing into the screen (WINDOW FIT, *Alt-F2*), zoom in (*F3*), zoom out (*F4*), redraw screen (WINDOW or *F2*), select new area.

> *To move the current drawing window, click the middle mouse button and move your mouse!*
> *WINDOW LAST returns to the previous display window.*

UNDO/REDO

These commands allow you to cancel previous commands and to execute commands which have previously been cancelled. Default function keys: *F9* and *F10*.

Stop Icon

Terminates the execution of EAGLE commands (*Edit/Stop command*).

Go Icon

Starts the execution of an active EAGLE command, which allows further parameters to be entered by the user, like it is with the AUTO or the MARK command.

The Command Toolbar of The Schematic Editor

INFO

Shows the properties of the selected object. If you know the name of the object, you can use it as a parameter in the command line. Depending on the selected object some of the properties can be altered in this dialog.

SHOW

Highlights the object to be selected with the mouse.

It's also possible to enter the object's or Gate's name (even several names at once) in the command line. You may use the characters `*` and `?` as wildcards, as well. *Ctrl* + SHOW toggles the show state of the selected object.

If you are looking for very small objects, it can be useful to use the SHOW command with the @ option, like in

```
SHOW @ C12;
```

The location of part *C12* will be recognized at once, because the part is marked with a surrounding frame.

If the searched object is not located on the current sheet, the *SHOW* window opens and informs you about the sheet where it is located. In case of objects that consist of more than one part, like elements with several gates or nets that spread over several sheets, the window will list several entries. Clicking on one of the entries center the selected object on the screen. If the searched object is not found in the whole schematic, the *Sheet* column will be marked with a minus sign '-'.

DISPLAY

Select and deselect the layers to be displayed. See the *Appendix* for the meaning of the layers
DISPLAY LAST shows the recently used layer combination that was previously selected for display.

For further details please see help function.

MARK

The following mouse click defines the new origin for the coordinate display. Relative coordinates *(`R` x-value y-value)* and polar values *(`P` radius angle)* are shown in addition to absolute coordinates in the coordinate display box. If you first click the MARK icon and then the traffic-light icon, only the absolute coordinate values will be displayed again.

MOVE

Move any visible object. The right mouse button rotates the object while it is attached to the mouse cursor.

If you move a net over a pin, no electrical connection will be established. If you move the pin of a Gate over a net or another pin, an electrical connection will be created.

To move groups of objects:
Define the group with the GROUP command, click the MOVE icon, press the *Ctrl* key, then click into the drawing with the right mouse button, and move it to the desired location.
If you don't press the *Ctrl* key, the context menu pops up after clicking with the right mouse button. It contains an entry *Move:Group* that allows you to move the group, too. The right mouse button rotates the group by 90 degrees while it is attached to the mouse cursor.

If you like to move the group onto another sheet, click the sheet combo box in the action toolbar or select it from the Sheets preview. Place the group there.

MOVE can be used in the command line with various options. See the help function for details.

COPY

Copy parts and other objects.
When copying nets and buses the names are retained, but in all other cases a new name is assigned.

COPY can be used with groups and works the same as it would be with CUT and PASTE within a drawing. The content of the group does not go to the clipboard of your operating system!

MIRROR

Mirror objects.

ROTATE

Rotate objects by 90 degrees (also possible with MOVE).

GROUP

Define a group which can then be moved, rotated, or copied with CUT and PASTE to another drawing or whose properties are to be changed. After the icon has been clicked, a rectangular group can be defined by holding down the left mouse button and dragging the cursor to the diagonal corner of the rectangle. If you want to define a group by a polygon, use the left mouse button to determine the corners of the polygon. Then click the right mouse button to close the polygon.

GROUP ALL in the command line selects all objects on the current sheet, if the respective layers are displayed.
The following command (ROTATE, CHANGE, MOVE...) has to be applied to the group with the right mouse button while the key is pressed.
Exception is CUT: Here a left mouse click is expected.

If you like to add further groups to an already existing one, press the *Shift* key and define the first corner of the selection area with a mouse click.
In case you want to add an object to or remove it from the group, press the *Ctrl* key and click onto the object in question.
Press *Ctrl* + *Shift* to toggle the membership of an object and its hierarchically superior objects: Clicking for example, on a net segment in the Schematic inverts the group membership of the whole net.

CHANGE

Change the properties of an object, e.g. the width of a line, the Package variant or the size of text. See help for details.

An object's properties can be checked and even changed, where applicable, by the *Properties* entry of the context menu. To access the context menu, click onto the object with the right mouse button.

CUT

Transfer the objects of a previously defined group into the paste buffer. Activate the CUT command and click with the left mouse button into the group to set a reference point.
All other commands that can be executed with a group expect a right mouse click while the *Ctrl* key is pressed.

> *Not identical to the Windows cut function!*

PASTE

Insert objects from the paste buffer into the drawing. Restrictions: see help function.

Not identical to the Windows paste function.

DELETE

Delete visible objects.

Also in combination with GROUP command. If a group has been defined, it can be deleted with the right mouse button while the *Ctrl* key is pressed.

The DELETE command deletes an entire part in the Schematic when clicking onto a Gate with the *Shift* key pressed. In that case, the tracks connected to the Package in the board, if already existing, will stay unchanged.

Clicking onto a net or bus wire with the *Shift* key pressed deletes the entire net or bus segment.

ADD

Add library elements to the schematic. A search function helps Devices to be found quickly. USE specifies which libraries are available.

A right-click onto the ADD icon opens a popup menu that lists recently fetched Devices.

PINSWAP

Swap two nets connected to equivalent pins of a Device, provided the pins have been defined with the same Swaplevel.

GATESWAP

Swap two equivalent Gates of a Device, provided the Gates have been defined with the same Swaplevel. In EAGLE terminology, a Gate is a part of a Device which can be individually placed on a schematic (e.g. one transistor from a transistor array).

REPLACE

Replace a component (Device) with another one from any library. This can only work if the new component has at least as many pins as the current one and the pins as well as the pads have identical names or the same positions.

A right-click onto this icon opens a popup menu that shows a list of recently replaced Devices.

NAME

Give names to components, nets, or buses.

VALUE

Provide values for components. Integrated circuits normally get the type (e.g. 74LS00N) as their value.
A right-click onto this icon opens a list of already used values. Select an entry and apply it to one or more components by clicking onto them successively.

SMASH

Separate name, value, and, if any, attribute texts from a Device, so that they can be placed individually. The size of detached (smashed) texts can also be individually changed. Also in combination with GROUP. If a group is defined, you can smash it with a right mouse click while the *Ctrl* key is pressed.

Use DELETE to hide smashed texts.

Keep the *Shift* key pressed while using the SMASH command in order to unsmash text. Text is not editable any more and appears at original position(s) after a window refresh (also possible in the context menu with *unSmash*).
Alternatively you can also switch on or off the option *Smashed* in the context menu's *Properties* entry.

MITER

Round off or bevel wire joints (also possible for nets, buses, polygon contours). The grade of mitering is determined by the miter radius. Positive sign results in a rounded joint, negative sign in a bevel.
The miter radius influences some wire bends, too (see help function: SET command, *Wire_Bend*).

SPLIT

Insert an angle into a wire or net.

INVOKE

Devices that consist of more than one Symbols (Gates) can be fetched Gate by Gate, for example in certain order (Gate D before Gate C), if wanted.
INVOKE can also be used to fetch power supply Gates that do not appear automatically in the Schematic. This is useful and required, for example, when you are adding decoupling capacitors to your design.

This command allows you also to add a Gate from a Device which is located on another sheet. In such a case, type the name of the Device (e.g. IC1) into the command line after the INVOKE command has been selected.

WIRE

Draw line (this command is called WIRE because it is used to define electrical connections, i.e. wires, in the Layout Editor). The type of line can be changed with CHANGE STYLE. Clicking the right mouse button changes the bend mode (SET WIRE_BEND).

WIRE can also be used to draw arcs.

Please note the particularities in combination with the *Ctrl* and *Shift* key in the help function:
If you press, for example, the *Ctrl* key while starting to draw a wire, the wire begins exactly at the end of an already existing wire nearby. Even if this wire is not in the currently set grid. Wire width, style and layer will be adopted from the already existing wire.

TEXT

Placing text. Use CHANGE SIZE to alter the height of the text. If the text is using a vector font, CHANGE RATIO will alter the thickness. CHANGE TEXT is used to alter the text itself. CHANGE FONT alters the typeface. You can change label texts by assigning a different name to the bus or to a net by means of the NAME command. See also LABEL command.

CIRCLE

Draw a circle. Circles with a width of 0 are drawn as filled circles.

ARC

Draw an arc (also possible with WIRE).
CHANGE CAP FLAT | ROUND defines straight or rounded ends for arcs.

RECT

Draw a rectangle.

POLYGON

Draw a polygon (copper areas in any shape).

BUS

Draw a bus line. The meaning of a bus is more conceptual than physical. It is only a means to make a schematic easier to read. Only nets define an electrical connection. Nets, however, can be dragged out of a bus.

NET

Draw a net. Nets with the same name are connected (even if located on different sheets).

Nets and pins which appear to the eye to be connected are not necessarily electrically connected. Please check with the SHOW command, the ERC, or by exporting a netlist or pinlist (EXPORT NETLIST or PARTLIST). See also the help for MOVE.

JUNCTION

Place the symbol for a net connection. In general, junctions are placed automatically, but nets which cross over can also be joined manually by the JUNCTION command.

LABEL

Place the name of a bus or net as a label. Labels cannot be changed with CHANGE TEXT but rather with the NAME command because the label represents the net name.

If the label option *XREF* (in the parameter toolbar or by CHANGE XREF ON) is set, a cross reference pointing to an further instance of the chosen net on the next sheet is generated automatically.
The cross reference label format can be defined in the menu *Options/Set/Misc*, *Xref label format*. See the help function of the LABEL command for the meaning of the placeholders that can be used.

For a proper location of the object you should use a drawing frame with classifications for columns and rows. Such frames can be defined with the FRAME command. The library *frames.lbr* already contains such frames.

ATTRIBUTE

Defines an attribute for a component. Attributes are free definable and can contain any information.
Through the menu *Edit/Global attributes..* you can define attributes that are valid for all components respectively for the whole schematic.

ERC

Perform an Electrical Rule Check and a consistency check for schematic and board, if already existing. A positive consistency check allows the Forward&Back Annotation engine to run.

Commands Not Available in the Command Toolbar

Menu items already explained in the Control Panel section are not discussed here.

The following commands can be entered into the command line as text inputs. Some of them are available as menu items. Most of them can be used in the Schematic and in the Layout and even in the Library Editor.

ASSIGN

Assign function keys.
The most convenient way of doing this is to use the *Options/Assign* menu.

CLASS

Select and define net classes (*Edit/Net classes...*). A net class specifies the width of a track, the clearance from neighbouring signals, and the diameter of vias for the Autorouter and the ROUTE command. These settings are also used in polygons. See also page 117.

CLOSE

Text command for closing an editor window (*File/Close*).

EDIT

Text command for loading a file or a library object. You can, for instance, load a board from the Schematic Editor (EDIT name.brd).

FRAME

Define a drawing frame for the Schematic (*Draw/Frame*). Also possible for a board drawing.

EXPORT

Output lists (especially netlists), directories, script files, or images (*File/Export...*).

LAYER

Choose or define the drawing layer. When using drawing commands the layer can be chosen in the parameter toolbar.

To create, for example, a new layer with number *200* and layer name *Mylayer*, type in the command line:

```
LAYER 200 Mylayer
```

In case you created a Layout, for example, with the EAGLE Light Edition and upgraded to the Standard Edition because you would like to use two additional inner signal layers, you have to create these layers with the LAYER command first:

```
LAYER 2 Route2
LAYER 15 Route15
```

MENU

Specifies the contents of the text menu. See also the example in the appendix. The text menu can be made visible with the aid of *Options/User Interface*. See help function for details.

OPEN

Text command for opening a library for editing (*Library/Open*). This command is not identical to the *File/Open* menu item of the Schematic Editor, which only lets you select schematics. You can use the OPEN command as an alternative to the *File* menu of the Control Panel.

PACKAGE

In case there is more than one Package variant defined in the library for a part (Device), a typical example would be a resistor from *rcl.lbr*, it is possible to change the currently used Package with the PACKAGE or with the CHANGE PACKAGE command. This can be done in the Schematic or in the Layout Editor.

PRINT

Call up the print dialog with the printer icon in the action toolbar or from the menu item *File/Print...*. Usually the PRINT command is used to print schematics or for checking the drawings needed for the PCB production.
The actual production data are generated with the CAM Processor.
If you want to output your drawing in black and white check the *Black* option (and *Solid*, if you don't want layers to be printed in their different fill styles). The caption text is suppressed unless you check *Caption*. Set *Page limit* to 1, if your drawing is to be fitted on one page. If you prefer to print the currently visible drawing window instead of the whole drawing, select *Window* instead of *Full* in the *Area* option.

QUIT

Quit EAGLE. Identical with the menu item *File/Exit* or *Alt-X*.

REMOVE

Delete files or schematic sheets.

```
REMOVE .S3 ←
```

for instance, deletes sheet 3 of the loaded schematic.

SET

Set system parameters and modes. Best done via the *Options/Set* menu item. Please note that not all of the possibilities are available through this dialog. Presettings can be defined in the script file *eagle.scr* by using text commands. Further information can be found in the help function.

TECHNOLOGY

If a part (Device) has been defined with various technologies in the library, see typical examples in *74xx.lbr*, it is possible to change the currently used technology with the TECHNOLOGY or with the CHANGE TECHNOLOGY command. This can be done in the Schematic or in the Layout Editor.

UPDATE

The UPDATE command checks the parts in a board or schematic against their respective library objects and automatically updates them if they are different. (*Library/Update...* or *Library/Update all*).
The context menu in the Control Panel's' tree view offers the Options *Use all* and *Use none* for a quick selection of libraries.

WRITE

Text command for saving the currently loaded file. Please note that, in contrast to *Save as*, the name of the currently edited file is never changed when the WRITE command is used.

Mouse Keys

The middle and right mouse button have a special meaning for a number of commands. You can use the middle mouse button only if the operating system knows your mouse is a 3-button mouse, that is your mouse must be installed this way.

If you are working with a wheel mouse, you can zoom into and out of the drawing with the help of the mouse wheel. The option *Mouse wheel zoom* in the *Options/User Interface* menu determins the zooming in/out factor per step. The value is set to 1.2 by default.
Selecting a value of 0 allows you to use the wheel for scrolling.

Keep the mouse wheel or the middle mouse button pressed for panning.

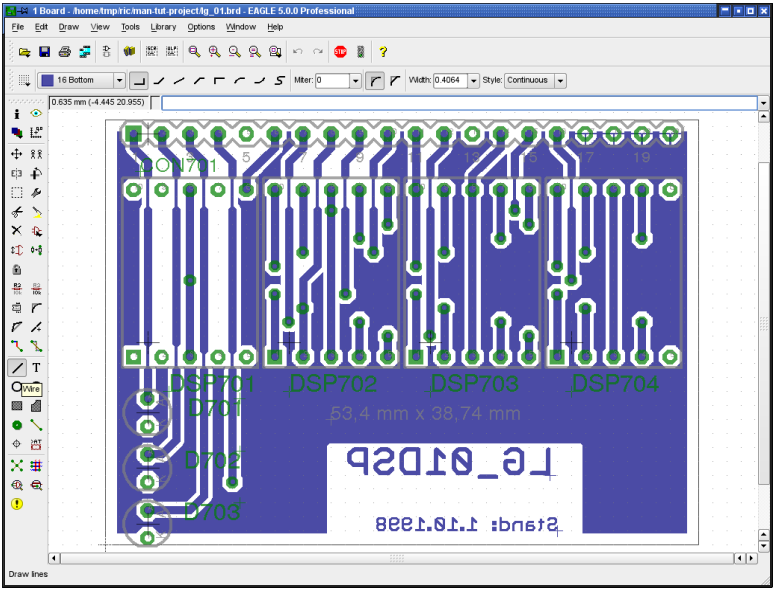
Mouse clicks in combination with the *Shift*, *Ctrl*, and *Alt* key can have various functions, for example, while selecting objects with MOVE or while drawing lines with WIRE.
The help section on *Keyboard and Mouse* and the help of the referring command gives you more details.

Selecting Neighbouring Objects

If one of two objects which are very close together is to be selected, the individual objects are highlighted one after the other. The user can select the highlighted object with the left mouse button, or proceed to the next one with the right mouse button. The status bar of the editor window shows information about the pre-selected object. See also help function (SET command, SELECT_FACTOR).

4.3 The Layout Editor Window

The Layout Editor window opens when you open an existing board file or create a new board. If you own the Schematic Editor you will normally draw a schematic first and then generate the board file with the BOARD command, or by clicking the *Board* icon.



➢ ***Layout Editor window***

The Layout Editor window appears very much like the Schematic Editor window. Even if you don't work with the Schematic Editor, you should study the previous section, as most of the information there applies to the Layout Editor, too.

Only the commands in the command toolbar are discussed again, as some commands differ in their use.

Descriptions of commands that cannot be reached through the command toolbar are also to be found in the section concerning the Schematic Editor window. All of the commands can also be reached through the pull-down menus in the menu bar. This also applies, of course, to the Schematic and Layout Editor windows.

The Commands on the Layout Command Toolbar

INFO

Shows the properties of the selected object. Typing `INFO IC1` in the command line results in the properties dialog of the object named IC1. Depending on the selected object some of the properties can be altered here.

SHOW

Highlights the object to be selected with the mouse.
It's also possible to enter the object's name (even several names at once) in the command line. `*` and `?` are allowed to be used as wildcards, as well.
Ctrl + SHOW toggles the show state of the selected object.

DISPLAY

Select and deselect the layers to be displayed. Components on the top side of the board can only be selected if the layer 23, *tOrigins*, is displayed. The same applies to components on the bottom side of the board and layer 24, *bOrigins*. Only those signal layers that are defined in the Layer setup are shown.

See *Appendix* for the meaning of the layers.

The DISPLAY command supports so-called aliases. This allows you to name certain combinations of layers and use it as a parameter with the LAYER command. A quick change from one view to another layer combination is possible with this command.

DISPLAY LAST switches to the last displayed layer combination.

Detailed information about the DISPLAY command can be found in the help function.

MARK

The following mouse click defines the new origin for the coordinate display. Relative coordinates *(`R` x-value y-value)* and polar values *(`P` radius angle)* are shown in addition to absolute coordinates in the coordinate display box.

If you first click the MARK icon and then the traffic-light icon, only the absolute coordinate values will be displayed again.

MOVE

Move any visible object. The right mouse button rotates the object.

The MOVE command cannot connect signals even if a wire (trace) is moved over another wire or a pad. Use ROUTE or WIRE to route signals.

Keeping the *Ctrl* key pressed while selecting an object selects it in a particular manner. Please consult the help function for details (CRICLE, ARC, WIRE, MOVE, ROUTE etc.).

For moving groups, please see *MOVE* in the Schematic Editor section.

COPY

Copy parts and other objects.

When copying objects, a new name will be assigned, but the value will be retained. When copying a single wire, the copy will have the same name.

COPY can be used with groups and works the same as it would be with CUT and PASTE within a drawing.

MIRROR

Mirror objects. Components can be placed on the opposite side of the board by using the MIRROR command.

ROTATE

Rotate objects (also possible with MOVE). Keep the left mouse button pressed to rotate the selected object by moving the mouse. The parameter toolbar shows the current angle. This can be done with groups (GROUP and right mouse button)as well.
ROTATE can be used with groups, as well. Activate ROTATE, press the *Ctrl* key and click with the right mouse button into the drawing to set the center of rotation. The group will be rotated counterclockwise by the given angle.
Alternatively type in the angle in the *Angle* box or in the command line. Details about the syntax can be found in the help function.

GROUP

Define a group which can then be moved, rotated, or copied with CUT and PASTE to another drawing or whose properties should be changed. After clicking the icon, a rectangular group can be defined by holding down the left mouse button and dragging the cursor to the diagonal corner of the rectangle, or if you want to define a non-rectangular group area, use the left mouse button to determine the corners of the polygon. Then click the right mouse button to close the polygon.

GROUP ALL in the command line selects all objects.

To be sure that all objects are selected DISPLAY ALL layers before. On the other hand, deselecting specific layers can exclude certain objects from the selection.

Further information about GROUP can be found in the section about the Schematic Editor and in the help function.

CHANGE

Change the properties of an object, for example the width of a wire or the size of a text. If the *Esc* key is pressed after changing a property, the previously used value menu will appear again. In this way a new value can be conveniently chosen. See also the help function.

Alternatively, object properties can be viewed and some of them even changed with the context menu's *Properties* entry. The context menu opens after a right mouse click onto the object.

CUT

Transfer the objects of a previously defined group into the paste buffer.
Activate the CUT command and click with the left mouse button into the group to set a reference point.

All other commands executed with a group expect a right mouse click while the *Ctrl* key is pressed.

Not identical to the Windows cut function.

PASTE

Insert objects from the paste buffer into the drawing. Restrictions: see help. Not identical to the Windows paste function.

DELETE

Delete visible objects.

If a group has been defined, it can be deleted with the right mouse button while the *Ctrl* key is pressed.

DELETE SIGNALS in the command line erases all tracks and signals in the layout, provided there is no consistent schematic loaded.

The DELETE command deletes an entire polygon when clicking on a polygon wire with the *Shift* key pressed.
Keeping the *Ctrl* key pressed while clicking with the left mouse button on a wire bend will delete the bend. A new direct connection between the next bends will be drawn now.

If objects cannot be deleted, the reason can lie with error polygons related to the DRC command. They can be deleted with the ERRORS command (ERRORS CLEAR). If layer 23, *tOrigins*, or 24, *bOrigins*, is not displayed, components cannot be deleted.

ADD

Add library elements to the drawing. It offers a convenient search function for Packages here. USE specifies which libraries are available.

A right-click onto the ADD icon opens a popup menu that contains a list of recently placed Devices.

PINSWAP

Swap two signals connected to equivalent pads of a component, provided the pins have been defined with the same Swaplevel.

REPLACE

Replace a component (or a Package, if there is no schematic) by another one from any library.
If you want to change the Package variant only and not the whole Device, use CHANGE PACKAGE or the PACKAGE command.

A right-click onto the REPLACE icon opens a popup menu that shows a list of recently replaced components.

LOCK

Locks the position and orientation of a component on the board. If a component is locked, you can't move it or duplicate it with CUT and PASTE. *Shift* + LOCK unlocks the component. This is also possible with the *unLock* entry of the context menu.
To be able to distinguish locked from unlocked components, the origin cross of a locked component is displayed like a 'x' instead of a '+'.

The position of a locked component can be changed, however, by typing in new coordinate values in the properties dialog.

NAME

Give names to components, signals, vias, and polygons.
With NAME it's possible to move a polygon from one signal to another.

VALUE

Provide values for components. A resistor, for example, gets *100k* as its value. A right-click onto this icon opens a list of already used values. Select an entry and apply it to one or more components by clicking onto them successively.

SMASH

Separate name, value, and attribute (if any) texts from a Device, so that they can be placed individually. The size of detached (smashed) texts can also be individually changed.
Also in combination with GROUP. If a group is defined, you can smash it with a right mouse click while the *Ctrl* key is pressed.

Use the DELETE command to hide smashed texts.

Keep the *Shift* key pressed while using the SMASH command in order to unsmash texts. They are not editable any more and appear at their original positions after a window refresh (also possible with *unSmash* in the context menu).

Alternatively you can switch on or off the option *Smashed* in the context menu's *Properties* entry.

MITER

Round off or bevel wire joints (also possible for polygon contours). The grade of mitering is determined by the miter radius. Positive sign results in a rounded joint, negative sign in a bevel.
The miter radius influences some wire bend modes, too (see help function: SET, *Wire_Bend*).

SPLIT

Insert a bend into a wire.
If you want to change, for example, the layer for a section of an already routed track, you can insert two wire bends with the SPLIT command and change the layer of the newly created segment with the CHANGE LAYER. EAGLE will set vias automatically at the position of the wire bends.

You can use the SPLIT command for a quick re-routing of an already existing track. Click onto the track to insert a wire bend. Now move the mouse and route it anew. To remove the previous track use the RIPUP command or DELETE in combination with the *Ctrl* key.

OPTIMIZE

Joins wire segments in a signal layer which lie in one straight line.

ROUTE

Route signals manually. Airwires are converted to wires. If your EAGLE license comes with the Autorouter module, the ROUTE command supports the Follow-me router mode which automatically processes the trace of a selected signal.

This command offers several options with the different mouse buttons, also in combination with the *Ctrl* and *Shift* key.

Ctrl + Left	starts routing at any given point along a wire or via
Shift + Left	if the airwire begins at an already existing wire and this wire has a different width, the new wire adopts this width
Center	selects the layer
Right	changes the wire bend style
Shift +Right	reverses the direction of switching bend styles
Ctrl + Right	toggles between corresponding bend styles
Shift + Left	places a via at the end point of the wire
Ctrl + Left	defines arc radius when placing a wire's end point

More information can be found in the help function.

RIPUP

Convert routed wires (tracks) into unrouted signals (airwires). Change the display of filled (calculated) polygons to outline view.
Using signal names in the command line allows you to ripup only certain signals, to exclude particular signals, or to execute the command exclusively for polygons. More details can be found in the help function.
Wires not connected to components must be erased with DELETE.

WIRE

Draw lines and arcs. If used in the layers 1 through 16, the WIRE command creates electrical connections.
The *Style* parameter (CHANGE) determines the line type. The DRC and the Autorouter always treat a WIRE as a continuous line, regardless of what *Style* is used.
Clicking the right mouse button changes the wire bend (SET WIRE_BEND).

Please note the particularities in combination with the *Ctrl* and *Shift* key in the help function:
If you press, for example, the *Ctrl* key while starting to draw a wire, the wire begins exactly at the end of an already existing wire nearby. Even if this wire is not in the currently set grid. Wire width, style and layer will be adopted from the already existing wire.

TEXT

Placing text. Use CHANGE SIZE to alter the height of the text. If the text is using a vector font, CHANGE RATIO will alter the thickness. CHANGE TEXT is used to alter the text itself. CHANGE FONT alters the typeface.
The option *Always vector font* (*Options/User Interface*) shows and prints all texts in vector font, regardless of which font is actually set for a particular text.

If you want to have inverted text in a copper layer, you have to enter the text in the layers 41, *tRestrict,* or 42, *bRestrict*, and draw a copper plane in *Top* or *Bottom* layer around the text with the POLYGON command. The polygon keeps the restricted areas (which is the text) free from copper.

> *It is strongly recommended to write texts in copper layers as vector font! So you can be sure that the CAM Processor's output is identical with the text shown in the Layout Editor. See also help function.*

CIRCLE

Draw a circle. This command creates restricted areas for the Autorouter/Follow-me router, if used in the layers 41, *tRestrict*, 42, *bRestrict*, or 43, *vRestrict*. Circles with wire width = 0 are drawn as filled.

ARC

Draw an arc (also possible with WIRE).
CHANGE CAP FLAT | ROUND defines straight or rounded ends for arcs.
If the arc is a part of a trace and both ends are connected to a wire, caps will be round.

Arcs with flat caps are emulated when generating manufacturing data in Gerber format with the CAM Processor. That means they will be drawn with small short straight lines. Arcs with round caps won't be emulated.

RECT

Draw a rectangle. This command creates restricted areas for the Autorouter or Follow-me router, if used in the layers 41, *tRestrict*, 42, *bRestrict*, or 43, *vRestrict*.

POLYGON

Draw a polygon.

Polygons in the signal layers are treated as signals. They keep an adjustable distance to objects belonging to other signals (copper pouring, flood fill). This enables you to realize different signal areas on the same layer and make isolated regions for your design.

The contour of a polygon in the outline mode is displayed as a dotted line.

The POLYGON command creates restricted areas for the Autorouter/Follow-me router, if used in the layers *tRestrict*, *bRestrict*, or *vRestrict*, .

For other possibilities of the POLYGON command see help.

VIA

Place a plated-through hole. Vias are placed automatically if the layer is changed during the ROUTE command. You can assign a via to a signal with the NAME command by changing it's name to the name of the signal. Vias can have different shapes in the outer layers (round, square, octagon) , but are always round in inner layers.

SIGNAL

Manual definition of a signal. This is not possible if the Forward&Back Annotation is active. In that case you have to define the connection with the NET command in the Schematic Editor.

HOLE

Define a mounting hole (not plated-through).

ATTRIBUTE

Defines an attribute for a component.
Through the menu *Edit/Global attributes..* you can define attributes that are valid for the whole layout.

RATSNEST

Calculate the shortest airwires possible and the real mode (filled) display of polygons.

Use the RATSNEST command with a signal name in order to calculate and display or hide a certain airwire. A preceding exclamation mark hides the airwires of the given signal name. More information can be found in the help function.

The polygon calculation can be deactivated with the SET command. Either through the menu *Options/Set/Misc* or by typing in the command line:
SET POLYGON_RATSNEST ON | OFF or in short: SET POLY ON | OFF.

RATSNEST will be executed automatically for the selected signal while drawing a wire with ROUTE.

While RATSNEST is active the status bar of the Layout Editor displays the name of the currently calculated signal.

AUTO

Start the Autorouter.
If you type AUTO FOLLOWME in the command line, the *Autorouter Setup* window opens in the follow-me mode, which allows to set the parameters for the follow-me router only.

ERC

Perform a consistency check for schematic and board.

DRC

Define Design Rules and perform Design Rule Check.

ERRORS

Show errors found by the DRC. If you haven't already processed a Design Rule Check for the board, it will be done automatically before showing the error list, if there are any errors found.

> *There are further commands for the Layout Editor, as they are in the Schematic, that are not available in the Command Menu. Please take a look at the section beginning with page 59. Most of them are valid in Schematic and Layout.*

4.4 The Library Editor Window

The Library Editor window opens when you load a library for creating or editing components. A library normally has three different elements: Packages, Symbols and Devices.

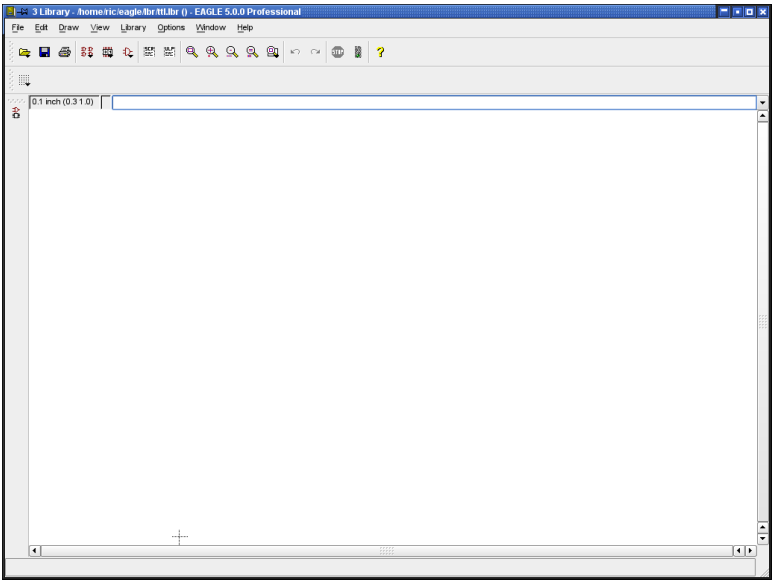
- A Package is a Device's housing, as will be used in the Layout Editor (on the board).
- The Symbol contains the way in which the Device will be shown in the schematic.
- The Device represents the link between one (or more) Symbol(s) and a Package. Here we define the connection between a pin of a Symbol and the referring pad of the Package.
  We call it a Device set if the component exists in more than one Package and/or technology variant.

Even if you do not have the Schematic Editor, you can still create and edit Symbols and Devices.

A library need not contain only real components. Ground or supply symbols as well as drawing frames can also be stored as Devices in a library. These Symbols do not normally contain any pins.
There are also libraries that only contain Packages. These libraries can only be used in the Layout Editor.

Extensive examples of the definition of library elements are to be found in a section entitled *Component Design Explained through Examples*, starting on page 201 in this manual.

When a library is loaded the following window appears first:



➢ ***Library Editor window: No element has yet been loaded***

Load or Rename Package, Symbol, or Device

The following commands are important for navigating within a library:

EDIT

Load Device or Package (if you only have the Layout Editor) for editing.

From the left: Load Device, load Package, load Symbol. These icons are shown in the action toolbar.
If you click on one of these icons with the right mouse button, or long-click with the left mouse button on one of theses icons, a list with the recently edited objects will pop up.

REMOVE

Delete Device/Package/Symbol from library. Available only through the *Library* menu or the command line. See help function or the chapter about *Library and Part Management* in this manual for more.

RENAME

Rename Device/Package/Symbol in library. Available only through the *Library* menu or the command line. See help function or the chapter about *Library and Part Management* in this manual for more.

The Package Editing Mode

The definition of a component is described briefly below. There is a more extensive guide in the *Component Design Explained through Examples* section.

The icons available in the command toolbar are equivalent to the identical icons of the Schematic or Layout Editor.

Design New Package

You change into Package editing mode through the *Package* icon in the action toolbar. Type in the name of a package, and reply to the confirming question *Create new package 'packagename'?* with *yes*.

Place pads (though-hole contacts) or SMDs (SMD contact areas) with the following commands which are only available in the Package Editor.

PAD

Place the pad of a conventional (through-hole) component.
The pad comes with a plated-through drill that goes through all signal layers. The pad shape can be round, square, octagon or long in the outer signal layers. In the inner signal layers pads are always round.

SMD

Place a SMD pad.

You can change the name of the pads or SMDs with the NAME command.

Use the WIRE, ARC, etc. commands to draw

- the symbol for the silkscreen on layer 21, *tPlace*,
- additional graphical information for the documentation print into layer 51, *tDocu*.

Draw restricted areas for the Autorouter, if needed, in layers 41, *tRestrict*, 42, *bRestrict*, or 43, *vRestrict*, or in layers 39, *tKeepout*, or 40, *bKeepout*, by using the commands CIRCLE, RECT, or POLYGON.

Place mounting holes with the HOLE command, if needed.

Use the TEXT command to place

- the string >NAME in layer 25, *tNames*, serving as a text variable containing the name of the component,

- the string >VALUE in layer 27, *tValues*, serving as a text variable containing the value of the component.

Use the DESCRIPTION command to add a description for the Package. HTML text format can be used for this. You will find further information in the help pages.

The Symbol Editing Mode

Defining a Symbol means defining a part of a Device which can be placed individually in a schematic. In the case of a 74L00 this could be one NAND gate and the two power pins, defined as another Symbol. In the case of a resistor, the Device contains only one Symbol which is the representation of the resistor.

You now change into Symbol editing mode through the *Symbol* icon in the action toolbar. Enter the name of the Symbol, and reply to the confirming question *Create new symbol 'symbolname'?* with *Yes*.

Design a New Symbol

Use the commands WIRE, ARC, etc. to draw the schematic representation of the Symbol into layer 94, *Symbols*.

Place the pins by using the following PIN command, which is only available in the Symbol editing mode:

PIN

Place pins.

You can adjust the pin parameters (*name, direction, function, length, visible, Swaplevel*) in the parameter toolbar while the PIN command is active, or later with the CHANGE command. The pin parameters are explained starting on page 206 and in the help pages under the keyword *PIN*. Pin names are changed using the NAME command.

Use the TEXT command to place

- the string >NAME in layer 25, *tNames*, serving as a text variable containing the name of the component,
- the string >VALUE in layer 27, *tValues*, serving as a text variable containing the value of the component.

The Device Editing mode

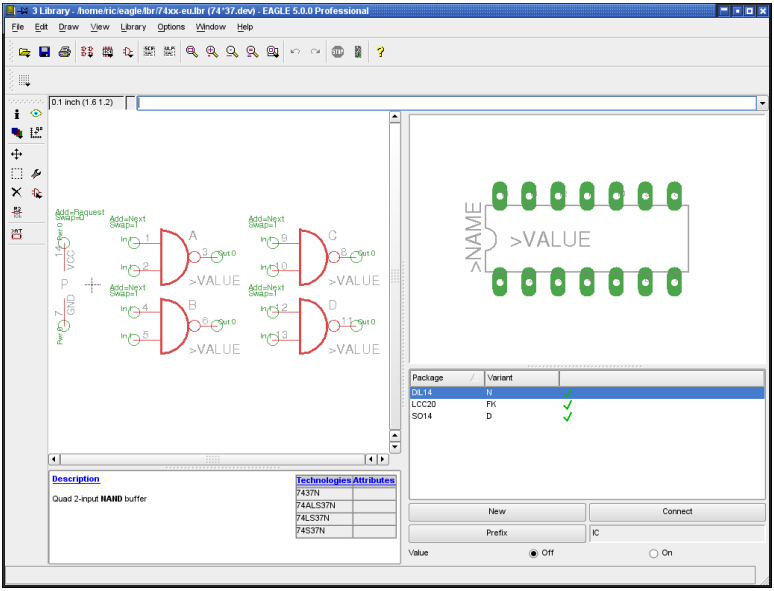
Components are defined as Devices. In the Device editing mode you do not draw anything, but you define the following:

- which Package variant is used,
- which Symbol(s) is/are used (called Gate within the Device),
- which names are provided for the Gates (e.g. A, B),

- which technologies are available (e.g. 74**L**00, 74**LS**00, 74**HCT**00),
- if the Device should have additional user-definable attributes,
- if there are equivalent Gates which can be interchanged (Swaplevel),
- how the Gate behaves when added to a schematic (Addlevel),
- the prefix for the component name, if a prefix is used,
- if the value of the component can be changed or if the value should be fixed to the Device name,
- which pins relate to the pads of the Package (CONNECT command)
- whether a description for this component should be stored in the library.

The following diagram shows the fully defined 7400 Device with four NAND gates and a supply gate in various Package and technology versions.

If you click onto one of the gates with the right mouse button, the context menu with the executable commands pops up. Furthermore you can display the *Properties* of the gate. Click on *Edit Symbol* to open the Symbol Editor.



➢ ***Device Editor window***

Create Actual Components from Symbols and Packages

Switch to the Device editing mode by clicking the *Device* icon in the action toolbar. Type in the Device name and confirm the question *Create new device 'devicename'?* with *Yes*.

Use the following commands to create a Device.

ADD

Add a Symbol to a Device. Gate name, Swaplevel, and Addlevel can be defined in the ADD command in the parameter toolbar, or redefined later with the CHANGE command.

The Swaplevel specifies whether there are equivalent Gates.

The Addlevel defines, for instance, if a Gate is to be added to the schematic only on the users request. Example: the power gate of an integrated circuit which is normally not shown on the schematic.

NAME

Change Gate name.

CHANGE

Change Swaplevel or Addlevel.

PACKAGE

Define and name Package variant(s). The PACKAGE command is started by clicking on the *New* button in the Device Editor window, or by typing on the command line. Choose the requested Package variant.

More information about this can be found on page 251.

CONNECT

Define which pins (Gate) relate to which pads (Package).

PREFIX

Provide prefix for the component name in the schematic (e.g. R for resistors).

VALUE

In the Device mode, VALUE is used to specify whether the component value can be freely selected from within the schematic diagram or the layout, or whether it has a fixed specification.

On: The value can be changed from within the schematic (e.g. for resistors). The component is not fully specified until a value has been assigned.

Off: The value corresponds to the Device name, including, when present, assignment of the technology and the Package version (e.g. 74LS00N).

Even if *Value* is *Off*, the value of a component can be changed. A query checks if this action is intended.
The altered value of the component remains unchanged, if the Technology or the Package version is altered later with CHANGE PACKAGE or CHANGE TECHNOLOGY.

TECHNOLOGY

If necessary, various technologies can be defined, for example, for a logic component. Click on *Technologies* therefore.

ATTRIBUTE

Click on *Attributes* to define any additional attribute for the Device. A detailed description can be found in the chapter about libraries in this manual.

DESCRIPTION

Compose a description of the Device which can also be examined by the search function associated with the ADD dialog.

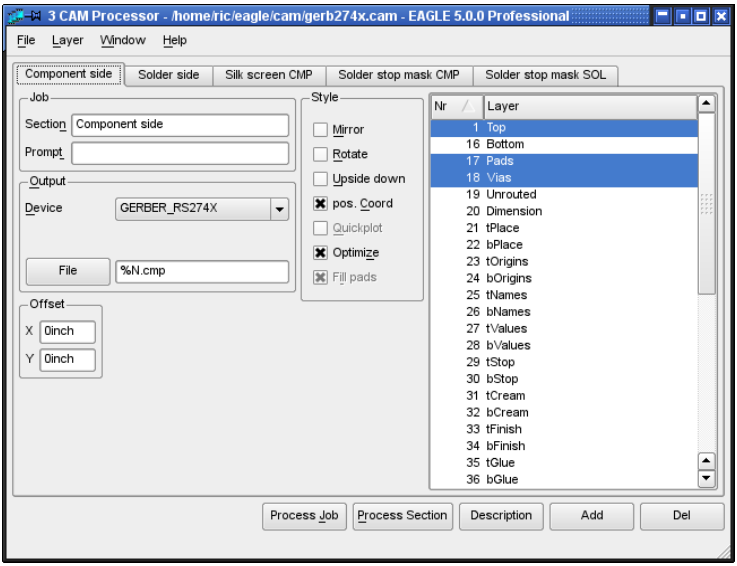
> *Information about Copying of Packages, Symbols and Devices can be found from page 256 on.*

4.5 The CAM Processor

Manufacturing data is generated by means of the CAM Processor. A number of drivers for the data output are available. The drivers are defined in the file *eagle.def*, which can be edited with any Text Editor.

Output to matrix printers, however, is not created with the CAM Processor but with a PRINT command.

The EAGLE license conditions allow you to give the CAM Processor to your board manufacturer. You only need to supply them with a copy of the EAGLE Light Edition. The CAM Processor of EAGLE Light runs without limitation. Alternatively, EAGLE Light can be downloaded from CadSoft's web server.



➢ ***The CAM Processor***

The CAM Processor can also be started directly from the command line. A number of command line parameters can be passed to it when it is called. These are listed in the appendix.

> *Please do not pass on your personal installation code! This is punishable by law!*

Generate Data

Starting the CAM Processor

There are different ways to start the CAM Processor:

You can do this directly from the Layout or Schematic Editor window with the CAM Processor icon in the action toolbar or through the menu *File/CAM Processor*. The current schematic or board will be loaded automatically from the Control Panel by clicking on one of the entries in the Tree View's *CAM Jobs* branch. Then the selected CAM Job will be loaded automatically. You still have to load the schematic or board from which you want to make the CAM Processor manufacturing data from through the *File/Open* menu by using the command prompt (Windows command prompt, Terminal or Console window) without graphical user interface. Particular information can be found in the appendix chapter about *EAGLE Options*.

Load Job File

A job defines the sequence of several output steps in an automatic data creation task. You can, for example, use a job to generate individual files containing the Gerber data for several PCB layers.

A job is loaded with the *File* menu of the CAM Processor or with a double-click on one of the Tree view's *CAM Jobs* entries in the Control Panel.

A job is not absolutely essential for output. All the data can be made step by step manually.

Load Board

Before you can generate an output you must open the *File* menu and load a board file, if not already loaded automatically while stating the CAM Processor from an Editor window. At the bottom left in the CAM Processor window you will see the file name the data is being generated from.

The output of data for a Schematic, for example for a certain plotter, is also possible.

Set Output Parameters

If a job file is loaded, the output parameters are already adjusted. A job can contain several sections with different parameter sets. The various peripheral devices accept different parameters.

If no job is loaded, set the parameters to whatever you need (see page 280).

Start Output

If you want to execute the job which has been loaded, click the *Process Job* button. If you just want to get an output using the currently visible parameter settings, click the *Process Section* button.

Define New Job

Perform the following steps to define a new job:

1. Click *Add*, to add a new section.
2. Set parameters.
3. Repeat 1. and 2. if necessary.
4. Save job with *File/Save job*.

The *Description* button allows to describe the job file. This description will be shown in the Control Panel.

The chapter on *Preparing the Manufacturing Data* contains detailed information on this subject.

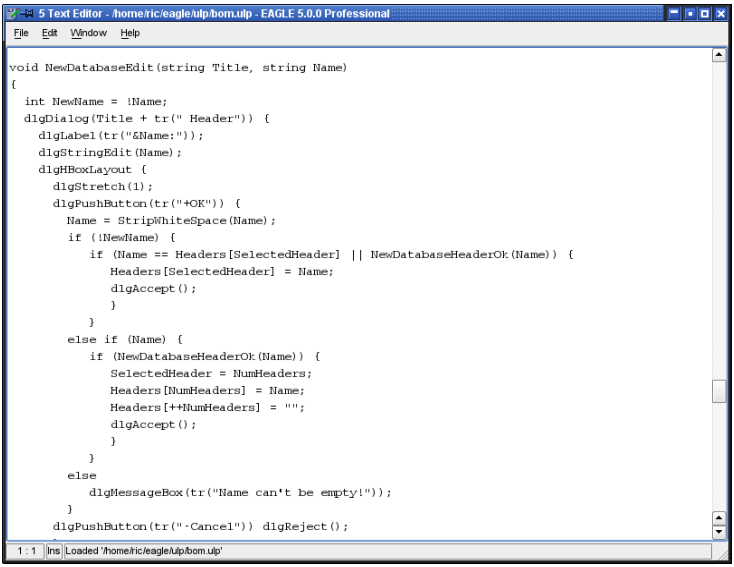
4.6 The Text Editor Window

EAGLE contains a simple Text Editor.

You can use it to edit script files, User Language programs or any other text file. The EAGLE Text Editor stores its files with UTF-8 encoding.

The menus bring you to a variety of functions, such as commands for printing, copying and cutting, searching, replacing (with support of Regular Expressions), changing font and size, and so on.
The keyboard shortcuts in the EAGLE Text Editor follow the platform specific standards.

When in the Text Editor, the right mouse button calls up a context menu.



➢ ***The Text Editor***

In case you prefer an external text editor, define the program call in the Control Panel's or in one of the Editor window's *Option/User interface* menu, *External text editor*. If you want to prevent EAGLE to start any text editor automatically, type in a minus sign '-' in the *External text editor* line. Clear the line for the built-in EAGLE text editor.

Please note further information about the usage of an external text editor in the help function, section *Editor windows/Text Editor*.

Chapter 5
Principles for Working with EAGLE

5.1 Command Input Possibilities

Usually the commands in EAGLE are executed by clicking an icon or an item in the menu bar and then clicking onto the object you want to edit. But there are also alternative to execute commands.

Possibilities for command input in Schematic, Layout, and Library Editor:

- clicking a command icon
- typing text commands in the command line
- through the context menu
- via function keys
- via script files
- via User Language programs

In any case it is necessary to understand the syntax of the EAGLE command language which is described in the following section.
A detailed description of the EAGLE commands can be found on the help pages.

Activate Command and Select Object

The classical way of working with EAGLE is to activate the command first, and then choose the object you want to have it executed on. For example, first activate the MOVE command by clicking the icon in the command menu or selecting the command in one of the menus, and finally click onto the object you want to move.

Command Line

As an alternative to the previously mentioned clicking onto an icon you can use the command line. When entering commands you may abbreviate key words as long as they cannot be mistaken for another key word, or you may use small or capital letters (the input is not case sensitive), for example:

```
CHANGE WIDTH 0.024
```

is equivalent to

```
cha wi 0.024
```

The actual unit for the values is set in the GRID menu. It's also possible to specify the unit directly in the command line without changing the currently set grid:

```
CHANGE WIDTH 0.6MM
```

or

```
cha wid 24mil
```

Most commands can be executed whilst declaring coordinate values in the command line.

Examples:

```
MOVE IC1>VALUE (2.50 1.75) ;
```

The value placeholder text for part IC1 moves to position 2.50 1.75 in the layout, provided it has been released with the SMASH command before.

```
MIRROR U1;
```

Part U1 will be mirrored to the bottom side of the board.

```
HOLE 0.15 (5 8.5) ;
```

Place a hole with drill diameter 0.15 at position 5 8.5.

```
VIA 'GND' 0.070 round (2.0 3.0) ;
```

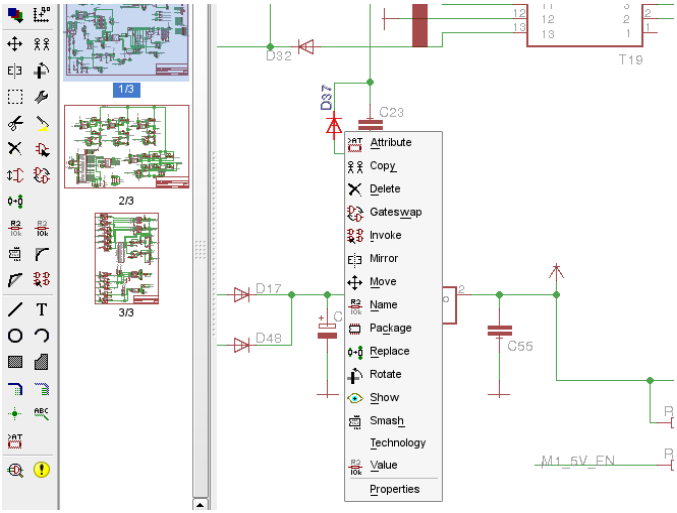
A round shaped via with a diameter of 0.070 belonging to signal GND will be placed at position 2.0 3.0.

History Function

You can recall the most recently entered commands by pressing Crsr-Up (↑) or Crsr-Down (↓) and edit them. The *Esc* key deletes the contents of the command line.

The Context Menu

Another way of using EAGLE is to work with the object-specific command menu. In this case you first click with the right mouse button onto the object and then you select the command that you want to have executed.
The context menu contains all commands that can be executed with the selected object. Additionally you can display all the object's properties by clicking onto the *Properties* entry. Some of them can be even changed directly in the *Properties* window.



➢ *The context menu for a Device in the Schematic*

Function Keys

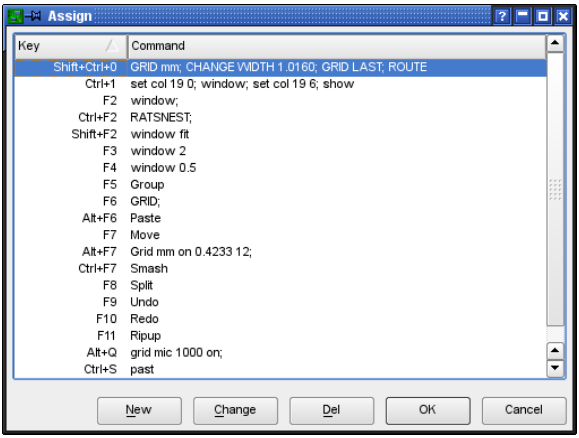
Texts may be allocated to the function keys and to combinations of those keys with *Alt, Ctrl* and *Shift* (for Mac OS-X additionally *Cmd),* if not occupied by the operating system or a Linux Window Manager (for example *F1* for help). If a function key is pressed, this corresponds to the text being typed in via the keyboard. Since every command is capable of being entered as text, every command, together with certain parameters, can be assigned to a function key. Even whole sequences of commands can be assigned to a function key in this way.

The command

```
ASSIGN
```

displays the current function key assignments. Changes to the key assignments can be carried out in the assign window.
The *New* button can be used to define a new key assignment. A click onto *Del* will delete a marked entry, while *Change* alters an existing definition. *OK* closes the dialog and saves the definitions, while *Cancel* aborts the dialog.

These settings can also be made via the *Options/Assign* menu in the Schematic or Layout Editor.



➢ ***The dialog for the ASSIGN command***

To predefine certain assignments you can also use the ASSIGN command in the file *eagle.scr* (see page 105).

Examples:

The combination of *Ctrl* + *Shift* + *G* displays a grid of 0.127mm:

```
ASSIGN CS+G 'GRID MM 0.127 ON;';
```

The combination of *Alt* + *F6* changes the layer to Top and starts the ROUTE command:

```
ASSIGN A+F6 'LAYER TOP; ROUTE';
```

The combination of the keys *Alt* + *R* displays only the layers Top, Pads, Vias and Dimension first and then starts the print out with the default printer:

```
ASSIGN A+R 'DISPLAY NONE 1 17 18 20; PRINT;';
```

`A`, `C`, `M`, and `S` are the modifiers for the *Alt*, *Ctrl*, *Cmd* (Mac OS-X only), and *Shift* key.

The combination of *Alt* + *0* brings the Control Panel into the foreground. The combinations *Alt* + *1* up to *9* are assigned to the various editor windows, according to the window number which is shown in the respective title bar.

Script Files

Script files are a powerful tool. They can contain long sequences of commands, such as the specification of specific colors and fill-patterns for all layers, as for example in *defaultcolors.scr*. On the other hand they might contain netlists converted from the data of other programs.

The SCRIPT command is used to execute script files.

Many User Language programs (ULP) create script files that can be read in order to modify a layout or a schematic.

EAGLE outputs an entire library, for instance, as a script file with the aid of the EXPORT command (*Netscript* option). This file can be modified with a text editor, after which it can be read in again. This allows changes to be made to a library quite easily.

There is more information about script files and export commands later in this chapter.

Mixed Input

The various methods of giving commands can be mixed together.

You can, for instance, click the icon for the CIRCLE command (which corresponds to typing `CIRCLE` on the command line), and then type the coordinates of the center of the circle and of a point on the circumference in this form

```
(2 2) (2 3) ←
```

in the command line.

The values used above would, if the unit is currently set to *inch*, result in a circle with a radius of one inch centered on the coordinate (2 2). It is irrelevant whether the CIRCLE command is entered by icon or by typing on the command line.

> *Some EAGLE commands are used in combination with the Shift, Alt or Ctrl keys. In case you are working with EAGLE for Mac OS-X, please use the Cmd key instead of Ctrl.*

5.2 The EAGLE Command Language

You only need a knowledge of the EAGLE command language if you want to make use of the alternative input methods discussed in the previous section.

The syntax of the EAGLE command language will be discussed in this section, and typographical conventions, which are important for understanding the descriptions, will be specified.

Typographical Conventions

Enter key and Semicolon

If EAGLE commands are entered via the command line they are finished with the *Enter* key. In some cases a command must have a semicolon at the end, so that EAGLE knows that there are no more parameters. It is a good idea to close all commands in a script file with a semicolon.

The use of the *Enter* key is symbolized at many places within this handbook with the ← sign.

However in the following examples neither the *Enter* key sign nor the semicolon are shown, since all of these commands can be used both on the command line and within script files.

Bold Type or Upper Case

Commands and parameters shown here in UPPER CASE are entered directly. When they are entered, there is no distinction made between upper and lower case. For example:

Syntax:

```
GRID LINES
```

Input:

`GRID LINES` or `grid lines`

Lower Case

Parameters shown here in lower case are to be replaced by names, numbers or keywords. For example:

Syntax:

```
GRID grid_size grid_multiple
```

Input:

```
GRID 1 10
```

This sets the grid to 1 mm (assuming that the current unit is set to mm). Every tenth grid line is visible. The figures 1 and 10 are placed into the command instead of the placeholders *grid_size* and *grid_multiple*.

Underscore

In the names of parameters and keywords the underscore sign is often used in the interests of a clearer representation. Please do not confuse it with an empty space. As can be seen in the example above, *grid_size* is a single parameter, as is *grid_multiple*.

If a keyword contains an underscore sign, such as COLOR_LAYER does in the command

```
SET COLOR_LAYER layer_name color_word
```

then the character is to be typed in just like any other. For example:

```
SET COLOR_LAYER BOTTOM BLUE
```

Spaces

Wherever a space is permissible, any number of spaces can be used.

Alternative Parameters

The | character means that the parameters are alternatives. For example:

Syntax:

```
SET BEEP ON | OFF
```

Input:

```
SET BEEP ON
```

or

```
SET BEEP OFF
```

The beep, which is triggered by certain actions, is switched on or off.

Repetition Points

The .. characters mean either that the function can be executed multiple times, or that multiple parameters of the same type are allowed. For example:

Syntax:

```
DISPLAY option layer_name..
```

Input:

```
DISPLAY TOP PINS VIAS
```

The layer number can alternatively be used:

```
DISPLAY 1 17 18
```

More than one layer is made visible here.

If a layer (in this case Bottom) is to be hidden:

```
DISPLAY -16
```

Mouse Click

The following sign ● usually means that at this point in the command an object is to be clicked with the left mouse button.

For example:

```
MOVE ● ●
```

Input:

```
MOVE ← (or click the icon)
Mouse click on the first object to be moved
Mouse click on the destination
Mouse click on the second object to be moved
and so on.
```

You can also see from these examples how the repetition points are to be understood in the context of mouse clicks.

Entering Coordinates as Text

The program sees every mouse click as a pair of coordinates. If it is desired to enter commands in text form on the command line, then instead of clicking with the mouse it is possible to enter the coordinates through the keyboard in the following form:

(x y)

where x and y are numbers representing units as selected by the GRID command. The textual input method is necessary in particular for script files.

The coordinates of the current cursor position can be fetched with `(@)`. For example:

```
WINDOW (@);
```

Examples of coordinate entry in text form:

You want to enter the outline of a circuit board with precise dimensions.

```
GRID MM 1;
LAYER DIMENSION;
WIRE 0 (0 0) (160 0) (160 100) (0 100) (0 0);
GRID LAST;
```

The first step is to switch to a 1 mm grid. The dimension layer is then activated. The WIRE command then first sets the line width to 0 and draws a rectangle with the aid of the four given coordinates. The last command returns the grid to whatever had previously been selected, since circuit boards are usually designed using inches.

Relative values:

It is possible to use relative coordinate values in the form (R x y) which refer to a reference point set with the MARK command before. If you don't set a reference point the absolute origin of the coordinate system will be taken.

Setting a via relative to the reference point:

```
GRID MM 0.5;
MARK (20 10);
VIA (R 5 12.5);
MARK;
```

First the grid is set to Millimetres, then the reference point at the position (20 10) is placed. The via is located at a distance of 5 mm in x and 12.5 mm in y direction from the this point. Then the reference point is removed.

Polar values:

Polar coordinates are given in the form of (P radius angle).

```
GRID MM;
MARK (12.5 7.125);
LAYER 21;
CIRCLE (R 0 0) (R 0 40);
PAD (P 40 0);
PAD (P 40 120);
PAD (P 40 240);
```

This examples shows how to set the reference point at position (12.5 7.125). Then a circle with a radius of 40 mm is drawn in layer 21, tPlace. Three pads are placed on the circumference with an angle of 120°.
Here the circle is easily drawn with the help of relative coordinates. So we do not have to worry about absolute values of a point on the circumference for the second coordinates pair to determine the circle.

Right Mouse Click:

The > character within parenthesis represents a right mouse click. That way one can move a whole group, for example:

```
MOVE (> 0 0) (10 0);
```

The previously selected group will be moved 10 units in x direction.

Modifier:

Within parenthesis one can use some modifiers. For combinations you don't have to care about the order:

- `A` represents the pressed *Alt* key, the alternative grid
- `C` represents the pressed *Ctrl* key, Mac OS-X: *Cmd*
- `S` represents the pressed *Shift* key
- `R` relative coordinates
- `P` polar coordinates
- `>` right mouse click

C and S cause miscellaneous commands to behave in different manners. More information can be found in the help function of the respective command.

If the commands are being read from a script file, each one must be closed with a semicolon. In the above cases the semicolons can be omitted if the commands are being entered via the keyboard and each is being closed with the *Enter* key.

Examples:
A component is to be transposed to a specified position.

```
GRID MM 1;
MOVE IC1 (120 25) ;
```

Alternatively you can use the object's coordinates:

```
MOVE (0.127 2.54) (120 25);
```

IC1 is located at coordinates (0.127 2.54) and is moved to position (120 25). The current position of a Device can be obtained with the aid of the INFO command.

```
INFO IC1
```

When a Symbol is defined, a pin is placed at a certain position.

```
PIN 'GND' PWR NONE SHORT R180 (0.2 0.4) ;
```

You draw a rectangular forbidden area in layer 41 *tRestrict*:

```
LAYER TRESTRICT;
RECT (0.5 0.5) (2.5 4) ;
```

5.3 Grids and the Current Units

EAGLE performs its internal calculations using a basic grid size of 1/10 000 mm (0.1 micron). Any multiple of that can be set as the working grid.

Microns, mils, inches and mm can be used for the unit. The current unit as set with the GRID command applies to all values.

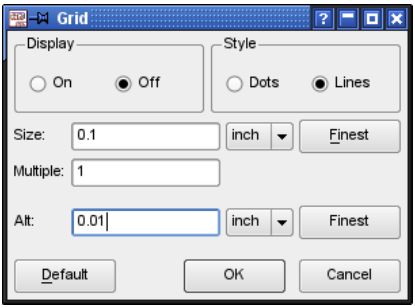
> *You should always use the pre-set 0.1 inch grid for schematic diagrams! This grid should also be used when defining Symbols in the Library Editor!*

When starting the design of circuit boards or libraries it pays to give prior thought to the question of which grid size (or sizes) will be used as a basis. For example, it is only the origin of a Package that will be pulled onto the board's placement grid. All other objects constituting the Package (such as pads) are placed relative to that point on the board, just as it was defined in the library.

> *The basic rule for boards is: always make the grid as big as possible and as small as necessary.*

Various grid sizes can be pre-set in the *eagle.scr* file for different types of editor windows (see page 99).

The current grid *Size* is set in the grid menu. The units chosen in the combo box are used.
The *Multiple* option indicates how many grid lines are displayed. If, for instance, the value 5 is entered at *Multiple*, every fifth line will be displayed.



➢ ***The Grid menu***

The *Alt* line allows to set an alternative grid which can be activated by pressing the *Alt* key (while, for example, MOVE, ROUTE, ADD, or WIRE is active). This can be very useful for placing parts in a dense layout or arranging labels in the schematic. If you decide not to place it in the alternative grid and release the *Alt* key before placing it, the object stays in its origin grid.

Style specifies the way it is displayed: *Lines* or *Dots*.
The options *On* and *Off* under *Display* switch the grid display on or off.

Finest sets the finest grid that is possible.
Clicking on *default* will select the editor's standard grid.

Beginning with a certain zooming limit, grid lines are not displayed anymore. This limit can be set in the menu *Options/Set/Misc, Min. visible grid size*.

Grid lines and grid dots can have any color. Click the colored button of the respective palette (depends on the background color) in the menu *Options/Set/Colors* and select the color as requested. This can also be done in the command line, for example:

```
SET COLOR_GRID BLUE
```

Instead of the color name the color number can be given, as well. It can be in the range 0 .. 63. The shown color depends on the (self-)defined colors of the current palette.

See also the hints concerning *Color settings* on page 100.

5.4 Aliases for DISPLAY, GRID, and WINDOW

For the commands DISPLAY, GRID, and WINDOW you can define so-called aliases. This is a set of parameters which you can save with any name and executed it with the command. To access such an alias simply click with the right mouse button onto the command icon.

The aliases are stored in the *eaglerc* file for Schematic, Layout, and Library separately. They are available for all Schematics, Layout, and Libraries then.

Example: DISPLAY Alias

- Display the layers you want to see in the Layout Editor with the DISPLAY command, for example *Top*, *Pads*, *Vias*, and *Dimension*
- Right-click onto the DISPLAY icon and a popup menu appears
- Select the *New..* entry
- Enter the name of the alias, for example *Top_view*
- Click the *OK* button

From now on the popup menu of the DISPLAY icon contains the entry Top view.

If you prefer the command line for activating this alias you have to enter:

`DISPLAY TOP_VIEW` or `disp top_v`

It does not matter if you write in upper or lower case letters hers. You may use abbreviations as long as the name is clear.

There are no limitation to the number of aliases used.

Use `DISPLAY LAST` in the command line or the entry *Last* of the DISPLAY's popup menu icon to return to the last layer selection.

More details can be found on the help page of the DISPLAY command.

Example: GRID Alias

The how to and the function of a grid alias is exactly the same as it is explained for the DISPLAY command. Set the appropriate grid in one of the Editor windows, right-click onto the GRID icon , and select the *New..* entry in the popup menu to define the alias.

This can be done in the command line as well. for the grid command it could look like this:

```
GRID = My_Grid inch 0.005 lines on
```

The command

`grid my_grid` or in short `gri my`

executes the alias. The command is case insensitive, the alias can be abbreviated.

Example: WINDOW Alias

The WINDOW command allows you to define an alias for a certain part of the drawing area. Aliases help you to navigate comfortably from one location to another in your drawing. The definition of a WINDOW alias is similar to the DISPLAY alias as described above:

- Select the appropriate display window in the drawing

- Right-click onto the *Select* icon of the WINDOW command to open the popup menu
- Click the *New..* entry now and name your alias

Let's assume the alias name is *upper_left:* You can restore this display detail, for example, in the command line with:

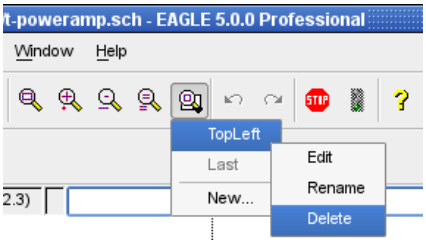
`WINDOW Upper_Left` or in short `win upper_l`

Alternatively right-click onto the *Select* icon of the WINDOW command and select the entry *upper_left* in the popup menu.

> *In a Schematic that consists of more than one sheet an alias is executed always on the currently active sheet, independent of where it was defined originally.*

Editing, Renaming, Deleting of an Alias

In the case you want to delete an alias, you can do this in the command icon's popup menu. First right-click onto the command icon to open the popup menu. Then use a right mouse click onto the alias entry. This opens a context menu. Click the *Delete* option there.



➢ ***Deleting a WINDOW alias***

The same methods can be used to *Rename* or *Edit* an alias.

These actions can be executed also via the command line. Further information can be found in the help pages about the DISPLAY, GRID, and WINDOW commands.

5.5 Names and Automatic Naming

Length

Names in EAGLE can have any desired length. There is no limit.

Forbidden and Special Characters

No names may contain spaces, semicolons or umlauts. Quotation marks and other exotic characters (above 127 in the ASCII table) should be avoided as far as possible.

Device names must not contain either question marks or asterisks, since these characters are used as placeholders for Package variants (?) and technologies (*).

Commas must be avoided in pad names.

Part-bus names must not contain colons, commas or square brackets.

The exclamation mark is a special character that starts and ends a bar over the text. See the help function for the TEXT command for details. If an exclamation mark should be visible in the text, it needs to be escaped by a leading backslash.

In order to have a backslash displayed in a name or text, you have to type it, for example with the NAME or TEXT command, twice.

Automatic Naming

If a name is given together with one of the commands PIN, PAD, SMD, NET, BUS or ADD, then other names will be derived from it as long as the command is still active.

The name is simply typed into the command line before placing the object (while it is attached to the mouse). Note that the name must be placed within simple quotation marks. Entry is completed with the *Enter* key (←) .

The examples illustrate how automatic naming functions:

```
ADD DIL14 'U1' ← •••
```

fetches three DIL14 Packages to the board and names them U1, U2 and U3 (corresponds to a mouse click).

```
PAD OCT '1' ← ••••
```

places four octagonal pads with the names 1, 2, 3, and 4.

If the name consists of only one character from A...Z, then the following objects receive the following letters of the alphabet as names:

```
ADD NAND 'A' ← ••••
```

fetches four NAND gates with the names A, B, C and D. If the generated name reaches Z, then names with the default prefix will again be generated (e.g. G$1).

5.6 Import and Export of Data

EAGLE provides a number of tools for data exchange.

- Script files for importing
- The export command for exporting

- EAGLE User Language programs for import and export.

The User Language is very flexible, but does call for a suitable program to be created. You will find further details in the section on *The EAGLE User Language*.

Script Files and Data Import

The SCRIPT command makes a universal tool available to the EAGLE user for data import.

Since every EAGLE operation can be carried out with the aid of text commands, you can import all types of data with the aid of a script file. A script file can in turn call other script files.

Script files can be created with a simple text editor. The prerequisite for the development of your own script files is that you understand the EAGLE command language. You will find the precise functioning and the syntax of the individual commands in the EAGLE help pages.

The file *euro.scr* in the *eagle/scr* directory, which draws the outline of a eurocard with corner limits, provides a simple example.

If a netlist is to be imported into a board design which already contains the appropriate components, then a script file of the following form is necessary:

```
SIGNAL GND IC1 7 IC2 7 J4 22;
SIGNAL VCC IC1 14 IC2 14 J4 1;
```

A *Netscript* of this sort can easily be created from the schematic diagram by the EXPORT command.

You will get a further impression of the power of importing, if you output a library with the EXPORT command (*Script* option). The script file that is generated provides an instructive example for the syntax of the script language. It can be examined with any text editor. If SCRIPT is then used to read this file into an empty library, a new library file will be created.

Comments can be included following a #-character.

The execution of a script file can be stopped by clicking the Stop icon in the action toolbar.

File Export Using the EXPORT Command

The EXPORT command has the following modes:

DIRECTORY

Outputs a list of the contents (Devices, Symbols, and Packages) of the currently loaded library.

NETLIST

Outputs a netlist for the currently loaded schematic or board in an EAGLE-specific format. It can be used to check the connections in a drawing.

There are also available several User Language programs that allow to export various net list formats. Please take a look into CadSoft's Download area in the web: http://www.cadsoftusa.com/download.htm.

NETSCRIPT

Outputs a netlist of the currently loaded schematic in the form of a script file. The netscript can be imported into the board file with the help of the SCRIPT command. This could be possibly suggestive if there are differences in the netlist between schematic and layout.
In the first step you have to delete all signals in the layout with the command DELETE SIGNALS. Be aware that all traces are lost! Now export the Netscript from the Schematic and import it with the SCRIPT command into the layout. The result is a Schematic/Layout file pair with an identical netlist.

PARTLIST

Outputs a component list for the schematic or board.

PINLIST

Outputs a pin/pad list for the schematic or board, listing the connected nets.

SCRIPT

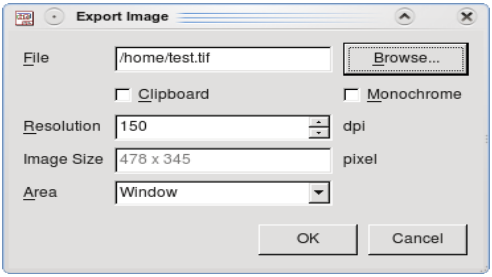
Outputs the currently loaded library in the form of a script file.
This script can be modified with a text editor in order to generate, for example, a user defined library, or to copy parts of one library into another. The modified script file can be imported into a new or an already existing library with the help of the SCRIPT command.
The script file also serves as a good example for the EAGLE command syntax.
In order to avoid loss of precision the grid unit in the script file is set to Millimetres.

IMAGE

The option *Image* allows you to generate files in various graphic formats.

The following formats are available:

bmp	Windows Bitmap file
png	Portable Network Graphics file
pbm	Portable Bitmap file
pgm	Portable Grayscale Bitmap file
ppm	Portable Pixelmap file
tif	Tag Image file
xbm	X Bitmap file
xpm	X Pixmap file



➢ ***Settings for graphic file output***

Click the Browse button, select the output path, and type in the graphic file name with its extension. The file extension determines the graphic file type.

To generate a black and white image activate the option *Monochrome*. To make the image available via the system's clipboard set the *Clipboard* option.

The *Resolution* can be set in dots per inch. The resulting *Image Size* will be shown in the lowest field.

The *Area* field allows a selection of *Full* or *Window*. *Full* prints the whole drawing, whereas *Window* prints the currently in the Editor window visible part of the drawing.

> *Further graphic formats, like HPGL, Postscript (PS), or Encapsulated Postscript (EPS), can be generated with the help of the CAM Processor. The User Language Program dxf.ulp generates Dxf data. The PRINT command supports PDF output.*

5.7 The EAGLE User Language

EAGLE contains an interpreter for a C-like User Language. It can be used to access any EAGLE file. Since version 4 it has also been able to access external data. It is possible, with very few restrictions, to export data from EAGLE, and import a wide range of data into EAGLE.

ULPs can, for example, manipulate a layout file or a library by generating and executing a Script file. The Script file contains all the necessary commands for the manipulation. The User Language's integrated exit() function allows it to execute these commands directly.

The program examples included (**.ulp*) will provide some insight into the capacity of the User Language. They are located in your installation's ULP directory. A description of the way in which a ULP works is located in the file header. This is also displayed in the Control Panel or in the usage box when the program is called. User Language programs must be written in a text editor that does not add any control codes.

A ULP is started with the RUN command, or by dragging a ULP from the Control Panel into an editor window (Drag&Drop). To cancel the execution of an ULP click the Stop icon.

EAGLE prompts a message in the status bar, *Run: finished,* if the User Language program has been ended.

The language is described in detail in the EAGLE help pages, under the keyword *User Language*.

Typical applications for ULPs:

- Creating parts lists in various formats.
  See also page 268.
- Output in graphical formats.
- Data output for component insertion machines, in-circuit testers etc.
- Linking to an external database.
- Manipulation of the silk screen print, the solder stop mask, and so on.
- Import of graphic data files (for example *import-bmp.ulp* for logos or the like)

A lot of valuable ULPs can be found on our web pages. Simply take a look at http://www.cadsoftusa.com/download.htm.

5.8 Forward&Back Annotation

A schematic file and the associated board file are logically linked by automatic Forward&Back Annotation. This ensures that the schematic and the board are always consistent.

As soon as a layout is created with the BOARD command , the two files are consistent. Every action performed on the schematic diagram is simultaneously executed in the layout. If, for instance, you place a new Device, the associated housing will appear on the layout at the edge of the board. If a net is placed, the signal lines are simultaneously drawn in the layout. Certain operations such as the placement or deletion of signals are only allowed in the schematic. The Layout Editor does not permit these actions, and issues an appropriate warning. Renaming Devices or changing their values, for example, are permitted in both files.

The EAGLE help pages contain a closer description of the technical details.

It is not necessary for you, as the user, to pay any further attention to this mechanism. You only have to ensure that you do not work on a schematic whose associated board file has been closed, and vice versa. This means that both files must always be loaded at the same time. Otherwise they loose consistency, and the annotation can no longer work.

If you have, however, once edited the board or the schematic separately, the Electrical Rule Check (ERC) will check the files for consistency when they are loaded. If inconsistencies are found, the ERC opens an Error window with appropriate messages about the Schematic and the Layout. Section 6.9, starting with page 175, shows how to proceed in such a case.

5.9 Configuring EAGLE Individually

There are a number of settings that permit the program to be adjusted for individual needs. We distinguish between program, user and project-specific settings.

Basic program settings that will apply to every user and every new project are made in the ***eagle.scr*** file. Under Windows, personal preferences are stored in the file ***eaglerc.usr***, or, under Linux, in ***~/.eaglerc***.
EAGLE remembers settings that only apply to one particular project in the ***eagle.epf*** project file.

Values that, for instance, only apply to one specific board, such as the Design Rules, special layer colors, unique newly defined layers or the grid setting are stored directly in the layout file. This also applies, of course, to schematic diagram and library files.

Configuration Commands

Most of the options are usually set by means of the *Options* menus of the individual EAGLE editor windows.

The Control Panel allows settings to be made for *Directories*, file *Backup* and the appearance of the editor window (*User interface*). These options are described in the chapter on the *Control Panel* under the
Options menu heading, starting on page 43.

Through the *User interface* settings it is possible to select the icon-based menu or a configurable text menu.
The MENU command allows the text menu to be given a hierarchical configuration by means of a script file. There is an example of this in the appendix.

The *Options* menu in the editor windows for schematic diagrams, layouts and libraries contains, in addition to the *User interface* item, two further entries:
Assign and *Set*.

The ASSIGN command alters and displays the assignment of the function keys. You will find information about this on page 83.

General system parameters are altered with the SET command.

The CHANGE command allows a variety of initial settings for object properties.

The GRID command sets the grid size and the current unit. Further information about this starts on page 90.

The Menu Options/Set (SET Command)

Most common options of the SET command are available in the *Settings* window of the menu *Options/Set*. This window can be reached also by entering on the command line:

```
SET
```

Display Certain Layers Only

The number of available layers shown in the DISPLAY or LAYER menu can be set with the option *Used_Layers*. That way it is possible to hide unused layers for clarity reasons.

```
SET USED_LAYERS 1 16 17 18 19 20 21 23 25 27 29 31
44 45 51;
```

stored in the file *eagle.scr* shows only the mentioned layers. After

```
SET USED_LAYERS ALL;
```

all layers are available again.

Contents of The Parameter Menus

The parameter menus for *Width, Diameter, Drill, SMD, Size, Isolate, Spacing,* and *Miter,* which are available for example through the CHANGE command, can be configured and filled with any values by the SET command. Simply list the values, separated by blanks, in the command line.

Example for the *Miter* menu:

```
SET MITER_MENU 0.1 0.2 0.3 0.4 0.5 0.6 1 1.5 2 3 4;
```

The units of the given values are determined by the currently used GRID in the Editor window. A maximum number of 16 entries is allowed.

Example for the *SMD* menu:

```
SET SMD_MENU 1.2mm 2.0mm 0.5mm 0.9mm 0.1in 0.14in;
```

For each entry of the three value pairs the unit is given directly. A maximum number of 16 value pairs is allowed.

The values in the menus are always shown in the currently selected GRID unit.

Write the SET command in the file *eagle.scr* to make it default for all you projects.

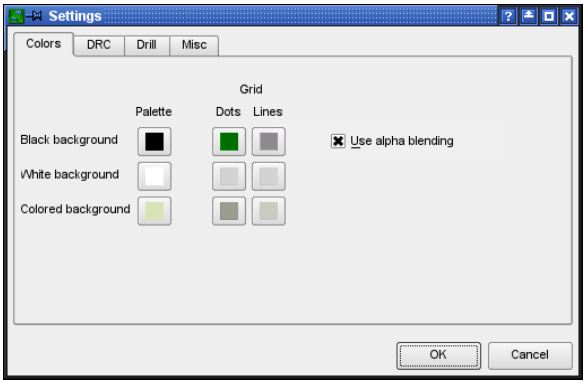
To return to the EAGLE default settings use for example for the *Width* menu:

```
SET WIDTH_MENU ;
```

Color Settings

The *Colors* tab contains settings for layer and background colors and colors for grid lines or dots.

Three color palettes are available: for black, white and colored background. Each palette allows a maximum of 64 color entries, which can be given any value for the Alpha channel and any RGB value.



➢ ***Settings window: Color settings***

If you prefer the old raster OP behaviour of previous EAGLE versions on black background, deactivate the *Use alpha blending* check box. In this case the alpha value is ignored when using a black background. Colors are mixed now using an OR function.

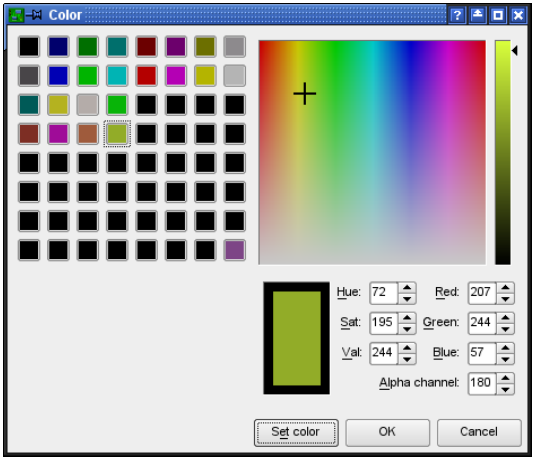
By default EAGLE uses the first 16 values (entries 0..15). Eight colors followed by further eight so-called highlight colors.
The first entry of the palette determines the background color. In the white palette, however, it is not possible to change the background color because it's needed for print-outs, which normally are made on white paper.

The image above shows three buttons in the *Palette* column. Click on one of them. For example, the button for *Colored* Background. The *Color* window opens now.

On the left an 8 x 8 matrix is visible that contains already 16 pre-defined colors. There are alternating eight 'normal' colors with their corresponding eight highlight colors. A color of the palette at position x can be given the corresponding highlight color at position x+8.

In order to define new values select an unused box of the matrix and adjust the new color with the help of the color selection area and the saturation bar on the right. Click *Set Color* to apply your color. Now select a new color box in the matrix and repeat the procedure for the next color.
You may also enter values for *Red*, *Green*, *Blue* or *Hue*, *Sat*, *Val* and *Alpha channel* directly.

Alpha channel determines the transparency of the color. The value 0 means the color is totally transparent (invisible), the maximum value 255 stands for non-transparent. For printouts the value of the alpha channel is set to 255 for each color.



➢ ***Color window: Defining colors***

The image above shows four further pairs of colors (normal and corresponding highlight color vertically arranged). In order to change the color palette for an editor window select the appropriate *Background* in the menu *Options/User Interface*.

> *You should always define at least one pair of colors: a normal color and its related highlight color.*

Alternatively, the color definition and change of palette can be made in a script file or in the command line.

```
SET PALETTE <index> <αrgb>
```

defines a color for the currently used palette, where the value for the alpha channel and the color value has to be given hexadecimal. Index stands for the color number, αrgb for the values for alpha channel, the colors red, blue, and green. Example:

```
SET PALETTE 16 0xB4FFFF0
```

sets the color number 16 to yellow, which corresponds to the decimal RGB value 255 255 0 which is hexadecimal `FF FF 00`. The first byte `B4` determines the value of the alpha channel (decimal 180).
Hexadecimal values are marked by a leading `0x`.

To activate the black color palette type in the command line:

```
SET PALETTE BLACK
```

The new palette will become visible after refreshing the drawing area with the WINDOW command.

The color assignment for layers is done with the DISPLAY command or with SET COLOR_LAYER.

```
SET COLOR_LAYER 16 4
```

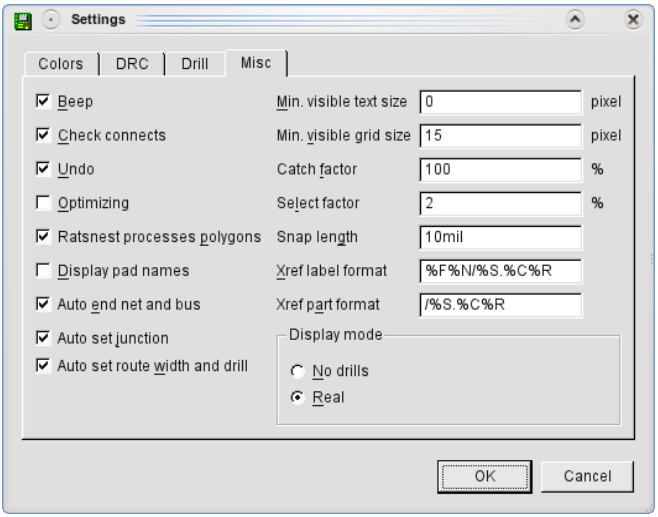
defines, for example, the color number 4 for layer 16.

More details about the syntax can be found in the SET command's help.

> *If you prefer to use the default color values again, start the script file defaultcolors.scr*

Miscellaneous SET Options

The *Misc* tab of the *Settings* window contains the most common options, which are switched on or off by check boxes. Some options allow entering values.



➢ ***Settings at Options/Set/Misc***

Options overview:

Beep:
Switches on/off the confirmation beep. Default: on.

Check connects:
Activates the package check while placing parts in the schematic.
Default: on.

Undo:
Switches on/off the undo/redo buffer. Default: on.

Optimizing:
Enables the automatic removal of bends in straight lines. Default: on.

Ratsnest processes polygons:
The contents of polygons will be calculated with the RATSNEST command. Default: on.

Display pad names:
Pad names are displayed in the Layout or Package Editor. Default: off.

Auto end net and bus:
If placing a net on a pin or a bus the net drops from the mouse cursor.
Default: on.

Auto set junction:
Ending a net on another net a junction will be set automatically.
Default: on.

Auto set route width and drill:
If this option is active, the Follow-me-Router uses the values for wire width and via drill diameter given by the Design Rules or the net classes for the tracks. These values will be set automatically as soon as you are clicking onto a signal wire.
If this option is switched off, EAGLE will take the value you have set with, for example, the previous CHANGE WIDTH command.

Min. visible text size:
Only texts with the given minimum size are displayed.
Default: 3 pixels.

Min. visible grid size:
Grid lines/dots which are closer than the given minimum distance are no longer displayed on the screen. Default: 5 pixels.

Catch factor:
Within this radius a mouse click can reach objects. Set the value to 0 in order to switch this limitation off. So you can reach even objects that are placed far beyond the area of the currently displayed window. Default: 5% of the height of the current display window.

Select factor:
Within this radius (given in % of the height of the current drawing window) EAGLE offers objects for selection. Default: 2%.

Snap length:
Defines the radius of the magnetic-pads function of pads and SMDs.
If you are laying tracks with the ROUTE command and approach a pad or a SMD beyond the given value – that is to say the dynamically calculated airwire becomes shorter than the given radius – the wire will be snapped to the pads/SMDs center. Default value: 20 mil.

Display mode:
Pads/vias are shown with a drill hole (*real*) or without it (*No drills*).
Default: real.

All SET options can be used in the command line. Entering

`SET POLYGON_RATSNEST OFF` or, in short `SET POLY OFF`

for instance switches off polygon calculation for the RATSNEST command.

The help system provides you with more instructions about the SET command.

The eagle.scr File

The script file ***eagle.scr*** is automatically executed when an editor window is opened or when a new schematic diagram, board or library file is created, unless a project file exists.

It is first looked for in the current project directory. If no file of this name exists there, it is looked for in the directory that is entered in the *Script* box in the *Options/Directories* dialog.

This file can contain all those commands that are to be carried out whenever an editor window (other than the Text Editor) is opened.

The *SCH, BRD* and *LBR* labels indicate those segments within the file which are only to be executed if the Schematic, Layout or Library Editor window is opened.

The *DEV, SYM* and *PAC* labels indicate those segments within the file which are only to be executed if the Device, Symbol or Package editor mode is activated.

Commands which are defined before the first label (normally *BRD:*) are valid for all Editor windows.

If, because of the specifications in a project file, EAGLE opens one or more editor windows when it starts, it is necessary to close these and to reopen them so that the settings in ***eagle.scr*** are adopted. It is, as an alternative, possible simply to read the file ***eagle.scr*** through the SCRIPT command.

Comments can be included in a script file by preceding them with #.

Example of an eagle.scr file:

```
# This file can be used to configure the editor windows.

Assign A+F3 'Window 4;';
Assign A+F4 'Window 0.25;';
Assign A+F7 'Grid mm;';
Assign A+F8 'Grid inch;';

BRD:
#Menu Add Change Copy Delete Display Grid Group Move \
#Name Quit Rect Route Script Show Signal Split  \
#Text Value Via Window ';' Wire Write Edit;

Grid inch 0.05 on;
Grid alt inch 0.01;
Set Pad_names on;
Set Width_menu 0.008 0.01 0.016;
Set Drill_menu 0.024 0.032 0.040;
Set Size_menu 0.05 0.07 0.12;
Set Used_layers 1 16 17 18 19 20 21 22 23 24 25 26 \
27 28 39 40 41 42 43 44 45;
```

```
Change width 0.01;
Change drill 0.024;
Change size 0.07;

SCH:
Grid Default;
Change Width 0.006;
#Menu Add Bus Change Copy Delete Display Gateswap \
#Grid Group Invoke Junction Label Move Name Net \
#Pinswap Quit Script Show Split Value Window ';' \
#Wire Write Edit;

LBR:
#Menu Close Export Open Script Write ';' Edit;

DEV:
Grid Default;
#Menu Add Change Copy Connect Delete Display Export \
#Grid Move Name Package Prefix Quit Script Show \
#Value Window ';' Write Edit;

SYM:
Display all;
Grid Default On;
Change Width 0.010;
#Menu Arc Change Copy Cut Delete Display Export \
#Grid Group Move Name Paste Pin Quit Script \
#Show Split Text Value Window ';' Wire Write Edit;

PAC:
Grid Default On;
Grid Alt inch 0.005;
Change Width 0.005;
Change Size 0.050;
Change Smd 0.039 0.039;
#Menu Add Change Copy Delete Display Grid Group \
#Move Name Pad Quit Script Show Smd Split Text \
#Window ';' Wire Write Edit;
```

The eaglerc File

Under Windows, user-specific data is stored in the file ***eaglerc.usr***, or, under Linux and Mac, in ***~/.eaglerc***. This file is stored in the user's home directory. If there is no *home* environment variable set, the following Windows registry entry is taken:

`HKEY_CURRENT_USER\Software\Microsoft\Windows\Current-Version\Explorer\Shell Folders\AppData`

It contains information about the:

- SET command (*Options/Set* menu)
- ASSIGN command (function key assignments)
- User Interface
- Currently loaded project (path)

EAGLE looks for the configuration file in various locations in the given sequence and executes them (if existing):

<prgdir>/eaglerc	(Linux, Mac, Windows)
/etc/eaglerc	(Linux , Mac)
$HOME/.eaglerc	(Linux, Mac)
$HOME/eaglerc.usr	(Windows)

These files should not be edited.

With the help of various User Language programs it is also possible to define settings in EAGLE. Visit our web site:
http://www.cadsoftusa.com/download.htm.

EAGLE Project File

If a new project is created (by clicking the right mouse button on an entry in the *Projects* branch of the tree view and then selecting *New/Project* in the context menu in the Control Panel), a directory is first created which has the name of the project. An ***eagle.epf*** configuration file is automatically created in every project directory.
EAGLE takes note of changes to object properties that are made with the CHANGE command during editing and the contents of the *Width*, *Diameter*, and *Size* menus in the project file.
It also contains information about the libraries *in use* for this project.
The position and contents of the active windows at the time the program is closed are also saved here. This assumes that the *Automatically save project file* option under *Options/Backup* in the Control Panel is active. This state will be recreated the next time the program starts.

Those who have previously worked on projects with **EAGLE 3.5x** can most easily adapt them to the structure that applies from Version 4.0 onwards as follows:

- Create a directory with the project name
- Copy the old project file under the name *eagle.epf* into this directory
- Copy all the files belonging to the project into this directory

The path that leads to these project directories is specified in the Control Panel under *Options/Directories/Projects*.

A project will be opened automatically, if you start EAGLE directly with an *eagle.epf* file or with a project name that exists as folder in the given project path.

This
page
has been
left free
intentionally.

Chapter 6
From Schematic to Finished Board

This chapter illustrates the usual route from drawing the schematic diagram to the manually routed layout. Particular features of the Schematic or Layout Editor will be explained at various points. The use of the Autorouter, the Follow-me router, and the output of manufacturing data will be described in subsequent chapters.

We recommend to create a project(folder) first. Details can be found on page 41.

6.1 Creating the Schematic Diagram

The usual procedure is as follows:

Devices are taken from existing libraries and placed on the drawing area. The connecting points (pins) on the Devices are then joined by nets (electrical connections). Nets can have any name, and can be assigned to various classes. Power supply voltages are generally connected automatically. In order to document all the supply voltages in the schematic diagram it is necessary to place at least one so-called *supply symbol* for each voltage.

Schematic diagrams can consist of a number of pages (but not in the Light edition). Nets are connected across all the pages if they have the same name.

It is assumed that libraries containing the required components are available. The definition of libraries is described in its own chapter.

It is possible at any time to create a layout with the BOARD command or with the *Board* icon. As soon as a layout exists, both files must always be loaded at the same time. This is necessary for the association of the schematic diagram and the board to function. There are further instructions about this in the section on *Forward&Back Annotation*.

Open the Schematic Diagram

You first start from the Control Panel. From here you open a new or existing schematic diagram, for instance by means of the *File/Open* or the File/New menus, or with a double click on a schematic diagram file in the directory tree. The schematic diagram editor appears.

Create more schematic sheets if needed. For that purpose, open the combo box in the action toolbar with a mouse click, and select the *New*. A new sheet will be generated (see page 50). Another way to get a second sheet is to type in

```
EDIT .S2
```

on the command line. If, however, you do not in fact want the page, the entire sheet is deleted with

```
REMOVE .S2
```

Set the Grid

The grid of schematic diagrams should **always** be 0.1 inch, i.e. 2.54 mm. Nets and Symbol connection points (pins) must lie on this common grid. Otherwise connections will **not** be created between nets and pins.

Place Symbols

First you have to make available the libraries you want to take elements from with the USE command. Only libraries which are *in use* will be recognized by the ADD command and its search function. More information concerning the USE command can be found on page 50.

Load Drawing Frame

It is helpful first to place a frame. The ADD command is used to select Devices from the libraries.

When the ADD icon is clicked, the ADD dialog opens.

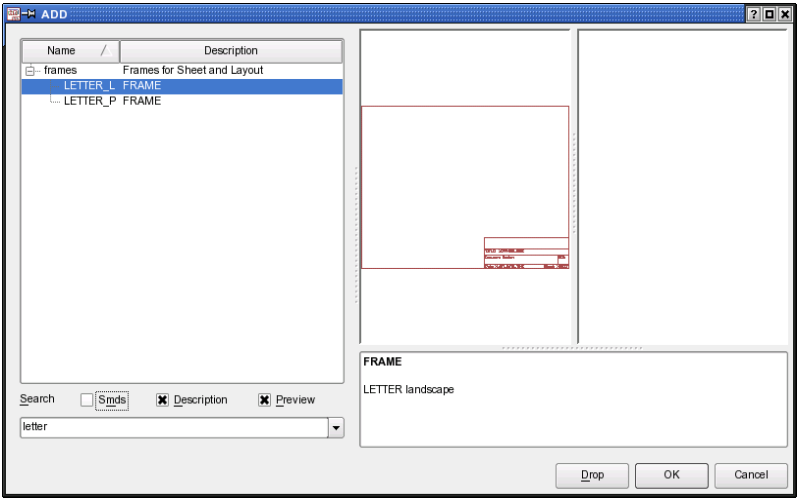
It shows all the libraries that are made available with the USE command, first. You can expand the library entries for searching elements manually or you can use the search function.

A letter format frame is to be used. Enter the search key *letter* in the *Search* line at the lower left, and press the *Enter* key. The search result shows a number of entries from *frames.lbr*. If you select one of the entries (*LETTER_P*), a preview is shown on the right, provided the *Preview* option is active. Disabling the option *Description* excludes the part's descriptive texts.

> *In the Schematic Editor you are searching for Device names and terms of the Device description, in the Layout Editor for Package names and terms of the Package description!*

Clicking *OK* closes the ADD window, and you return to the schematic diagram editor. The frame is now hanging from the mouse, and it can be put down. The bottom left hand corner of the frame is usually at the coordinate origin (0 0).

Library names, Device names and terms from the Device description can be used as search keys. Wildcards such as * or **?** are allowed. A number of search keys, separated by spaces, can be used.



➢ **ADD dialog: Results from the search key** *letter*

The ADD command may also be entered via the command line or in script files. The frame can also be placed using the command:

```
add letter_p@frames.lbr
```

Wildcards like `*` and `?` may also be used in the command line. The command

```
add letter*@frames.lbr
```

for example opens the ADD windows and shows various frames in letter format to select.

The search will only examine libraries that are *in use*. That means that the library has been loaded by the USE command (*Library/Use*).

If a certain library should not be listed in the ADD dialog anymore, select the library name in the tree and click the *Drop* button. Now it is not *in use* anymore.

> *Drawing frames are defined with the FRAME command.*
> *This can be done in a library, where the frame can be combined with a document field. EAGLE can also use the FRAME command in the Schematic as well as the Board Editor. Details about defining a drawing*

frame can be found on page 249.

Place Circuit Symbols (Gates)

All further Devices are found and placed by means of the mechanism described above. You decide a Package variant at this early stage. It can easily be changed later if it should turn out that a different Package form is used in the layout.

If you have placed a Device with ADD, and then want to return to the ADD dialog in order to choose a new Device, press the *Esc* key or click the ADD icon again.

Give the Devices names and values (NAME, VALUE).

If the text for the name or the value is located awkwardly, separate them from the Device with SMASH, and then move them to whatever position you prefer with MOVE. Clicking with DELETE on a text makes it invisible.
Use the *Shift* key with SMASH to get the texts at their original positions. The texts are now no longer separated from the Device (unsmash). Deactivating the *Smashed* option in the context menu's *Properties* window is the same.

MOVE relocates elements, and DELETE removes them. With INFO or SHOW information about an element is displayed on the screen.

ROTATE turns gates by 90°. The same can be done with a right mouse click while the MOVE command is active.

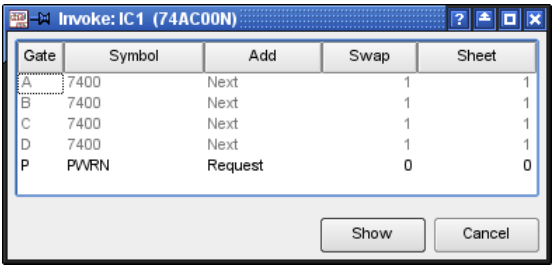
Multiple used parts may be copied with the COPY command. COPY places always a new part even if it consists of several gates and not all of them are already used.

Repeated circuitry can be reproduced in the schematic diagram with the aid of the GROUP, CUT and PASTE commands. First make sure that all the layers are made visible (DISPLAY ALL).

Hidden Supply Gates

Some Devices are defined in the libraries in such a way that the power supply pins are not visible on the schematic diagram. Visibility is not necessary, since all the power pins with the same name are automatically connected, regardless of whether or not they are visible.

If you want to connect a net directly to one of the hidden pins, fetch the gate into the schematic diagram with the aid of the INVOKE command. Click onto the INVOKE icon, and then on the Device concerned, assuming that it is located on the same sheet of the schematic diagram. If the gate is to be placed on a different schematic diagram sheet, go to that sheet, activate INVOKE, and type the name of the Device on the command line. Select the desired Gate in the INVOKE window, then place it. Then join the supply gate to the desired net.



➢ ***INVOKE: Gate P is to be placed***

Devices with Several Gates

Some Devices consist not of one but of several Gates. These can normally be placed onto the schematic diagram one after another with the ADD command. To place a certain Gate you can use the Gate name directly.

Example:

The Device *74*00* from the *74xx-eu* library with Package variant *N* and in *AC* technology consists of for NAND gates named *A* to *D* and one power gate named *P*. If you want to place the Gate *C* first, use the Gate name with the ADD command:

```
ADD 'IC1' 'C' 74AC00@74xx-eu.lbr
```

See also help function for the ADD command.

As soon as one Gate has been placed, the next one is attached to the mouse (Addlevel is *Next*). Place one Gate after another on the diagram. When all the Gates in one Device have been used, the next Device is brought in.

If the Gates in one Device are distributed over several sheets, place them first with ADD, change to the other sheet of the schematic diagram, and type, for example

```
INVOKE IC1
```

on the command line. Select the desired Gate from the INVOKE window.

> *If you select one of the already placed Gate entries in the INVOKE window, the OK button changes to Show. Click the Show button, and the selected Gate is shown in the center of the current Schematic Editor window.*

Designlink.ulp – Access to Farnell's Online Product Database

In case you are looking for information about electronic components you can use *designlink.ulp*. It can access the Farnell webserver and provides you with valueable information, for example, about electronical components. This ULP is currently under development and will be updated from time to time.

Wiring the Schematic Diagram

Draw Nets (NET)

The NET command defines the connections between the pins. Nets begin and end at the connection points of a pin. This is visible when layer 93, *Pins*, is displayed (DISPLAY command).

Nets are always given an automatically generated name. This can be changed by means of the NAME command. Nets with the same name are connected to one another, regardless of whether or not they appear continuous on the drawing. This applies even when they appear on different sheets.

If a net is taken to another net, a bus, or a pin connecting point, the net line ends there and is connected. If no connection is made when the net is placed, the net line continues to be attached to the mouse. This behavior can be changed through the *Options/Set/Misc menu* (using the *Auto end net and bus* option). If this option is deactivated, a double click is needed to end a net. Nets are shown on layer 91, *Nets*.

Nets must end exactly at a pin connecting point in order to be joined. A net is not joined to the pin if it ends somewhere on the pin line.

EAGLE will inform you about the resulting net name or offer a selection of possible names if you are connecting different nets.

The JUNCTION command is used to mark connections on nets that cross one another. Junctions are placed by default. This option, (*Auto set junction*), can also be deactivated through the *Options/Set/Misc* menu.

Nets must be drawn with the NET command, not with the WIRE command.

Do not copy net lines with the COPY command. If you do this, the new net lines won't get new net names. This could result in unwanted connections.

If the MOVE command is used to move a net over another net, or over a pin, no electrical connection is created.
To check this, you can click the net with the SHOW command. All the connected pins and nets will be highlighted. If a Gate is moved, the nets connected to it will be dragged along.

A simple identifier (without *XREF* option, see next section about Cross References) can be placed on a net with the LABEL command. Provided you have defined a finer alternative grid, labels can be arranged comfortably in the finer grid with the *Alt* key pressed.

Defining Cross-References for Nets

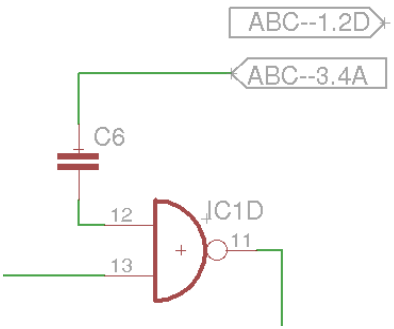
If you place a LABEL with active *XREF* option for a net, a cross-reference will be shown automatically. It points to the next sheet where the net occurs again. Depending on the rotation of the label the cross-reference refers to a previous or a following schematic sheet. If the label itself goes towards the right or bottom border of the drawing, the cross-reference shows the next higher page number. If the label points to the left or top border, the previous pages are taken into consideration. In the case that the net is only available on one sheet, this cross reference is shown, independently from the rotation of the label.
If the net is only on the current sheet, only the net name and possibly the label's frame around it is shown. This depends on the *Xref label format* definition which can be done in the menu *Options/Set/Misc* (can be defined via SET, too).

The *XREF* option can be activated in the parameter toolbar of the LABEL command or after placing the label with CHANGE XREF ON.

The following placeholders for defining the label format are allowed:

- `%F` enables drawing a flag border around the label
- `%N` the name of the net
- `%S` the next sheet number
- `%C` the column on the next sheet
- `%R` the row on the next sheet

The default format string is `%F%N/%S.%C%R`. Apart from the defined placeholders you can also use any other ASCII characters. If `%C` or `%R` is used and there is no frame on that sheet, they will display a question mark '?'. See also page 249.



➢ ***Cross-reference with a XREF label***

The lower label in the picture points to the right and refers to the net *ABC* on the next page *3*, field 4A, the upper XREF label points to the left (beginning with the origin point) and refers to the previous page *1*, field *2D*.

If a *XREF* label is placed on a net line directly, it will be moved together with the net.

More information about cross-references can be found in the help function for the LABEL command.

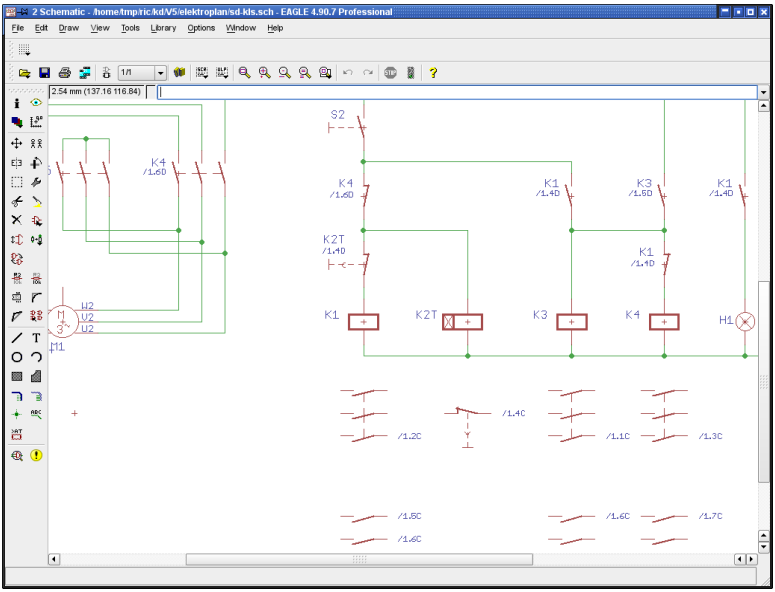
Cross-References for Contacts

In case you are drawing an Electrical Schematic and using, for example, electro-mechanical relays, EAGLE can display a contact cross-reference. In order to do that, place the text *>CONTACT_XREF* inside the Schematic's drawing frame. This text is not displayed in the drawing (excepted its origin cross), but its position (the y coordinate) determines from where on the contact cross-reference will be drawn on the current sheet. As soon as this text is placed the contact cross-reference will be displayed.

The format of the contact cross-references can be defined - as for net cross-references – in the *Options/Set/Misc* menu. It uses the same format variables as described in the previous section *Defining cross-references for nets*. The default setting is: `/%S.%C%R`, which means */Pagenumber.ColumnRow*.

The variables *%C* for column and *%R* for row can only work with a drawing frame that was defined with the FRAME command and comes with a column/row graduation.

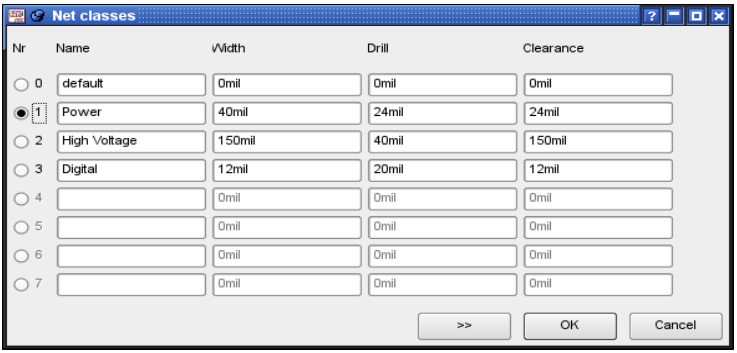
For a proper display of the contact cross-references in the Schematic the elements have to be defined in the Library Editor according to certain rules. More information about this can be found in the help function under Contact cross-reference and in the chapter about *Libraries and Component Design* later in this manual.



➢ *Electrical Schematic with contact cross-reference*

Specifying Net Classes

The CLASS command specifies a net class (*Edit/Net classes* menu). The net class specifies the minimum track width, the minimum clearance to keep away from other signals and the minimum hole diameter for vias in the layout. Each net primarily belongs to net class 0. By default all values are set to 0 for this net class, which means that the values given in the Design Rules are valid. You can use up to eight net classes.

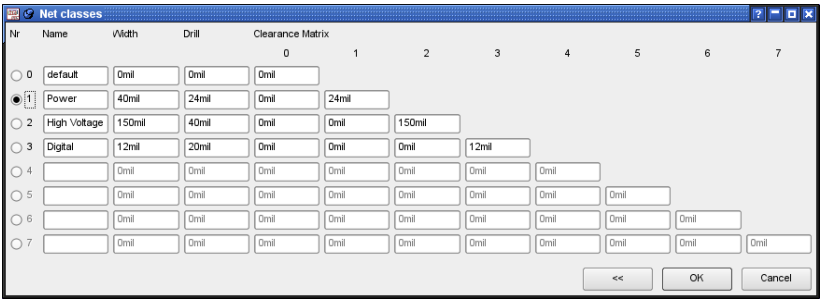


➢ *Net classes: Parameter settings*

The image shows three additional net classes defined:
All nets that belong to class 0, *default*, will be checked by the settings of the Design Rules.
Net class number 1, for example, has got the name *Power* and defines a minimum track *width* of 40 mil.
The minimum *drill* diameter for vias of this class is set to 24 mil.
The *clearance* between tracks of net class 1 and tracks that belong to other net classes is also set to 24 mil.

The left column *Nr* pre-defines the net class of the next net that is drawn with the NET command. This selection can be made in the parameter toolbar of the NET command, as well.

If you would like to define special clearance values between certain net classes, click the button marked with >>. The *Clearance Matrix* opens. Enter your values here.



➢ ***Net classes: The Clearance Matrix***

To return to the simple view, click the << button. This is only possible, however, if there are no values defined in the matrix.

The net classes can be changed later by means of the CHANGE command (the *Class* option) in the Schematic and in the Layout Editor.

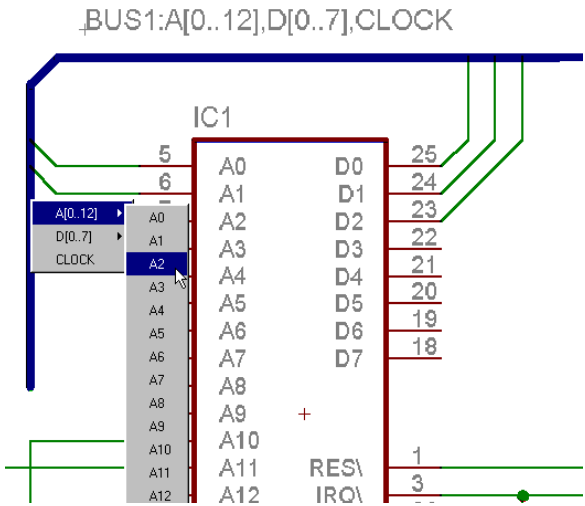
Net class definition can be done in the Layout Editor, as well.

Drawing a bus (BUS)

Buses receive names which determine which signals they include. A bus is a drawing object. It does not create any electrical connections. These are always created by means of the nets and their names. The associated menu function is a special feature of a bus. A menu opens if you click onto the bus with NET. The contents of the menu are determined by the bus name.

The bus in the diagram is named *Bus1:A[0..12],D[0..7],Clock*.

Clicking on the bus line while the NET command is active, opens the menu as illustrated above. The name of the net that is to be placed is selected from here.



➢ ***Bus menu***

The index of a partial bus name may run from 0 to 511.

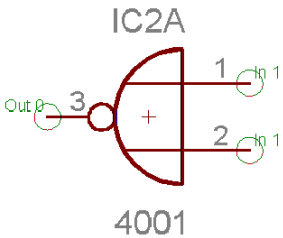
The help function gives further information about the BUS command.

Pinswap and Gateswap

Pins or Gates that have the same Swaplevel can be exchanged with one another. These properties are specified either when the Symbol is defined (Pinswap) or when the Device is created (Gateswap).

Provided the Swaplevel of two pins is the same, they can be exchanged for one another. Display layer 93, *Pins*, in order to make the Swaplevel of the pins visible.

Pins or Gates may not be swapped if the Swaplevel = 0.



➢ ***Swaplevel: Pins layer is visible***

Input pins 1 and 2 have Swaplevel *1*, so they can be exchanged with one another. The output pin, 3, which has Swaplevel *0*, cannot be exchanged.

You can find the Swaplevel of a Gate by means of the INFO command, for example, type in the command line `INFO IC2A`. Alternatively via the context menu, *Properties* entry.

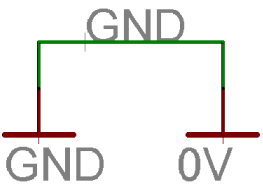
Power Supply

Pins defined as having the direction *Pwr* are automatically wired up. This is true, even if the associated power gate has not explicitly been fetched into the schematic. The name of the Pwr pin determines the name of the voltage line. This is already fixed by the definition of the Symbols in the library.

If nets are connected to a Device's Pwr pins, then these pins are not automatically wired. They are joined instead to the connected net.

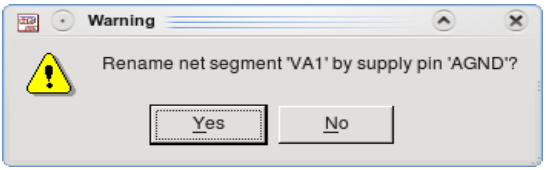
For every Pwr-pin there must be at least one pin with the same name but the direction *Sup* (a supply pin). There must be one on every sheet. These Sup pins are fetched into the schematic in the form of power supply symbols, and are defined as Devices in a library (see *supply*.lbr*). These Devices do not have a Package, since they do not represent components. They are used to represent the supply voltages in the schematic diagram, as is required by the Electrical Rule Check (ERC) for the purposes of its logical checks.

Various supply voltages, such as 0 V or GND, which are to have the same potential (GND, let's say), can be connected by adding the corresponding supply symbols and connecting them with a net. This net is then given the name of that potential (e.g. GND).



➢ ***Supply symbols***

If you place a supply pin (direction *Sup*) onto a net (with ADD or MOVE), you will be asked for a new net name. Should it be the name of the supply pin or should the net name remain unchanged?



➢ ***Supply pin name as new net name?***

Click *Yes* (default) for renaming the net with the name of the supply pin (in the image above: *AGND*). Click *No* to preserve the current net name (*VA1*).

If the net has an automatically generated name, like *N$1*, you may suppress this warning message. Use the SET command in the command line:

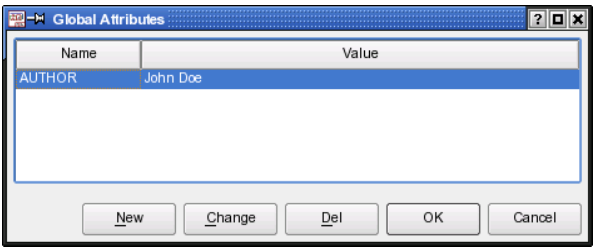
```
SET Warning.SupplyPinAutoOverwriteGeneratedNetName 1;
```

If the last supply pin of a net is deleted, the net will get an automatically generated name, like *N$1*.

Define Attributes

Global Attributes

It is possible to define Global Attributes in the Schematic, for example, for the author or a project identification number, that can be placed anywhere in the schematic, often used in the docfield of the drawing frame.
Open the dialog through the *Edit/Global Attributes...* menu. Click the button *New* to define a new Global Attribute. It consists of the attribute's name and its value.



➢ ***Global Attributes: The Author attribute is created***

If you want to make a global attribute visible in the schematic, write a placeholder with the TEXT command. For the *AUTHOR* attribute, write the text *>author*.
It does not matter, if it is written in lower or upper case letters. The > character in front of the text indicates that this is special text.

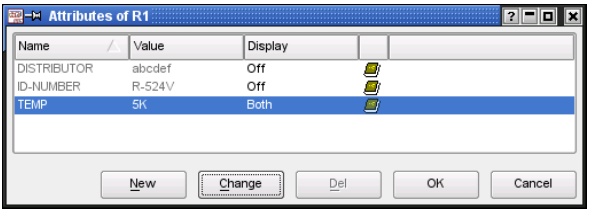
It is possible to define the placeholder text already in the Library, for example, in a Symbol of a drawing frame. In this case the global attribute will be shown on each schematic sheet containing this frame.

Global Attributes can be defined in the Schematic and Layout separately.

More information on this can be found in the ATTRIBUTE command's help.

Attributes for Elements

The ATTRIBUTE command allows you to define attributes for Devices. An attribute consists of the attribute name and its value that may provide any information. If there already exists an attribute that has been defined in the library, you may alter the value in the schematic.



➢ ***Attribute dialog***

Clicking the ATTRIBUTE icon and then onto a Device opens a dialog window. It lists the part's attributes already defined in the schematic or in the library.

The image above shows the attributes *DISTRIBUTOR*, *ID-NUMBER*, and *TEMP* for part R1. The icons on the right indicate where the attribute comes form:

- globally in the Schematic
- globally in the Layout
- in the Library's Device Editor
- for the element in the Schematic
- for the Package in the Layout

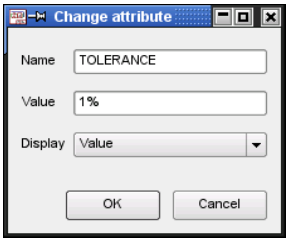
The three attributes of the example dialog above come from the library.

> *Attributes that are defined in the Layout Editor are not shown in the Schematic Editor. A newly defined attribute in the Schematic adopts the value of an already existing attribute in the Layout.*

Defining a New Attribute

Click onto the *New* button to define a new attribute in the schematic. In the following dialog you can define *Name*, *Value*, and the *Display* mode.

In this image the attribute's name is *TOLERANCE,* its value is *1%*.



➢ ***Create and change attributes***

With the *Display* option you manage the way the attribute is displayed in the drawing. There are four options available:

Off: The attribute is not visible
Value: Only the attribute's value is visible (*1%*)
Name: Only the attribute's name is visible (*TOLERANCE*)
Both: Name and value are visible (*TOLERANCE = 1%*)

If the *Display* option is not set *Off*, the respective text will be displayed at the Device's or Gate's origin. The layer which is preset in the Schematic, for example with CHANGE LAYER before creating the attribute, determines the text's layer. Location and layer can be changed any time.
If there is an already defined placeholder text for an element in the library, the text shows up at the given location. It is possible to unfix such texts with the SMASH command. Now you can move it, change its layer, the font, its size and so on.

Changing an Attribute's Value

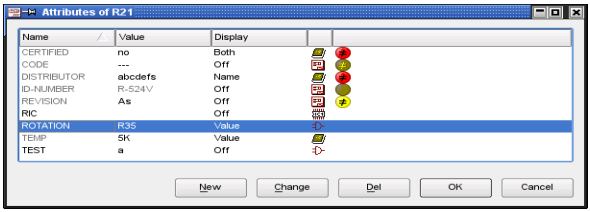
Values of attributes that are already defined in the library can be changed in the Schematic Editor. After changing an attribute's value, the attributes dialog displays special icons that indicate the attribute's status. The icons have the following meaning:

the yellow icon indicates that the attribute initially was defined with a *variable* value and that the value has been changed.

the red icon indicates that the value of the attribute which was initially defined as *constant* has been changed after a confirmation prompt.

the plain brown icon indicates that a global attribute was overwritten by a part attribute. The value, however, remained unchanged.

the brown icon with the unequal sign indicates that a global attribute was overwritten by a part attribute and the value has been changed.



➢ ***Attribute dialog with different attributes***

Grayed text in the Attibutes' dialog indicates that it can't be changed or rather the element's attribute value was defined as *constant* in the library.

The icons inform you about the attribute's origin and its current status. Move the mouse cursor onto one of the icons to let EAGLE display tool tip texts to explain its meaning, provided the Bubble help in *Options/User interface* is active.

More details on defining attributes can be found in the library chapter beginning with page 240.

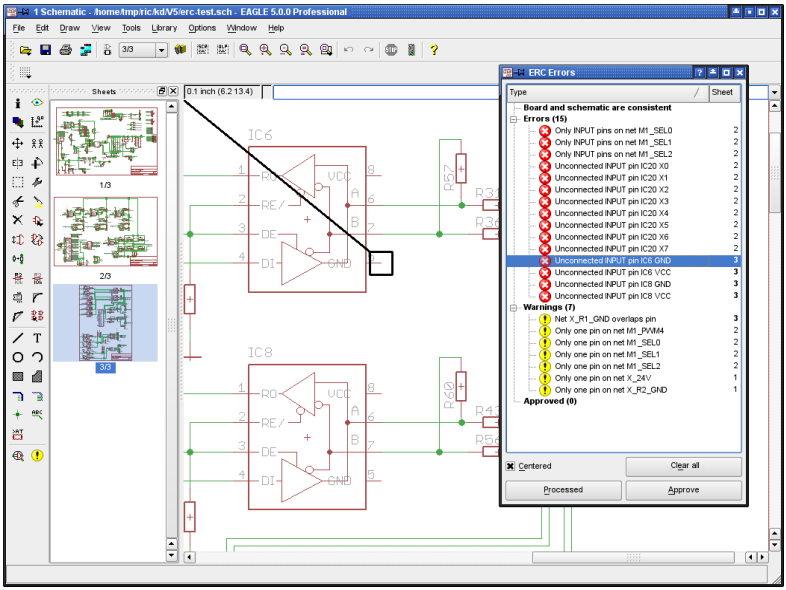
ERC – Check and Correct Schematic

A schematic diagram must be checked with the aid of the Electrical Rule Check (ERC), when the design of the schematic diagram has been completed, if not before. It is actually a good idea to run the (ERC) many times during your design process to catch errors immediately. To start the Electrical Rule Check click onto the ERC icon in the command menu or the entry *Erc...* in the *Tools* menu.

All the errors and warnings are listed in the ERC Error window. Errors are marked with a red icon, warnings with a yellow icon.

In the case of a corresponding board file, the ERC also checks the consistency between schematic and board. If there are no differences, ERC reports *Board and schematic are consistent*. Otherwise the ERC Errors window contains a branch with *Consistency errors*. For further information on this see page 175.

It is possible to sort the errors and warnings, ascending or descending, by error types or sheet numbers. click onto the column headers *Type* or *Sheet* therefore.



➢ ***The ERC Errors window***

If you select an entry in the *Errors* or *Warnings* branch, a line points to the corresponding location in the schematic diagram. In case you zoomed into the drawing, you can click the option *Centered.* The currently selected error is shown in the middle of the drawing window now.

Please check each error and every warning.

In some situations it may be the case that you want to tolerate an error or a warning. Use the *Approve* button for this. The error/warning entry will be removed from the *Errors* or *Warnings* branch and appears in the *Approved* branch.

If you want to have the capability of displaying an approved error/warning occurrence in the *Errors* or *Warnings* branch, expand the *Approved* branch, select the error entry and click the *Disapprove* button. Now it is treated as a normal error/warning and is marked in the schematic.

An approved error/warning retains its approved status as long as you do not disapprove it by clicking the *Disapprove* button. Even a new ERC won't change this status.

If the *Errors* window lists approved errors or warnings only, it won't open automatically after running the Electrical Rule Check again. The status line of the Schematic Editor, however, will show the following hint:

ERC: 2 approved errors/warnings

Moving an entry from one branch into the other, marks the schematic file as changed and not saved.

While correcting the error in the board, the ERC Errors window may remain open. After correcting an error or warning you can mark the entry as *Processed* in the error list by clicking onto the *Processed* button. The error/warning icon turns gray now. Entries marked as processed will be remembered as long as you don't start ERC again. Re-opening the ERC Errors window with the ERRORS command, shows the same status as you left it at last.

If you click onto the *Clear all* button, the *Errors* and *Warnings* branches will be cleared. Approved errors and warnings, however, will remain in the *Approved* branch. The message *List was cleared by user* is shown then.

If you did not run an ERC before, the ERRORS command will start it automatically before opening the errors window.

> *The ERC checks the schematic diagram according to a rigid set of rules. It can sometimes happen that an error message or warning can be tolerated.*
>
> *If necessary, make an output of net and pin lists with the EXPORT command.*
>
> *SHOW allows nets to be traced in the schematic diagram.*

Organize Schematic Sheets

If your schematic is a bit more complex or you want to use more than one sheet, for example, for better readability, you can add (and also remove) sheets with the help of the sheet thumbnails' context menu. Click with the right mouse button onto one of the thumbnails that are located on the left side of the Schematic Editor window.
A new sheet is always added as the last one.

The schematic sheets can be sorted by dragging and dropping the thumbnails. Therefore click with the left mouse button on a thumbnail and drag it to its new position.

Alternatively you can sort the sheets with the EDIT command in the command line:

```
EDIT .s5 .s2
```

moves sheet number 5 at the position before sheet number 2. Further information about this can be found in the EDIT command's help function.

Go to the *Options/User interface* menu in order to switch on/off the sheet preview.

> *Adding, removing, and reordering schematic sheets clears the UNDO/REDO buffer.*

When switching between schematic sheets, the current zoom level of each sheet will be maintained.

Points to Note for the Schematic Editor

Superimposed Pins

Pins will be connected if the connection point of an unconnected pin is placed onto the connection point of another pin. Pins will not be connected if you place a pin that is already connected to a net line onto another pin.

Open Pins when MOVEing

If a Gate is moved then its open pins will be connected to any nets or other pins which may be present at its new location. Use UNDO if this has happened unintentionally.

Merge Different Schematics

The commands GROUP, CUT, and PASTE can be used to copy a schematic or parts of it into another one. To do this open a schematic, display all layers (DISPLAY ALL), select all objects that should be copied with GROUP, activate the CUT command, and left-click into the drawing. Open another schematic now and use the PASTE command.

While inserting a group EAGLE checks the objects' names and compares them with those already existing in the schematic. If there are matches the affected objects to be inserted get new unequivocal names.

This works also with board files, provided there is no consistent schematic available.

6.2 Considerations Prior to Creating a Board

Checking the Component Libraries

The EAGLE component libraries are developed by practising engineers, and correspond closely to present-day standards. The variety of components available is, however, so wide that it is impossible to supply libraries which are suitable for every user without modification.

There are even different Packages which are supplied by various manufacturers using the same identification! Manufacturers recommend very different sizes for SMD pads, and these depend again on the soldering procedure being applied.

In short: You cannot get away without checking the components, in particular the Package definitions, being used when laying out.

In the case of SMD components, please take particular care to ensure that the Package from the library agrees with the specifications of your component. Housings from different manufacturers with the same name but different dimensions are often found.

Agreement with the Board Manufacturer

If you plan to have your PCB professionally manufactured, now is the time to inquire at your board manufacturer whether they stipulate any particular values for the following parameters:

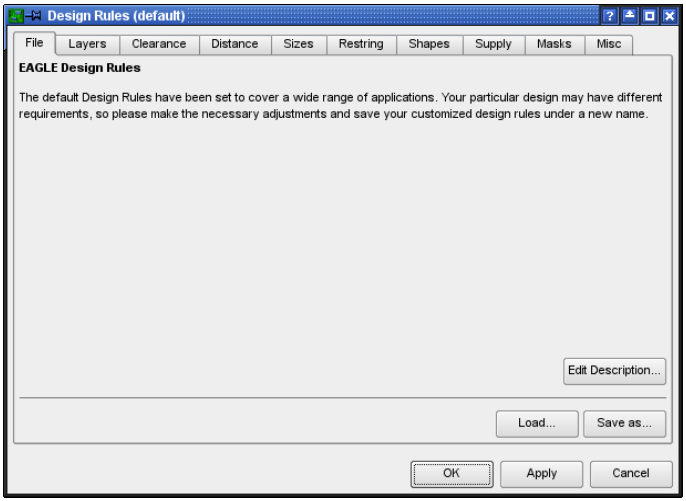
- track width
- shape of solder lands
- diameter of solder lands
- dimensions of SMD pads
- text size and thickness
- drill hole diameters
- number of signal layers
- in case of multilayer boards: manufacturing directions for Blind and Buried vias and composition of the board (see page 158)
- clearance values between different potentials
- parameters concerning solder stop mask and cream frame

You will save yourself time and money if you take these stipulations into account in good time. You will find more details on this in the section on the *Preparing of Manufacturing Data* (Chapter 9).

Specifying the Design Rules

All the parameters relevant to the board and its manufacture are specified in the Design Rules.

Use the menu *Edit/Design Rules..* to open the Design Rules window shown below:



➢ ***DRC: Adjusting the Design Rules***

General Principles

The first time that you call this dialog, the Design Rules are provided by the program. If necessary, adjust the values to suit your or your Board house's requirements.

The *Apply* button stores the values that are currently set in the layout file. Changes to various Design Rules, like the settings concerning the Restring, are immediately displayed in the Layout Editor after clicking *Apply*.

The Design Rules can be saved in a special Design Rules file (**.dru*) by the use of the *Save as..* button. So you can easily use this set of rules for another board.
To apply a set of Design Rules to a board, you can drag any *dru* file of the *Design Rules* branch of the tree view in the Control Panel into the Layout Editor window or click the *Load..* button in the *File* tab of the Design Rules window.

Edit Description.. can be used to alter the descriptive text for the current parameter set. The description usually appears in the *File* tab, as can be seen in the image above. HTML text can be used. You will find notes on this in the help system.

The Design Rules dialog offers a range of different options that can be selected through the tabs. The options include:

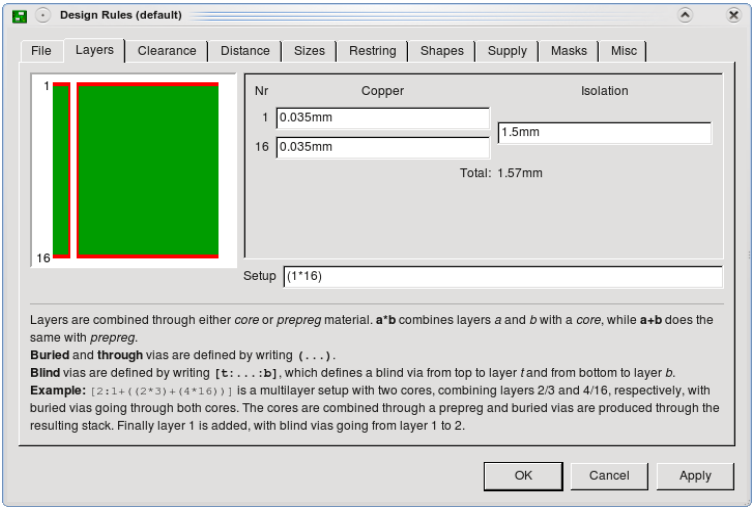
File	Manage the Design Rules
Layers	Number of copper layers, structure of multilayer boards, kind and length of vias, thickness of copper and isolation layers

Clearance	Distances between objects in the signal layers representing signals that may be different or the same
Distance	Distances from the board edge and between holes
Sizes	Minimum track width and hole diameter, particularly for Micro and Blind vias
Restring	Width of the remaining ring at Pads and (Micro) vias and the diameter of Thermal and Annulus symbols in supply layers
Shapes	Shapes of Pads and SMDs
Supply	Thermal and insulation symbols in supply layers
Mask	Values for solder stop and solder cream masks
Misc	Additional checks

Most parameters are explained with the help of a small image. As soon as you click into a parameter line, the associated display appears.

Layers

Define the number of signal layers and the kind of vias (Blind or Buried vias) here. With the help of a mathematical expression in the *Setup* line the proper structure of the board, the appropriate combination of cores and prepregs and the resulting facilities for vias can be defined.



➢ ***Design Rules: Layer Setup***

In most cases (for simple two or more layer boards) the vias are drilled through all layers. The image above shows the default setup for a two layer board. The expression `(1*16)` defines one core with layers 1 and 16, which

can be connected with vias. Parenthesis around the expression define through-hole (continuous) vias.

Basic examples:

1 layer:

`16` Only layer 16, no vias.

4 layers, vias through all layers:

`(1*2+3*16)` Two cores are affiliated with each other.

6 layers, vias through all layers:

`(1*2+3*4+5*16)` Three cores are affiliated with each other.

The fields for *Copper* and *Isolation* are used to define the thickness of copper and isolation layers. These settings are only relevant for complex multilayer boards that use Blind or Micro vias.

The commands DISPLAY, LAYER, WIRE, and ROUTE work only with those signal layers defined in the Layer Setup.

Further information and examples about the *Layer setup* can be found in the section *Multilayer Boards* beginning with page 158.

Loading a board file that was made with an older version causes EAGLE to check which signal layers contain wires. These layers appear in the layer setup. Please adjust it if necessary.

Minimum Clearance and Distance

Clearance refers to the minimum distances between tracks, pads, SMDs and vias of different signals, and between SMDs, pads and vias of the same signal. Setting the values for *Same signal* checks to 0, disables the respective check.

Distance allows settings to be made for the minimum distances between objects in layer 20, *Dimension*, in which the board outline is usually drawn, and between holes.

Setting the value for Copper/Dimension to 0 switches off the minimum clearance check between copper and dimension.
In this case EAGLE does not recognize holes that are placed on wires. Polygons don't keep their distance to objects in layer 20, Dimension, either!

If a net belongs to a special net class, the values for C*learance* and for the drill diameter of vias (*Drill),* defined by means of the CLASS command, are taken into consideration, provided these values are higher than those given in the Design Rules (*Clearance* and *Minimum Drill* in the *Sizes* tab).

Sizes

The minimum values for track width and for hole diameter allowed in the layout are selected here.

If additionally net classes are defined and values for clearance, width, or minimum drill, are set, the respectively higher value is taken into consideration.

Here you set the aspect ratio of drill depth to drill diameter for boards that contain Blind vias. Please contact your board house for this information! If the board house specifies, for example, an aspect ratio of `1:0.5` you have to enter the value `0.5` in the line *Min. Blind Via Ratio*.

For micro vias you have to set the minimum drill diameter in the line *Min. MicroVia*. Setting this value higher than the value in *Minimum Drill* means that there are no micro vias used (default).

Restring (Pad and Via Diameter)

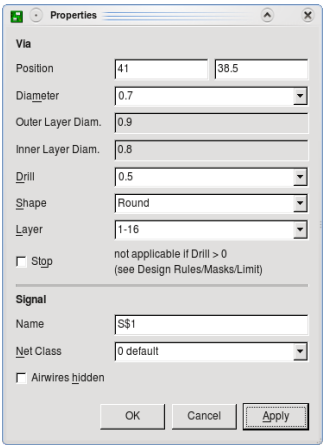
The settings made under *Restring* determine the width of the ring remaining at pads, vias, and micro vias. The remaining ring refers to the ring of copper that remains around a hole after a pad or via has been drilled. Different selections can be made for the width of the remaining ring in the inner and outer layers. Pads may also differ between the *Top* and *Bottom* layers.

Usually the value is expressed as a percentage of the hole diameter. Minimum and maximum values can additionally be specified.

The resting settings also affect the diameter of Thermal and Annulus symbols in supply layers, provided the *Restring* flag (in the Design Rules' *Supply* tab) is set for Thermal/Annulus symbols.

As soon as you change a parameter and click the *Apply* button you can directly see the effects in the layout. If you want to use different values for the upper and lower layer (or different shapes, see *Shapes* tab), it is recommended to set the layer color for layers 17, *Pads*, and 18, *Vias*, the same as the background color (black or white). In this case you can recognize the real size and shape of the pad/via in its respective layer.

The INFO command which has the same dialog as the context menu's *Properties* entry, informs you about the via diameter in outer and inner layers, and about the initial user-defined value. For example, in the following image:



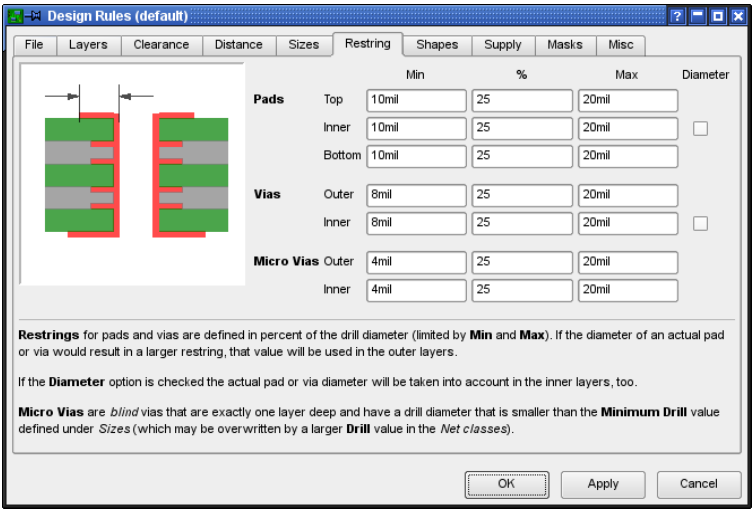
➢ ***Displaying Via properties with INFO***

Pre-defined value (by CHANGE DIAMETER): `0.7`
Actual calculated diameter in the outer layers: `0.9`
Actual calculated diameter in the inner layers: `0.8`

Here the resulting via diameter is bigger than the pre-defined value, according to the given minimum value in the Design Rules' Restring settings for Vias.

The following image illustrates the template for setting the width of the residual ring. The standard value for the restring around holes is 25 % of the hole diameter. Since the width of the ring on small holes specified this way would soon fall below a technically feasible value, a minimum value (here: 10 mil for pads, 8 mil for vias, 4 mil for micro vias) is specified here. It is also possible to specify a maximum value.

Example:
The ring around a hole with 40 mil diameter is 10 mil (25 %). It therefore lies in between the maximum and minimum values.
If the hole is only 24 mil in diameter (e.g. for a via), the calculation yields a restring value of only 6 mil. For a board made in standard technology this is extremely fine, and cannot easily be made. It might well involve extra costs. In this case a minimum value of 10 mil is given.



➢ ***Design Rules: Restring settings***

If you like to define a restring with a fixed width, use the same value for minimum and maximum. The value in percent has no effect in this case.

Diameter check box:
In case you defined a diameter for a pad in the library or for a via in the Layout Editor, and you want to have this given diameter taken into consideration for the inner layers, activate the *Diameter* option. This can be of interest if a pre-defined pad or via diameter exceeds the value calculated by the Design Rules. Otherwise the pad or via in the inner layers would be smaller than in the outer layers. If you want pads/vias to have the same diameter in all layers, set the option *Diameter*.
The option is set off, by default, for new created boards, but will be set on for boards that are updated from version 3.5 or prior because in these versions pads and vias had the same diameter in all layers. Thus the update process does not change the original layout.

All the values can also be given in Millimetres (for example 0.2mm).

Shapes

SMDs:
A rounding factor can be specified here for SMD pads. The value can be between 0 % (no rounding) and 100 % (maximum rounding).

A square SMD has been placed instead of an oblong one on the far right of the diagram. After assigning the property *Roundness* = 100 %, the SMD becomes round.

➢ ***Roundness: 0 - 10 - 25 - 50 - 100 [%].*** ***Right: 100 %, square***

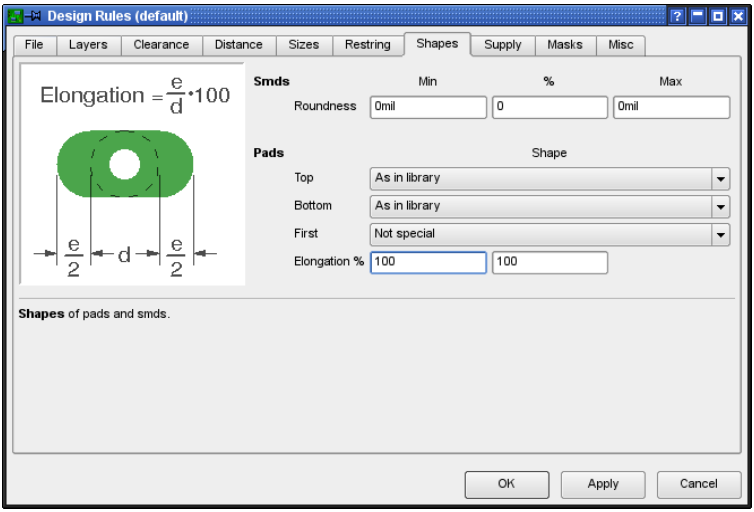
Pads:
This is where the form of the pads is specified. It is possible to give different settings for the top and bottom layers.

The *As in library* option adopts the form defined in the Package Editor. Clicking on *Apply* shows the change immediately in the Layout Editor.

> *Pads and Vias within inner layers are always round, no matter what they are in Top or Bottom layer. The diameter is determined by the restring settings.*

Provided a pad was given the *First* flag in the library one can specify a certain shape for all those pads in the layout.

Elongation defines the aspect ratio of length to width of *Long* and *Offset* pads (see image). The value is given in percent. Click with the mouse into the field *Long* or *Offset* and the image on the left shows the corresponding calculation rule.
100 % is equivalent to an aspect ratio of 2:1. 0 % results in a normal octagon pad with an aspect ratio of 1:1. The maximum is 200 % (ratio 4:1).



➢ ***Design Rules: Adjusting pad shapes***

Notes on the display in the Layout Editor:

If pads or vias have different shapes on different layers, the shapes of the currently visible (activated with DISPLAY) signal layers are displayed on top of each other.

If the color selected for layer 17, *Pads*, or 18, *Vias*, is 0 (which represents the current background color), the pads and vias are displayed in the color and fill style of their respective layers. If no signal layer is visible, pads and vias are not displayed.

If the color selected for layer 17, *Pads*, or 18, *Vias*, is **not** the background color and no signal layers are visible, pads and vias are displayed in the shape of the top and bottom layer.

This also applies to printouts made with PRINT.

Supply

Specifies the settings for Thermal and Annulus symbols. These special symbols are generated in supply layers that are created automatically (layer name starting with a $-sign).

EAGLE calculates the required dimensions for Annulus and Thermal apertures based on the pad's drill diameter and the Annulus and Thermal parameters set in the Design Rules:

Gap defines the gap between the insulation bridges on a Thermal symbol. The specification is made as a % of the hole diameter, and is constrained by maximum and minimum values.

The *Isolate* values for *Thermal* and *Annulus* determine the width of the thermal bridge or ring.

The *Restring* option determines whether the insulation bridge of the Thermal symbol should be drawn immediately at the edge of the hole or at a distance from the hole given by the restring value (*Restring* tab, *Inner* setting).
If the *Restring* option for *Annulus* is deactivated, a filled circle is generated instead of the Annulus ring. This is the default setting.

The isolate value for *Thermal* also applies to polygons. It determines the distance between a polygon and the restring of the pad or via that is joined to the polygon through a Thermal symbol.

The *Generate thermals for vias* flag permits Thermal symbols at through-holes. Otherwise vias are fully connected to the copper plane. This applies also for polygons. But you can disable this option for individual polygons with CHANGE THERMALS OFF and a mouse click onto the polygon's contour.

> *Inside hatched polygons EAGLE doesn't generate Thermal symbols for vias that do not have a direct contact to one of the polygon lines.*

Calculations for automatically created symbols in supply layers:

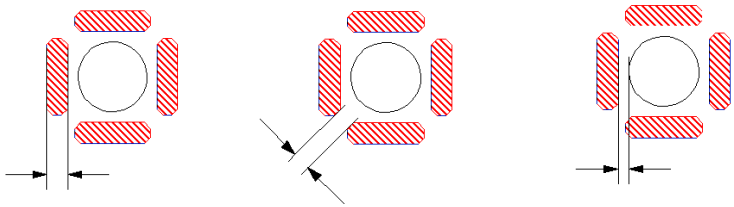
Annulus:

```
Inner diameter = pad hole diameter + 2 * restring

Outer diameter = max(pad-hole diameter, inner diameter)
+ 2*isolate
```

Default: restring = 0 (fully filled), isolate = 20 (mil).

Thermal:

```
Inner diameter = pad hole diameter + 2 * restring

Outer diameter = inner diameter + 2 * isolate
```



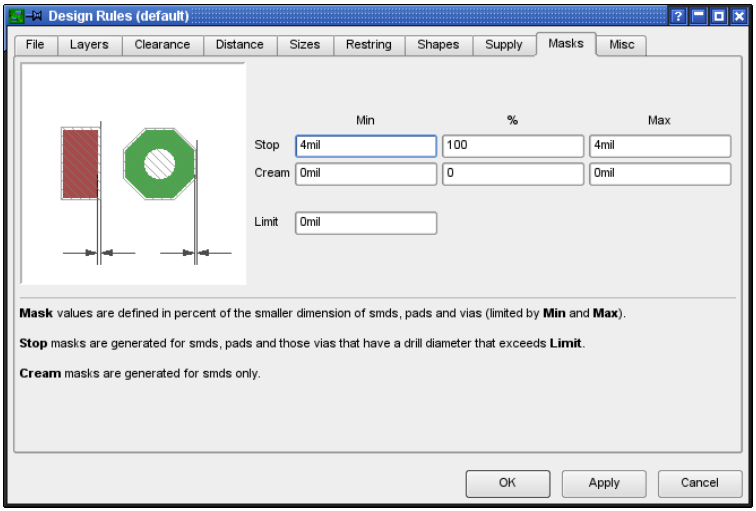
➢ ***Isolate*** ➢ ***Gap*** ➢ ***Restring***

Pads or SMDs marked with the flag *NOTHERMALS* (CHANGE THERMALS OFF) in the Package Editor will be connected basically without Thermal symbols.

Masks

Settings for the overmeasure of the solder stop mask (*Stop*) and the solder cream mask (*Cream*) are made here.

The default value for solder stop is 4 mil, i.e. minimum value is maximum value is 4 mil. The percent value has no effect in this case.
The value for the cream frame is set to 0, which means that it has the same dimensions as the SMD.



➢ ***Design Rules: Settings for Solder Stop and Cream Frame***

If values are given in percent, in the case of SMDs and pads of the form *Long* or *Offset*, the smaller dimension is the significant one. The values are constrained by minimum and maximum values.

The value for *Cream* is given positively, as is *Frame*, although its effect is to reduce the size of the solder cream mask (cream frame).
The solder cream mask is only generated for SMDs, and is displayed on layer 31, *tCream*, or layer 32, *bCream*.
The solder stop mask is drawn in layers 29, *tStop*, or 30, *bStop*.

Setting the flag *STOP* or *CREAM* (only for SMD) to *OFF* for a pad or SMD at the Package definition forbids EAGLE to generate a solder stop mask or a cream frame for it.

Limit determines, together with the hole diameter, whether or not a via is to be covered with solder stop lacquer.

Example:
The default value for *Limit* is 0. This means all vias get a solder stop symbol. They are free of solder stop lacquer.
Set the *Limit* = 24:
All through-plated holes with diameters up to 24 mil don't get a solder stop symbol (they are lacquered), but vias with larger hole diameters get a solder stop symbol.
For vias with hole diameters below the *Limit* the *STOP* flag can be set (CHANGE STOP ON). EAGLE generates a solder stop mask then.

Misc

Here you can select/deselect various checks which are made by the Design Rule Check:

Check grid
examines whether objects lie precisely on the grid currently set by the GRID command. This test is not always worthwhile, since in many cases Devices built to both metric and imperial grids are in use at the same time. No common grid can be found in such a case.

Check angle
ensures that all tracks are laid at whole multiples of 45 degrees. This test is normally switched off, but can be activated if required.

Check font
(de-)selects the font check.
The DRC checks if texts are written in vector font. Text which is non-vector font is marked as an error. This check is necessary due to the fact that the CAM Processor can't work with others than vector font for the generation of manufacturing data.
Assumed you use proportional font text in the bottom layer, place it between two tracks, and use the CAM Processor to generate Gerber files, it could happen that the tracks are shorted by the text (height and length of the text can change)!
Default: on.

Check restrict
can be set off if copper objects should not be checked against restricted areas drawn in layers 39, *tRestrict*, and 40, *bRestrict*.
If restricted areas and copper objects are defined in a common Package, EAGLE does not check them against each other.
Default setting: on.

6.3 Create Board

After you have created the schematic, click the *Board* icon.

An empty board is generated, next to the components that are to be placed, joined together by airwires. Supply pins are connected by those signals which correspond to their name, unless another net is explicitly joined to them.

The placement grid for components is set to 50 mil (1.27 mm) by default.
If you prefer a different placement grid, you are allowed to specify it optionally with the BOARD command in the Schematic Editor's command line.

To have the components placed, for example, in a 1 mm grid, type:

```
BOARD 1mm
```

The unit has to be specified in the command line directly.

The board is linked to the schematic by the Forward&Back Annotation engine provided that both files are always loaded. If both loaded during editing they are guaranteed to remain consistent. Alterations made in one file are automatically carried out in the other.

If you already generated a board from your schematic and continue placing components in the schematic, the referring packages in the board are placed in the current grid setting of the Layout Editor.

> *If, for example, the Schematic is loaded and edited without the Layout, consistency can be lost. The Forward&Back Annotation Engine no longer functions. Differences must then be rectified manually with the aid of the error messages provided by the ERC (see page 175).*

Without the Schematic Editor

If you work without a Schematic Editor, you must generate a new board file, place the Packages with the ADD command and define the connections with the SIGNAL command.

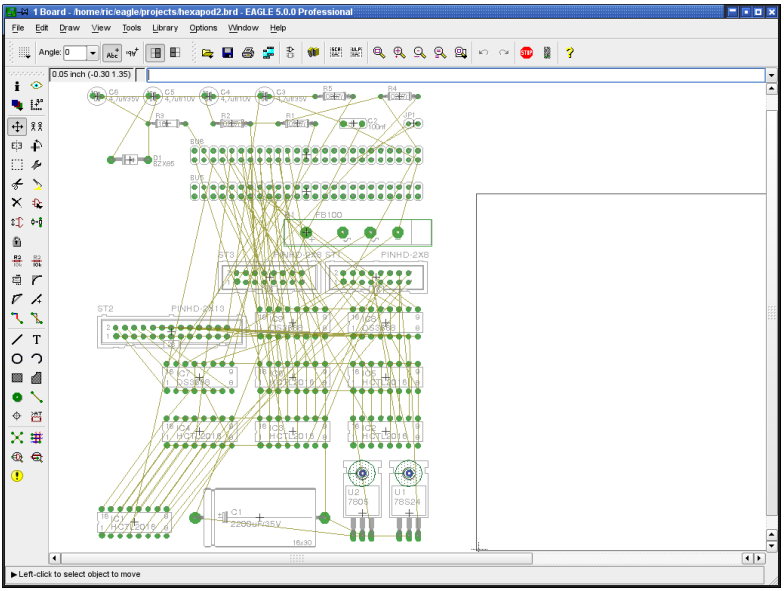
To understand this process, please read the section on *Placing Components* on page 110, and the section on *Specifying Net Classes* on page 117. These two points apply to the Layout Editor as well as to the Schematic Editor.

You are also allowed to define attributes in the Layout Editor. Please read the chapter about defining Attributes beginning with page 121.

The remaining procedures are identical for users with and without the Schematic Editor.

Specify the Board Outline

A board that has just been generated from a schematic diagram initially appears as in the following diagram:



➢ ***Board command: Create the layout from the schematic***

The Devices are automatically placed at the left of the board. The board outline is drawn as a simple line in layer 20, *Dimension*. The outline of an eurocard is displayed in the Professional and Standard editions, a half-eurocard in the Freemium and Light edition.

If you wish, you may alter the size or shape of the empty board with the MOVE and SPLIT commands. Please note the hints in the help function for WIRE and MITER. It's easily possible to draw round outlines, too.
You can also delete the outline and place a frame from a library (such as *19inch.lbr*) with ADD.
A script file can, alternatively, be read by the SCRIPT command. The *euro.scr* file, for instance, can be used. Simply type

```
SCRIPT EURO
```

on the command line.

The board outline serves simultaneously as a boundary for the autorouter or Follow-me router (if present).

If your board has additional cut-outs, you should draw the necessary milling contours in a separate layer, for example in 46, *Milling*. Use the WIRE command with wire width = 0 to define your lines.

Arrange Components

Drag the various components to the desired positions. Use the MOVE command for this purpose. Devices can be clicked on directly, or addressed by name.

If you type, for example,

```
MOVE R14
```

in the command line, the Device named R14 will be attached to the mouse cursor, and can be placed.

Precise positioning results from input such as:

```
MOVE R14 (0.25 2.50)
```

R14's locating point is now located at these coordinates.

> *Keep the Ctrl key pressed while selecting a component in order to let its origin jump at the mouse cursor and move it onto the currently used grid.*

A group of Devices can be transposed by combining the GROUP and MOVE commands. Click onto the GROUP icon and then draw a frame around the desired elements, click MOVE, and then click within the group with the right mouse button in order to select it. With a click of the left mouse button you can place the group at the desired location.

ROTATE, or a click with the right mouse button while the MOVE command is active turns a Device through 90 degrees. This also applies to groups.

In order to place a component in any angle you may specify the rotation directly with the ADD command or later with ROTATE or MOVE in the parameter toolbar.

➢ ***Parameter toolbar for ROTATE, MOVE, ADD, COPY, TEXT***

Next to the *Angle* box are the buttons for the *Spin* and *Mirror* flag.

The left-hand *Spin* icon is selected , if the spin flag is not set (default). This means that texts are displayed always readable from the right or from the bottom side of the drawing.

If the spin flag is activae – the right-hand *Spinned* icon is marked – the texts can be displayed in any rotation, also upside down.

The *Mirror* icons used with components determine where a component is placed: on the top side (default) or on the bottom side of the board. If a component is placed on top, the left-hand icon is active. If you want to place it on the bottom side, click onto the right-hand *Mirrored* icon.

As an alternative you can work with the command line:

```
ROTATE R45 'IC1' ;
```

adds a rotation of 45° to the current position of part IC1. Assumed you tried, for example, to rotate the component with the ROTATE command and pressed mouse button, and you decided that it is not possible to obtain the exact rotation angle this way (because of a too coarse grid) type in the command line:

```
ROTATE =R45 'IC1' ;
```

The rotation of IC1 is now exactly 45°. The = sign stands for absolute values. The initial position does not matter.

If, for example, a SMD should be placed on the bottom side of the board you may add the *Mirror* flag, as in:

```
ROTATE =MR45 'IC1' ;
```

An additional *Spin* flag causes texts to be written upside down (by a rotation of 180°), that means they can be read from the top view:

```
ROTATE =SMR180 'IC1' ;
```

The *Spin* flag is alternating, i. e. using it again causes the text to be displayed 'normal' again.

Check frequently whether the placement is optimal. To do this, use the RATSNEST command. This calculates the shortest connections of the airwires between two pints.

In boards that contain a huge number of signals it may be useful to hide some of the airwires or display only a few of them. If you want to hide, for example, the signals VCC and GND, type in the command line

```
RATSNEST ! VCC GND
```

if want to see them all again, type:

```
RATSNEST *
```

More information about this can be found in the EAGLE help section.

The position of particular Devices can be displayed by typing the Device name onto the command line or by clicking directly on an object while the SHOW command is active.

INFO shows detailed information about the selected object. Depending on the object you clicked on, some of its properties can be altered in the dialog.

The LOCK command allows you to fix components on the board. They can't be moved any more then. *Shift*+LOCK releases the component again. LOCK can be used with groups as well.

If the text for the name or the value is located awkwardly, separate them from the Device with SMASH and move them to whatever position you prefer with MOVE. At the same time EAGLE shows a line from the text's origin to the belonging object. Clicking with DELETE on either of the texts makes it invisible.

Activate the SMASH command, hold down the *Shift* key, and click onto the component to have the texts displayed at their original positions again. They are no longer editable and *unsmashed*, again. Another way to archive this is to deactivate the option *Smashed* in the context menu's *Properties* entry.

> *Please keep in mind that the CAM Processor* ***always*** *uses vector font for generating manufacturing data.*
> *We recommend to write texts in the layout always in vector font (at least in the signal layers). If you do so the shown text meets exactly reality. Further information can be found on pages 45 and 157.*

Attributes for Components and Global Attributes

If you want to assign any further information than name and value to a component in the Board, you can do this with the ATTRIBUTE command.

The definition of attributes for a component can be made in the Schematic as well as in the board file. If Back&Forward Annotation is active, any attribute change will affect both files, schematic and board.

Global attributes are not valid for single components but for the whole board. They can be defined in Board and Schematic separately.

You will find more information about this in the *Creating the Schematic* chapter on page 121.

Boards with Components on Both Sides

If the board is also going to have components on the Bottom layer, the MIRROR command is used. It causes Devices on the underside to be inverted. SMD pads, the silk screen and the layers for the solder stop and solder cream masks are automatically given the correct treatment here.

While ADD, COPY, MOVE, or PASTE is active it is possible to mirror an object or a selected group with the middle mouse button.

> *Define components in the Package Editor always on the top side!*

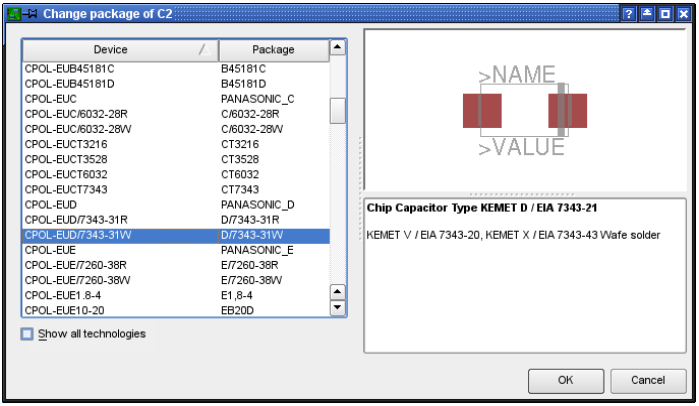
Exchanging Packages

If, as the layout is developed, you want to replace the selected Package variant with a different one, then you can use either the PACKAGE or the REPLACE command, depending on the situation.

PACKAGE Command

It is assumed that the layout and the schematic diagram are consistent and the Device has been defined with more than one Package variant.

Type in the command line `PACKAGE` and click onto the Package to be replaced or alternatively click onto the Package with the right mouse button and select the *Package* entry from the context menu. A third variant would be to click onto the CHANGE icon and select the *Package* option.
Now you select the desired Package, and confirm it with *OK*, in the dialog that then appears.



➢ ***CHANGE package dialog***

If the *Show all technologies* option is active, the Package versions for all the technologies available for this Device are displayed. If this option is not active you will only see Packages that are defined in the selected technology.

The Package can also be exchanged from within the schematic diagram.

Devices that don't have alternative Package variants defined, can be modified in the Library Editor. Add further Package variants as needed and update your drawing with the new library definition. See page 231 *Choosing the Package Variants* for further information.

If you change the Package variant of a Device which you gave a new value with the help of the VALUE command, although it has been defined with VALUE *Off,* the value will remain unchanged. See also page 76.

If you would like to change the Package variant for several identical parts, you can do this in the command line.
Define a GROUP with all parts that shall get a new Package variant, first. Now type in the command line

```
CHANGE PACKAGE 'new-device-name'
```

and click with *Ctrl* + right mouse button into the drawing.
The name of the new Package variant has to be enclosed in inverted commas.

REPLACE command

Consistent Schematic/Layout Pair

The REPLACE command allows you to substitute one component with another. The well-known ADD dialog window opens where you can select the new part. Now click onto the part you want to have replaced in the Schematic or Layout. The old and new Device must be compatible, which means that their used Gates and connected pins/pads must match, either by their names or their coordinates. Otherwise the substitution is not possible.

Layout without Schematic

If you have a layout without an associated schematic diagram, you exchange the Package with the aid of the REPLACE command. REPLACE opens the window that is familiar from the ADD dialog, in which it is possible to search for Devices. When the Package has been chosen you click on the part that is to be replaced in the layout.

The REPLACE command operates in the Layout Editor in two ways, chosen with the SET command:

SET REPLACE_SAME NAMES; (default)

SET REPLACE_SAME COORDS;

The first mode permits Packages whose pad or SMD names are identical to be exchanged. The connecting areas can have any position.

In the second case (replace_same coords) the pads or SMDs in the new Package must be located at the same coordinates (relative to the origin). The names may differ.

The text for the name and value of a Device is only exchanged if they have not been separated from the Device with SMASH.

The new Package can come from a different library, and can contain additional pads and SMDs. Connections on the old Package that were connected to signals must exist correspondingly in the new Package. The new Package can even have fewer connections, provided that this condition is satisfied.

Changing the Technology

It is possible to change the technology of a Device in the layout at any time , provided there are various technologies defined in the library. Use the CHANGE command with the *Technology* option or the *Technology* command of the context menu (available by clicking onto the Package with the right mouse button). This procedure is identical to the one described before in which Package forms are exchanged using PACKAGE.

Define Forbidden Areas

Areas in the form of rectangles, polygons or circles in layers 41, *tRestrict*, and 42, *bRestrict*, are forbidden for the Autorouter/Follow-me router. No copper objects may be drawn in the top or bottom layers inside these areas. These

regions are tested by the Design Rule Check and taken into consideration by the Autorouter/Follow-me router.

Layer 43, *vRestrict*, is for drawing restricted areas where the Autorouter or the Follow-me router may not set vias. vRestrict regions are not examined by the DRC.

Routing – Placing Tracks Manually

The ROUTE command allows the airwires to be converted into tracks. A click on the middle mouse button while a track is being laid allows a change of layer. A via is placed automatically then. Clicking with the right mouse button changes the way in which the track is attached to the mouse and how it is laid (SET command, *Wire_Bend* parameter). Among them are modes which allow to use a wire as 90-degree or as free-definable arc.

If your EAGLE license provides the Autorouter Module, two additional wire bends (*Wire_Bend* 8 and 9) are available that cause the ROUTE command to work as Follow-me router. The Follow-me router can route a selected airwire automatically. The position of the mouse cursor determines the trace of the connection. The settings of the Design Rules and the relevant Autorouter settings are taken into consideration.
Please check the Autorouter chapter for more information about function and usage of the Follow-me router.

The signal's name and net class will be displayed in the status bar. When a signal line has been completely laid, EAGLE confirms that there is a correct connection with a short beep as it is placed.

The signal name can be used in the command line directly, for example `ROUTE VCC`. EAGLE starts the wire at a signal's connection point which is nearest to the current mouse position.

If you want to have the wire started at a via, press the *Ctrl* key and click onto the via.

If there is no longer enough room for routing a signal, other tracks can be relocated with MOVE and SPLIT, or the properties of tracks (width, layer) can be modified with CHANGE.

SPLIT can be used to re-route already existing tracks, if necessary, arrange segment for segment of the wire anew and remove the present track with the commands *Ctrl* + DELETE and/or RIPUP.

If a plated-through hole (a via) is to be placed at a certain point, this can be done with the VIA command. Use the NAME command to assign the via with the signal it should be connected to.

If you want to begin a wire of a partially routed signal at another position than the start or end point of an airwire (for example, at a farther position on a wire that belongs to the same signal), press the *Ctrl* key and and click the left mouse button at the position where the wire should begin. EAGLE generates a new airwire from this position then (see also help function for ROUTE).

Airwires with length of 0 (for example, from *Top* to *Bottom* layer) are drawn as a cross in layer 19, *Unrouted*.

Ending a wire at the same position where another wire of the same signal but in another layer already exists and pressing the *Shift* key at the same time causes EAGLE to place a via. Otherwise it won't.

If you intend to design a multilayer board and use Blind and Buried or Micro vias, please note the details (also for the VIA command) in the section about *Multilayer Boards* beginning at page 158.

While laying out wires EAGLE calculates the shortest connection to the closest point of the current signal automatically. This connection is represented by an airwire.

Pads and SMDs that belong to the currently routed signal have the so-called magnetic-pads function:
Within a certain radius around the pad the wire will be snapped to the pad's center automatically. That is to say as soon as the length of the automatically calculated airwire is shorter than the given value for *snap length*, the wire jumps into the pad's or SMD's center point. It doesn't matter whether the pad or SMD is exactly loacted at the currently used grid. The snap point is always the center point.
As soon as you move the mouse cursor away from this pad beyond the limits, the airwire will be shown and the wire to be routed follows the mouse again. The snap length can be defined in the menu *Options/Set/Misc*. Default value is 20 mil.

As the routing proceeds it is helpful to run the RATSNEST command frequently, in order to recalculate all the airwires.

For more complex boards it may be useful to adjust the *Snap Length* in the *Options/Set/Misc* menu as described on page Fehler: Referenz nicht gefunden.

Use RIPUP if you want to convert the whole or part of a track or a via that has been laid back to a signal line. By clicking on a track it is decomposed between the nearest bends. If you click on this location again (on the airwire), the whole signal branch back to the nearest pads is decomposed. If you want to undo the whole of the signal, click RIPUP and enter the name of the signal on the command line. More than one may be entered at the same time.

The command

```
RIPUP GND VCC +5V
```

converts the three signals GND, VCC and +5V back to airwires.

```
RIPUP ! GND VCC
```

on the other hand converts all signals apart from GND and VCC to airwires.

```
RIPUP ;
```

converts all signals (that are visible in the editor) into airwires. To truly include every track, all the layers in which tracks have been drawn must be visible (DISPLAY).

If you want to use wires as arcs or try to smooth the wire bends see the hints concerning the MITER command in the help function. The miter radius determines how the wire joints are mitered. A positive value generates a rounding, a negative one a straight line. The miter radius influences some bend modi (0, 1, 3, 4; see SET command) and is shown additionally in the parameter toolbar of the commands SPLIT, ROUTE, WIRE, and POLYGON.

While WIRE or ROUTE is active it is possible to click through the previously mentioned wire bends (bend modes) with the right mouse button. EAGLE knows ten different modes (0..9) which are shown as icons in the parameter toolbar. Mode 8 and 9 are special modes for the Follow-me router. They are not available, if your license doesn't support the Autorouter module.
Holding down the *Shift* key while clicking the right mouse button reverses the direction of selection.
Holding down the *Ctrl* key allows to toggle between complementary wire bends.

If you want to have only some wire bends available for the right mouse button, you can define this, for example, in the *eagle.scr* file.
Supposed you want to work with wire bends number 2, 5, 6, and 7 use the following syntax:

```
SET WIRE_BEND @ 2 5 6 7 ;
```

However, if you want to use another bend mode you can always chose it from the parameter toolbar.

> *It is also possible to leave the track laying to the Autorouter which has its own chapter in this manual.*
>
> *Laying tracks with the Follow-me router is explained in a subsection of the Autorouter chapter.*

Defining a Copper Plane with POLYGON

EAGLE can fill regions of a board with copper. Simply draw the borders of the area with the POLYGON command. The polygon is displayed as a dotted line in the outline mode. You give the polygon a signal name, using NAME followed by a click onto the border of the polygon. Then all the objects that carry this signal are connected to the polygon. Both pads and, optionally, vias (as specified in the Design Rules) are joined to the copper plane through Thermal symbols. Elements not carrying this signal are kept at a specified distance.

RATSNEST calculates and displays the surface area of all polygons in the layout. If you call RATSNEST with a signalname, for example

```
RATSNEST GND ;
```

only the GND polygon(s) will be calculated. All other polygons in the layout will remain unchanged in the outline mode.

RIPUP, followed by a click on the polygon border, makes the content invisible again. If there are several polygons in your layout, and you want to have them displayed in the outline mode again, type in the command line:

```
RIPUP @ ;
```

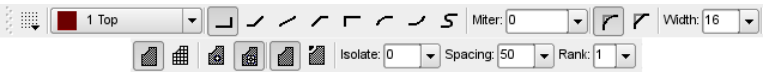
To have all polygons of a particular signal switched to outline mode, specify the signal name, like

```
RIPUP @ GND ;
```

More information about the syntax can be found in the help about RIPUP.

The content of the polygon is not saved in the board file. When you first load the file, you will only see the dotted outline of the polygon. It is only calculated and displayed again by RATSNEST.

Various options can be changed via the parameter toolbar, either as the polygon is being drawn or, with CHANGE, at a later stage.



➢ ***POLYGON command: Parameter toolbar (split into two lines)***

Width:

Line thickness with which the polygon is drawn. Select the largest possible width. That avoids unnecessary quantities of data when the board is sent for manufacture. If the wire width is lower than the resolution of the output driver in the CAM Processor, a warning is issued.
A finer line width permits the polygon to have a more complex **shape.**

Pour:

Specifies the filling type. The whole area (*Solid*) or a grid (*Hatch*).

Rank:

Overlapping polygons must not create any short-circuits. *Rank* can therefore be used to determine which polygons are to be subtracted from others. A polygon with rank = 1 has the highest priority in the Layout Editor, no other polygon drawn in the layout is ever subtracted from it, while one with rank = 6 has the lowest priority. As soon as there is an overlap with a higher rank, the appropriate area is cut out from the polygon with rank = 6. Polygons with the same rank are compared by the DRC.
The rank property works only for polygons with different signals. For overlapping polygons with the same signal name it is without effect. They will be drawn one over the other.

Polygons that are created in the Package Editor can be assigned rank 0 or 7. Rank = 0 has highest priority and will not be overwritten (i.e. obscured by polygons in the layout). Polygons with rank = 7 will always be overwritten by overlapping polygons of rank 1 to 6 in the layout.
Among other uses, rank = 7 allows a surface mount pad to be surrounded by a polygon (thereby making the polygon electrically the same as the surface mount pad) that is, in turn, swallowed by a copper plane and is therefore especially useful for grounded heat sinks.

Spacing:

If the option *Hatch* is chosen for *Pour*, this value determines the spacing of the grid lines.

Isolate:

Defines the value that the polygon must maintain with respect to all other objects not part of its signal. If higher values are defined for special objects in the Design Rules or net classes, the higher values apply.

In the case of polygons with different Ranks, *Isolate* always refers to the drawn contour which is shown in the outline mode of the polygon, even if the calculated polygon has got another contour, for example, due to a wire that supersedes the polygon. The actual clearance can become greater than the given *Isolate* value.

Thermals:

Determines whether pads in the polygon are connected via Thermal symbols, or are completely connected to the copper plane. This also applies to vias, assuming that the option has been activated in the Design Rules.

The width of the thermal connectors is calculated as the half of the pad's drill diameter. The width has to be in the limits of a minimum of the wire width and a maximum of twice the wire width of the polygon.

The length of the thermal connectors is defined by the *Isolate* value for *Thermals* in the Design Rules' *Supply* tab.

Don't choose the polygon's width too fine, otherwise the thermal connectors won't handle the current load.
This is also true for bottlenecks in the board! The polygon's wire width determines the smallest possible width of the copper area.

Orphans:

Determines if a polygon may contain areas (islands) which are not electrically connected to the polygon's signal.
If Orphans is set *Off* such un-connected areas won't be drawn.

When drawing a polygon, please take care to ensure that the outline is

not drawn more than once (overlapping) anywhere, and that the polygon outline does not cross over itself. It is not possible for EAGLE to compute the contents of the area in this case.
An error message 'Signalname' contains an invalid polygon! is issued, and the RATSNEST command is aborted.
If this message appears, the outline of the polygon must be corrected. Otherwise, manufacturing data cannot be created by the CAM Processor.
The CAM Processor automatically computes the polygons in the layout before generating its output.

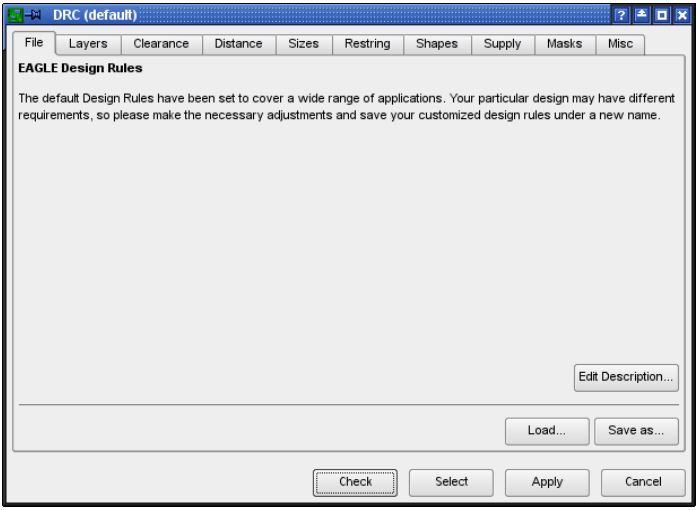
If the polygon stays in the outline mode after calculating it with RATSNEST, you should check the parameters for width, isolate, and orphans and the polygon's name. Probably the polygon's filling is not able to reach one of the objects that should be connected with its signal.

Renaming a polygon with the NAME command, connects it with another signal!

6.4 DRC – Checking the Layout and Correcting Errors

The Design Rule Check (DRC) is carried out at the end of the board design, if not before. If you have not yet specified any Design Rules for the layout, this is your last opportunity. See the section on *Specifying the Design Rules* from page 128. To start the Design Rule Check click onto the DRC icon in the command toolbar or the entry *DRC...* in the menu *Tools*.

Usually one sets the common Design Rules with the Edit/Design Rules.. menu first and starts the Design Rule Check when required with the DRC command. But it is also possible to adjust the Design Rules if you use the DRC command. Some settings, like those for Restring, affect the layout directly.



➢ ***Starting the Design Rule Check***

When you have finished the adjustments, start the error check by clicking *Check*. At the same time the Design Rules are stored in the board file itself.

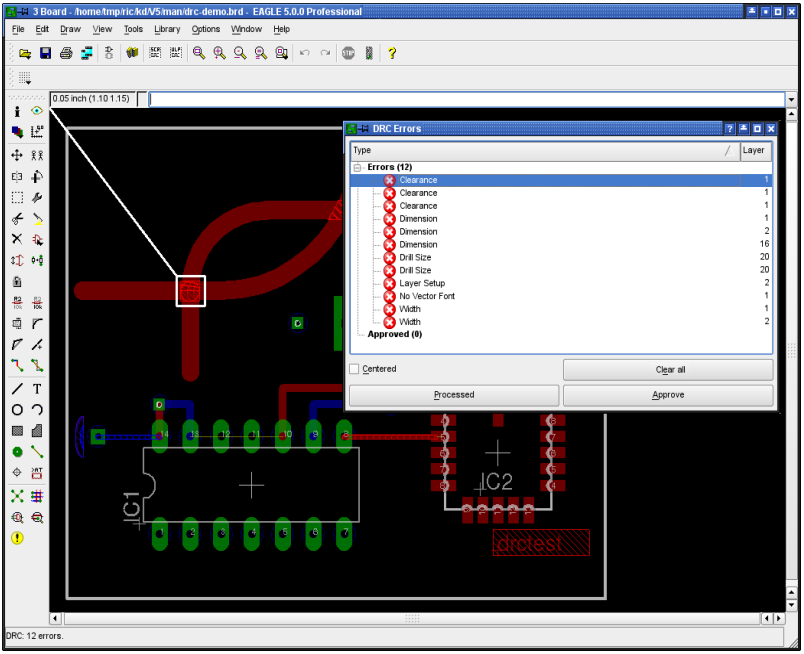
By clicking *Select* you specify the region of the layout that is to be examined. Simply drag a rectangle over the desired region with the mouse. The error check will then star automatically.

Clicking on *Apply* transfers the settings to the board file. This means that the values that have so far been chosen are not lost if you do not immediately start the error check and if you want to leave the DRC dialog via the *Cancel* button.

> *All signal layers are always examined by the Design Rule Check, no matter if visible or not (DISPLAY command).*

The DRC Errors Window

If the Design Rule Check finds errors, an error window opens automatically. It lists all the errors found. The window can be opened at any time by means of the ERRORS command.



➢ ***DRC Errors list in the Layout Editor***

Each error is marked with an error polygon. Its size tells you, for example in the case of a clearance error, about how much the limit is exceeded. The error polygons are visible in the Layout Editor, only. They won't be printed nor exported with the CAM Processor. It's not possible to erase them with the DELETE command. Click the *Clear all* button to delete them. Or type in the command line:

```
ERRORS CLEAR
```

Errors are marked with a red icon in the errors window. If an error in the list is selected, a line points to the corresponding location in the board.

It is possible to have the error list sorted, ascending or descending, by error types or layer numbers. therefore click onto the column headers *Type* or *Layer*.

The errors dialog shows only errors that occur in the currently displayed layers.

In case you zoomed into the drawing and there is only a partial view of the board, you can click the option *Centered*. The currently selected error is shown in the middle of the drawing window now. If you prefer to have the *Centered* option deactivated for browsing the error list, you are nevertheless able to center an error in the middle of the drawing area by pressing the *Enter* key.

While correcting the error on the board, the DRC Errors window may remain opened. After correcting one error you can mark it as *Processed* in the error list by clicking onto the *Processed* button. The red error icon turns gray now.

In some situations it may be the case that you want to tolerate an error. Use the *Approve* button for this. The error entry will be removed from the *Errors* branch and appear in the *Approved* branch and the error polygon is no longer shown in the Layout Editor.
If you want to treat an already approved error as a quite normal error, select it in the *Approved* branch, and click onto the *Disapprove* button. Now it is a member of the *Errors* branch again.

Clicking the *Clear all* button does not delete approved errors. They remain in the *Approved* branch.

Moving an entry from one branch into the other, marks the board file as changed and not saved.

EAGLE's minimum resolution is 1/10.000 Millimetre. That is why the DRC can report only distance and clearance errors which are larger than 1/10.000 mm.

Error Messages and their Meaning

Angle:

Tracks are not laid in an angle of 0, 45, 90 or 135°. This check can be switched on or off in the Design Rules (*Misc* tab).
Default: off.

Blind Via Ratio:

The limit of the ratio of via length (depth) to drill diameter is exceeded. In this case you have to adjust the via's drill diameter (Design Rules, *Sizes* tab) or the layer thickness of your board (Design Rules, *Layers* tab).

Clearance:

Clearance violation between copper objects. The settings of the Design Rules' *Clearance* tab and the value for *Clearance* of a given net class are taken into consideration. Of these two values the higher one is taken for

checking.
In addition the *Isolate* value will be taken into consideration for polygons with the same rank and polygons which are defined as a part of a Package.
To deactivate the clearance check between objects that belong to the same signal, use the value 0 for *Same signals* in the *Clearance* tab.
Micro Vias are treated like wires. The clearance value for wire to wire applies in this case.

Dimension:

Distance violation between SMDs, pads, and connected copper objects and a dimension line (drawn in Layer 20, *Dimension*), like the board's outlines. Defined through the value for *Copper/Dimension* in the Design Rules' *Distance* tab.
Setting the value *Copper/Dimension* to *0* deactivates this check.
In this case polygons do not keep a minimum distance to objects in layer 20, *Dimension*, and holes!
The DRC will not check if holes are placed on tracks then!

Drill Distance:

Distance violation between holes. Defined by the value *Drill/Hole* in the Design Rules (*Distance* tab).

Drill Size:

Drill diameter violation in pads, vias, and holes. This value is defined in the Design Rules' *Sizes* tab, *Minimum Drill.*
It is also possible to define a special drill diameter for vias in a given net class (CLASS command, *Drills*). In this case the higher one is used for the check.

Invalid Polygon:

Reason is a not properly drawn polygon contour. As soon as the contour lines are overlapping or even crossing, the polygon can't be calculated correctly. Change the polygon's contour in the Layout Editor or in the Library, if it is part of a Package.
The RATSNEST command shows this error message, as well.

Keepout:

Restricted areas for components drawn in layer 39, *tKeepout,* or 40, *bKeepout*, lie one upon another. This check is executed only if layers 39 and 40 are displayed and if the keepout areas are already defined in the Package Editor of the library.

Layer Abuse:

Layer 17, *Pads*, or 18, *Vias*, contain objects which are not automatically generated by EAGLE. Probably you drew something manually in these layers, although they are reserved for pads and vias. Better move such objects into another layer.
In case there are wires connected to pads or vias in a Supply layer, the

Design Rule Check reports this error message, as well. You are not allowed to draw signals in Supply layers.

Layer Setup:

This error is shown if an object in a layer is found that is not defined by the Layer setup. The same for vias that do not follow the settings of the Layer setup, for example, if a via has an illegal length (Blind/Buried vias).
If there is a supply layer defined and this layer is not used in the Layer Setup, EAGLE reports this error, as well.

Micro Via Size:

The drill diameter of the micro via is smaller than the value given for *Min. Micro Via* in the *Sizes* tab.

No Vector Font:

The font check (Design Rules, *Misc* tab) recognizes text in a signal layer which is not written in EAGLE's internal vector font.
If you want to generate manufacturing data with the help of the CAM Processor the texts, at least in the signal layers, ought to be written in vector font. This is the only font the CAM Processor can work with. Otherwise the board will not look the same as it is shown. Change the font with the help of the command CHANGE FONT or use the option *Always vector font* in the Layout Editor's *Options/User Interface* menu:
If activated, the Layout Editor shows all texts in vector font. This is the way the manufactured board will look like.
Activating the sub-option *Persistent in this drawing* saves the setting in the drawing file. If you send the layout file, for example, to the board house you can be sure that the vector font will be displayed also at his system.

No real vector font:

The font check (Design Rules, *Misc* tab) recognizes text in a signal layer which is not written in EAGLE's internal vector font although it is displayed as vector font in the Layout Editor window. This situation arises if the option *Always vector font* in the menu *Options/User Interface* is active.
See error message *No vector font* for further details.

Off Grid:

The object does not fit onto the currently chosen grid.
This check can be switched on or off in the Design Rules' *Misc* tab. The default setting is off, because as soon as trough-hole and surface-mount parts are used together it's not easily possible to find a reasonable common grid. The check is set off by default.

Overlap:

DRC reports this error as soon as two copper elements with different signals touch each other.

Restrict:

A wire drawn in layer 1, *Top,* or 16, *Bottom,* or a via lies in a restricted area which is defined in layer 41 or 42, *t/bRestrict*.
If restricted areas and copper objects are defined in a common Package, the DRC does not check them!

Stop Mask:

If there are silkscreen objects drawn in layers 21, 25, 27 for components on the Top layer, and 22, 26, and 28 for components on the Bottom layer overlapping the area of a solder stop symbol generated in layer 29 and 30, the DRC reports a Stopmask error.
You have to display the corresponding layers to activate this check!
Please keep in mind that this check always takes the vector font as basis for the calculation of the required space. This is the font type the CAM Processor uses for manufacturing data generation.

Width:

Minimum width violation of a copper object. Defined by *Minimum Width* in the Design Rules (*Sizes* tab) or, if defined, by the track parameter *Width* of a referring net class. The higher one of the given values will be taken for this check.
Also the line width of vector font texts in signal layers will be checked.

Wire Style:

The DRC treats a line (wire) whose *Style* is LongDash, ShortDash or DashDot in the same way as a continuous line. If a wire drawn with one of these styles is laid as a signal, the DRC reports a *Wire Style* error.

For further investigations, net, part and pin lists can be output by means of the EXPORT command or by various User Language programs.

6.5 Multilayer Boards

You can develop multilayer boards with EAGLE. To do this, you use one or more inner layers (*Route2* to *Route15*) as well as the layers *Top* and *Bottom* for the top and undersides. You display these layers when routing.

Before starting the routing of the layout you should be aware of the number of signal layers to use, if vias should go through all layers, or if you have, due to the complexity of the layout, to work with Blind, Buried or Micro vias. In this case you really ought to contact your board manufacturer to inform you about the possible structure of the board and the costs to be expected.
Also think about how to use the inner layers: As a normal inner signal layer or as a supply layer? On this, for example, depend reasonable via lengths that have to be defined in the Layer Setup.

Inner Layer

Inner layers are used the same way as the outer layers *Top* and *Bottom*. They can be filled with copper areas (polygons) as well. A special status have the so-called supply layers which are generated automatically.

Before using inner layers you must define them in the Design Rules, *Layers*-Tab. More details can be found in the following sections and on page 130.

Signal Layers

You use the ROUTE command as usual to place tracks in those inner layers which are provided for signals. Eagle will itself ensure that the tracks are connected by way of plated-through holes to the appropriate signals in the outer layers. The settings of the Layer setup in the Design Rules apply for this procedure.

Supply Layers with Polygons and More than One Signal

Areas of the board can be filled with a particular signal (e.g. ground) using the POLYGON command. The associated pads are then automatically connected using Thermal symbols. The isolate value for the Thermal symbols is specified in the Design Rules (DRC command, *Supply* tab). The width of the connecting bridge depends on the line thickness with which the polygon is drawn (see page 151). You can also specify whether or not vias are to be connected through Thermals. The minimum clearances from objects carrying other signals specified in the Design Rules are maintained (*Clearance* and *Distance* tabs). Changes are shown in the layout when the polygon is next computed (RATSNEST command).

In this way you can also create layers in which several areas are filled with different signals. In that case you can assign different ranks (priorities) for the polygons. The rank property determines which polygon is subtracted from others if they overlap. Rank = 1 signifies the highest priority in the layout; nothing will be subtracted from such a polygon (with the exception of polygons with rank = 0, drawn as a part of a Package in the Package Editor). Rank = 6 signifies the lowest priority. Polygons with the same rank are compared by the DRC.

Please read the notes regarding polygons in the section on *Defining a Copper Plane* on page 149.

> *Do not choose the wire width for polygons too fine! This can lead to huge amounts of plot data and problems for the manufacturing process.*

Power Supply Layer with One Signal

A power supply layer with one signal is implemented by renaming one of the Route2...15 layers in such a way that the new name consists of the corresponding signal name prefixed by a $ sign. If, for example, the signal

called GND is to be realized as a power supply layer, then a layer is specified as having the name *$GND*.

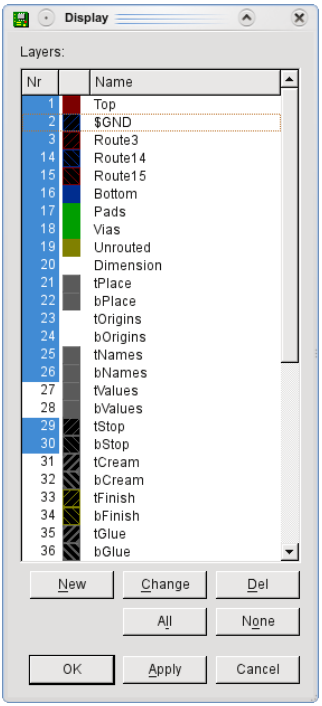
For this purpose you activate the DISPLAY command. Select layer *2* from the menu with the mouse. Then click on *Change* button.

Enter the signal name for the layer, and activate the *Supply layer* check box. If the layer is to be visible in the layout immediately, then click also the *Displayed* check box.

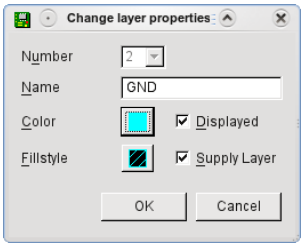
The syntax for the LAYER command in the command line corresponding to this example would be:

```
LAYER 2 $GND
```

This specifies that layer number 2 (previously known as Route2) is henceforth known as $GND, and that it will be treated as a power supply layer.



➢ ***The Display menu***



➢ ***Change layer properties***

The DISPLAY menu shows only those layers defined in the Layer Setup of

the Design Rules!

Pads are connected to power supply layers with what are known as Thermal symbols, or are isolated with Annulus symbols. Thermal symbols usually just have four thin bridges as a conductive connection to the through-plated hole. They are used because the high thermal conduction of a continuous copper plane would result in the pad being no longer solderable. The shape and size of the Annulus and Thermal symbols are specified in the Design Rules (DRC command, *Supply* tab).

To keep the board edges free of copper draw an isolating wire around the layout. This avoids possible short circuits between adjacent (supply) layers. If you want to use the Autorouter, however, these wires must not be drawn until the routing process has been finished. Automatically generated supply layers have to be disabled in the Autorouter/Follow-me setup.

Supply layers are displayed and plotted inversely!

No additional signals or signal polygons may be drawn in these layers. That would lead inevitably to an unusable circuit board!

Supply Layers and Autorouter

The Autorouter takes into account all inner layers, and so delivers the full set of patterns for multilayer boards. It connects SMD pads to inner layers with vias.

Assumed only vias that can connect all layers (from layer 1 to 16) are used (through vias), the preferred direction for the Autorouter has to be set to *N/A* for power supply layers (this means the layer is not available for routing). This is only possible for layers that do not contain any wires.

If you are using, among vias through all layers, Blind, Buried, or Micro vias the Autorouter can't work with supply layers. In that case you have to change the supply layer into an inner layer with polygon:
Therefore deselect the *Supply* flag in the *Change* window of the DISPLAY menu and draw a polygon in the inner layer. The polygon has to be given the name of the supply signal.

Multilayer Boards with Through Vias

This type should be preferred if possible. Vias go through all signal layers and will be drilled at the end of the production process. The production costs are relatively moderate.

Layer Setup

The settings concerning layer composition and number of signal layers are made in the Design Rules, *Layers* tab, *Setup*. See page 130.

For through vias the setup is very simple. No considerations about thickness of copper and isolation layers are necessary.

Simply join two layers by an asterisk (like *1*2* or *3*16*) to one core and combine several cores. This is symbolized by a plus character (like in *1*2+3*4*). The isolation layer between two copper layers is called prepreg. To express the possibility to have vias through all layers the whole expression is set into parenthesis.

Examples:

4 layers:	`(1*2+3*16)`
6 layers:	`(1*2+3*4+5*16)`
8 layers:	`(1*2+3*4+5*6+7*16)`

Here vias always have the length 1-16. They are reachable from all layers (see also the help function for VIA).

Multilayer with Blind and Buried Vias

In high density boards it is often necessary to use Blind and Buried vias. These kinds of vias don't connect all layers, but are only reachable from a certain number of layers. How these layers are connected depends on the manufacturing process of the board which has to be determined in the Layer setup in the Design Rules.

> *Please contact your board house before starting your work! Check which Layer Setup is suitable for your purpose and what the manufacturing costs are.*

Disambiguation

Core:
The non-flexible kernel which is coated with copper on one or on both sides. Is represented by a * in the Layer Setup. For example *5*12*: Layer 5 and 12 are the board's core.

Prepreg:
Flexible glueing or isolating layer which is used in the manufacturing process of a multilayer board to press inner and outer layers onto each other.
Is represented by a + in the Layer Setup. *1+2* tells us that layer 1 is a prepreg and combined with layer 2.

Layer Stack:
A pack of any number of layers consisting of cores and prepregs which are handled together in the current step of production.

Buried Via:
The production process of this via does not differ from a through (normal) via. The current layer stack will be drilled through completely. In the following production steps the already drilled vias can be covered (buried) by pressing further cores and prepregs on the current layer stack. If the via is not

visible on the completed board we call it a buried via.
This is represented by parenthesis, for example in *1+(2*15)+16* where the Buried Via goes from layer 2 to 15.

Blind Via:
A Blind via connects an outer layer with any inner layer but doesn't go through all copper layers. The speciality of a Blind via lies in the production process. The current layer stack is not drilled all through. The drill hole has a certain depth depending on the number of layers that should be allowed to be connected with each other. Blind vias have to follow a given ratio of depth to drill diameter. Please contact your board house to get information about this. This ratio has to be defined in the *Sizes* tab as *Min. Blind Via Ratio.*
This is represented by brackets and the target layer marked by a colon before or after the bracket. The example ***[3:**1+2+3*14+15+16]* allows Blind vias from layer 1 to 3.
Blind vias may be shorter than defined. In this example you are allowed to use vias from layer 1 to 2. The Autorouter is also allowed to use shorter Blind vias.

Micro Via:
The micro via is a special case of a Blind via. It has a maximum depth of one layer and a very small drill diameter. See page 169.

Displaying Vias

It makes sense to set the layer color of layer 18, *Vias*, to the background color (DISPLAY menu, *Change*, *Color*) if you are working with vias that have different lengths and shapes. In doing so it is possible to recognize layer affiliation.

Layer Setup

Combining cores and prepregs allows many variants. In the following section some examples show the function of the Layer setup.
Please read this paragraph entirely. Even if you intend to design a four layer board, for example, it is most advisable to read also all the other examples for a better understanding.

4-Layer Board

Example 1:

Layers 1, 2, 3 and 16 are used.

Board structure: One core inside, outside prepregs.

Connections: 1-2 (blind vias), 2-3 (buried vias) and 1-16 (through vias)

The setup expression looks like this:

[2:(1+(2*3)+16)]

Explanation:

2*3

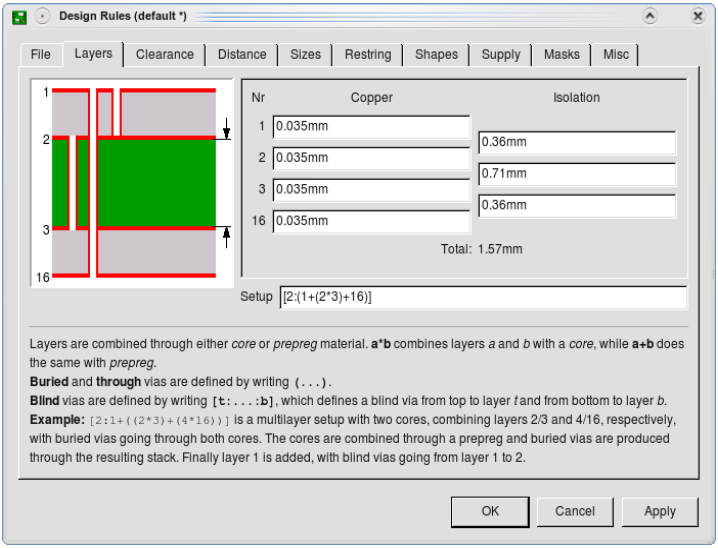
Layers 2 and 3 form the core.

(2*3)
Parenthesis allow buried vias from 2 to 3.

(1+(2*3)+16)
On both sides of the core copper layers are pressed on with prepregs.
The outer parenthesis define continuous vias from 1-16.

[2:(1+(2*3)+16)]
In square brackets and separated by a colon blind vias are defined. Here from layer 1 to 2.

The following image shows the related setup expression in the *Layers* tab of the Design Rules.



➢ ***Example 1: Layer Setup for a 4 layer Board***

Blind vias have to keep a certain ratio of via depth to drill diameter. For this reason it is necessary to specify values for the layer thickness.
These values are given by your board house! You are supposed to contact it in either case before starting the layout!

Type in the values in the *Copper* (thickness of copper layer) and *Isolation* (thickness of isolation layer) fields as shown in the image. The total thickness of the board is shown below the *Copper* and *Isolation* fields.

Example 2:

Layers 1, 2, 3, and 16 are used.

Board structure: One core inside, outside prepregs.

Connections: 1-2, 3-16 (blind vias), 1-16 (through vias)

Setup expression:

[2:(1+2*3+16):3]

Explanation:

2*3

Layers 2 and 3 form the core.

1+2*3**+16**

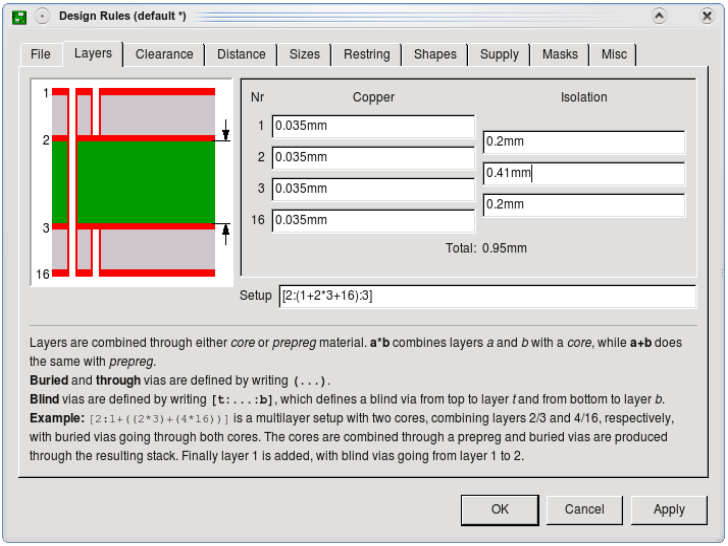
On both sides of the core copper layers are pressed on with prepregs.

(1+2*3+16**)**

The outer parenthesis define through vias from 1-16.

[2:(1+2*3+16)**:3]**

In square brackets and separated by a colon blind vias are defined. Here from layer 1 to 2 and 16 to 3.



➢ *Example 2: Layer Setup for a 4 layer Board*

6-Layer Board

Example 3:

Layers 1, 2, 3, 4, 5, and 16 are used.

Board structure: Two cores, prepregs outside.

Connections: 2-3, 4-5 (buried vias), 1-16 (through vias)

Setup expression:

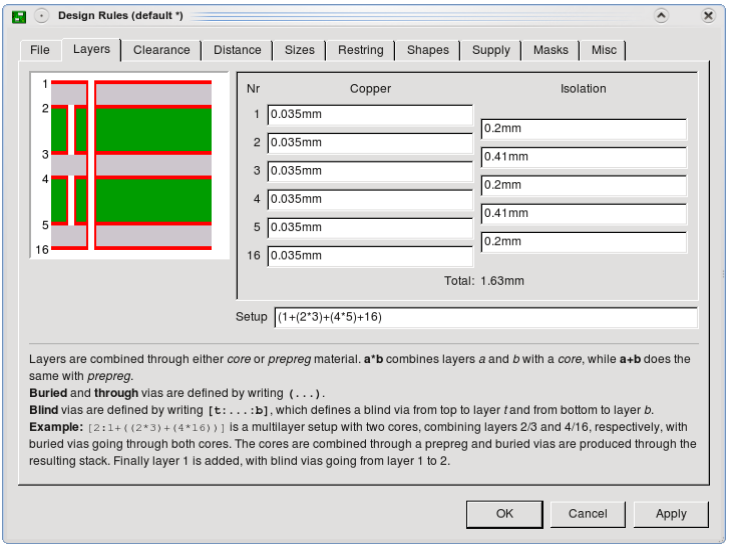
(1+(2*3)+(4*5)+16)

Explanation:

(2*3)+(4*5)
Two cores with buried vias are pressed together.

1+(2*3)+(4*5)**+16**
This layer stack is covered with outer layers 1 and 16 which are isolated with prepregs.

(1+(2*3)+(4*5)+16**)**
The whole expression in parenthesis defines through vias from 1-16.



➢ ***Example 3: Layer Setup for a 6 layer Board***

The values for layer thickness for copper and isolation used in these examples are fictive. Please contact your board house to get the allowed values.

Example 4:

Layers 1, 2, 3, 4, 5, and 16 are used.

Board structure: One core, on each side two prepregs.

Connections: 3-4 (buried vias), 2-4 (blind vias in inner layer stack), 1-16 (through vias)

Setup expression:

(1+[4:2+(3*4)+5]+16)

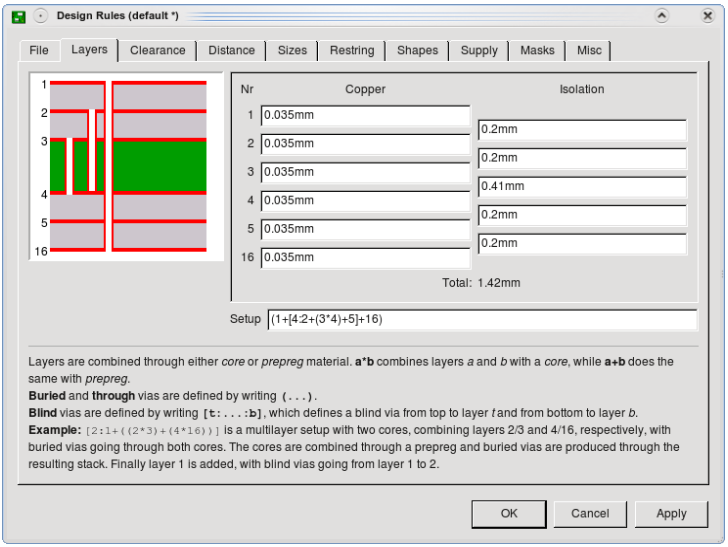
Explanation:

2+(3*4)+5
The core with buried vias. One prepreg on each side.

[4:2+(3*4)+5**]**
Blind vias from layer 2 to 4.

1+[4:2+(3*4)+5]**+16**
On this layer stack a prepreg on each side is pressed on.

(1+[4:2+(3*4)+5]+16**)**
Parenthesis allow through vias from 1 to 16.



➢ ***Example 4: Blind Vias in the inner layer stack***

8-Layer Board

Example 5:

Layers 1, 2, 3, 4, 5, 6, 7, and 16 are used.

Board structure: Three cores, prepregs outside.

Connections: 1-3, 6-16 (blind vias), 2-3, 4-5, 6-7 (buried vias), 1-16 (through vias).

Setup expression:

[3:(1+(2*3)+(4*5)+(6*7)+16):6]

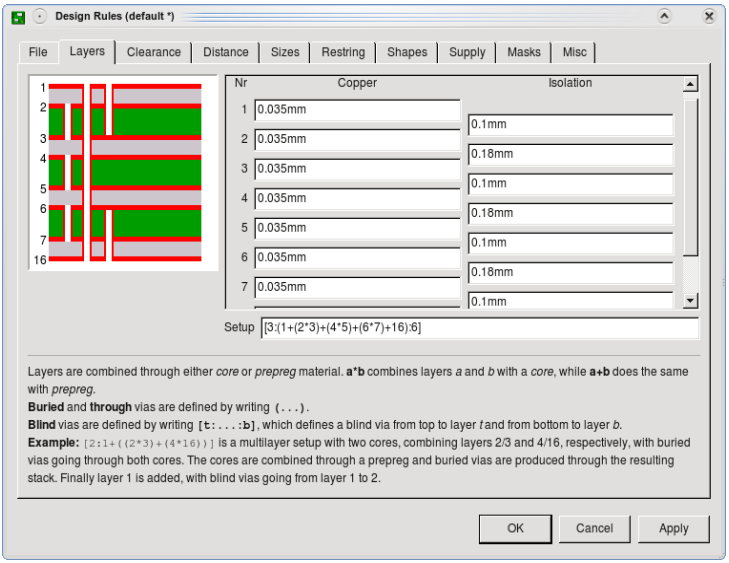
Explanation:

(2*3)+(4*5)+(6*7)
Three cores, each with buried vias, are pressed together and isolated with prepregs.

1+(2*3)+(4*5)+(6*7)**+16**
Outer copper layers 1 and 16 which are isolated through prepregs are pressed onto this layer stack.

(1+(2*3)+(4*5)+(6*7)+16**)**
Parenthesis allow through vias from 1-16.

[3:(1+(2*3)+(4*5)+(6*7)+16)**:6]**
Blind vias from 1-3 and 16-6.



➢ *Example 5: Layer Setup for an 8 layer board*

Hints For Working With Blind, Buried, and Micro Vias

VIA command

Depending on the Layer setup vias can have different lengths. The parameter toolbar of the VIA command shows all available lengths in the *Layer* box. When routing manually (ROUTE command) EAGLE takes the shortest possible via length in order to change layers. It is also possible that vias at the same position are elongated.

The via length can be changed with the CHANGE VIA command. Select the value from the according menu and click the via with the left mouse button.

Alternatively use the command line:

```
CHANGE VIA 2-7
```

and a click onto the via changes the length from layer 2 to 7.
If the given via length is not defined in the Layer setup it will be elongated to the next possible length or, if this is not possible, an error message will be generated.

```
VIA 'GND' 1-4 (1.05 2)
```

places a via that belongs to the signal GND and reaches from layer 1 to 4 at position (1.05 2).

ROUTE Command

If you want to change the layer while laying-out the board, EAGLE always takes the shortest possible via (CHANGE LAYER command; also in Follow-me mode). It is also possible that a via at the same position is elongated automatically.

> *If Micro vias are enabled in the Design Rules by setting a minimum value for the drill diameter (Sizes tab, Min. Micro Via) and defining a proper Layer setup, EAGLE sets a Micro via when routing from a SMD and immediately changing to the next inner layer.*
>
> *In Follow-me mode, however, EAGLE can't place Micro vias. The Follow-me router is powered by the Autorouter engine and therefore it has to follow its properties and restrictions.*

Autorouter

The Autorouter supports Blind and Buried vias.

If you decide to use the Autorouter for a multilayer board that uses Blind and Buried vias it is not allowed to work with automatically generated supply layers (see page 159). In this case inner layers can be defined with the help of polygons.

Already existing supply layers can be transformed into layers with supply polygons by

- renaming the layer (no $ character at the beginning of the name),
- removing possibly existing isolation wires,
- drawing a polygon covering the whole layout, and
- naming the polygon with the favoured signal name.

Micro Via – A Special Case of Blind Via

In contrast to Blind vias that can reach several layers deep into the board the Micro via connects an outer layer with the next inner layer. The drill diameter of a micro via is relatively small. Presently the usual values are about 0.1 to 0.05 mm.

For manufacturing reasons Micro vias, as Blind vias, have to follow a certain A*spect ratio* of depth to drill diameter. This ratio defines the maximum via depth for a certain drill diameter.
The proper value can be learned from your board house.

Set this value in the Design Rules, *Sizes* tab, *Min. Blind Via Ratio.*
Assumed the board house demands the ratio as 1:0.5 you have to enter `0.5` for *Min. Blind Via Ratio*.

Additionally the Design Rule Check verifies the minimum drill diameter for Micro vias given in *Min. MicroVia*. If this value is higher than the value for *Minimum Drill* (default), micro vias won't be checked.

The diameter of micro vias is set in the *Restring* tab of the Design Rules.

If you change the layer from an outer to the next inner one while you are routing a track out of a SMD, EAGLE automatically places a Micro via, provided the Design Rules allow it.

The Autorouter can't set Micro vias!

6.6 Updating Components (Library Update)

The UPDATE command allows components in a schematic diagram or a layout to be replaced by components defined in accordance with the current libraries. This function is of particular interest for existing projects. If, in the course of development, the definitions of Packages, Symbols or Devices in the libraries are changed, the existing project can be adapted to them.

The menu item *Library/Update* causes all the components in a project to be compared with the definitions in the current libraries. If EAGLE finds differences, the components are exchanged.

Those libraries on the path specified for *Libraries* in the Control Panel under *Options/Directories* will be examined.

It is also possible to update components from one particular library. Type the UPDATE command on the command line, stating the library, for instance as:

```
UPDATE linear
```

or

```
UPDATE /home/mydir/eagle/library/linear.lbr
```

or select the library in the File dialog of the *Library/Update...* menu item.

In the case you want to replace parts from one library with parts from another library you can use the command:

```
UPDATE old-lbr-name = new-lbr.name
```

Old-lbr-name represents the name of the library as shown by the INFO command in the layout or schematic. *New-lbr-name* stands for the library from which you want to take elements. You may add paths as well.
Please see the help function for more information.

In many cases you will be asked whether Gates, pins or pads should be replaced according to name or according to position. This always happens if library objects are renamed, or if their position (sequence) is changed.

If too many changes are made in the library at one time (e.g. pin names and pin positions are changed) it is not possible to carry out an automatic adaptation. In such a case it is possible either to carry out the modifications to the library in two steps (e.g. first the pin names and then the pin positions), or the library element can be given a new name, so that it is not exchanged.

Changing a Device's prefix in the library does not update the part names of already placed elements in your drawing.

If Forward&Back Annotation is active, the components are replaced in the schematic diagram and in the layout at the same time.

You will find further information on the program's help pages.

After any library update, please carry out both an ERC on the schematic and a DRC on the layout!

Individual components can, for instance, be updated with the aid of the ADD command. If you use ADD to fetch a modified component from a library, you will be asked whether all the older definitions of this type should be updated.

After the update you can delete the component that you just fetched.

Again here it is wise to carry out an ERC and a DRC after the update!

6.7 Print Out Schematic and Layout

Schematic diagrams, boards and also library elements can be printed out with the PRINT command.

Using DISPLAY you should first select the layers that you want to print.

The basic rule is: If you can see it in the editor, you will see it on the print.

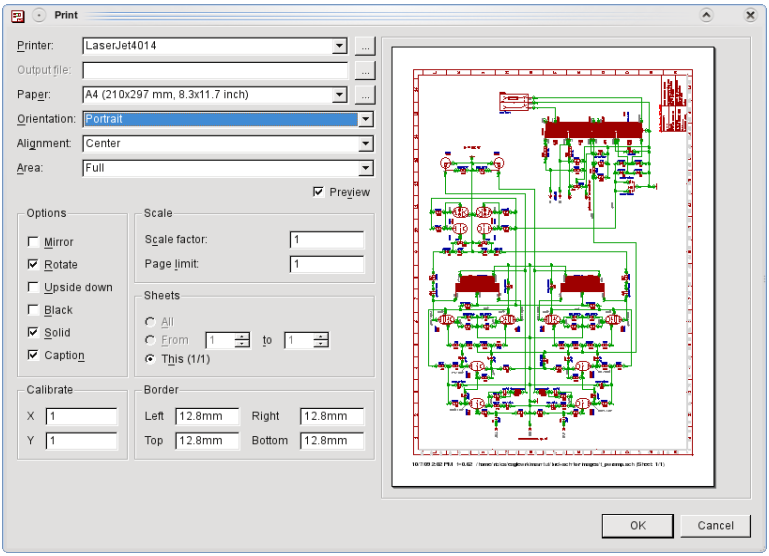
Exceptions to the rule above are:

- Origin crosses for texts
- Grid lines or grid dots
- Polygons that can't be calculated by RATSNEST and therefore only show their contours in the Layout Editor
- Error polygons of the Design Rule Check

Settings of the Print Dialog

When the printer icon on the action toolbar is clicked, the PRINT dialog opens.

The currently selected printer is shown at the top of the window in the *Printer* line. The small button on the right, at the end of the line, can be used to select another printer or activate one of the print-to-file options. If a printer is selected, the button with the three dots ... leads you to the printer properties.



➢ ***The PRINT window***

In case you selected a print-to-file option the *Output file* line shows the path to the output file. If you want to change it, click onto the … button.

Below these two lines you will find settings about *Paper* format, *Orientation* and *Alignment* of your print. The … button in the *Paper* line allows you to define a user-specific format, provided the selected printer supports this.

Alignment defines the location of the print-out on the paper. Changing this will directly result in a modified *Preview*, if active.

In the *Area* line, you determine what to print: *Window* prints the drawing window which is currently visible in the Editor window. *Full* on the other hand, prints the whole drawing. In this case all drawing objects (displayed or not) are relevant for the calculation of the resulting printing area.

Printing Options

Mirror inverts the drawing from left to right about the Y axis, *Rotate* turns it 90 degrees counterclockwise, and *Upside down* turns it through 180 degrees. If both are activated, a rotation of 270 degrees is the result.
If the *Black* option is chosen, a black-and-white printout is made. Otherwise the print will be either in color or gray scale, depending on the printer.
Solid causes each object to be entirely filled. If you want to see the different filling patterns of the individual layers, then deactivate this option.
The *Caption* option switches the appearance of the title, printing date, file name and the scale of the print on or off.

In the *Scale* section of the window the *Scale factor* specifies the scale of the drawing. It may be in the range of 0.001 and 1000.
If *Page limit* is set to 0, the printer will use whatever number of pages is

needed to print the output at the selected scale. If a different value is selected, EAGLE will adjust the scale of the drawing to fit it onto the stated number of pages. This can mean that, under unfavourable circumstances, the selected scale cannot be maintained.
Otherwise you have the possibility to select *Page Limit 1,* and a *Scale factor* that would request more than one page for printing to get a maximum filling of the page.

It is possible to select which sheets from a schematic diagram are printed using the *Sheets* box. This only appears in the Schematic Editor. This selection also determines which sheet is shown in the preview.

The edges of the print can be defined with the aid of the four entry boxes under *Border*. The values may be entered in mm or in inches. If you have changed the values and want to use the printer driver's standard settings again, simply enter a *0*.

Calibrate allows correction factors for the aspect ratio of the printout. This allows linear errors in the dimensional accuracy of the print to be corrected. The values can be specified in the range of 0.1...2.

> *Note concerning colored printing:*
> *EAGLE always takes the white palette as basis for colored printouts. If you are working with a black or colored background and using self-defined colors, it is recommended to define these colors also for the white palette. So the printer can use your colors, too.*

If, when a layout is printed, the drill holes in the pads and vias are not to be visible, select the *No Drills* option for the *Display mode* by way of the menu item *Options/Set/Misc*.

Generating PDF and Postscript files

If you want to generate a PDF file (resolution 1200dpi) or a Postscript file from your drawing, click onto the small selection button in the *Printer* line and choose the option *Print to file (PDF)* or *Print to file (Postscript)*. Go to the *Output file* line then and specify path and name of the PDF or Postscript output file.
All texts that are not written in EAGLE vector font are searchable in the PDF file by means of your PDF viewer.

Visibility and Sequence of Printed Layers

EAGLE prints its layers in a certain sequence, one over the other. If you are using, for example, self-defined layers that are hidden by other layers in the print-out, you can use a SET command option – SET Option.LayerSequence – for bringing them into the foreground, or in general, for defining the layer printing sequence. This affects printing into a PDF or Postscript file, as well.
Details about this can be found in the help function of the SET command, *Help/Editor commands/SET*.

The PRINT command can also be given directly on the command line, or can be run by a script file. Information about the selection of options is available on the help pages for PRINT.

> *For all those who have experiences with EAGLE versions before 4.0: EAGLE comes with an User Language Program named cam2print.ulp that converts your previous EAGLE 3.5x CAM jobs into script files in order to use them with the PRINT command.*
>
> *Another User Language Program cam2dxf.ulp allows to use a previous CAM job for generating DXF data.*

6.8 Combining Small Circuit Boards on a Common Panel

In order to save costs, it may be worth supplying, for example, a smaller board to the board manufacturer in the form of a multiple board. So you can have several boards made in one step.

You can reproduce the layout or combine different layouts to create a multiple board with the GROUP, CUT and PASTE commands. Please note that this will change the board's silk screen, since elements receive new names, if a certain designator is already used in the board when pasting from the buffer. If you don't need the silkscreen this does not matter. Otherwise a User Language program can help. *Panelize.ulp* copies the texts written in the layers 25 and 26 (*t/bNames*) into two new layers 125 and 126. When combining the boards the names of the parts will change anyway, the copied texts in those new layers however will remain unchanged.
Tell the board manufacturer that they have to take layers 125 and 126 instead of the original layers 25 and 26 to generate the silkscreen from.

Procedure:

- Load the board file.
- Run *panelize.ulp* to copy name texts.
- DISPLAY all layers.
- Use GROUP to select all objects to be copied.
  To select the whole layout you could also use GROUP ALL.
- Activate CUT and left-click with the mouse into the drawing area.
- Edit a new board file with *File/New* .
- Use PASTE to place the layout as often as wanted. Should the situation arise, it is possible to specify an orientation for the group before fixing it.

- Please make sure that the new board has the same set of Design Rules as the original board file has. It is possible to export Design Rules into a file (*.dru) and then import it into another board file (*Edit/Design rules* menu, *File* tab).
- Save the new board file.
- Tell your board house that they have to use layers 125/126 instead of 25/26.

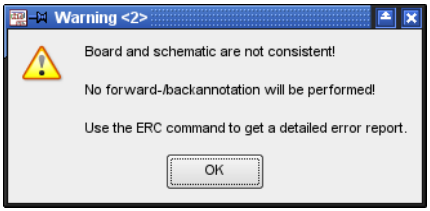
This way you can also combine boards from different projects.

> *In the case the board has Supply layers, you have to check the names of the power signals. For example, the signal GND will be displayed as GND1, GND2 and so on. The connections to the Supply layer $GND no longer exist. You have to name the signals with their original name GND again.*

6.9 Consistency Lost between Schematic and Layout

It is very important during the design that the content of the schematic and the layout exactly correspond to allow for design congruency. Eagle uses a Forward&Back annotation to perform this task. General information about this can be found in the chapter about Forward&Back Annotation beginning with page 98.

The interconnection between Schematic Editor and Layout Editor ensures that both are in lock-step from a design standpoint automatically, provided both files are always loaded at the same time. If you close one of them, either the schematic or layout file, and continue your work in the remaining opened file the consistency will be lost. EAGLE will not be able to transfer the modifications into the other file. So differences will arise between Schematic and Layout.



➢ ***Consistency loss between Schematic and Layout***

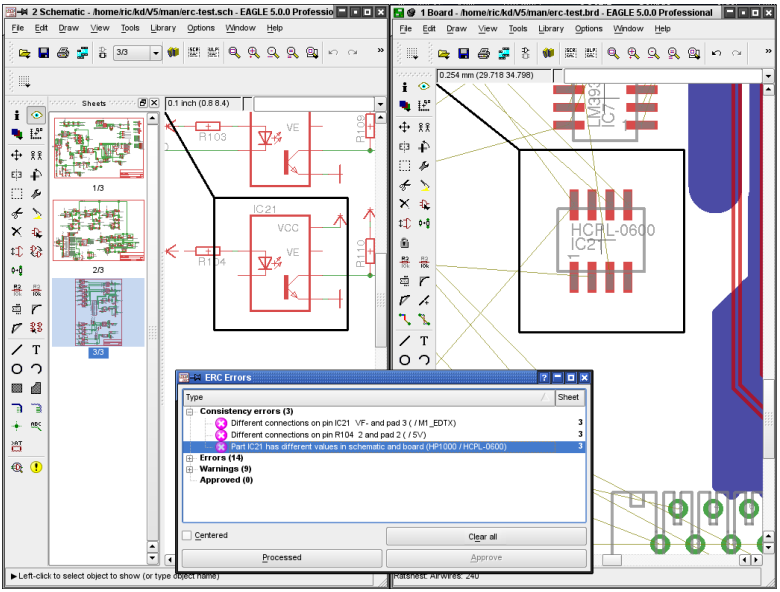
EAGLE will prompt a warning as soon as you want to switch to the referring file or start this project again.

Start the Electrical Rule Check (ERC) immediately. It compares both files and reports differences in the ERC Errors window's *Consistency Errors* branch. If you click onto one of these entries, EAGLE marks the affected object in Schematic and Board, if possible.

Process each message and resolve the difference in the Schematic or in the Layout Editor window, according to requirements. Finally you can mark the entry in the list as done with the *Processed* button.

Launch the ERC everytime a change has been made for design verification and to get an overview of progress. All differences are cleared, if ERC reports consistency. Now the Annotation will work again and the board and schematic are again in lock-step with each other.

Don't forget to save the files now and remember to leave both files loaded simultaneously all the time.



➢ ***The differences are marked in both editor windows***

Chapter 7
The Autorouter

7.1 Basic Features

- Any routing grid (min. 0.02 mm)
- Any placement grid (min. 0.1 micron)
- SMDs are routed on both sides
- The whole drawing area can be the routing area (provided enough memory is available)
- The strategy is selected via control parameters
- Simultaneous routing of various signal classes with various track widths and minimum clearances
- Common data set (Design Rules) for the Design Rule Check and the Autorouter
- Multilayer capability (up to 16 layers can be routed simultaneously, not only in pairs)
- Support of Blind and Buried vias
- The preferred track direction can be set independently for each layer: horizontal and vertical, true 45/135 degrees (important for inner layers!)
- Ripup and retry for 100 % routing strategy
- Optimization passes to reduce vias and smooth track paths
- Prerouted tracks are not changed
- Serves a basis for the Follow-me router, a special operating mode of the ROUTE command that allows automatic routing of selected signals

7.2 What Can be Expected from the Autorouter

The EAGLE Autorouter is a "100%" router. This means that boards which, in theory, can be completely routed will indeed be 100% routed by the Autorouter, provided - and this is a very important restriction - the Autorouter has unlimited time. This restriction is valid for all 100%

Autorouters whatsoever. However, in practice, the required amount of time is not always available, and therefore certain boards will not be completed even by a 100% Autorouter.

The EAGLE Autorouter is based on the ripup/retry algorithm. As soon as it cannot route a track, it removes prerouted tracks (ripup) and tries it again (retry). The number of tracks it may remove is called ripup depth which is decisive for the speed and the routing result. This is, in principle, the previously mentioned restriction.

Those who expect an Autorouter to supply a perfect board without some manual help will be disappointed. The user must contribute his ideas and invest some energy. If he does, the Autorouter will be a valuable tool which will greatly reduce routine work.

Working with the EAGLE Autorouter requires that the user places the components and sets control parameters which influence the routing strategy. These parameters must be set carefully if the best results are to be achieved. They are therefore described in detail in this section.

7.3 Controlling the Autorouter

The Autorouter is controlled by a number of parameters. The values in the current Design Rules, the net classes and special Autorouter control parameters all have an effect.

The Design Rules specify the minimum clearances (DRC commands for setting *Clearance* and *Distance*), the via diameter (*Restring* setting) and the hole diameter of the vias (*Sizes* setting). The minimum track width is also specified.

The net classes - if any are defined - specify special minimum clearances, track widths and the hole diameters for vias carrying particular signals.

There is also a range of special cost factors and control parameters that can be changed via the Autorouter menu. They affect the route given to tracks during automatic routing. Default values are provided by the program. The control parameters are saved in the BRD file when the layout is saved. You can also save these values in an Autorouter control file (**.ctl*). This allows a particular set of parameters to be used for different layouts. Neither Design Rules nor the data for various net classes are part of the control file.

A routing process involves a number of separate basic steps:

Bus Router

Normally the bus router starts first.
It deals with signals which can be routed in the preferred direction with only slight deviation in x and y direction allowed. The bus router takes only those signals into consideration that belong to net class 0.
This step may be omitted.

Buses, as understood by the Autorouter, are connections which can be laid as straight lines in the x or y direction with only a few deviations.
It has nothing in common with buses in the meaning of electronics, for example, address buses or the like.

Routing Pass

The actual routing pass is then started, using parameters which make a 100% routing as likely as possible. A large number of vias are deliberately allowed to avoid paths becoming blocked.

Optimization

After the main routing pass, any number of optimization passes can be made. The parameters are then set to remove superfluous vias and to smooth the track paths. In the optimization passes tracks are removed and rerouted one at a time. This can, however, lead to a higher degree of routing, since it is possible for new paths to be freed by the changed path of this track.
The number of optimization passes must be specified before starting the Autorouter. It is not possible to optimize at a later stage. Once the routing job has been completed all the tracks are considered to have been prerouted, and may no longer be changed.

Any of the steps mentioned above may be separately activated or deactivated.

7.4 What Has to be Defined Before Autorouting

Design Rules

The Design Rules need to be specified in accordance with the complexity of the board and of the manufacturing facilities available. You will find a description of the procedure and of the meanings of the individual parameters in the section on *Specifying the Design Rules* on page 128.

Track Width and Net Classes

If you have not already defined various net classes in the schematic diagram you now have the opportunity, before running the Autorouter, of specifying whether particular signals are to be laid using special track widths, particular clearances are to be observed, or whether certain drill diameters are to be used for vias for particular signals. Please consult the help pages (CLASS command) or the section on *Specifying Net Classes* on page 117 for information about the definition of net classes.

If no special net classes are defined, the values from the Design Rules apply. The value *Minimum width* in the *Sizes* tab determines the track width, the values for minimum clearances/distances are taken from the *Clearance* and

Distance tabs. The diameter of vias is defined by the values in the *Restring* tab.

> *Did you set values in the Design Rules and for net classes? In this case the Autorouter follows the higher value.*

Grid

The Design Rules determine the routing and placement grid. The minimum routing grid is 0.02 mm, which is about 0.8 mil.

Placement Grid

Although the Autorouter does permit any placement grid, it is not a good idea to place the components on a grid that is too fine. Two good rules are:

- The placement grid should not be finer than the routing grid.
- If the placement grid is larger than the routing grid, it should be set to an integral multiple of the routing grid.

These rules make sense if, for example, you consider that it might be possible, within the Design Rules, to route two tracks between two pins of a component, but that an inappropriate relationship between the two grids could prevent this (see diagram).

Routing Grid

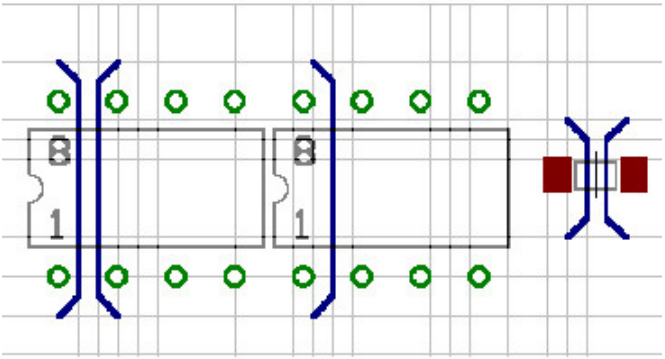
Please note that the Autorouter grid has to be set in the AUTO command's *Autorouter Setup* Window, *General* tab, *Routing Grid*. This is not the same as the currently used grid in the Layout Editor window that you have selected with the GRID command.

Bear in mind that for the routing grid the time demand increases exponentially with the resolution. Therefore select as large a grid as possible. The main question for most boards is how many tracks are to be placed between the pins of an IC. To answer this question, the selected Design Rules (i.e. the minimum spacing between tracks and pads or other tracks) must of course also be considered.

The result is:

> *The two grids must be selected so that component's pads are located on the routing grid.*

There are of course exceptions, such as with SMDs to which the opposite may apply, namely that a position outside of the routing grid leads to the best results. In any event the choice of grid should be carefully considered in the light of the Design Rules and the pad spacing.



➢ ***Track patterns with different placement grids***

The example above may clarify the situation:

For the component on the left, the pads are placed on the routing grid. Two tracks can be routed between two pads. The pads of the component in the middle are not on the routing grid, and therefore only one track can be routed between them.

On the right you see the exception from the rule shown for SMD pads, which are placed between the routing grid lines so that one track can be routed between them.

When choosing the grid, please also ensure that each pad covers at least one grid point. Otherwise it can happen that the Autorouter is unable to route a signal, even though there is enough space to route it. In this case the Autorouter issues the message *Unreachable SMD at x y* as it starts. The parameters x and y specify the position of the SMD pad.

The default value for the routing grid is 50 mil. This value is sufficient for simple through-hole layouts. Working with SMD components demands a finer routing grid.
Usual values are 25, 12.5, 10, or 5 mil.

> *Please remember that finer routing grids require significantly more routing memory.*

Memory Requirement

The amount of routing memory required depends in the first place on the selected routing grid, the area of the board and the number of signal layers in which tracks are routed.

The static memory requirement (in bytes) for a board can be calculated as follows:

```
number of grid points x number of signal layers x 2
```

Note that a supply layer generated by giving the appropriate name (*$name*) does not occupy any autorouting memory. On the other hand, a supply layer that has been created with the aid of one or more polygons consumes as much routing memory as any other signal layer.

Space is also required for dynamic data, in addition to the static memory requirement. The dynamic data require in a very rough estimate about 10% up to 100% (in some cases even more!) of the static value. This depends heavily on the layout.

Total memory requirement (rough approximation):

```
static memory x (1.1..2,0) [bytes]
```

This much RAM should be free before starting the Autorouter. If this is insufficient, the Autorouter must store data on the hard disk. This lengthens the routing time enormously, and should be avoided at all costs. Short accesses to the hard disk are normal, since the job file on the hard disk is regularly updated.

> *Try to choose the coarsest possible routing grid. This saves memory space and routing time!*

Layer

If you want to design a double-sided board, then select Top and Bottom as route layers. You should only use the Bottom layer for a single-sided board. In the case of inner layers, it is helpful to use the layers from the outside to the inside, i.e. first 2 and 15 and so on.
Inner layers are converted to supply layers if they are renamed to ***$name***, where *name* is a valid signal name. These layers are not routed.
Supply layers with more than one signal can be implemented with polygons. These layers are treated as normal signal layers.

In the case of boards that are so complex that it is not certain whether they can be wired on two sides, it is helpful to define them as multilayer boards, and to set very high costs for the inner layers. This will cause the Autorouter to avoid the inner layers and to place as many connections as possible in the outer layers. It can, however, make use of an inner layer when necessary.

These settings are made in the Autorouter menu (see page 183).

The autorouter shows the message *Unreachable SMD in layer...*, if a layer that contains SMDs is not active. Clicking *OK* starts the autorouter nevertheless. If you want to change the autorouter setup click *Cancel*.

Preferred Directions

On the two outside layers the preferred directions are normally set to 90 degrees from each other. For the inner layers it may be useful to choose 45 and 135 degrees to cover diagonal connections. Before setting the preferred direction it is well worth examining the board (based on the airwires) to see

if one direction offers advantages for a certain side of the board. This is particularly likely to be the case for SMD boards.

> *Please also follow the preferred direction when pre-placing tracks. The defaults are vertical for the Top (red) and horizontal for the Bottom (blue) layer.*

Experience has shown that small boards containing mainly SMD components are best routed without any preferred direction at all (set * in the Autorouter setup). The router then reaches a usable result much faster.

Single sided boards should be routed without a preferred direction.

Restricted Areas for the Autorouter

If the Autorouter is not supposed to route tracks or place vias within certain areas, you can define restricted areas by using the commands RECT, CIRCLE, and POLYGON in the layers 41, *tRestrict*, 42, *bRestrict, and 43, vRestrict.*

tRestrict: Restricted areas for Wires and Polygons in the Top layer.
bRestrict: Restricted areas for Wires and Polygons in Bottom layer.
vRestrict: Restricted areas for Vias.

Such restricted areas can already be defined in a Device or Package (around, for instance, the fixing holes for a connector, or for a flat-mounted transistor under which there should not be any tracks).

Wires drawn in layer 20, *Dimension*, are boundary lines for the Autorouter. Tracks cannot be laid beyond this boundary.
Typical application: board boundaries.

An area drawn in layer 20 can also be used as a restricted region for all signals. It should, however, be noted that this area should be deleted before sending the board for manufacture, since layer 20 is usually output during the generation of manufacturing data.

Cost Factors and Other Control Parameters

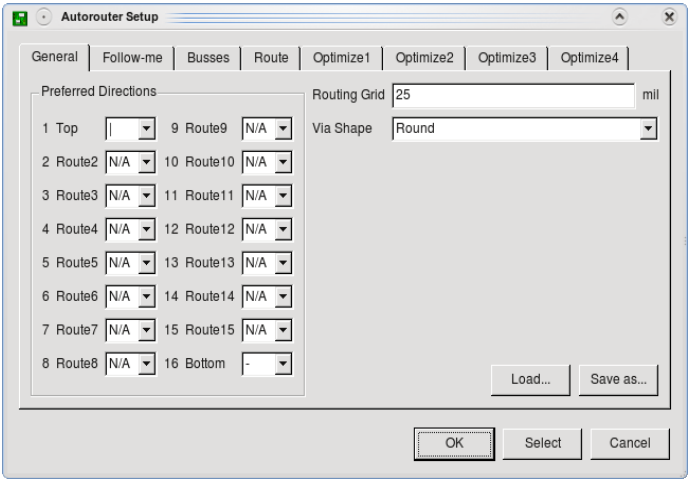
The default values for the cost factors are chosen on the basis of our experience in such a way as to give the best results.

The control parameters such as *mnRipupLevel*, *mnRipupSteps* etc. have also been set to yield the best results according to our experience.

We want to emphasize, that we recommend working with the default values. If you nevertheless do want to experiment with these parameters, please consider the description of the cost factors in the following section. In the case of many parameters even small alterations can have large effects.

7.5 The Autorouter Menu

When running the Autorouter with the AUTO command, the setup menu appears first. All the necessary settings are made there.



➢ ***Autorouter setup: General settings (General tab)***

This is where you specify the layers that may be used for routing and which preferred directions apply. Click in the appropriate combo box with the mouse, and select the desired value.

Setting the preferred directions:

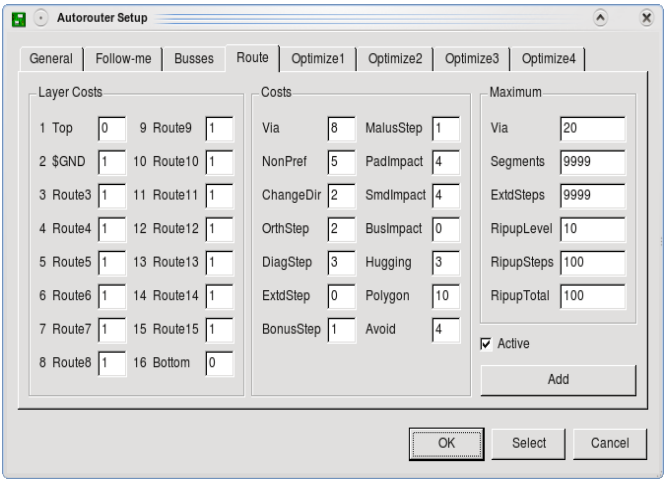
-	horizontal
|	vertical
/	diagonal at 45°
\	diagonal at 135°
*	none
N/A	Layer not active

Define a suitable *Routing grid* and select the shape of the through-plated holes (*Via shape*).

You may use the *Load...* and *Save as....* buttons to load a different parameter set from an Autorouter control file (**.ctl*) or to save the current settings for further projects.

You now make the settings for the individual routing steps. Click onto the corresponding tab and make any settings you may want.

> *The Follow-me tab is not relevant for the Autorouter pass itself. You may, however, define the settings for the Follow-me router in this tab, because it uses some of the Autorouter parameters, too. More details about the Follow-me router can be found in a later section of this chapter.*



➢ ***Autorouter setup: Settings for the Route pass***

The parameters in the *Layer costs*, *Costs* and *Maximum* groups can be different for each pass. Further information about this can be found in the next section.

The *Active* check box specifies whether this step should be executed or not.

Additional optimization passes can be inserted with the *Add* button.

Clicking onto the *Select* button allows certain signals to be selected for autorouting. Select these with a mouse click onto the respective airwires. Then start the Autorouter with a click on the traffic-light icon in the action toolbar.
It is, alternatively, possible to enter the signal names on the command line. Examples:

```
VCC GND ;
```

The signals VCC and GND will be routed.
The semicolon at the end of the line starts the Autorouter immediately. It is alternatively possible to click on the traffic-light icon.

If you type in the command line

```
! VCC GND ;
```

all signals except VCC and GND will be routed.

You may use wildcards for the signal selection, as well. Allowed is

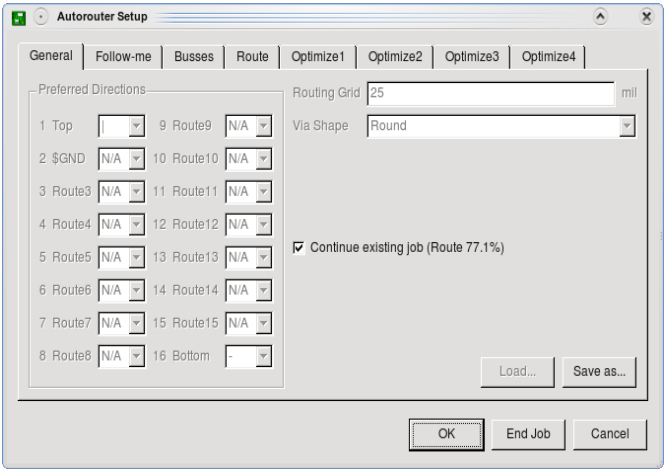
*	which matches any number of any characters.
?	which matches exactly one character.
[...]	which matches any of the characters between the brackets, for example [a-f], for all characters from a to f.

The Autorouter is started for all the signals that have not yet been laid by clicking on the *OK* button.

The *Cancel* menu item interrupts the AUTO command without storing any changes.

You are not allowed to make any changes to the parameters, if you want to restart an interrupted routing job. Use the *Continue existing job* check box to decide whether you want to continue with an existing job, or whether you want to choose new settings for the remaining unrouted signals.

End job ends the autorouting job and loads the previous routing result.

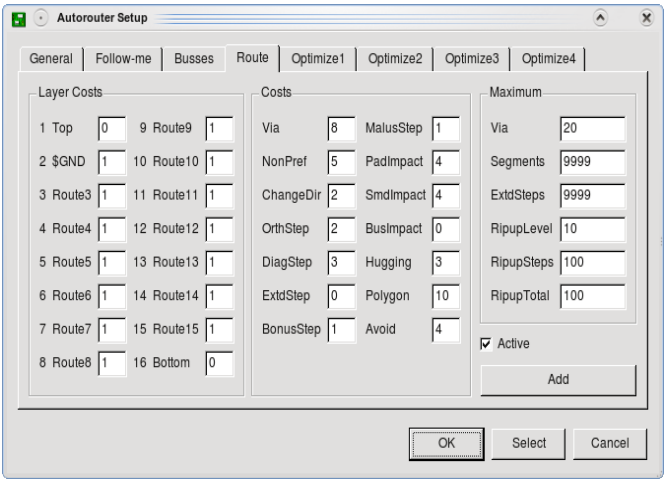


➢ ***Autorouter setup: Restarting an interrupted job***

7.6 How the Cost Factors Influence the Routing Process

Values between 0..99 are possible for each cost factor (***cf****xxx*), but the full range is not useful for all parameters. Sensible values are therefore given with each parameter.
The control parameters (***mn****xxx*) accept values in the range 0..9999. Reasonable figures are also provided under each parameter.

The parameter can be set by the Autorouter Setup Menu. The settings for *Route* and the *Optimize* passes can be configured separately. The menu is split into three sections, *Layer Costs*, *Costs*, *Maximum*.



➢ ***Autorouter: Parameter for Route***

The following section shows the available parameters and their effects. The names of the parameters are the same as they would be used in an Autorouter control file *.ctl. Details about this can be found in *Parameters of a Control File* beginning with page 193.

Layer Costs

cfBase.xx: 0..20

Base costs for one step on the corresponding layer. Recommendation: outside layers (Top, Bottom) always 0, inside layers greater than 0.

Costs

cfVia: 0..99

Controls the use of vias. A low value produces many vias but also allows the preferred direction to be followed. A high value tries to avoid vias and thus violates the preferred direction. Recommendation: low value for the routing pass, high value for the optimization.

cfNonPref: 0..10

Controls following of the preferred direction. A low value allows tracks to be routed against the preferred direction, while a high value forces them into the preferred direction.
If *cfNonPref* is set to 99, track sections can only be placed in the preferred direction. Only select this value if you are certain that this behavior is really wanted.

cfChangeDir: 0..25

Controls how often the direction is changed. A low value means many bends are allowed within a track. A high value produces virtually straight tracks.

cfOrthStep, cfDiagStep

Implements the rule that the hypotenuse of a right-angled triangle is shorter than the sum of the other two sides. The default values are 2 and 3. That means that the costs for the route using the two other sides are 2+2, as against 3 for the hypotenuse. These parameters should be altered with great care!

cfExtdStep: 0..30

Controls the avoidance of track sections which run at an angle of 45 degrees to the preferred direction, and which would divide the board into two sections. A low value means that such sections are allowed while a high value tries to avoid them. In combination with the parameter *mnExtdStep* you can control the length of these tracks. If *mnExtdStep* = 0, each grid step at 45 degrees to the preferred direction causes costs that are defined in parameter *cfExtdStep*. Choosing for example *mnExtdStep* = 5 allows a track to run five steps at 45 degrees without any additional costs. Each further step causes costs defined in *cfExtdStep*.
In this way, 90 degree bends can be given 45 degree corners. Settings like *cfExtdStep* = 99 and *mnExtdStep* = 0 should avoid tracks with 45 degree angles.
This parameter is only relevant to layers which have a preferred direction. Recommendation: use a lower value for the routing pass, and a higher value for the optimization.

cfBonusStep, cfMalusStep: 1..3

Strengthens the differentiation between *preferred* (*bonus*) and *bad* (*malus*) areas in the layout. With high values, the router differentiates strongly between *good* and *bad* areas. When low values are used, the influence of this factor is reduced. See also *cfPadImpact, cfSmdImpact*.

cfPadImpact, cfSmdImpact: 0..10

Pads and SMDs produce *good* and *bad* sections or areas around them in which the Autorouter likes (or does not like) to place tracks. The *good* areas are in the preferred direction (if defined), the *bad* ones perpendicular to it. This means that tracks which run in the preferred direction are routed away from the pad/SMD. With high values the track will run as far as possible in the preferred direction, but if the value is low it may leave the preferred direction quite soon.
It may be worth selecting a somewhat higher value for *cfSmdImpact* for densely populated SMD boards.

cfBusImpact: 0..10

Controls whether the ideal line is followed for bus connections (see also *cfPadImpact*). A high value ensures that the direct line between start and end point is followed. Only important for bus routing.

cfHugging: 0..5

Controls the hugging of parallel tracks. A high value allows for a strong hugging (tracks are very close to each other), a low value allows for a more generous distribution. Recommendation: higher value for routing, lower value for the optimization.

cfAvoid 0..10

During the ripup, areas are avoided from which tracks were removed. A high value means strong avoidance.
Not relevant to the optimization passes.

cfPolygon 0..30

If a polygon has been processed with the RATSNEST command and therefore is displayed as a filled area before you start the Autorouter, every step within the polygon is associated with this value. A low value makes it easier for the Autorouter to route traces inside the polygon area. The probability, however, that the polygon is broken into several pieces is higher. A higher value causes the Autorouter to make fewer connections inside the polygon.

If a polygon is in outline mode and not processed by RATSNEST before you start the Autorouter, it won't be taken into consideration at all. *cfPolygon* does not play a role for such polygons.

Maximum

mnVia 0..30

Controls the maximum number of vias that can be used in creating a connecting track.

mnSegments 0..9999

Determines the maximum number of wire pieces in one connecting track.

mnExtdSteps 0..9999

Specifies the number of steps that are allowed at 45 degrees to the preferred direction without incurring the value of *cfExtdStep*.
See also *cfExtdStep*.

Additionally can be found the parameters *mnRipupLevel*, *mnRipupSteps,* and *mnRipupTotal.* Those are described in the following section.

7.7 Number of Ripup/Retry Attempts

Due to the structure of the Autorouter there are some parameters which influence the ripup/retry mechanism. They are set in such a way that they offer a good compromise between time demand and routing result. The user should therefore only carefully change the values for
mnRipupLevel, *mnRipupSteps* and *mnRipupTotal* when needed.

As a rule, high parameter values allow for many ripups but result in increased computing times.

To understand the meaning of the parameters you need to know how the router works.

To begin with the tracks are routed one after the other until no other path can be found. As soon as this situation occurs, the router removes up to the maximum number of already routed tracks (this number has been defined with *mnRipupLevel*) to route the new track. If there are eight tracks in the way, for example, it can only route the new track if *mnRipupLevel* is at least eight.

After routing the new track, the router tries to reroute all the tracks which were removed. It may happen that a new ripup sequence must be started to reroute one of these tracks. The router is then two ripup sequences away from the position at which, because of a track which could not be routed, it started the whole process. Each of the removed tracks which cannot be rerouted starts a new ripup sequence. The maximum number of such sequences is defined with the *mnRipupSteps* parameter.

The parameter *mnRipupTotal* defines how many tracks can be removed simultaneously. This value may be exceeded in certain cases.

If one of these values is exceeded, the router interrupts the ripup process and re-establishes the status which was valid at the first track which could not be routed. This track is considered as unroutable, and the router continues with the next track.

7.8 Routing Multi-Layer Boards

There are two different ways of implementing supply layers.

- Supply layers with a **$**-sign at the start of the layer name
- Supply layers with polygons

Supply Layers

The definition of supply layers has already been described in the previous chapter on page 159.

The following points must be considered when autorouting:

- The layer name specifies the signal that is carried by this layer. The layer $VCC, for example, only carries the VCC signal.
- The layer is represented with an inverse display.

- The layer is not active during autorouting (setting *N/A* in the Autorouter setup, *General* tab).
- After routing, you should draw an isolation wire around the layout. This prevents short-circuits at the edges of the board.
- No other signals or polygons may be drawn!

The Autorouter can't use supply layers in combination with Blind and Buried vias. Use inner layers with polygons instead!

The Autorouter cannot set Micro vias!

The Autorouter is allowed to set Blind vias that are shorter than the maximum depth defined in the Layer Setup.

Polygons as Supply Layers

It is possible with polygons to create supply layers that contain more than one supply voltage, and a few individual wires as well. Please note the instructions on page 158, *Ground Planes and Supply Layers with Several Signals.* These are not the kind of supply layer identified by a ***$*** in the name, but are in fact ordinary layers.

- Define the polygons before running the Autorouter.
- Give the appropriate signal names to the polygons.
- Use the RATSNEST command to let EAGLE calculate the polygon.
- Select the preferred directions and base costs (*cfBase*) for the layer in the Autorouter setup. A higher value of *cfBase* for the polygon layer causes the Autorouter to avoid these layers more strongly.
- After routing, check that the polygon still connects all the signal points. It is possible that the polygon was divided as a signal was laid. RATSNEST recomputes polygons, and issues the message *Ratsnest: Nothing to do!*, if everything is in order.

7.9 Backup and Interruption of Routing

As, with complex layouts, the routing process may take several hours, a backup is carried out at intervals (approx. every 10 minutes). The file *name.job* always contains the last status of the job. If the job is interrupted for any reason (power failure etc.) the computer time invested so far is not lost, since you can recall the status saved in *name.job*. Load your board file in the Layout Editor, and then enter:

```
AUTO;
```

Answer the prompt as to whether the Autorouter should recall (*Continue existing job?*) with *Yes*. The Autorouter will then continue from the position at which the job was last saved (a maximum of 10 minutes may be lost).

If the autorouting is interrupted via the stop icon, the *name.job* file also remains intact and can be recalled. This may be useful when you have started a complex job on a slow computer and want to continue with it on a fast computer as soon as one is available.

Please note that changing the parameters before recalling will not influence the job, since it will have been saved together with the parameters which were valid at the time of the initial Autorouter start.

When the Autorouter has finished, the routed board is saved as *name.b$$*. You can rename it to *name.brd* and use it, for instance, if a power failure occurred after the autorouting run and you could not save the board file. This file is deleted automatically after the board has been saved.

7.10 Information for the User

Status Display

During the routing, the Autorouter displays information on the actual routing result in the status bar.

➢ ***Autorouter: Status Bar***

The displayed values have the following meaning:

Route:

Result in % (hitherto maximum, best data)

Vias:

Number of vias in the layout

Conn:

Number of Connections `total/found/not routable`
Connections here means 2-point connections.

Ripup:

Number of `Ripups/current RipupLevel/cur. RipupTotal`

Number of ripups:
This indicates the number of connections that have already been routed during the foregoing routing procedure that have been (can be) removed in order to be able to route new signals.

Current RipupLevel:
This indicates the number of connections that have been removed or converted in airwires in order to lay the track for the current signal.

Current RipupTotal:
After a signal's routes have been ripped up it can be broken down into a large number of two-point connections. These connections are then routed again. This variable indicates the number of such two-point connections still to be routed.

Signals:

Signals `found/handled/prepared`,
if so followed by: `(routing_time signalname)`

In case the Autorouter needs more than about 5 seconds to lay-out a connection, EAGLE shows in parenthesis the routing time and the name of the currently processed signal.

Log file

For each routing pass the Autorouter generates a file called *name.pro*, containing useful information. Example:

```
EAGLE AutoRouter Statistics:

Job : d:/eagle4/test-design/democpu.brd

Start at : 15.43.18 (24.07.2000)
End at : 16.17.08 (24.07.2000)
Elapsed time : 00.33.48

Signals : 84 RoutingGrid: 10 mil Layers: 4

Connections : 238 predefined: 0 ( 0 Vias )

Router memory : 1121760

Passname:       Busses   Route    Optimize1  Optimize2  Optimize3  Optimize4

Time per pass:  00.00.21 00.08.44  00.06.32   00.06.15   00.06.01   00.05.55
Number of Ripups:   0       32         0          0          0          0
max. Level:         0        1         0          0          0          0
max. Total:         0       31         0          0          0          0

Routed:            16      238       238        238        238        238
Vias:               0      338       178        140        134        128

Resolution:       6.7 %  100.0 %   100.0 %    100.0 %    100.0 %    100.0 %

Final: 100.0 % finished
```

7.11 Parameters of a Control File

We see here how the individual parameters in an Autorouter control file (*name.ctl*) are used.

Parameter	Default	Meaning
RoutingGrid =	50Mil	Grid used by the Autorouter for tracks and via-holes
		Cost factors for...

```
cfVia         =     8     Vias
cfNonPref     =     5     Not using preferred direction
cfChangeDir   =     2     Changing direction
cfOrthStep    =     2     0 or 90 deg. Step
cfDiagStep    =     3     45 or 135 deg. Step
cfExtdStep    =    30     Deviation 45 deg. against preferred direction
cfBonusStep   =     1     Step in bonus area
cfMalusStep   =     1     Step in handicap area
cfPadImpact   =     4     Pad influence on surrounding area
cfSmdImpact   =     4     SMD influence on surrounding area
cfBusImpact   =     4     Leaving ideal bus direction
cfHugging     =     3     Wire hugging
cfAvoid       =     4     Previously used areas during ripup
cfPolygon     =    10     Avoiding polygons


cfBase.1       =    0     Basic costs for a step in the given layer
cfBase.2       =    1
...
cfBase.15      =    1
cfBase.16      =    0


                          Maximum number of...
mnVias         =   20     Vias per connection
mnSegments     = 9999     Wire segments per connection
mnExtdSteps    = 9999     Steps 45 deg. against preferred direction
mnRipupLevel   =  100     Ripups per connection
mnRipupSteps   =  300     Ripup sequences per connection
mnRipupTotal   =  200     Ripups at the same time


                          Track parameters for...
tpViaShape     = Round    Via shape (round or octagon)


PrefDir.1      =    |     Preferred direction in the given layer
PrefDir.2      =    0     Symbols: 0 - / |  \ *
                          0 : Layer not used for routing
PrefDir.15     =    0     * : No preferred direction
PrefDir.16     =    -     - : X is preferred direction
                          | : Y is preferred direction
                          / : 45 deg. is preferred direction
                          \ : 135 deg. is preferred direction
```

7.12 Practical Hints

This section presents you with some tips that have, over a period of time, been found useful when working with the Autorouter.

Look on these examples as signposts suggesting ways in which a board can be routed. None of these suggestions guarantee success.

General

The layer costs (*cfLayer*) should increase from the outer to the inner layers or be the same for all layers. It is unfavourable to use lower values in the inner layers than in the outer layers. This could increase the needed routing memory enormously.

> *The Autorouter can't layout wires as arcs!*
>
> *The Autorouter can't set micro vias!*
>
> *The Autorouter can't set Blind and Buried vias if inner layers are defined as Supply layers!*

Single-Sided Boards

There are two procedures, depending on the kind of layout:

In the simplest case, only layer 16, *Bottom*, is active. No preferred direction is defined. Select a suitable grid and run the Autorouter.

If the layout is rather more complex, it may be possible to achieve a usable result with special parameter settings. Please take a look at the project named *singlesided*, which can be found in the *eagle/projects/examples* directory. This example project comes with various control files (**.ctl), which* are optimized for singlesided routing.

The Autorouter may use the Top layer as well. The tracks laid there will be realized as wire bridges on the board. In layer 41, *tRestrict*, you can define restricted areas around the components and in regions where wire bridges are not allowed.

Feel free to experiment with the parameter settings for your layout.

SMD Boards With Supply Layers

The following procedure has been found effective:

The supply signals are routed first. In general, a short track is wanted from a SMD component to a via that connects to the inner layer.

Before altering the parameters, save the current (default) values in an Autorouter control file (CTL file). Click on the button *Save as..* in the *General* tab of the Autorouter setup window and input any name, for example, *standard.ctl*.

Now switch off the bus router and all the optimization passes in the Autorouter setup. Only the routing remains active. Alter the following cost factors:

```
cfVia = 0            Vias are welcome
mnVia = 1            Max. one via per connection
cfBase.1/16 = 30..99  Fewer tracks in Top/Bottom
mnSegments  =  2..8   Short tracks
```

Start the Autorouter, using the *Select* button, and choose the signals to be routed. After the routing pass it is possible, if appropriate, to optimize the result manually.

The rest of the connections are routed after this. Use AUTO to open the Autorouter setup menu, and load the previous stored control parameters with the *Load..* button (*standard.ctl*). Adjust the values to any special wishes you may have, and start the Autorouter.

This procedure is identical for both types of supply layer (automatically generated supply layer, or a supply layer generated with polygons).

What can be done if not all signals are routed?

If this happens, check your settings.

- Has a sufficiently fine routing grid been selected?
- Have the track widths got appropriate dimensions?
- Can the vias have smaller diameters?
- Have the minimum clearances been optimally chosen?

If it is either impossible or unreasonable to optimize these values any further, an attempt to achieve a higher level of routing may be made by increasing the ripup level. Observe the notes in the section on the *Number of Ripup/Retry Attempts* on page 190.

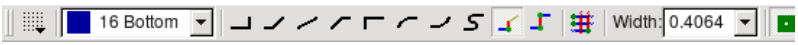
7.13 The Follow-me Router

To simplify the routing of airwires on the board, the ROUTE command offers two follow-me operating modes that can route a selected signal automatically. The position of the mouse cursor in the layout determines the trace of the connection.

For this function your license must provide the Autorouter module.

Partial and Full Mode

To start the Follow-me router, activate the ROUTE command and select the wire bend mode 8 or 9 from the parameter toolbar.



➢ ***Parameter toolbar of the ROUTE command***

After clicking onto an airwire, EAGLE calculates an appropriate trace and displays the connection. Moveing the mouse cursor will change the current trace. Trace processing depends on the complexity of the layout and may last some moments. It is recommended not to move the mouse cursor until the connection is displayed.

If you select wire bend mode 8 , the so-called *partial* mode, EAGLE calculates the trace of the selected signal, beginning with the mouse cursor position to the nearer end of the airwire, and display it. Fix the result with a mouse click. The remaining part of the airwire will be calculated dynamically. This means, that the airwire may point to another object that belongs to the signal, depending on the current mouse cursor position.

With wire bend mode 9 , the *full* mode, the Follow-me router calculates the trace in both directions simultaneously. A complete connection will be estblished. As soon as you are clicking onto an airwire, EAGLE begins to calculate the trace of the connection from the nearer end of the airwire to the current mouse position. It is not mandatory that the farer end of the airwire points always to its original position. Depending on the mouse cursor position this end point may direct you to another (nearer) location.

If it is not possible to draw a connection from the current mouse cursor position, the cursor turns into a small prohibition sign. Move the mouse and try to find a possible way for the connection. Maybe it is sufficient to change the layer at the current position. It could also be adviceable to adjust the Design Rules. Please keep in mind that restricted areas in the layers *t/bRestrict* or wires in the *Dimension* layer can hinder EAGLE to establish a connection.

Configuration

The Follow-me router respects Design Rules settings:
Values for *Clearance*, *Distance*, and *Size* will be taken in consideration, as well as particular values for net classes, if defined. Please be sure that the *Layer setup* in the *Layers* tab is properly set.

The current grid setting in the Layout editor serves as routing grid. Use the GRID command in order to change it. If there is already a signal assigned to mouse cursor, drop it, and select it again. Otherwise the grid change does not affect the connection.

The layer setting, which can be checked and changed in the parameter toolbar of the ROUTE command, displays the layer which has to be used at the mouse cursor position.

The Follow-me router reacts immediately on changes concerning wire width or drill diameter of vias. If the option *Auto set route width and drill* in the *Options/Set/Misc* menu is set, the Follow-me router adapts the given values

for wire width and via drill diameter from the Design Rules and from the net classes as soon as an airwire is selected.

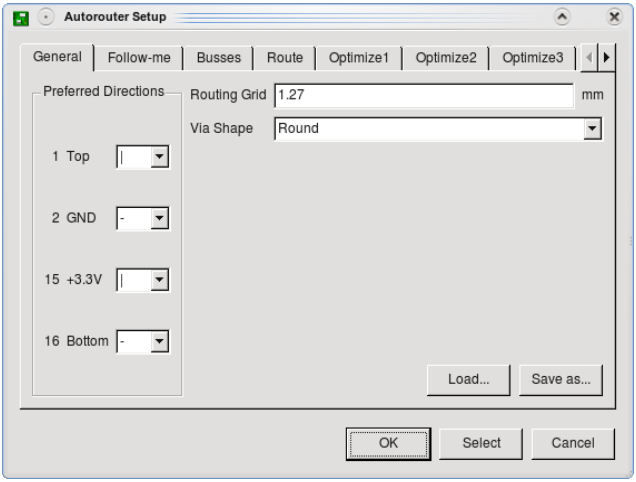
Routing Parameters

Parameters that affect the routing strategy are set by clicking onto the AUTO icon , which is available in the parameter toolbar after entering one of the follow-me modes. Click this icon in order to open the known *Autorouter Setup* window. There are two tabs that affect the Follow-me router: *General* and *Follow-me*.
The *Busses* tab and the *Optimize* tabs are also available, but you can't modify their values. They are only for Autorouter use.

Alternatively you can open this setup window from the command line. Type:

```
AUTO FOLLOWME
```

Select the *General* tab for defining preferred directions for the signal layers. They determine the main direction of signal wires in this layer.

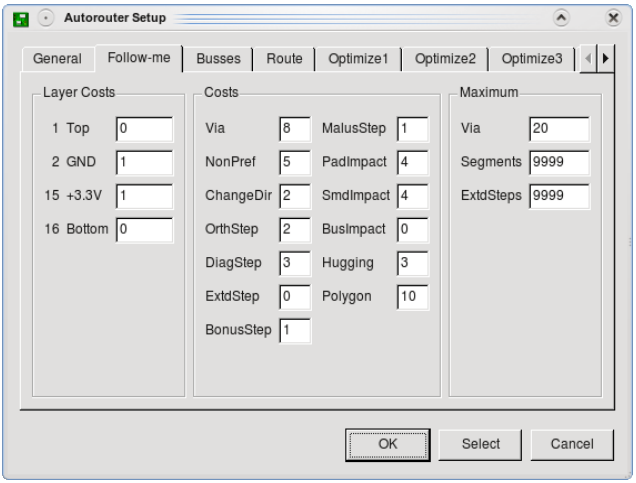


➢ ***Follow-me router: Defining preferred directions***

Check the possible options on page 184.

It is possible to choose *N/A – not available* – for a signal layer. The Follow-me router ignores such a layer. Objects that are already drawn in this layer won't be taken into consideration. This may lead to collisions and result in Design Rule Check errors. The Follow-me router explicitely warns you that such problems can appear.
Supply layers, however, must be set *N/A*. Such layers are generally forbidden for signal wires.

Settings that concern routing modalities are made in the *Follow-me* tab: *Layer Costs* control the usage of each layer. The higher the value (from 0 up to 99), the more the layer is avoided.



➢ ***Follow-me router: Setting routing parameters***

The effects of these parameters are explained in section 7.6, beginning with page 186.

In the *Maximum* section, you can define the number of *Vias* the router may use for one connection. If this value is set to 0, the Follow-me router is not allowed to set vias automatically. However, you are able to manually set a via by changing the layer.

The value for *Segments* defines the maximum number of wire segments a connection may consist of. If you choose it too small, it may happen that no connection will be established.

Notes

The Follow-me router supports round and octagon via shapes only. Square shaped vias are not possible.

If you are working in *Full* mode, the Follow-me router works in both directions independently, beginning with the mouse cursor position. So it could happen that the router places two vias very close to or even overlapping each other near the current position of the mouse cursor. In this case move the mouse cursor slightly, until the vias are optimized and the trace looks good.

It's recommended to draw a *Dimension* line in layer 20 in order to limit the board area and therefore required time and memory.

Depending on the complexity of your design, it may be wise to increase the cost factor for *Vias* and decrease it for *NonPref*. This avoids frequent layer changes.

Chapter 8
Component Design Explained through Examples

When developing circuits with EAGLE, components are fetched from libraries and placed into the schematic or, if the Schematic Editor is not being used, into the layout. All the component information is then saved in the schematic or board file. The libraries are no longer needed for continued work with the data. So when you want to pass your schematic to a third party to have a layout made from it, you do not also have to supply the libraries. An alteration in a library has no effect on a schematic or board.

The most important procedures for designing components (Devices) and working with libraries are explained from page 73 on. Please read this paragraph before you continue to read the current chapter!

Some practical examples follow, from which the effective application of the relevant commands and parameters will be seen. First we will take the example of a resistor and go through the whole process of designing a simple component.
The second example provides a full description of the definition of a complex component, including various Package variants and technologies. After that we shall discuss the special features which have to be taken into account with more complicated components.

Starting at page 256 hints concerning library and Device management can be found. How to create my own library? How to copy elements from one library into another?

First attempts at editing Packages, Symbols, or Devices may result in the need to delete various library elements. To do this, use the REMOVE command (see page 260).

8.1 Definition of a Simple Resistor

First open a new library in the EAGLE Control Panel using the *File/New/Library* menu.
Alternatively you can type the command

```
OPEN
```

on the command line of the Schematic or Layout Editor windows. Then enter a library name in the file dialog. The Library window opens.

Resistor Package

Define a New Package

Select the Package editing mode via the icon in the action toolbar, and enter the Package name *R-10* in the *New* field. Answer the question *Create new package 'R-10'?* with *Yes*. Later when creating a new Symbol and a new Device you will again have to answer the corresponding questions with *Yes*.

Set the Grid

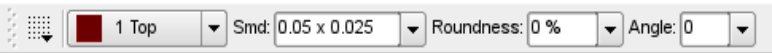
Use the GRID command to set an appropriate grid size for the pad placement. 0.05 inch (i.e. 50 mil) is usual for standard components with lead wires.

Solder Pads

For a resistor with lead-wires, select the PAD command, and set the pad shape and the drill diameter in the parameter toolbar. The default value for the pad diameter is *auto* (respectively 0). This value should be kept. The actual diameter is specified by the Design Rules for the layout. Then place two pads at the desired distance. The origin of the drawing will later be the identifying point with which a component is selected. For this reason it should be somewhere near the center of the Device.

> *You should not draw any objects in layer 17, Pads, or 18, Vias! They will not be recognized, nor by the DRC, neither by polygons drawn in the layout, and can lead to short circuits!*

For a SMD resistor, select the SMD command, and set the pad dimensions in the parameter toolbar. You can either select one of the offered values, or directly type the length and breadth into the entry field.



➢ ***SMD command: Parameter toolbar***

All properties can be altered after placement using the CHANGE command or by typing the command directly on the command line.

Select *Top* as the layer, even if the component will later be placed on the underside of the board. SMD components are located on the other side of a board using the MIRROR command. This moves the objects in all the *t*..-layers into the corresponding *b*..-layers.

Place the two SMD pads (which in EAGLE are just called SMDs) at the desired distance. It may be necessary first of all to alter the grid setting to a suitable value. The SMD can be rotated with the right mouse button before it is placed.

The parameter *Roundness* specifies whether the corners of the SMDs are to be rounded. By default this value is set to 0 % (no rounding). This value is usually kept, since the final roundness of SMDs is specified in the Design Rules. The help system provides you with more information about this parameter.

Angle determines the rotation of the SMD pad.

> *The INFO command or the Properties entry of the context menu provides you with a quick summary of the current properties of a SMD or Pad.*

Pad Name

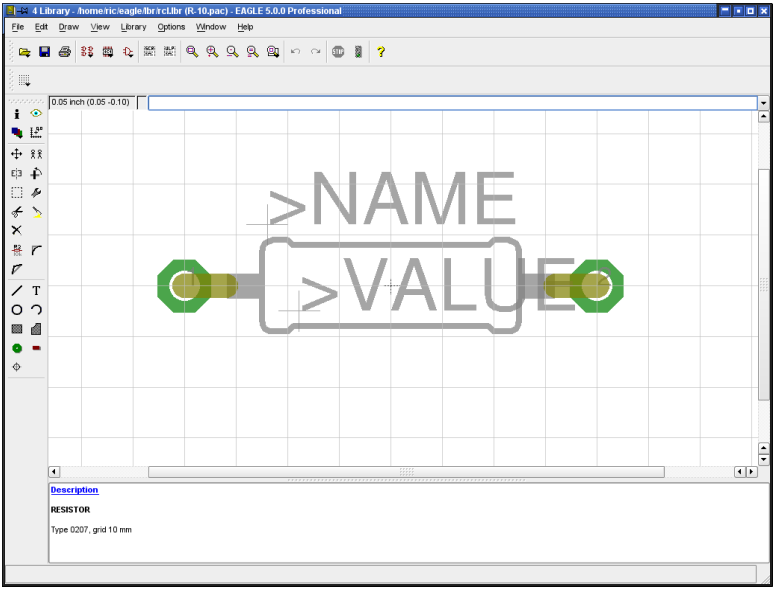
You can now enter the names, such as *1* and *2*, for the pads or SMDs using the NAME command.

Silkscreen and Documentation Print

Now use the commands WIRE, ARC, CIRCLE, RECT, and POLYGON to draw the silkscreen Symbol in layer 21, *tPlace*. This layer contains what will be printed on the board. It is up to you how much detail you give to the Symbol. Set a finer grid size if it helps.

Take the information provided in *library.txt* as a guideline for the design of components. The line thickness for the silk screen is usually 0.008 inch (0.2032 mm), for smaller components 0.004 inch (0.1016 mm).

Layer 51, *tDocu*, is not used to print onto the board itself, but is a supplement to the graphical presentation which might be used for printed documentation. Care must be taken in layer 21, *tPlace*, not to cover any areas that are to be soldered. A more realistic appearance can be given, however, in the *tDocu* layer, which is not subject to this limitation. In the example of the resistor, the Symbol can be drawn in layer 21, *tPlace*, but the wires, which go over the pads, are drawn in layer 51, *tDocu*.



➢ ***The Package Editor***

Labeling

With the TEXT command you place the texts >NAME (in layer 25, *tNames*) and >VALUE (in layer 27, *tValues*) in those places where in the board the actual name and the actual value are to appear. 0.07 inch for the text height (size) and 10 % for the ratio (relationship of stroke width to text height, which can only be set, using CHANGE, for vector fonts) are recommended.

> *We recommend to write these texts in vector font. So you can be sure that it looks exactly the same on the printed circuit board and in the Layout Editor.*

SMASH and MOVE can be used later to change the position of this text relative to the package symbol on the board.

In the case of ICs, for instance, the value corresponds to what will later be the Device name (e.g. *74LS00*).

> *When working with the Layout Editor only, the value is specified in the board.*

Restricted area for components

In layer 39, *tKeepout*, you should create a restricted area over the whole component (RECT command). This allows the DRC to check whether components on your board are too close or even overlapping.

Description

Finally, you click on the *Description* box. Text can then be entered in the lower part of the window which then opens. HTML text can be used. This is a syntax somewhat similar to HTML, which permits formatting of the text. You will find detailed information in the help system under *HTML Text*.

Example:

```
<b>R-10</b>
<p>
Resistor 10 mm grid.
```

Keywords from this text can be searched for from the ADD dialog in the layout.

> *Do not forget to save the library from time to time!*

Note

The CHANGE command can be used at a later stage to alter object properties such as the stroke thickness, text height, pad shape, or the layer in which the object is located.

If you want to change the properties of several objects at one go, define a group with the GROUP command , click the CHANGE command, select the parameter and the value, and click on the drawing surface with the *right* mouse button while the *Ctrl* key is pressed.

Example:
Use GROUP to define a group that contains both pads, then select CHANGE and *Shape/Square*. Press the *Ctrl* key, and click on the drawing surface with the right mouse button. The shape of both pads changes.

Resistor Symbol

Define a New Symbol

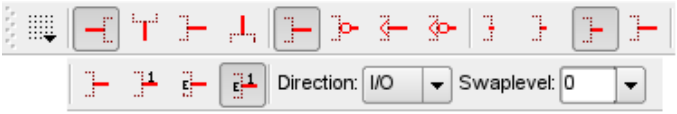
Select the Symbol editing mode, and enter the Symbol name *R* in the *New* field. This name only has a meaning internal to the program, and does not appear in the schematic.

Set the Grid

Now check that 0.1 inch is set as the grid size. The pins in the Symbol **must** be placed on this grid, since this is what EAGLE expects.

Place the Pins

Select the PIN command. You can now set the properties of these pins in the parameter toolbar, before placing them with the left mouse button. All these properties can be changed at a later stage with the CHANGE command. Groups can again be defined (GROUP) whose properties can then be altered with CHANGE and the right mouse button. See also page 205.



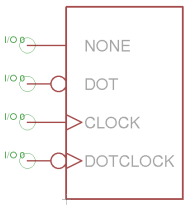
➢ ***Pin command: Parameter toolbar (split into two lines)***

Orientation

Set the direction of the pins (*Orientation* parameter) using the four left-hand icons in the parameter toolbar or, more conveniently, by rotating with the right mouse button.

Function

The function parameter is set with the next four icons on the parameter toolbar. This specifies whether the Symbol is to be shown with an inversion circle (Dot), with a clock symbol (Clk), with both (DotClk) or simply as a stroke (None). The diagram illustrates the four representations on one Package.



➢ ***Pin functions***

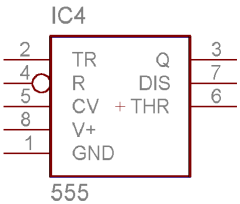
Length

The next four icons on the parameter toolbar permit setting of the pin length (0, 0.1 inch, 0.2 inch, 0.3 inch). The 0 setting is used if no pin-line is to be visible, or if, as in the resistor Symbol, a pin shorter than 0.1 inch is desired. In that case the pin is to be drawn with the WIRE command as a line in layer 94, *Symbols*.

The SHOW command can be used to check whether a net is connected to a pin in the schematic diagram. The pin line and the net are displayed more brightly if they are connected. If a pin with length 0 is used, or if it was drawn as a line with the WIRE command, it cannot be displayed brightly.

Visible

The next four icons in the parameter toolbar specify whether the pins are to be labeled with pin names, pad names, both or neither. The diagram illustrates an example in which pin names are shown inside and pad-names outside. The location of the label relative to the pin is fixed. The text height is also fixed (at 60 mil).



➢ ***Pin labeling***

Direction

The Direction parameter specifies the logical direction of the signal flow:

NC	Not connected
In	Input
Out	Output
I/O	Input/output
OC	Open Collector or Open Drain
Hiz	High impedance output
Pas	Passive (resistors, etc.)
Pwr	Power pin (power supply input)
Sup	Power supply output for ground and supply symbols

The Electrical Rule Check executes, depending on the pin direction, various checks. It expects for the direction

NC	a not connected pin
In	a net connected to this pin and not only *In* pins connected to this net
Out	not only *Out* pins connected to the net, no *Sup* or *OC*

	pin at the same net
OC	no *Out* pin at the same net
Pwr	a *Sup* pin set for this net
I/O, Hiz, Pas	no special checks

The *Pwr* and *Sup* directions are used for the automatic connection of supply voltages (see page 238).

Swaplevel

The Swaplevel is a number between 0 and 255. The number 0 means that the pin cannot be exchanged for another pin in the same Gate. Any number bigger than 0 means that pins can be exchanged for other pins which have the same Swaplevel and are defined within the same Symbol. The pins can be swapped in the schematic or in the board with the PINSWAP command.

The two pins of a resistor can have the same Swaplevel (e.g. 1), since they are interchangeable.

If the layer 93, *Pins*, is being displayed, the connection points on nets are shown with green circles. The *Direction* and *Swaplevel* parameters moreover (here *Pas* and *1*) are displayed in this layer.

> *The connections of a diode, for instance, cannot be exchanged, and are therefore given Swaplevel 0.*

Pin Names

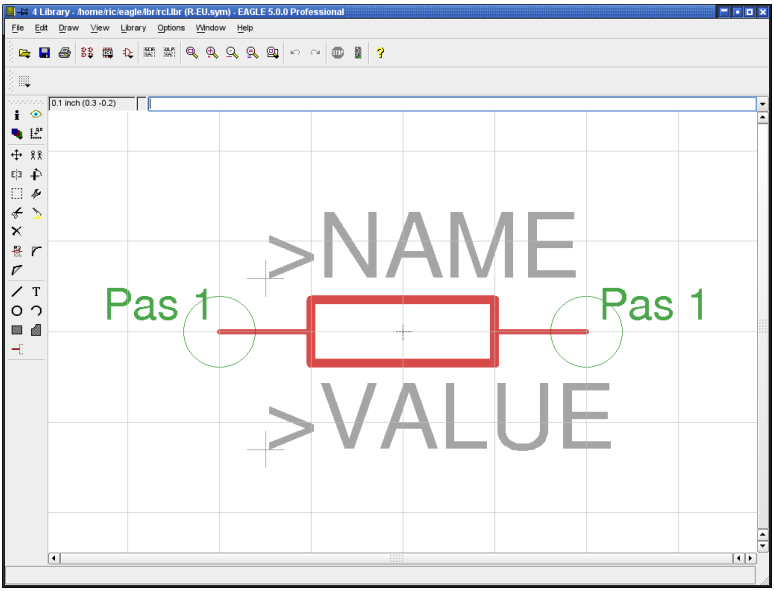
The NAME command allows you to name pins after they have been placed. The automatic name allocation, as described on page 93 also operates.

Schematic Symbol

The schematic Symbol is drawn in the Symbols layer using WIRE and the other drawing commands. Place the texts *>NAME* and *>VALUE* in layers 95, *Names*, and 96, *Values* (TEXT command). Place them where the name and value of the component are to appear in the schematic.

Precise placement of the text can be achieved by setting the grid finer, which can even be done while the TEXT command is active. Afterwards, however, set the again grid to 0.1 inches.

Layer 97, *Info,* may be used for additional information and hints.



➤ *The Symbol Editor*

Resistor Device

Define a New Device

Create the new Device *R-10* with this icon. When you later use the ADD command to fetch the component into the schematic, you will select it by using this name. It is only a coincidence that in this case the name of the Package and the name of the Device are the same.

So enter the name *R* on the *New* line. The Device Editor opens after the confirming question *Create new device 'R'?*.

Selecting, Naming and Configuring Symbols

The previously defined resistor Symbol is fetched into the Device with the ADD command.

If a Device consists of several schematic Symbols which can be placed independently of one another in the circuit (in EAGLE these are known as *Gates*), then each Gate is to be individually brought into the schematic with the ADD command.

Set an Addlevel of *Next* and a Swaplevel of *0* in the parameter toolbar, and then place the Gate near the origin. There are further explanations about Addlevel from page 244 on.

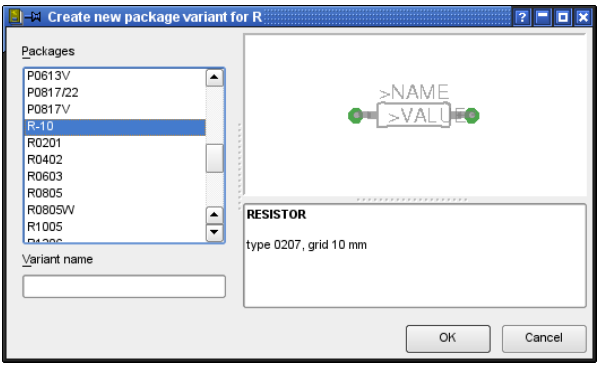
The Swaplevel of a Gate behaves very much like the Swaplevel of a pin. The value of 0 means that the Gate cannot be exchanged for another Gate in the Device. A value greater than 0 means that the Gate can be swapped within the schematic for another Gate in the same Device and having the same Swaplevel. The command required for this is GATESWAP.
Only one Gate exists in this example; the Swaplevel remains 0.

You can change the name of the Gate or Gates with the NAME command. The name is unimportant for a Device with only one Gate, since it does not appear in the schematic.
Keep the automatically generated name!

In the case of Devices with several Gates, the name of the particular Gate is added to the name of the Device.
Example:
The Gates are called A, B, C and D, and the name of the component in the schematic is IC1, so the names which appear are IC1A, IC1B, IC1C and IC1D.

Selecting the Package

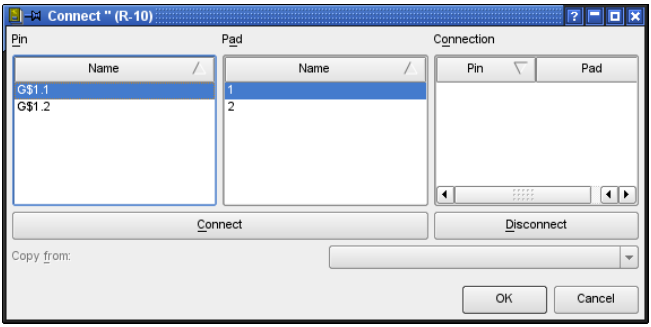
Now click on the *New* button at the lower right of the Device Editor window. Choose the R-10 Package from the selection window, and enter a name for the version. If only one Package version is used, it is usual to use two single quote marks (") for the name of the Package version. It is, however, quite possible to assign a particular name.



➢ *The Package selection*

Connections Between Pins and Pads

With the CONNECT command you specify which pins are taken to which package pads.



➢ ***The CONNECT window***

The resistor gate in this example is automatically identified as *G$1*, for which reason the pins *G$1.1* and *G$1.2* of this gate appear in the *Pin* column.
The two connections of the housing are listed in the *Pad* column. Mark a pin and the associated pad, and click on *Connect*.
If you want to undo a connection that you have made, mark it in the *Connection* column and click *Disconnect*.
Clicking on a column's header bar changes the sorting sequence.

Finish the CONNECT command by clicking on *OK*.

Define Prefix

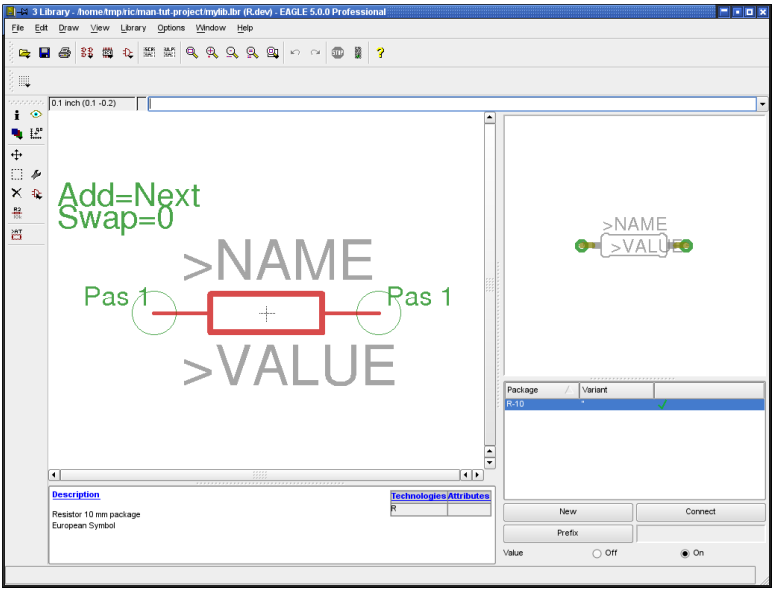
The PREFIX command is used to specify a prefix for a name. The name itself will initially be automatically allocated in the schematic. For a resistor this would, naturally enough, be *R*. The resistors will then be identified as R1, R2, R3 etc..

The names can be altered at any time with the NAME command

Value

On: You are allowed to change the value in the schematic (for example for resistors). Without a value the part will not be specified exactly.

Off: The value will be generated from the Device name and includes technology and Package variant (e. g. 74LS00N), if available.
Also recommended for supply symbols.



➢ ***The Device Editor: Fully defined resistor***

Description

Click on *Description* in the description box. You can enter a description of the component here. The search facility of the ADD command in the schematic diagram will search through this text.

You can use HTML Text, as in the Package description. You will find notes about this in the help system under the keyword *HTML Text*.

It can look like this:

```
<b>R-10</b>
<p>
Resistor 10mm package
```

> *Hyperlinks contained in the description of library objects are opened with the appropriate application program.*

Save

This completes definition of the resistor, and it can be fetched into the schematic diagram. If you have not already saved the library, please do it at this stage!

Library Description

Not only Packages and Devices can have descriptions, but the Library as a whole can have one as well. This description is shown in the Control Panel as soon as you expand the *Libraries* branch of the Tree view and select a library entry there.

No matter which editor mode (Symbol, Package, Device) is currently active, click the *Library/Description* menu to edit the description. You can use HTML text, if you like.

Use Library

The newly created library has to be made available for the schematic or layout with the help of the USE command. This command has to be used in the Schematic or Layout Editor. It is also possible to mark a library as *in Use* in the Control Panel's tree view. See help for details.
Now the library will be recognized by the ADD command and its search function.

8.2 Defining a Complex Device

In this section we use the example of a TTL chip (541032) to define a library element that is to be used in two different Packages (pin-leaded and SMD). It is a quad OR gate. The schematic diagram symbol is to be defined in such a way that the individual OR gates can be placed one after another. The power supply pins are not initially visible in the schematic diagram, but can be fetched into the diagram if needed.

The definition proceeds in the following steps:

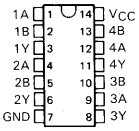
- Creating a new library
- Drawing the pin-leaded housing (DIL-14)
- Creating the SMD housing (LCC-20)
- Defining the logic symbol
- Creating the power supply symbol
- Associating the Packages and Symbols to form a Device set

SN54ALS1032A, SN54AS1032A, SN74ALS1032A, SN74AS1032A
QUADRUPLE 2-INPUT POSITIVE-OR BUFFERS/DRIVERS

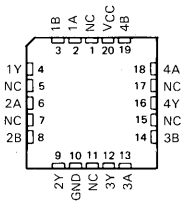
D2661, DECEMBER 1982—REVISED MAY 1986

- 'ALS1032A is a Buffer Version of 'ALS32
- 'AS1032A is a Driver Version of 'AS32
- 'AS1032A Offers High Capacitive Drive Capability
- Package Options Include Plastic "Small Outline" Packages, Ceramic Chip Carriers, and Standard Plastic and Ceramic 300-mil DIPs
- Dependable Texas Instruments Quality and Reliability

SN54ALS1032A, SN54AS1032A . . . J PACKAGE
SN74ALS1032A, SN74AS1032A . . . D or N PACKAGE
(TOP VIEW)



SN54ALS1032A, SN54AS1032A . . . FK PACKAGE
(TOP VIEW)



NC—No internal connection

description

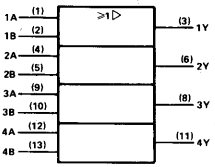
These devices contain four independent 2-input OR buffers/drivers. They perform the Boolean functions $Y = A + B$ or $Y = \overline{\overline{A} \cdot \overline{B}}$ in positive logic.

The SN54ALS1032A and SN54AS1032A are characterized for operation over the full military temperature range of −55°C to 125°C. The SN74ALS1032A and SN74AS1032A are characterized for operation from 0°C to 70°C.

FUNCTION TABLE
(each gate)

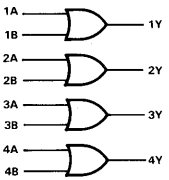
INPUTS		OUTPUT
A	B	Y
H	X	H
X	H	H
L	L	L

logic symbol†



†This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, and N packages.

logic diagram (positive logic)



PRODUCTION DATA documents contain information current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

TEXAS INSTRUMENTS

Copyright © 1982, Texas Instruments Incorporated

➢ *Data sheet for the 541032*

All the data for this component has been extracted from a data book published by Texas Instruments, whom we thank for permission to reproduce it.

Creating a New Library

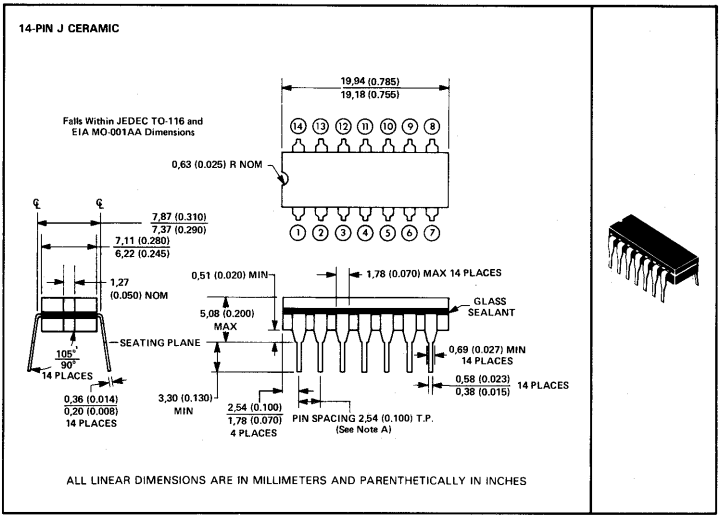
Click on the *File/New/Library* menu in the EAGLE Control Panel. The Library Editor window appears, containing a new library, *untitled.lbr*.

It is, of course, also possible to expand an existing library. In that case you would use *File/Open/Library* to select the library you want, or you would click on the *Libraries* entry in the Control Panel's tree view, selecting the desired library with a click of the right mouse button. This will open a context menu, one of whose options is *Open*. The Library Editor is opened.

Drawing the Pin-Leaded Package

The component is manufactured in a pin-leaded Package. This is a DIL-14 housing with a pin spacing of 2.54 mm (0.1 inch) and a width of 7.62 mm (0.3 inch).

If there is a suitable Package in another library, it can be copied into the current library. A new definition would not be necessary.



NOTE A: Each pin centerline is located within 0,25 (0.010) of its true longitudinal position.

➢ ***DIL-14 data sheet***

Click onto the *Edit a package* icon in the action toolbar, and enter the name of the Package in the *New* box of the *Edit* menu, which is *DIL-14* in our present example. Click *OK*, and confirm the question *Create new package 'DIL-14'?* by answering *Yes*.

The Package Editor window now opens.

Set the Grid

First set the appropriate grid (50 mil in this case) using the GRID command, and let the grid lines be visible.
The grid can easily be shown and hidden with the *F6* function key.

Place Pads

Use the PAD command, and place the solder pads in accordance with the specifications on the data sheet. The pads should be arranged in such a way that the coordinate origin is located somewhere near the center of the Package.

Each pad can have individual properties such as *Shape*, *Diameter*, and *Drill* hole diameter. Available shapes are: *Square*, *Round*, *Octagon*, *Long*, and *Offset* (Long with offset drill).

Select the desired pad shape and specify the hole diameter.

The pad diameter usually is defined with the standard value *auto* (respectively 0), since the size is finally determined in the layout by means of the Design Rules, *Restring* tab. The pad appears in the library with the default value of 55 mil.
You may, however, assign an individual value. If, for instance, you specify 70 mil, the consequence is that the diameter of the pad on the board cannot be less than 70 mil (independent of the calculated value of the Design Rules). You select this value when the PAD command is active (i.e. the pad is attached to the mouse cursor) using the parameter toolbar. It is also possible to specify the drill hole diameter and the pad shape.

➢ ***The parameter toolbar when the PAD command is active***

The properties of pads that have already been placed can be altered at a later stage by means of the CHANGE command. Click onto the CHANGE icon and select the property and the appropriate value. Then click onto the pads whose properties are to be altered. CHANGE can also be applied to groups (using the GROUP command). After the property has been selected, click inside the group with the right mouse button.

As soon as a pad has been placed, EAGLE automatically generates solder stop symbols in layers 29 and 30, *t/bStop*. The dimensions of the solder stop symbols is specified in the Design Rules, *Mask* tab, *Stop* parameter.

Pads can be marked with special flags (*First*, *Stop*, *Thermals*). They can be altered with CHANGE subsequently. Giving one pad of a Package the *First* flag (CHANGE FIRST ON) allows to define a special shape for it in the Design Rules, *Shapes* tab, option *First*, in order to mark it as the number '1' pad of the Package.

Setting the *Thermals* flag *off* prevents generating a Thermal symbol in a copper area.
CHANGE STOP OFF prevents automatic solder stop mask generation for a pad.

Pad Name

EAGLE automatically assigns pad names, *P$1, P$2, P$3* etc., as placement proceeds. Assign the names in accordance with the information in the data book.

The names can be checked easily by clicking the *Options/Set/Misc* menu and choosing the *Display pad names* option. All pad names are displayed after refreshing the screen (*F2*).
Alternatively type in the command line:

```
SET PAD ON
```

To hide the pad names again:

```
SET PAD OFF
```

The following procedure is recommended for components that have a large number of sequentially numbered pads:

Select the PAD command, type in the name of the first pad, e.g. '1', and place the pads in sequence. The single quote marks must be typed on the command line. See also the section on *Names and Automatic Naming* on page 93.

Draw the Silk Screen Symbol

A simple silk screen symbol that is to be visible on the board is drawn in layer 21, *tPlace*. Use the commands WIRE, ARC, CIRCLE, RECT, and POLYGON.
Ensure that it does not cover soldered areas, since this can cause problems when the boards come to be soldered. If necessary, use the GRID command to set a finer grid or use the *Alt* key for the alternative grid (see GRID command). The standard width (CHANGE WIDTH) for lines in the screen print is 8 mil or 4 mil, depending on the size of the component.

It is also possible to create an additional and rather better-looking silk screen for documentation purposes in layer 51, *tDocu*. This may indeed cover soldered areas, since it is not output along with the manufacturing data.

Package Name and Package Value

The labelling now follows. Use the TEXT command and write

```
>NAME
```

in layer 25, *tNames*, for the name placeholder, and

```
>VALUE
```

in layer 27, *tValues*, as the placeholder for the value, and place this at a suitable location. We use proportional font with a text height of 70 mil as default.

If you want to have texts upside down by a Package rotation of 180°, you have to use the *Spin* flag (see help function for TEXT command).

The texts can be relocated at a later stage using SMASH and MOVE.

> *We recommend to write these texts in vector font. So you can be sure that it looks exactly the same on the printed board as it is in the Layout Editor.*

Areas Forbidden to Components

In layer 39, *tKeepout*, you should create a restricted area over the whole component using the RECT command or draw a frame around the Package with WIRE. This allows the DRC to check whether components on your board are too close or even overlapping.

Description

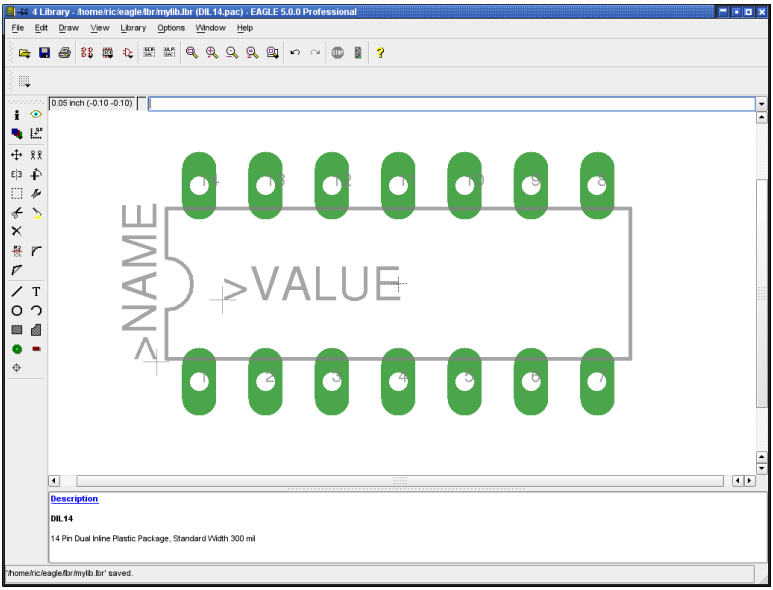
Click on *Description* in the description box. A window opens in whose lower part it is possible to enter text, while the formatted appearance of the description is displayed in the upper part (*Headline*). The text can be entered in HTML format. EAGLE works with a subset of HTML tags that allow the text to be formatted. You will find detailed information in the help system under *HTML Text*.

The descriptive text for our DIL-14 might look like this:

```
<b>DIL-14</b>
<p>
14-Pin Dual Inline Plastic Package, Standard Width
300 mil
```

It is also possible to add, for instance, the reference data book,
the e-mail address of the source or other information here. The search facility in the Layout Editor's ADD dialog also looks in this text for keywords.

> *Hyperlinks contained in the description of library objects are opened with the appropriate application program.*



➢ *Package Editor with DIL-14*

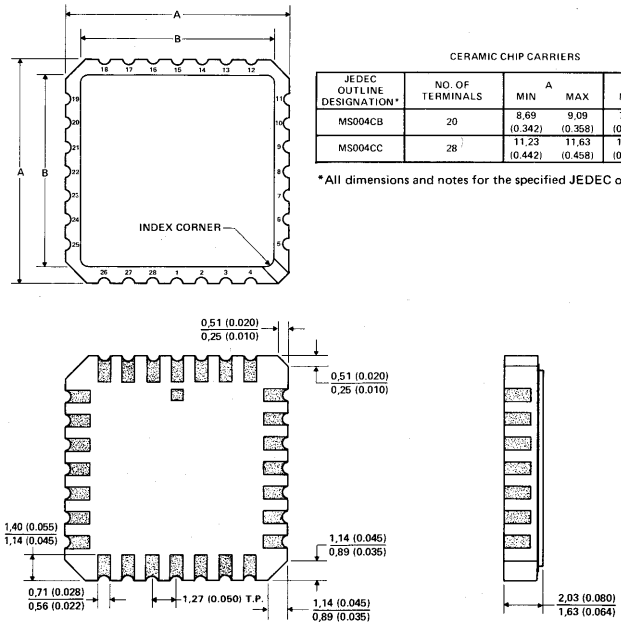
Save

At this stage if not before the library should be saved under its own name (e.g. *my_lib.lbr*).

Defining the SMD Package

The second type of housing for this component may be seen in the following scale drawing.

FK CERAMIC CHIP CARRIER
(28-terminal package shown)



CERAMIC CHIP CARRIERS

JEDEC OUTLINE DESIGNATION*	NO. OF TERMINALS	A MIN	A MAX	B MIN	B MAX
MS004CB	20	8,69 (0.342)	9,09 (0.358)	7,80 (0.307)	9,09 (0.358)
MS004CC	28	11,23 (0.442)	11,63 (0.458)	10,31 (0.406)	11,63 (0.458)

*All dimensions and notes for the specified JEDEC outline apply.

ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES

➢ ***SMD package, FK version***

The size of the soldering areas is to be 0.8 mm x 2.0 mm. The SMD 1, at 0.8 mm x 3.4 mm, is larger.

Click again onto the *Edit a package* icon, and enter the name of the Package in the *New* box in the edit menu. The Package is to be called LCC-20. Click *OK* and confirm the question *Create new package 'LCC-20'?* by answering *Yes*.

Set the Grid

Adjust the grid to 0.635 mm (0.025 inch), and let the grid lines be visible. It is useful to define an alternative grid of 0.05 mm for designing this Package.

Placing SMD Solder Pads

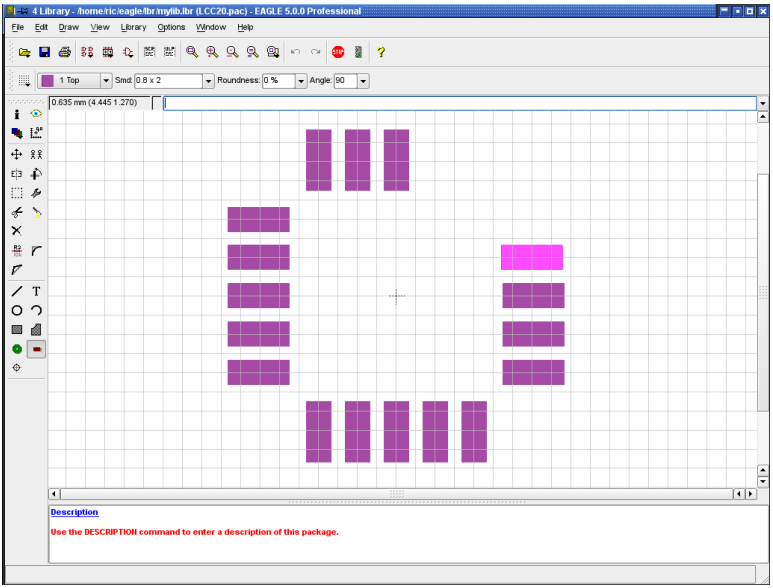
SMD devices are generally defined on the top of the board; SMDs are therefore always in layer 1, *Top*.
If you do want to have components on the solder side, the item is if needed reflected on the board with the MIRROR command. See also the section on page 250.

Begin by placing 5 SMDs at a distance of 1,27 mm from each other in two horizontal rows close to the coordinate origin. Since the value 0.8 x 2.0 is not contained in the SMD menu, this must be entered as 0.8 2.0, either on the command line or in the SMD box on the parameter toolbar.

Click therefore onto the SMD icon, and type

```
0.8 2 ←
```

in the command line. Create two vertical rows as well. The SMDs can be rotated in 90 degree increments with the right mouse button.



➢ ***Placing the SMDs***

The *Roundness* parameter (CHANGE command) specifies whether curves should be given to the corners of the solder pads. The default value is 0 %, which means that there is no rounding.
See also the section on page 134.
If a square SMD is selected, and if *Roundness* is defined as 100 %, the result is a round SMD, as is needed when creating ball grid array housings (BGA). *Roundness* is usually chosen to be 0 % when a Package is being defined. A

general value can be specified in the Design Rules if slightly rounded solder pads are preferred.

Drag the 4 SMD rows into the correct position. Therefore use the finer alternative grid of 0.05 mm by pressing the *Alt* key. The commands GROUP and MOVE, followed by a right mouse click on the marked group while the *Ctrl* key is pressed can be used to drag the marked group into the correct position. The size of the central SMDs in the upper row can be altered with the CHANGE SMD command. Since the value 0.8 x 3.4 is not contained in the menu as standard, type

```
change smd 0.8 3.4 ←
```

onto the command line, then click the SMD. Drag it with MOVE so that it is located at the correct position.

The INFO command is first choice for checking the positions and properties of the solder pads and modifying them, if needed.

When a SMD is placed (in the Top layer), symbols for solder stop and solder cream are automatically created in layer 29, *tStop*, and layer 31, *tCream*, respectively.
If the component in the layout is mirrored onto the bottom side, these are changed to the layers with the corresponding functions, namely 30, *bStop* and 32, *bCream*.

SMDs can have special flags (*Stop*, *Cream*, Thermals) that can be modified with the CHANGE command.
Setting the *Thermals* flag off avoids a Thermal symbol for the SMD copper areas.
CHANGE STOP OFF or CHANGE CREAM OFF prevents EAGLE from generating a solder stop mask or a cream frame for the SMD automatically. See also help function about CHANGE and SMD.

> *If you have to design a larger area, for example a cooling plate, that exceeds the limits of the SMD command, you can draw a POLYGON that covers more or less the SMD. Please keep in mind, that you have to adjust the masks for solder stop and cream frame in the layer tStop and tCream, too. The DRC will report an overlap error at this position, which you can approve.*

SMD Names

If no names are visible in the SMD pads, click the *Options/Set/Misc* menu and activate the *Display pad names* option.

Alternatively you can type the following onto the command line:

```
set pad_names on ←
```

Use the NAME command to adjust the names to match the specifications of the data sheet.

It is alternatively possible to assign names as the SMDs are being placed, if the component has a large number of pads with sequential numbers. Select the SMD command, type in the name of the first SMD, e.g. '1', and place the pads in the correct sequence. The single quote marks must be entered on the command line.
See also the section on *Names and Automatic Naming* on page 93.

You can also combine several statements on the command line, for example:

```
smd 0.8 2 '1' ←
```

A SMD of 0.8 mm x 2.0 mm named *1* is now attached to the mouse cursor.

Draw the Silk Screen

First set the grid to a suitable value such as 0.254 mm (10 mil).

Draw the silk screen print in layer 21, *tPlace*.

> *Note that the silk screen print must not cover soldered areas, as this will cause problems when the board comes to be soldered.*

The default value for the line width is 8 mil (0.2032 mm), for smaller components 4 mil (0.1016 mm).

It is also possible to create an additional, more detailed, silk screen for documentation purposes in layer 51, *tdocu*. This may indeed cover soldered areas, since it is not output along with the manufacturing data.

Package Name and Package Value

The labeling now follows. Use the TEXT command and write

```
>NAME
```

in layer 25, *tNames*, for the name placeholder, and

```
>VALUE
```

in layer 27, *tValues*, as the placeholder for the value, and place this at a suitable location. The texts can be separated and relocated at a later stage using SMASH and MOVE.

> *We recommend to write these texts in vector font. So you can be sure that it looks exactly the same on the printed board as it is in the Layout Editor.*

Area Forbidden to Components

In layer 39, *tKeepout*, you should create a forbidden area over the whole component (RECT command) or draw a frame around the Package with the WIRE command. This allows the DRC to check whether components on your board are too close, or even overlapping.

Locating Point (Origin)

As soon as you have finished drawing the package, please check where the coordinate origin is located. It should be somewhere near the middle of the Package. If necessary, use GRID to choose a suitable grid (e.g. 0.635 mm), and shift the whole Package with GROUP and MOVE.
First make sure that all the layers are made visible (DISPLAY ALL). That is the only way to be sure that all the objects have indeed been moved.

Description

Then click on *Description* in the description box. You can insert a detailed description of this Package form here. HTML Text can be used. This format is described in the program's help system under *HTML Text* .

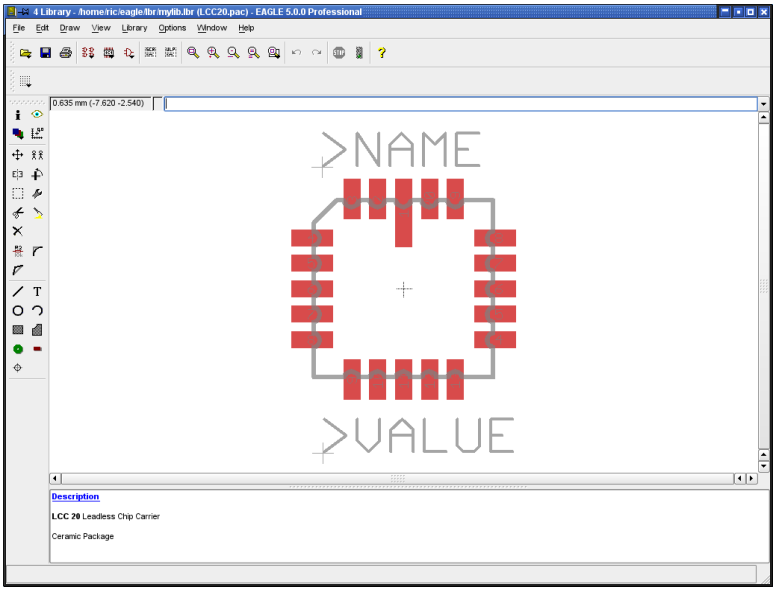
The entry of the LCC-20 in HTML text format could look like this:

```
<b>LCC-20</b>
<p>
 FK ceramic chip carrier package from Texas
 Instruments.
```

The ADD dialog in the Layout Editor can search for this description or for keywords within it.

Save

Please do not forget to save the library from time to time!

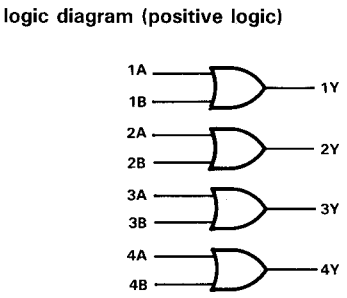


➢ ***The fully defined LCC-20***

> *Supposed you found a Package that is exactly the one you need in another library file, simply copy it into your current library. More information about this on page 256.*

Defining the Logic Symbol for the Schematic Diagram

Our Device contains four OR gates, each having two inputs and one output. We first create an OR symbol.



➢ ***Logical appearance of the 541032***

Click onto the *Edit a symbol* icon. Enter a name for the Symbol on the *New* line, such as *2-input_positive_or*, and click *OK*. Confirm the question *Create new symbol '2-input_positive_or'?* by answering *Yes*. You now have the Symbol Editor window in front of you.

Check the Grid

Check that the grid is set to the default value of 0.1 inch. Please try to use only this grid, at least when placing the pins.

> *It is essential that pins and net lines are located on the same grid. Otherwise there will <u>not be any</u> electrical connection between the net and the pin!*

Place the Pins

Select the PIN command, and place 3 pins. The pin properties can be changed by means of the parameter toolbar as long as the pin is attached to the mouse cursor and has not been placed. If a pin has already been placed, its properties can be altered at a later stage with the CHANGE command. A number of pins can be handled at the same time with the GROUP and CHANGE commands followed by a click into the drawing with the right mouse button while the *Ctrl* key is pressed. The parameters *Orientation*, *Function*, *Length*, *Visible*, *Direction* and *Swaplevel* have been thoroughly described when the example of the resistor symbol was examined (see p. 205).
The coordinate origin should be somewhere near the center of the Symbol, and, if possible, not directly under a pin connection point. This makes it easy to select objects in the schematic diagram.

Pin Name

You assign pin names with the NAME command. In our Symbol the two input pins are named *A* and *B*, and the output pin is named *Y*.

Pins carrying inverted signals (active low) can be displayed with a bar over the name text. An exclamation mark starts and ends the bar. Examples can be found in the help function of the TEXT command.

Draw the Symbol

Use the WIRE command to draw the Symbol in layer 94, *Symbols*. The standard line thickness for the Symbol Editor is 10 mil. You may also choose any other line thickness.

Placeholders for NAME and VALUE

For the component labeling, use the TEXT command in the schematic diagram to write

```
>NAME
```

in layer 95, *Names* and

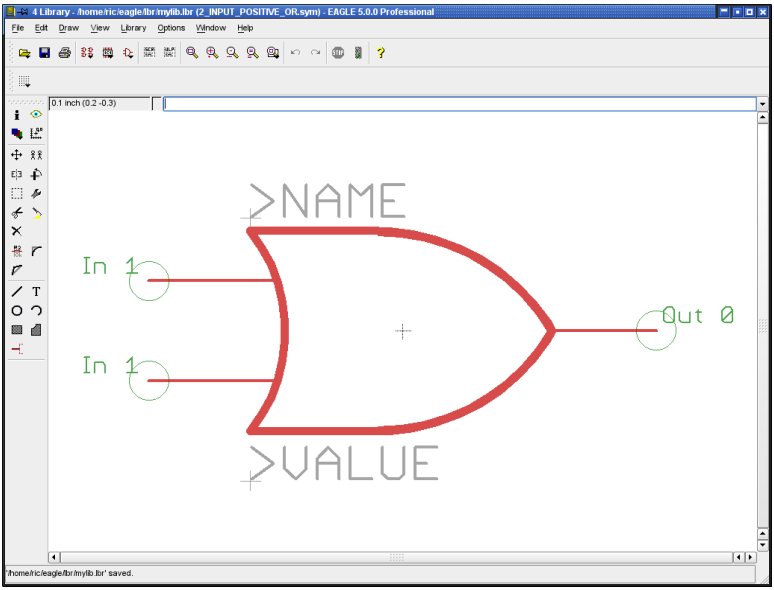
```
>VALUE
```

in layer 96, *Values*. Place the two texts at a suitable location. It is possible to move the texts again in the schematic diagram after using SMASH to separate it. The Symbol should now have the appearance shown in the following diagram.

Save

This is a good moment to save the work that you have done so far.

> *Supposed you found a Symbol that is exactly the one you need use GROUP, CUT, and PASTE to copy it into the current library. See also page 258.*



➢ *The Symbol Editor: Logic symbol (American representation)*

Defining a Power Supply Symbol

Two pins are needed for the supply voltage. These are kept in a separate Symbol, since they will not initially be visible in the schematic diagram.

Click onto the *Edit a symbol* icon. Enter a name for the Symbol on the *new* line, such as *VCC-GND*, and click *OK*. Confirm the question *Create new symbol 'VCC-GND'*? with *Yes*.

Check the Grid

First check that the grid is set to the default value of 0.1 inch. Only ever use this grid when placing pins!

Place the Pins

Fetch and place two pins with the PIN command. The coordinate origin should be somewhere near the center of the Symbol.

Both pins are given PWR as their direction. To do this, click with the mouse on CHANGE, select the *Direction* option, and choose *PWR*. Now click onto the two pins to assign this property.

The green pin label is updated, and now shows *Pwr 0*. It is only visible when layer 93, *Pins*, is active!

Pin Name

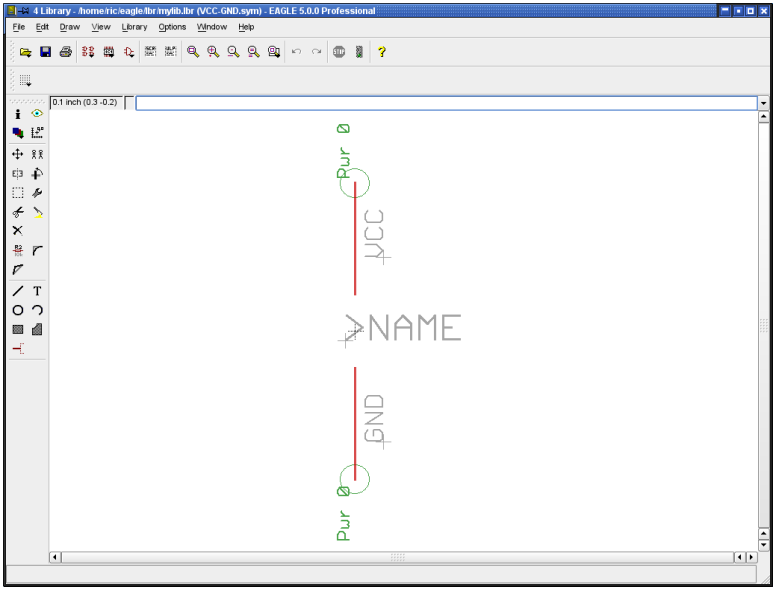
You use the NAME command to give the two pins the names of the signals that they are to carry. In this case, these are GND and VCC.

For reasons of appearance, the pin property *Visible* is set to *Pad* in the Symbol shown below, and the pin label has been placed on layer 95, *Names*, using TEXT.

Placeholders for NAME and VALUE

For the component labeling, use the TEXT command in the schematic diagram to write

```
>NAME
```



➢ ***The Symbol Editor: Supply symbol***

in layer 95, *Names*. Place the text at a suitable location. No placeholder is necessary for value here.

Associating the Packages and Symbols to Form a Device Set

We now come to the final step, the definition of the Device set. A Device set is an association of Symbols and Package variants to form real components

A Device set consists of several Devices, which use the same Symbols for the schematic but different technologies or Package variants.

Defining a Device set or a Device consists essentially of the following steps:

- Select Symbol(s), name them and specify properties
- Assign Package(s) or specify variants
- Specify the assignment of pins to pads using the CONNECT command
- Define technologies (if desired/necessary)
- State prefix and value
- Describe the Device

Click onto the *Edit a device* icon. Enter the name for the Device on the *New* line.
In our example this is a 541032A. This Device is to be used in two different technologies, as the 54**AS**1032A and as the 54**ALS**1032A. The * is used as a placeholder at a suitable location in the Device name to represent the different technologies. Enter, therefore, the name 54*1032A, and confirm the question *Create new device '54*1032A'*? with *Yes*.

The Device Editor window opens.

> *A question mark **?** as part of the Device name is used as a placeholder for the Package Variant name. If you don't use a **?**, EAGLE adds the Package Variant name at the end of the Device name automatically.*

Select Symbols

First use ADD to fetch the Symbols that belong to this Device. A window opens in which all the Symbols available in the current library are displayed. Double-click onto the *2-input_positive_or* symbol and place it four times.
Click again on the ADD icon, and select the '*VCC-GND*' Symbol from the list. Place this too onto the drawing area.

Naming the Gates

A Symbol that is used in a Device is known as a Gate. They are automatically given generated names (G$1, G$2 etc.). This name is not usually shown on the schematic diagram.

It is nevertheless helpful to assign individual Gate names when components are composed of a number of Gates. To distinguish the individual OR gates, you use the NAME command to alter the Gate names. Assign the names *A*, *B*, *C* and *D,* and name the power supply gate *P*.

Specify Addlevel and Swaplevel

The Addlevel can be used to specify how the gates are placed in the schematic diagram by the ADD command. You can see the current Addlevel for each Gate written above left in layer 93, *Pins*.

Assign the Addlevel *Next* for Gates A to D, and the Addlevel *Request* to the power supply gate. Do this by clicking onto the CHANGE icon, selecting the *Addlevel* entry, and then selecting the desired value for a gate. Then click on the Gate you want to change.

This means that as soon as the first OR gate has been placed on the schematic diagram, the next one is attached to the mouse cursor. All 4 gates can be placed one after another. The power gate does not automatically appear. You can, however, fetch it into the schematic diagram if necessary, by making use of the INVOKE command.

The parameter ADDLEVEL is described in full detail in the section entitled *More About the Addlevel Parameter* on page 244.

The Swaplevel determines whether a Device's gates can be swapped within the schematic diagram. The value that is currently set is like the Addlevel displayed above left in layer 93, *Pins*, for each gate. The default value is 0, meaning that the gates may not be exchanged. The Swaplevel can have a value from 0 to 255. Gates with the same Swaplevel can be exchanged with one another.

Our Device consists of four identical Gates that may be swapped. Click onto CHANGE, select the *Swaplevel* entry, and enter the value *1*. Click on the four OR gates. The information text in layer 93, *Pins*, changes correspondingly.

Choosing the Package Variants

In the Device Editor window, click the *New* button at the lower right. A window opens that displays the Packages defined in this library. Select the *DIL-14* package and give the version name *J*. Click *OK*.
Repeat this procedure, select the *LCC-20*, and give the version name *FK*.

In the list on the right you will now see the chosen Package variants, with a simple representation of the selected Package above it.
Clicking on a Package variant entry with the right mouse button will open a context menu. This allows variants to be deleted, renamed or newly created, Technologies to be defined, the CONNECT command to be called, or the Package editor to be opened.

Both entries are marked by a yellow symbol with an exclamation mark. This means that the assignment of pins and pads has not yet been (fully) carried *out*.

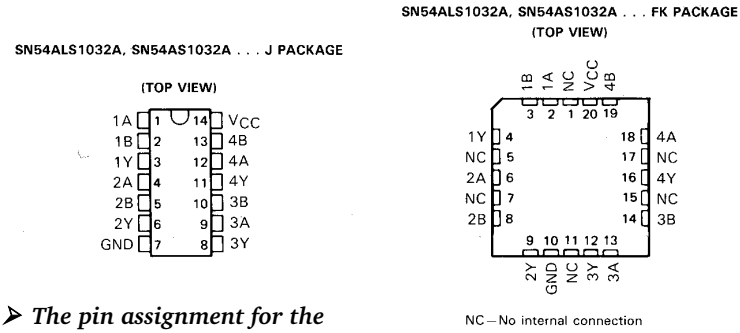
Supposed you don't find the appropriate Package variant in the current library you may use Packages from another library. Use the PACKAGE command to copy the Package into the current variant and to define a new

variant.
Example:

```
PACKAGE DIL14@d:\eagle\lbr\ref-packages.lbr  J
```

This command copies the Package named DIL14 from ref-packages.lbr into the current library. Simultaneously the variant J is generated for the Device. See also page 251.

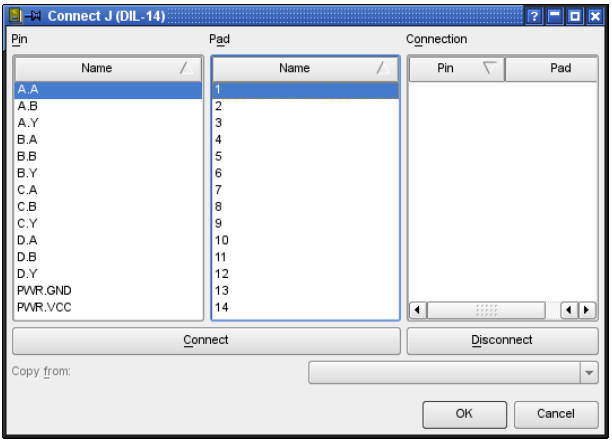
The Connect Command



➢ ***The pin assignment for the packages***

This must be the most important step in the library definition. CONNECT assigns each pin to a pad. The way in which nets in the schematic diagram are converted into signal lines in the layout is defined here. Each net at a pin creates a signal line at a pad. The pin assignment for the 541032 is specified in the data sheet. Check the connects in the library with care. Errors that may pass unnoticed here can make the layout useless.

Select the *J* version from the Package list and click the CONNECT button. The connect window opens.



➢ ***CONNECT dialog***

The list of pins is on the left, and the pads are in the center. Click onto a pin-entry, and select the associated pad. Both entries are now marked. You join them with the connect button. This pair now appears on the right, in the *Connection* column. Join each pin to its pad in accordance with the data sheet. Finish the definition by clicking *OK*.

Please note that in our example the Gates are named A, B, C, and D while they are named 1, 2, 3, and 4 in the data sheet.

Define the connections for the second Package version, *FK*, in the same way. Select the version, and click the *Connect* button. The usual dialog appears in the connect window. Proceed exactly as described above.
Please note that six pads are not connected in this version. They are left over in the *Pad* column. Finish the process by clicking *OK*.

There is now a green tick to the right of both Package variants, and this indicates that connection is complete. This is only true when every pin is connected to a pad.

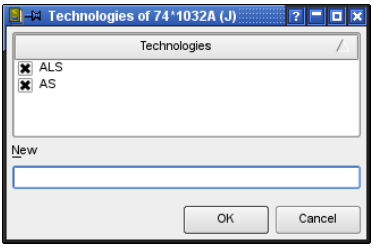
> *One pin has to be connected to exactly one pad!*
> *It is not possible to connect several pins with a common pad!*
> *A Device may contain more pads than pins, but not the other way around!*
> *Pins with direction NC (not connected) must be connected to a pad, as well!*

Defining Technologies

As noted above, the 541032 is to be used in two different technologies *AS* and *ALS*. By including a * as a placeholder in the Device name we have already taken the first step towards this. In the schematic diagram the code

for the chosen technology will appear instead of the *. The data sheet shows that both technologies are available in both Packages.

Select the *J* Package from the list on the right of the Device Editor window. Then click onto *Technologies* in the description box. The technologies window opens. Define the technology in the *New* line, and confirm the entry with *OK*. When the entry has been completed, the *AS* and *ALS* entries are activated with a tick.



➢ ***Technologies for package variant J***

Close the window by clicking *OK* again.

Select the FK version from the Package list. Click onto *Technologies* in the description box again. You will now see that AS and ALS are available as selections in the technologies window. Activate both of these by clicking into the small box to the left, so that a tick is displayed. Finish the definition by clicking *OK*.

The technologies available for the selected Package version are now listed in the description area of the Device Editor.

Specifying the Prefix

The prefix of the Device name is defined simply by clicking on the *Prefix* button. *IC* is to be entered in this example.

Value

The setting of *value* determines whether the VALUE command can be used to alter the value of the Device in the schematic diagram and in the layout.

On: You are allowed to change the value in the schematic (for example for resistors). Defining the value is necessary to specify the part.

Off: The value will be generated from the Device name which can include technology and Package variant name (e.g. 74LS00N).

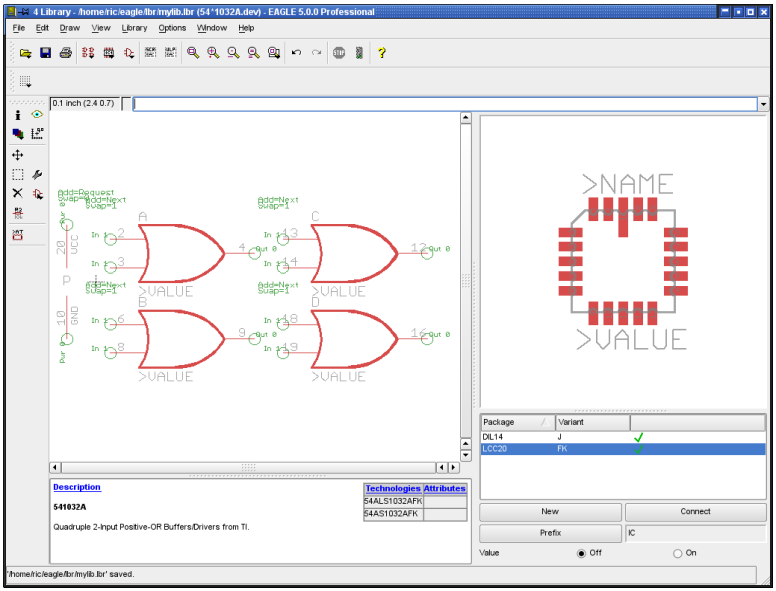
Even if *Value* is set *Off*, it is possible to change the value of a component after confirming a warning message.
If you change the initial value and decide to use another Technology or Package variant later with CHANGE PACKAGE or CHANGE TECHNOLOGY, the user-defined value will remain unchanged.

Description

Click onto *Description* in the description box. You can enter a description of the Device in the window which now opens. Use typical terms that you might apply for a keyword search. The search facility of the ADD command in the schematic diagram will also search through this text.

You can use HTML text. The syntax is described in the help system under the keyword *HTML Text*.

The description can look like this:



➢ ***Device Editor: 54*1032A.dev***

```
<b>541032A</b>
<p>
 Quadruple 2-Input Positive-OR Buffers/Drivers
 from TI.
```

Save

This completes definition of the Device set. If you have not already saved the library, please do it at this stage!

8.3 Supply Voltages

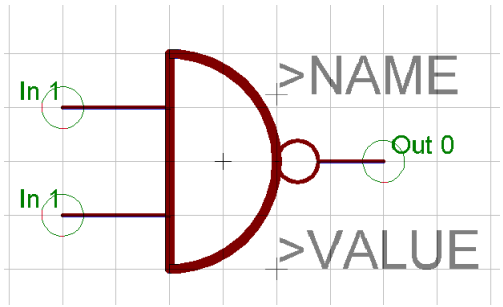
Component Power Supply Pins

The components' supply pins are to be given the direction *Pwr* in the Symbol definition. The pin name determines the name of the supply signal. Pins whose direction is *Pwr* and which have the same name are automatically wired together (even when no net line is shown explicitly). Whether the pins are visible in the schematic diagram or are fetched by means of a hidden Symbol is also not relevant.

Invisible Supply Pins

We do not want as a rule to draw the supply connections for logic components or operational amplifiers in the schematic. In such a case a specific Symbol containing the supply connections is defined. This can be demonstrated with the example of a 7400 TTL component:

You first define a NAND gate with the name *7400*, and the following properties in the Symbol Editor:

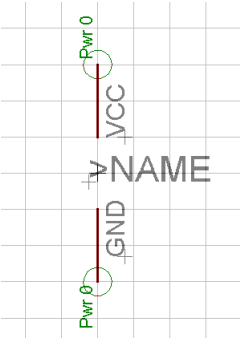


➢ ***NAND Symbol 7400 (European Representation)***

The two input pins are called *I0* and *I1* and are defined as having direction *In*, Swaplevel *1,* visible *Pin* and function *None*.

The output pin is called *O* and is defined with direction *Out*, Swaplevel *0*, visible *Pin*, and function *Dot*.

Now define the supply gate with the name *PWRN*, and the following properties:



➢ ***Power gate***

The two pins are called *GND* and *VCC*. They are defined with direction *Pwr*, Swaplevel 0, function *None*, and visible *Pad*.

Now create the *7400* Device in the Device Editor:

Specify the Package with PACKAGE (which must already be present in the library) and use PREFIX to specify the name prefix as *IC*.

Use the ADD command to place the *7400* Symbol four times, with Addlevel being set to *Next* and Swaplevel to *1*. Then label the Gates as A, B, C and D with the NAME command.

The Addlevel of *Next* means that as these Gates are placed into the schematic, they will be used in that sequence, i.e., the sequence in which they were fetched into the Device.

Then place the *PWRN* Symbol once, using Addlevel *Request* and Swaplevel 0. Name this Gate *P*.

Addlevel *Request* specifies two things:

- The supply gate will only be fetched into the schematic if requested, i.e. with the INVOKE command. The ADD command will only be able to place NAND gates.
- The supply gate will not be included when names are allocated to the schematic. Whereas an IC with two *Next* Gates appears in the schematic as something like IC1A and IC1B, an IC with one *Next* Gate and one *Request* Gate will only be identified as IC1.

So use the CONNECT command to define the housing pads to which the supply pins are connected.

Pins with the Same Names

If you want to define components having several power pins of the same name, let's suppose that three pins are all to be called GND, then proceed as follows:

- set pin direction *Pwr* for each power pin
- name these pins GND@1, GND@2, and GND@3

Only the characters in front of the "@" are visible in the schematic, and the pins are treated as if they were all called GND. In the board the referring pads are connected with airwires automatically.

8.4 Supply Symbols

Supply symbols, such as might be used in the schematic for ground or VCC, are defined as Devices without a Package. They are needed for the automatic wiring of supply nets (see page 120).

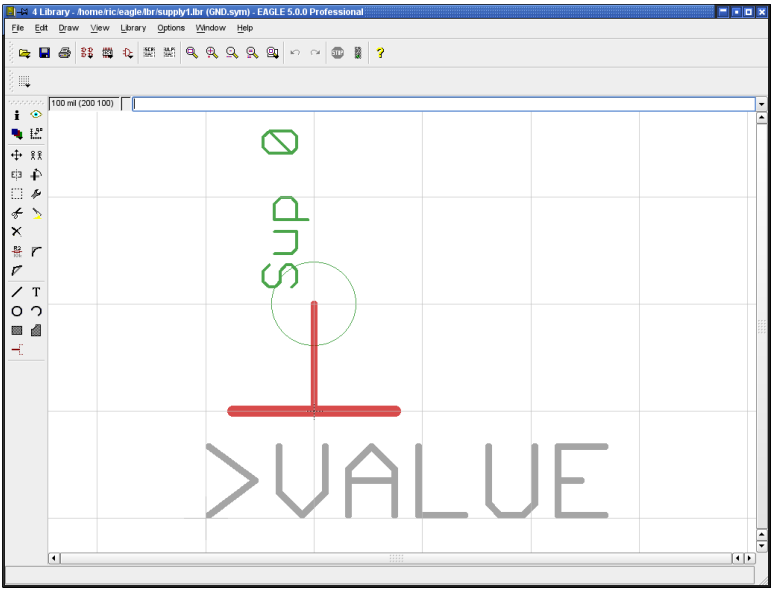
The following diagram shows a GND symbol as it is defined in one of the supplied EAGLE libraries.

Note that when defining your own supply symbols, the pin and the Device name need to agree.

The pin is defined with direction *sup* and has the name *GND*. This specifies that the Device containing this Symbol is responsible for the automatic wiring of the GND signal. The text variable for the value (>VALUE) is chosen for the labeling. The Device also receives the name *GND*. Thus the label GND appears in the schematic, since by default EAGLE uses the Device name for the value.

It is very important that the labeling reproduces the pin names, since otherwise the user will not know which signal is automatically connected.

The pin parameter *Visible* is therefore set to *off*, since otherwise the placing, orientation and size of the pin name would no longer be freely selectable. Directly labeling with the text *GND* would also have been possible here. With the chosen solution however, the Symbol can be used in various Devices (such as for DGND etc.).



➢ ***Supply symbol for GND***

> *The Supply symbol has no Package assigned!*

As has been explained above, the Device receives the name of the pin that is used in the Symbol. The corresponding Device is defined with Addlevel *Next*. If you set Value to *off* you can be sure that the labeling is not accidentally changed. On the other hand, you have more flexibility with Value set to *on*. You can alter the label if, for instance, you have a second ground potential. You must, however, then create explicit nets for the second ground.

Quick guide to define a Supply Symbol:

- Create a new Symbol in the library
- Place the pin, with direction Supply
- Pin name corresponds to the signal name
- Set Value placeholder

- Create a new Device
- Device name is signal name
- Package assignment not necessary

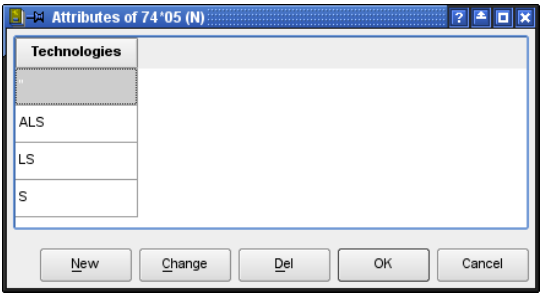
8.5 Attributes

You are allowed to define, additionally to *>name* and *>value*, further properties, the so-called attributes. It's possible to define attributes for each technology and Package variant in the Device editor. This chapter will guide you through the process of defining attributes with the help of an example.

Therefore open the library 74xx-us.lbr and save a copy of it with *Save as...* in an arbitrary directory. We don't want to change the original library for this. Edit the Device *74*05*.

Define Attributes

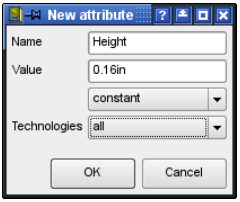
Let's define some attributes for the Package variant *N*, which is the *DIL14* Package. Therefore click onto entry DIL14 (Variant N) in the Package list on the right-hand side of the Device Editor window. Now click the ATTRIBUTE command icon in the menu bar or onto the text *Attributes* in the description window below the representation of the Device. The following Attribute window will appear.



➢ ***Attributes' dialog***

This dialog initially shows the Technologies available for the Package variant *N*. Clicking the *New* button opens the *New Attribute* window. Please enter, for example, *Height* for the attribute's *name* and *0.16in* for the attribute's *value*. The line below determines whether it is allowed to modify the value of the attribute (*variable*) or not (*constant*) in the drawing. Select *constant* in our example here.
Now you have still to define for which Technologies the attribute should be valid: for the currently selected one only (*current*) or for *all*. Select *all* here.

Click the *OK* button and the new attribute is shown in the list now.



➢ ***Defining the Height Attribute***

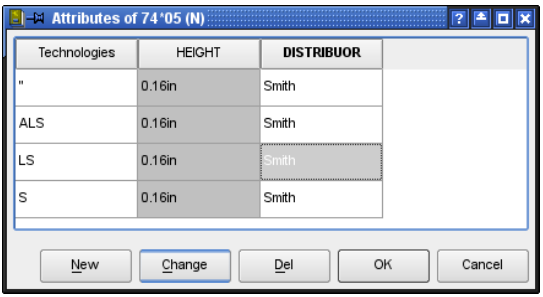
Let's define a second attribute that should have different values for the Technologies. Click the *New* button in the Attributes' dialog again. Enter the following parameters:

Name: *Distributor* Value: *Smith*, *variable* Technologies: *all*

Click *OK* now. A further column for the *Distributor* attribute is shown. All technologies have the *Smith* entry.

> *Attribute names are written in upper case letters automatically!*

But in our example the LS technology has to be distributed by *Miller* exclusively. Click into the field of the *Distributor* attribute that belongs to the LS technology.



➢ ***The Distributor field for LS is selected***

Click onto the *Change* button now. The window for changing the properties of the attribute opens. Set the following options:

Name: *Distributor* Value: *Miller exclusively*, *constant* Technologies: *current*

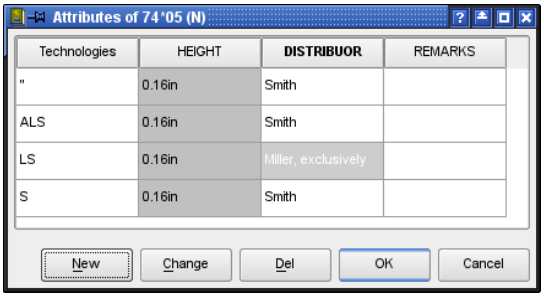
Click the *OK* button, and the exception for the LS technology is defined. This value can not be altered in the Schematic/Layout.
The *Change* dialog allows three possibilities in the *Technologies* field: *current*, *same*, *all*. This means that the currently changed properties will be valid for the currently selected (*current*), for all the technologies with the same

attribute value as the currently selected (*all with same value*) , or for *all* technologies.

Finally let's define a further attribute for remarks. This attribute will have no initial value and will be *variable.* So we can use it in the Schematic or in the Layout, if necessary.
Therefore click again the *New* button in the Attributes dialog and make the following settings:

Name: *Remarks* Value: -, *variable* Technologies: *all*

Click OK. The attributes window looks like this now:



➢ ***All the Attributes for 74*05, Variant N***

Attributes with a fixed value are colored gray in the table.

The definition of attributes for the Package variant *N* is finished now. Click *OK* to close the Attributes window now. The attributes are shown in addition to *Technologies* in the Device Editor window.

If you like to define attributes, for example, for the Package variant *D* (*SO14*), click onto the entry in the Package list of the Device Editor window and proceed as described above for variant *N*.

It's also possible to define attributes via the command line or with the help of a Script file. Please take a look into the help function about the ATTRIBUTE command for details.

Display Attributes

If you would use the Device *74*05* without further changes in the Schematic or Layout Editor, it would bring along its attributes and their values. The attributes are not visible in the drawing and can be check with the ATTRIBUTES command.
Information about how to display attributes in Schematic or Layout can be found on page 123 in this manual.

Placeholders in Symbol and Package

Already in the library you may define whether an attribute will be displayed together with the Device in the Schematic or the Package in the Layout. Simply write a placeholder text in the Symbol or Package with the TEXT command. Such a placeholder text begins with the > character, as it is with >*name* and >*value*. For our example attributes we defined above, you have to write:

```
>Distributor
>Height
>Remarks
```

Place this text at a suitable location in the Symbol or Package Editor and select a proper layer for each text. It doesn't matter if you write it with upper or lower case letters.

As soon as you add a part with pre-defined attribute placeholder texts and set a value for an attribute in Schematic and Board respectively, the attribute's value will be displayed at the placeholder text's location.
These texts can be separated from the Device/Package with the SMASH command. From then on the *Display* property of the *Attribute* dialog takes effect. The possible options are *Off*, *Value*, *Name*, or *Both*.

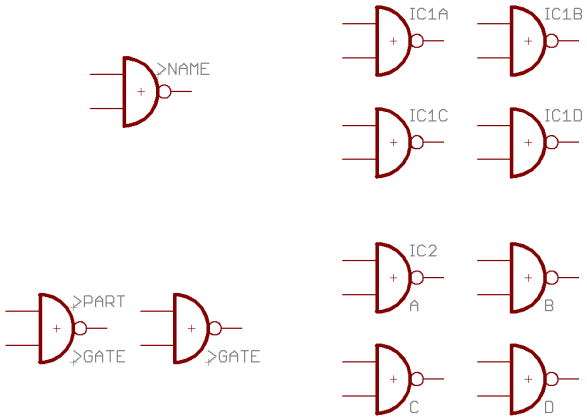
See page 123 for details about display options of attributes.

8.6 Labeling of Schematic Symbols

The two text variables >*NAME* and >*VALUE* are available for labeling Packages and schematic Symbols. Their use has already been illustrated. There are two further methods that can be used in the schematic: >PART and >*GATE*.

The following diagram illustrates their use, in contrast to >*NAME*. The Symbol definition on the left, the appearance in the schematic diagram on the right.

In the first case all the symbols are labeled with >NAME. In the second case, the symbol of the first gate is labeled with >PART and >GATE, the other three with >GATE only.



➢ ***Labeling of a schematic symbol***

8.7 More about the Addlevel Parameter

The Addlevel of the Gates that have been fetched determines the manner in which these Gates are fetched into the schematic, and under what conditions it can be deleted from the schematic.

Summary

Next: For all Gates that should be fetched in sequence (e.g. the NAND Gates of a 7400). This is also a good option for Devices with a single Gate. The ADD command first takes unused *Next*-Gates from components which exist on the current sheet before "opening" a new component.

Must: For Gates which must be present if some other Gate from the component is present. Typical example: the coil of a relay. *Must*-Gates cannot be deleted before all the other Gates from that component have been deleted.

Can: For Gates which are only used as required. In a relay the contacts may be defined with Addlevel *Can*. In such a case the individual contacts can be specifically fetched with INVOKE, and can later be deleted with DELETE.

Always: For Gates which as a general rule will be used in the schematic as soon as the component is used at all. Example: contacts from a multi-contact relay, of which a few are occasionally left unused. These contacts can be removed with DELETE, provided that they were defined with Addlevel *Always*.

Request: For supply gates of components.
The difference from *Can* is: A Device with exactly one *Next*-Gate and a *Request*-Gate will be named, for example IC1. The Gate name does not appear

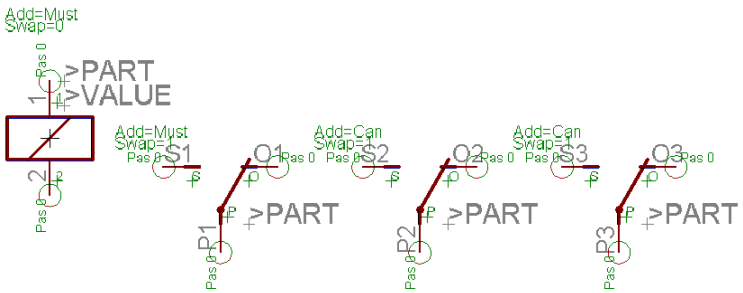
in the name of the part in the schematic. The *Request*-Gate's name, however will consist of *Prefix+Number+Gate name*, for example, *IC1P*.

Relay: Coil and First Contact must be Placed

A relay with three contacts is to be designed, of which typically only the first contact will be used.

Define the coil and one contact as their own Symbols. In the Device, give the coil and the first contact the Addlevel *Must*. All the other contacts are given the Addlevel *Can*.

If the relay is fetched into the schematic with the ADD command, the coil and the first contact are placed. If another contact is to be placed, this can be done with the INVOKE command. The coil cannot be deleted on its own. It disappears when all the contacts have been deleted (beginning with those defined with Addlevel *Can*).

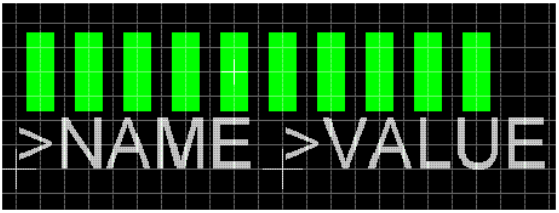


➢ ***Relay with one coil and three contacts***

Connector: Some Connection Pins can be Omitted

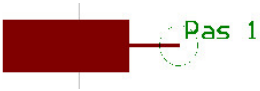
A PCB connector is to be designed in which normally all the contact areas are present. In some cases it may be desirable for some of the contact areas to be omitted.

Define a Package with 10 SMDs as contact areas, giving the SMDs the names 1 to 10.



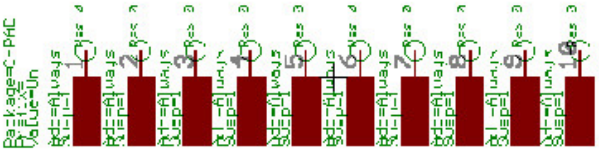
➢ ***Package of a circuit board connector***

Now define a symbol representing one contact area. Set visible to *Pad*, so that the names 1 to 10, defined in the Package, appear in the schematic.



➢ ***Connector symbol for the Schematic***

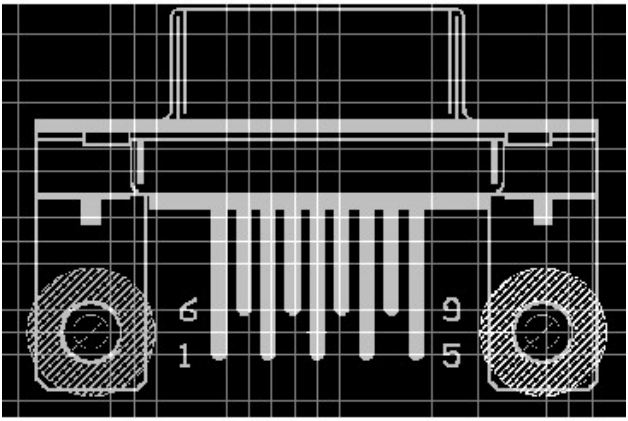
Then fetch the Symbol ten times into a newly created Device, setting the Addlevel in each case to *Always*, and use the CONNECT command to create the connections between the SMDs and the pins. When you now fetch this Device into a schematic, all the connections appear as soon as it is placed. Individual connections can be removed with DELETE.



➢ ***After ADD, all the connections are visible in the schematic***

Connector with Fixing Hole and Restricted Area

A connector is to be defined having fixing holes. On the solder side (bottom), the Autorouter must avoid bringing tracks closer to the holes than a certain distance.



➢ ***Fixing holes with restricted areas***

The drill holes are placed, with the desired diameter, on the Package using the HOLE command. The drilling diameter can be retrospectively changed with CHANGE DRILL.

The forbidden area for the Autorouter/Follow-me router is defined in layer 42, *bRestrict*, using the CIRCLE command. For reasons of representational clarity the circle is shown here with a non-zero width. Circles whose width is 0 are filled. In this case it has no effect on the Autorouter, since it may not route within the circle in either case. These forbidden areas are also taken into account by a polygon in layer 16, *Bottom*.

8.8 Defining Components with Contact Cross-References

If you have to design a component that consists of a coil gate and several contact gates for an Electrical Schematic, for example an electro-mechanical relay, you can define the contact symbols with a placeholder text that will generate cross-references for components. The contact overview in the Schematic will show the cross-references then.

For a proper display of the contact cross-references in the Schematic, please stick to the following rules for Symbol, Device, and Package definition.

Define Symbol

For defining an electro-mechanical relay you have to use one Symbol for the coil and one or more Symbols for the contacts.

Please note the following rules for the **contact symbols**:

- The center of the contact symbol should be located at position (0 0)

- Arrange the pins in vertical direction, i.e. they are pointing up or down
- In order to get automatically generated cross-references, use the TEXT command to define the placeholder text *>XREF* and place it. The text should be written in layer 95, *Names,* like *>NAME* and *>VALUE*.

There are no special rules for the **coil symbol**. The placeholder text *>XREF* is not needed here.

Define Device

Our electro-mechanical relay consists of multiple Gates: one Gate for the coil and several Gates for the contacts. The placement of the Gates in the Device Editor has to follow some rules. Otherwise the presentation of the cross-references in the Schematic would not be optimal.

- The origin of the first contact gate should be located at the x-coordinate 0. The lower pin of the Gate should be located completely in the positive coordinates range. The y-coordinate is typically 0.1 inch.
- Each further contact gate is placed to the right of the first one at the same y-coordinate (the same height).
  The distance between the contact gates in the Device Editor finally determines the distance of the contacts in the graphical representation of the contact cross-references in the Schematic. The contact gates will be rotated by 90° and aligned vertically one by one there.
- The coil gate may be placed anywhere in the Device drawing. The Addlevel for this Gate must be *Must*.

The representation of the contact cross-references shows all Gates that come with the *>XREF* text. The cross-references consisting of sheet numbers and column/row coordinates will be shown on the right of the Gates, if you placed a drawing frame defined with the FRAME command on the Schematic's sheets.
All other texts defined in the Symbol are not visible in the cross-reference representation.

Define Package

Due to EAGLE's library structure and in order to avoid error messages you have to define a Package, as well. This can be a simple dummy Package that simply has the same number of Pads as number of Pins in the Device.

Select the Package with the *New* button in the Device Editor and assign Pins with Pads with the CONNECT command.

Further information about the subject *Electrical Schematics* can be found on CadSoft's web site.

8.9 Drawing Frames

It may be true that drawing frames are not components, but they can be defined for schematics as Devices with neither Packages nor pins. Such Devices in EAGLE's *frames* library contain a Symbol consisting merely of a frame of the appropriate size, and a documentation field, which is also defined as a Symbol.

A drawing frame is defined with the FRAME command. This command can be found in the *Draw/Frame* menu.

The parameter toolbar offers settings for the number of columns and rows where you can define how your drawing should be fielded. A positive value for columns labels the frame from the left to the right, beginning with *1*, for rows from top to bottom, beginning with *A*. Negative values inverse the direction of the labelling. The following four icons determine on which position the labelling of the frame shall be visible.

➢ ***Parameter toolbar of the FRAME command***

The position of the drawing frame is fixed by two mouse clicks or by typing the coordinates of its corners in the command line.

Columns and Rows can be used to determine a Device's or a net's position, for example with the help of an ULP, or to have cross-references calculated automatically (see LABEL command).

Is the frame already defined but you want to change its properties?
Then use the CHANGE command with its options *Border*, *Rows* and *Columns* to determine the frame's position of the labelling and its number of rows or columns.

> *Due to the special nature of the frame object, it doesn't have a rotation of its own!*
>
> *The FRAME command is also available in Schematic or Board. But it is common practice to define a drawing frame in the Library.*

The library *frames.lbr* also contains documentation fields you can use together with a frame. Of course you are allowed to draw your own.

The text variables *>DRAWING_NAME*, *>LAST_DATE_TIME* and *>SHEET* are contained, as well as some fixed text. The drawing's file name, date and time of the last change appear at these points together with the sheet number in the schematic (e.g., 2/3 = sheet 2 of 3).

In addition, the following variables are available:
>PLOT_DATE_TIME contains the date and time of the last printout,
>SHEETS shows the total number of sheets in the schematic,
>SHEETNR shows the current sheet number.

All of these text variables can be placed on the schematic, and (with the exception of *>SHEET/S/NR*) on the board.

TITLE: >DRAWING_NAME		
Document Number:		REV:
Date: >LAST_DATE_TIME	Sheet: >SHEET	

➢ ***Text variables in the documentation field***

The frame is defined in the Device with Addlevel *Next*, and the documentation field with Addlevel *Must*. This means that the documentation field cannot be deleted as long as the frame is present.

There are frames defined as Packages available for the Layout Editor which can be placed even if there is a consistent schematic/layout pair. These frames don't have any electrical significance because they are defined without pads or SMDs.

The variable *>CONTACT_XREF* has a special meaning for Electrical Schematics. The position of this text, which is not displayed in the Schematic, determines the reserved area for the representation of the contact cross-references. More details about this can be found in the help function in the section *Contact cross-references*.

8.10 Components on the Solder Side

SMD components (and leaded ones too) can be placed on the top or bottom layers of a board. For this reason EAGLE makes a set of predefined layers available which are related to the top side (*Top*, *tPlace*, *tOrigins*, *tNames*, *tValues* etc.) and another set of layers related to the bottom side (*Bottom*, *bPlace* etc.).

SMD components are always defined in the layers associated with the top.

In the board, a component of this sort is moved to the opposite side with the MIRROR command . Therefore click onto the component with the mouse or enter the component's name in the command line. This causes objects in the *Top* layer to be reflected into the *Bottom* layer, while all the objects in the *t..* layers are reflected into the corresponding *b..* layers.

If one of the commands ADD, COPY, MOVE, or PASTE is active the component can be mirrored by clicking the middle mouse button.

8.11 Components with Oblong Holes

If the board manufacturer have to mill oblong holes, you have to draw the milling contour of oblong holes in a separate layer. Usually this is layer 46, *Milling*.

The milling contour for components that need oblong holes can be drawn with WIRE (and possibly ARC) with a very fine wire width near or even 0 in the Package Editor. Take a pad that has a drill diameter which lies inside the milling contour, or SMDs, for example in Top and Bottom layer, as basis for the oblong hole.

In case of a multilayer board you should draw a WIRE in the used inner layers at the position of the oblong holes so that it covers the milling contour and leaves a kind of restring around the opening.

Please inform your board manufacturer that they have to take care on the milling data drawn in this layer. Also tell them whether they should be plated-through or not.

> *Any other cut-outs in the board are drawn in the same way:*
> *Use a separate layer, typically layer 46, Milling, and draw the milling contours. Tell your board manufacturer that they have to take care with this information and make special note.*

8.12 Creating New Package Variants

Most components are manufactured in various Package variants. Supposed you do not find the appropriate Package for a certain Device in one of the libraries, it is very easy to define a new Package.

To describe this procedure clearly we want to come back again to our example Device *541032A* from paragraph 8.2.

> *The third Package variant to be designed here only serves as an example for practice and does not meet the specifications of the manufacturer!*

Please notify the explanations concerning this topic, in particular if the appropriate Package already exists in the current library beginning with page 231.

Package from Another Library

In the most favourable case you can use an already existing Package from another library. The easiest way to define the new Package variant is to use the PACKAGE command directly in the Device Editor.

After searching all libraries, either in the *Libraries* branch of the tree view in the Control Panel or with the help of the search function of the ADD command in the Layout Editor (only there, not in the Schematic, one can search for Packages), it is possible to Drag&Drop the Package from the Control Panel directly into the library. As an alternative you could use the PACKAGE command.

Open the library (here: *my_lib.lbr* from paragraph 8.2) that contains the Device you want to define the new Package variant for. For example, by the menu *File/Open/Library* of the Control Panel.

Click the *Edit-a-Device* icon and select the Device *54*1032A* from the menu. The Device Editor opens.

Defining the Package Variant

The new variant should be named *Test*. The Package must have a minimum of 14 pads because both Gates together have 14 pins. As an example, we take the *SO14* Package from the *smd-ipc.lbr* library.

If Control Panel and Library Editor window are arranged side by side, select the *SO14* Package and Drag&Drop it into the opened Library Editor window. After releasing the mouse button you will be asked for the new Package variant name. Enter it and confirm it by clicking *OK*. The new variant is now shown in the Package list.

It is also possible to define the Package variant in the Device Editor directly with the PACKAGE command.

Type in the command line:

```
PACKAGE SO14@smd-ipc.lbr TEST
```

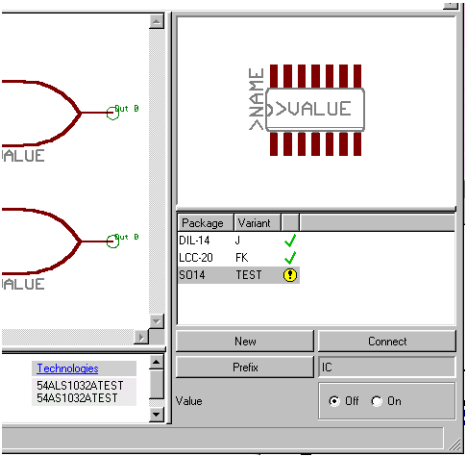
Or include the path (if necessary):

```
PAC SO14@d:\eagle\lbr\smd-ipc.lbr TEST
```

If the path contains spaces include the path name in single quotes, for example:

```
PAC 'SO14@\With Spaces\smd-ipc.lbr' TEST
```

Now on the lower right of the Device Editor window a new entry for the Package *SO14* and the variant name *TEST* appears.
On the left a black exclamation mark on yellow ground is shown which indicates that there are no connections between pins and pads defined yet.



➢ ***Device Editor: List of Package Variants***

> *The PACKAGE command copies the complete Package definition into the current library and makes available the new variant with the given name for the Device.*
>
> *If you decide to erase a newly defined variant, you can do this with the UNDO function (as far as possible) or by using the context menu of the Package entry (right mouse click, Delete entry).*

Connect Command

Click the *Connect* button now. The *Connect* window opens. Connect pins with pads by clicking on the pin and pad entries belonging together as described in paragraph 8.2.
It is also possible to adapt the pin/pad connections from an already existing Package variant. In our example the assignment does not differ from the DIL14 Package. Therefore select the entry *DIL14* from the *Copy from:* combo box.
After clicking *OK* the CONNECT command is finished.

Defining Technologies

The Device *54*1032A* is available in two technologies (ALS and AS). These still have to be set up for the new Package variant.

Select the Package variant *Test* from the list on the lower right of the Device Editor window. A click onto *Technologies* in the description field opens a window. Click the *New* button and set up technology ALS with a following click onto *OK*, and AS again with a following click onto *OK*. Both entries are

shown with a tag now. A further click onto the *OK* button closes the window again.

Save

The definition of the Package variant is finished. Now it is time to save the library.

Using a Modified Package from Another Library

If there is no appropriate but a similar Package available in another library you should copy the Package in the current library first, then modify it, and use it afterwards as new variant for the Device.

Copying the Package

We want to use a Package named *SOP14* from the *smd-ipc.lbr* library here. This Package should get a new name, *MYSOP14*, in the library *my_lib.lbr*.

Using the COPY command

Type in the command line of the Library Editor window (it does not matter which editor mode is active) the following:

```
COPY SOP14@smd-ipc.lbr MYSOP14
```

Or with the whole path:

```
COPY SOP14@d:\eagle\lbr\smd-ipc.lbr MYSOP14
```

If the path contains spaces use single quotes for it, for example:

```
COPY 'SOP14@\P A T H \smd-ipc.lbr' MYSOP14
```

The Package Editor window opens and the Package can be modified as needed.

In the Control Panel

First of all open a Library Editor window with the library that should contain the new Package (*File/Open/Library*). It is not necessary to select a certain editing mode. Now switch to the Control Panel (e. g. *Window* menu) and expand the *Libraries* branch of the tree view. Choose the library which contains the requested Package and select it. On the right half of the Control Panel a preview of the Package is visible now.

If the Control Panel and the Library Editor window are arranged in a way that both windows are visible you can move the Package into the Library window by keeping the left mouse button pressed. After releasing the mouse button (Drag&Drop) the Library Editor will be in the Package editing mode. The Package is shown there.

Alternatively you could use a right mouse click to open the context menu of the Package entry in the tree view. Select *Copy to Library* now. The Library Editor needs not to be visible.

Now the Package can be modified. The Package name is adopted from the source library. To change the Package name use the RENAME command.

Don't forget to save the library.

Defining the Variant

We want to define a further variant for our example Device. Switch to the Device editing mode, for example, by the menu *Libraries/Device*. The *Edit* window opens. Select the entry *54*1032A*. Click *OK* to open the editor window.

Use the *New* button to define a new variant. Select the Package *MYSOP14* in the selection dialog and enter, for example, *TEST2* as *variant name*. After clicking *OK* a new entry is shown in the Package list.

To complete the definition execute the CONNECT command and define Technologies (as described in the previous paragraph) now.

8.13 Defining Packages in Any Rotation

Components can be defined in any rotation with a resolution of 0.1 degrees in the Package Editor. Usually the Package is defined in normal position first and rotated afterwards as a whole. The definition of Packages has been already explained in this chapter. Here we only want to elaborate on the rotation of Packages.

> *Packages can be defined in any rotation! Schematic Symbols can be rotated in 90-degrees steps only!*

Rotating a Package as a Whole

To come back to the example of this chapter, please open the library *my_lib.lbr* and edit the Package *LCC-20*.

Display all layers with `DISPLAY ALL` to make sure you have all objects rotated. Now use `GROUP ALL` to select everything.

Use the ROTATE command to rotate the group:
Now click with the left mouse into the *Angle* box of the parameter toolbar and type in the requested angle. Then use a right mouse click into the group to define the rotation point.
The Package is shown now in the given angle.

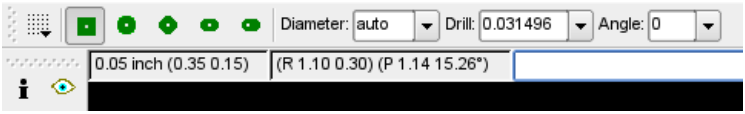
Alternatively you can work with the command line:

```
ROTATE R22.5 (> 0 0)
```

rotates, for example, the previously selected group 22.5° further around the point (0 0). The > sign (right angle bracket) within the parenthesis for coordinates causes the rotation of the whole group (as a right mouse click at the point (0 0) would do).

Packages with Radial Pad Arrangement

It is possible to work with polar coordinates to place pads or SMDs in a radial arrangement. Set a suitable reference point, for example, in the center of the Package with the MARK command first. The command line shows now additional information about the cursor position.



➢ ***Package Editor: Relative and Polar Coordinates Display***

Values marked with an `R` are relative values referring to the previously set reference point. The leading `P` indicates polar values referring to the reference point.

Example:
Three pads are to be placed on the circumference of a circle with a radius of 50 mm. The center of the part is at position (0 0).

```
GRID MM;
MARK (0 0);
PAD '1' (P 50 0);
PAD '2' (P 50 120);
PAD '3' (P 50 240);
```

Depending on the used pad shape it may be useful to place the pads rotated (for example for *Long* pads or SMDs).

It is possible to enter the angle directly in the parameter toolbar or in the command line while the PAD or SMD command is active.

Example:

```
GRID MM ;
MARK (0 0);
PAD '2' LONG R120 (P 50 120) ;
```

8.14 Library and Part Management

Copying of Library Elements

Within a Library

If you want to use a **Symbol** or a **Package** which already exists in a related manner for a Device definition you can copy it within the library with the commands GROUP, CUT, and PASTE. Afterwards it can be modified as requested.

The following sections explain every single step with the help of an example Package taken from *linear.lbr*.

Open Library

Use the menu *File/Open/Library* in the Control Panel to open the library *linear.lbr* or select the entry *Open* from its context menu of the tree view's expanded *Libraries* branch.

Edit Existing Element

Open the *Edit* window with *Library/Package* and select the Package *DIL08*. After clicking *OK* it is shown in the Package Editor window.

Use DISPLAY to show all layers.

Draw a frame around all objects to be copied with GROUP or type `GROUP ALL` in the command line.

Now click the CUT icon followed by a left mouse click into the group.
Alternatively you may enter a reference point in the command line for CUT, as for example:

```
CUT (0 0) ;
```

Define New Element

Click the *Edit-a-package* icon in the action toolbar.

Enter the name *DIL08-TEST* in the *New* field of the *Edit* window and confirm with *OK*.

Click the PASTE icon followed by a click at the drawing's reference point. The Package will be placed.
Or use again the command line:

```
PASTE (0 0) ;
```

Now it can be modified as requested.

The given coordinates for CUT and PASTE can be used to move a group by a certain value in the coordinates system. This may be valuable for elements that have been drawn in the wrong grid.

> *This procedure can be applied to Symbols too!*
> *Devices can't be copied within a library!*

From One Library into Another

Devices

If there is a proper Device or Device set that you want to use in your current library you can copy it in two different ways.

In the Control Panel:
Move (with Drag&Drop) the requested Device set from the Control Panel's tree view into the opened Library Editor window. The complete Device set with Symbol(s) and Package(s) will be copied and newly defined.
As an alternative you could use the entry *Copy to Library* in the context menu of the Device entry.

With the COPY command:
Type, for example,

```
COPY 75130@751xx.lbr
```

or with the whole path

```
COPY 75130@d:\eagle\lbr\751xx.lbr
```

in the command line, the Device *75130* from library *751xx.lbr* is copied into the currently opened library.

If the path contains spaces use single quotes for it, for example:

```
COPY '75130@d:\P A T H\751xx.lbr'
```

If the Device should be stored in the current library under a new name simply enter it, like here:

```
COPY 75130@751xx.lbr 75130NEW
```

Symbols

As Symbols are not shown in the tree view of the Control Panel it is not possible to copy them from there into the Library Editor.
Symbols are copied from one library into another the same way as one would do it within the same library. Therefore we use the commands GROUP, CUT, and PASTE (see previous paragraph).

The only difference to copying a Symbol within a library is that you have to open the other library (*File/Open*) after CUT and click into the group. Then you define a new Symbol that may have the same or another name as before.

In short:

- OPEN the source library and EDIT the Symbol
- DISPLAY all layers
- Select all objects with GROUP ALL
- Use CUT and left mouse click into the group to set a reference point
- OPEN your target library with *File/Open*
- EDIT a new Symbol
- Use PASTE to place it
- Save library

Packages

The procedure to copy Packages is nearly the same as to copy Devices.

Either move (with Drag&Drop) the requested Package from the Control Panel's tree view into the opened Library Editor window. The complete Package will be copied and newly defined in the current library. As an alternative you could use the entry *Copy to Library* in the context menu of the Package entry.

Or use the COPY command. Type, for example,

```
COPY DIL16@751xx.lbr
```

in the command line, the Package DIL16 from library *751xx.lbr* is copied into the currently opened library. If the library is not in the current working directory you have to enter the whole path, as for example, in:

```
COPY DIL16@\eagle\mylbr\751xx.lbr
```

If the path contains spaces use single quotes for it:

```
COPY 'DIL16@D:\P A T H \mylbr\751xx.lbr'
```

If the Package should be stored in the current library under a new name simply enter it directly in the command line:

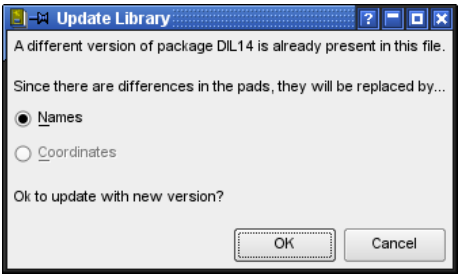
```
COPY DIL16@\eagle\mylbr\751xx.lbr DIL16NEW
```

The Package is stored with the new name DIL16NEW now.

If you want to copy a Package that already exists with the same name in the target library the Package will be simply replaced.

If the Package is already used in a Device and the position or the name of one or more pads/SMDs changes, EAGLE prompts a message in which mode the pads/SMDs are to be replaced. This procedure can also be cancelled. The Package remains unchanged then.

If the enumeration and position of the pads are unchanged but the order is, EAGLE will ask you for the appropriate update mode. Depending on your selection the pin/pad connections of the Device may change (see CONNECT command).



➢ ***Requesting the Update Mode***

Composition of Your own Libraries

The previously mentioned methods to copy library elements make it very easy to compose your own libraries with selected contents.

Provided the Control Panel and the Library Editor window are arranged in a manner that both are visible on the screen at the same time, it is very easy to make user-defined libraries while browsing through the library contents in the Control Panel. Simply use Drag&Drop or the context menu *Copy to Library* of the current Device or Package.

Removing and Renaming Library Elements

Devices, Symbols, and Packages can be removed from a library with the REMOVE command. Defining a new library element can't be cancelled by UNDO.

Example:
You would like to remove the Package named DIL16.
Open the menu *Library/Remove...*. A dialog window opens where you can enter the name of the element to be deleted.

This can be done also at the command line:

```
REMOVE DIL16;
```

Packages and Symbols can be removed only if they are not used in one of the library's Devices. In this case the message *Package is in use!* or *Symbol is in use!* appears. Remove the corresponding Device first or delete the particular Package or Symbol in the Device (set).

Would you like to change the name of an element in your library? Then use the RENAME command.
Switch to the Package editing mode so that the element that should be renamed is shown first and open the menu *Library/Rename*. A dialog window opens where you can enter the new name of the element.

This can also be done at the command line:

```
RENAME DIL16 DIL-16;
```

The Package DIL16 gets the new name DIL-16.

The Device, Symbol, or Package name may also be given with its extension (*.dev*, *.sym*, *.pac*), for example:

```
REMOVE DIL16.PAC
```

In this case it is not necessary to switch to the related editing mode before.

Update Packages in Libraries

As already mentioned in the section *Copying of Library Elements* it is possible to copy Packages from one library into another one. An already existing Package is replaced in that case.
Each library contains Packages which are needed for Device definitions. In many libraries identical types of Packages can be found. To keep them uniform over all libraries it is possible to replace all Packages of a library with those of another library with the help of the UPDATE command. An existing Package with the proper name will be replaced by the current definition.

If you have, for example, special requirements for Packages you could define them in a custom-built Package or SMD library. The UPDATE command could transfer them to other libraries.

Therefore open the library to be updated and select *Library/Update...*. Now select the library which you want to take the Packages from.

Having finished the update EAGLE reports in the status bar:
Update: finished - library modified!
If there was nothing to replace: *Update: finished - nothing to do.*

It is also possible to use the command line for this procedure.

If you want to update your library with Packages from, for example, *ref-packages.lbr*, type:

```
UPDATE ref-packages.lbr
```

To transfer Packages from different libraries, type in one after another:

```
UPDATE ref-packages.lbr rcl.lbr smd-special.lbr
```

To update a single Package, type in the Package name:

```
UPDATE SO14@ref-packages
```

The extension *.lbr* is not necessary. You may also use the whole library path. See page 258 for further information.

This
page
has been
left free
intentionally.

Chapter 9
Preparing Manufacturing Data

Data output for board manufacturing is made with the help of the CAM Processor. PCB manufacturer usually work with drill data in *Excellon* format and plot data in *Gerber* format. How to generate such data and which data you have to pass on to your PCB manufacturer will be explained in this chapter.

A lot of PCB manufacturers generate these data with EAGLE by themselves. In such a case you have to pass on the board file only and you need not care about data generation. You will find links to such firms on our Internet pages.

If, however, your board maker is not set up to process EAGLE board files directly, you will have to supply them with a set of files. What will be required will be discussed in the following sections.

Additional useful User Language Programs (ULPs) are available on CadSoft's Internet pages. They can be used, for example, for the generation of glue mask data, for the calculation of milling contours, or for data regarding automatic mounting and testing equipment.

9.1 Which Data do we Need for Board Manufacture?

The PCB manufacturer requires specific information pertaining to each step in the manufacturing process of your board. This special information is described in a file containing plot and / or drilling information.
For example, one file for each signal layer, for the silkscreen, for the solder stop mask, the cream frame, for a gold application, for a glue mask (for SMT devices), or for milling data regarding cut-outs in the board.
Double-sided boards with parts on top and bottom side require a silkscreen on both sides, or in case of SMT devices, a cream frame or a glue mask for each side.

Additionally the board manufacturer needs drilling data in a separate file.

If you want a milled prototype board, milling contours have to be calculated first, and generated in a specific data format for fabrication milling machines.

If you want to have the parts automatically mounted, you need additional files in appropriate data format that depict centroid and rotational angular information.

A bill of materials or a legend for the drill symbols can be helpful, too.

Gerber Plot Data

All PCB manufacturers use Gerber format. There are two options of Gerber format available, *Extended Gerber* format, also known as *Gerber RS-274X* (in short *RS-274X*), which is the most commonly used today in industry.
The CAM Processor offers this device option as *GERBER_RS274X*.

It may be the case that the PCB manufacturer works with the second option *RS-274D*. This will require the generation of data with the devices *GERBERAUTO* and *GERBER* of the CAM Processor.

Gerber data (RS-274D) basically consist of two parts:
The so-called *Aperture file* or *Wheel file*, a special tool table, and the plot data that contain coordinates and plotting information for the Gerber plotter.

The Gerber devices of the CAM Processor (RS-274X and RS-274D) have a resolution of 1/10,000 of an Inch (data format: 2.4, inches).
Exceptions are the devices *GERBERAUTO_23* and *GERBER_23*. They have a lower resolution of 1/1,000 of an Inch (data format 2.3, inches). *GERBER_RS274X_25* offers a higher resolution of 1/100,000 of an Inch (data format: 2.5, inches)

> *Please inquire which format your PCB manufacturer prefers. The more convenient to use is Extended Gerber, RS-274X.*

GERBER_RS274X

This device generates files in Extended Gerber format (RS-274X) where the aperture table is integrated in the output file. Simply generate Gerber files with the GERBER_RS274X device and pass them on to your board manufacturer. This is the most efficient and easiest way to generate Gerber data.

GERBERAUTO and GERBER

The prior RS-274D format works with a separate aperture file which is necessary to generate all Gerber files you will need for board manufacturing.

First you have to generate the aperture table (wheel file) with *GERBERAUTO*. This file is a tool table which defines the shape and the size of the Gerber plotter's apertures (tools). This table must contain the whole aperture definition we need for generating all Gerber files that describe the board.
For generating the Gerber files with the *GERBER* device, we refer to the previously generated aperture file, made with *GERBERAUTO*.

Drill Data

The generation of drill data is very similar to the generation of plot data. Typical formats used in industry are *Excellon* or *Sieb&Meyer 1000* or *3000*. They are supported by the CAM Processor. The most common one is *Excellon*.

The simplest case is to generate one common drill data file for all drill holes.
If you have to distinguish plated from non-plated drill holes, two drill data files must be generated. EAGLE differentiates between plated drills of Pads and Vias in layer 44, *Drills*, and non-plated holes in layer 45, *Holes*, which are placed by the HOLE command.

If you have to generate drill data for a multilayer board that uses Blind and Buried vias with different via lengths that result in different drilling depths, the CAM Processor takes care on this automatically. For each via length it generates a separate drill data file.
Further information about this can be found in chapter 9.5 from page 279 on.

EXCELLON

Using this device the CAM Processor generates a drill file that contains the drill table and the drill coordinates. This file format is the most common in the industry and will be recognized by most board manufacturers.
The default resolution of the EXCELLON device is 1/10,000 inches, no leading zeros (data format: 2.4, Inch).

EXCELLON_RACK

This device can be used, if your board manufacturer insists on two separate files. A drill table (rack file) and the drill data file. This was default in prior EAGLE versions.

In the first step one has to generate the drill table in the Layout Editor with the help of the User Language program *drillcfg.ulp*, and refers to this drill table when generating drill data with the CAM Processor. The board manufacturer gets two files then, drill data and drill table.

If you have to distinguish between plated and non-plated drillings you have to supply one drill table and two drill data files (one for layer *Drills*, one for layer *Holes*).

SM1000 and SM3000

These devices generate drill data in Sieb&Meyer 1000 or in Sieb&Meyer 3000 format. *SM1000* has a resolution of 1/100 mm, *SM3000* 1/1000 mm.

Data output is exactly the same as it is with *EXCELLON_RACK*. First you have to generate the drill table with *drillcfg.ulp*, then use the CAM Processor to generate drill data.

Further Drill Data Devices

The CAM Processor supports two further devices for drill data generation.
GERBDRL generates Gerber drill code. Here we need a separate drill table (RUN drillcfg.ulp), as it is with *EXCELLON_RACK*.
SMS68 is a further drill data device that generates a HPGL code.

Prototype Manufacture With a Milling Machine

With the help of various User Language programs you can generate outline data for milling a prototype board.

outlines.ulp

A simple example for contour data calculation is *outlines.ulp*. Start it with the RUN command. Select the layer for which outline data shall be generated, define the diameter of the milling tool (*Width*), and select the output file format (*Script* or *HPGL*) in the ULP's dialog window.
The *Script* file containing the outline data can be imported into EAGLE with the SCRIPT command. Thus it's possible to visualize the calculated contours in the Layout Editor. You can even modify them, if necessary.
Finally, the milling data output is made with the CAM Processor. Select the layer where the contours are to be drawn in and use, for example, the *HPGL*, the *PS* (Postscript) or one of the Gerber devices for the output.

Further information can be found in the help function, *Outline data*.

mill-outlines.ulp

Another User Language program that calculates outline and drill data is *mill-outlines.ulp*. It offers various configuration parameters. Simply start it with the RUN command in the Layout Editor. Consult the ULPs integrated help function for details.

This ULP exports HPGL formatted data (further formats are planned) or generates a Script file which can be imported into the layout again. The milling contours can be viewed, or even modified, if required. Generate the milling data with the CAM Processor and one of its devices, like Gerber, HPGL or PS then.

Film Generation Using PostScript Files

A high-quality alternative to Gerber is the data generation for PostScript raster image recorders. The exposed film serves as master for board manufacturing.

With the *PS* driver, the CAM Processor generates files in PostScript format. These files can be processed directly by appropriate service companies (most of which operate in the print industry).

For PostScript recorders the *Width* and *Height* parameters should be set to very high values (e.g. 100 x 100 inches), so that the drawing is not spread over several pages.

For generating Postscript files you have to select the proper layers, the same way it is performed for Gerber data generation. Use these files for the generation of your artwork at your particular PCB subcontractor.

Films that relate to the bottom side are usually output in mirrored form (*Mirror* option in the CAM Processor). In doing so the coating of the foil rests directly on the copper layer of the board which is to be exposed.

The *EPS* driver generates Encapsulated PostScript files. They can be processed with Desktop Publishing programs.

Printing on a Film

For boards of limited complexity, one can use a laser or ink jet printer and print on a transparent foil with the PRINT command. This method is used, for example, by hobbyists and results in a shorter fabrication time and a less expensive board fabrication process.

The layers that are displayed in the Layout Editor while printing will appear on the film. Check the options *Black* and *Solid* in the print dialog.

The drills of pads and vias are visible on the printout. This will allow an easy visual indication of where you have to drill manually on the board. Experience shows that the opening of a pad or a via should not be too big to allow for a good centering of the drill bit. This issue can be solved with the help of an User Language program, named *drill-aid.ulp*. Start it before printing, and let it draw a ring inside each pad and via in a separate layer. The inner diameter of this ring can be defined and is usually set to 0.3mm. Of course, you have to display this additional layer for printing on the film.

Data for Pick-and-place Machines and In-circuit Testers

EAGLE includes some ULPs which create data for various automatic placement machines and in-circuit testers that are typically used by PCB manufacturers.

The description of an ULP can be viewed in the *User Language Programs* branch of the tree view of the Control Panel by selecting one of the ULP entries with the mouse. The describing text appears on the right side of the Control Panel window. It's also possible to edit the ULP file with a text editor. The description usually is written in the file header.

ULPs for pick-and-place data (selection):

mount.ulp	Generates one file with coordinates of the centered part origins
mountsmd.ulp	Centered origins for SMT devices; one file for top and one file for bottom side

ULPs for circuit tester (selection):

dif40.ulp	DIF-4.0 format from Digitaltest
fabmaster.ulp	Fabmaster format FATF REV 11.1
gencad.ulp	GenCAD format for Teradyne/GenRad in circuit tester
unidat.ulp	UNIDAT format

Execute the RUN command in the Layout Editor window to start the particular ULP.

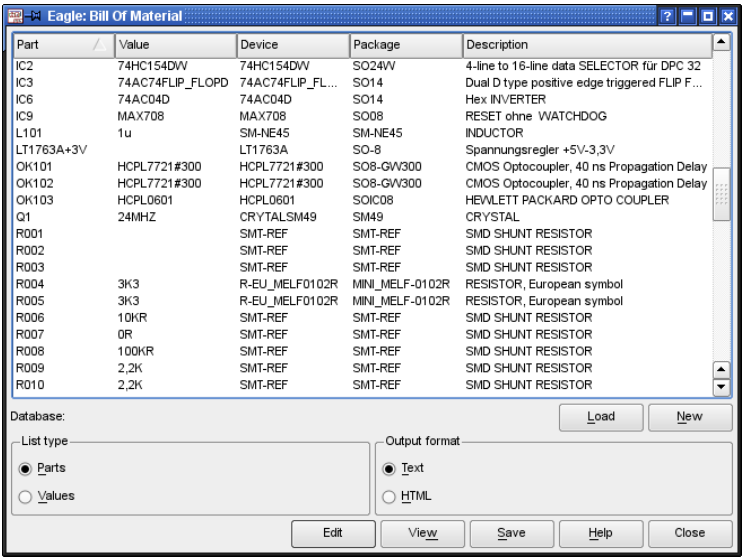
Note also the wide range of programs that are made available on the CadSoft web server http://www.cadsoftusa.com/download.htm, ULP directory.

Documentation

Many documentation items can be generated with the aid of User Language programs. Note also the wide range of programs that are made available on our web server. The bom.ulp, the program for generating a bill of materials, has been used as a basis for lots of user-contributed ULPs.

Parts List

The parts list can be created by *bom.ulp*. Start it from the Schematic Editor, using the RUN command. The *Bill Of Material* window with the parts summary opens first.



➢ ***bom.ulp: Dialog window***

It is possible to import additional information from a database file into the parts list (*Load*), or to create a new database with its own properties such as manufacturer, stores number, material number or price (*New*).
You can obtain further details about the current version of the ULP by clicking the help button.

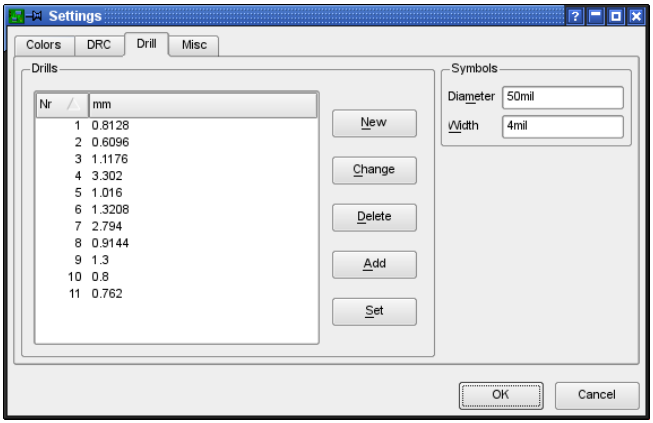
> *A simple parts list can also be created from a board or schematic by means of the EXPORT command (Partlist option).*

Drill Plan

Printing a drill plan enables you to check the drill holes and their diameters. It shows an individual symbol for each diameter of hole, via, and pad used in your design. EAGLE uses 19 different symbols: 18 of them are assigned to a certain diameter; one (∅) appears, if no symbol has been defined for the diameter of this hole. The symbols appear in layer 44, *Drills*, at the positions where pads or vias are placed, and in layer 45, *Holes*, at the positions where holes are placed.

The relation between diameters and symbols is defined through the Layout Editor's *Options/Set/Drill* dialog.

The buttons *New*, *Change*, *Delete* and *Add* can be used to create a new table, to modify certain entries, delete them or to add new ones.
The *Set* button extracts all the hole diameters from the layout and automatically assigns them to a drill symbol number. The values of *Diameter*



➢ ***Configuration of the drill symbols***

and *Width* determine the diameter and line thickness of the drill symbol on the screen and the printout.

The image above shows that drill symbol 1 is assigned to a drill diameter of 0.01 inch. In the following image you can see how the related symbol drawn in layer 44, *Drills*, or 45, *Holes*, looks like. The symbol number 1 looks like a plus character (+).

➢ ***Assignment of the drill symbols***

The dill symbol assignment is stored in the user-specific file *eaglerc.usr* (*.eaglerc* for Linux and Mac).

Drill Legend

Documenting the drill symbol assignment is quite simple with the help of a handy User Language program named *drill-legend.ulp*.

In the first step we let EAGLE generate the drill symbol assignment for the current layout with the *Set* button in the *Options/Set/Drill Symbols* menu.

Now we start *drill-legend.ulp*. It draws a table with the proper drill symbol assignment and the drill symbols at their positions in the board in the newly generated layer 144. For printing, it can be helpful to display layer 20, *Dimensions*, additionally.

If you want to delete this all, simply use GROUP and DELETE in layer 144.

9.2 Rules that Save Time and Money

- Each layer should without fail be uniquely identified (e.g. CS for Component Side, BS for Bottom Side).
- It may be wise to use fiducial or crop marks which can be defined in layer 49, *Reference*. This will allow easy alignment of PCB generated films for both inspection and fabrication. When generating manufacturing data, this layer has to be active additionally with all signal layers. Please contact you board manufacturer concerning this matter. Fiducials can be found in *marks.lbr*. A minimum of three fiducials or crop marks (three corners) is required for proper film alignment reference.
- For cost reasons you should, if at all possible, avoid tracks that narrow to below 8 mil.
- Usually the contour of the board is drawn in layer 20, *Dimension*. But it is also possible to draw angles at the corners to delimit the board in each signal layer. Please contact your board manufacturer what they prefer.
- If your board has milled edges, please contact your board manufacturer to clarify in which layer these contours have to be drawn. See also page 278.
- You should always leave at least 2 mm (about 80 mil) around the edge of the board free of copper. This is especially important for multilayer boards to avoid internal shorts between these layers.
  In the case of supply layers on multilayer boards, which are plotted inverse, you do this by drawing a wire around the edge of the board. This will act as a copper keep-out in this area.
- Please take care of the wire width for polygons. It should not be set too fine or even 0. These reduced wire widths result in huge file sizes and can lead to problems for board manufacturing, as well.
- As already mentioned in the section of the TEXT command, texts in copper layers ought to be written in vector font. So you can really be sure that the text on your board looks the same as it does in the Layout Editor window.
  To play safe, you could activate the options *Always vector font* and *Persistent in this drawing* in the *Options/User Interface* menu before passing on your board file to the board manufacturer.

- For the sake of completeness we want to point out here again that all questions concerning layer setup, layer thickness, and drill diameter for multilayer boards with Blind, Buried, or Micro vias have to be pre-examined.
- Supply an informational text file to your PCB manufacturer that contains information about specific features in the board. For example, information about used layers, milling contours, and so on. This saves time and avoids trouble.

9.3 Quick Guide for Data Output

The CAM Processor provides an automated job mechanism aiding in the creation of the output data for a board. It is possible to generate all data by a single mouse click.

The Control Panel's tree view (*CAM Jobs* branch) lists all jobs and shows a brief description.

If you are not yet familiar with the use of the CAM Processor, please scroll back to the chapter about *The CAM Processor* on page 77. There you will learn about the basic operation of the CAM Processor.

> *The pre-defined jobs gerb274x.cam and gerber.cam are designed for a two layer board which has components on the top side only. They will generate files for the signal layers, the silk screen for the component side, and the solder stop mask for top and bottom.*

Job gerb274x.cam

This job can be used to generate manufacturing data in *Extended Gerber* format.

Proceed as follows:

- Start the CAM Processor, for example, with the CAM Processor icon in the Layout Editor or with the *File/CAM Processor* menu.
- Load the board into the CAM Processor (*File/Open/Board*) if it was not loaded automatically during the start sequence.
- Load the pre-defined job *gerb274x.cam* with the *File/Open/Job* menu.
- Now click *Process Job*. EAGLE generates five files, one by one, automatically which you have to pass on to the board manufacturer. Each Gerber file contains the aperture table and the respective plot data.

The following files will be generated:

%N.cmp	***Component side***

%N.sol	***Solder side***
%N.plc	***Silk screen component side***
%N.stc	***Solder stop mask component side***
%N.sts	***Solder stop mask solder side***
%N.gpi	***Info file, not needed here***

%N is the placeholder for the board file name without its extension.

If other layers are also to be generated, e.g. silkscreen for the bottom side, or a solder cream mask, the Gerber job can be extended with modifications as required. Extending the job is discussed later in this chapter.

Job excellon.cam

The easiest way to generate drill data is to use the pre-defined CAM job *excellon.cam,* used with the CAM Processor a file that contains drill data and the respective drill table will be generated automatically. This job does not distinguish between the layers *Drills* and *Holes*. Both will be output into a common file. Usually all drillings will be plated-through then.

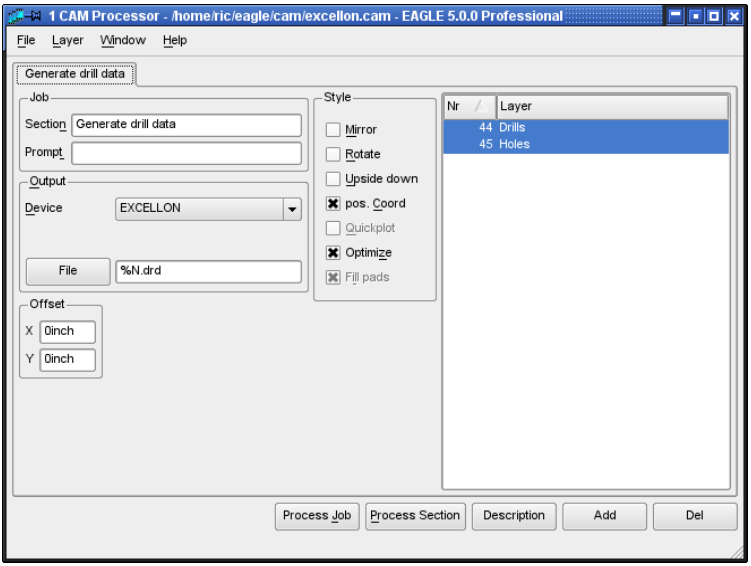
Proceed as follows:

- Start the CAM Processor (for example *File/CAM Processor* menu) in the Layout Editor.
- Load the board into the CAM Processor (*File/Open/Board*), if it has not been loaded automatically during the start sequence.
- Load the pre-defined job *excellon.cam*, e.g. with *File/Open/Job*.
- Execute the job with a mouse click on *Process Job*. Drill data output starts now.

Excellon.cam does not distinguish between plated-through drills and non-plated-through holes!

The following files will be generated:

%N.drd	***Drill data***
%N.dri	***Info file, for board manufacturer, if required***



➢ ***CAM Processor: Generate drill data with the excellon.cam job***

The default unit for the drill table is inch. If the drill table would be preferred in Millimetres the device definition can be changed in the file *eagle.def*. More details concerning this can be found in the section *Units for aperture and drill tables*, beginning with page 289.

Excellon.cam can be used for multilayer boards, as well as for those with Blind, Buried or Micro vias. In this case several drill data files will be generated. See chapter 9.5 for details.

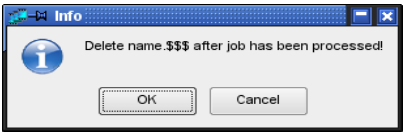
Job gerber.cam

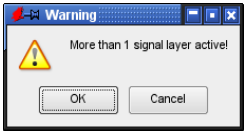
If your board manufacturer can not handle Extended Gerber format and expects Gerber data with a separate aperture file, use the job *gerber.cam*. These files can be created with the devices *GERBERAUTO* and *GERBER* and generates data in RS-274D format.

How to proceed:

- Start the CAM Processor, for example, with the CAM Processor icon in the Layout Editor or with the *File/CAM Processor* menu.
- Load the board into the CAM Processor (*File/Open/Board*) if it was not loaded automatically during the start sequence.
- Load the pre-defined job *gerber.cam* with the *File/Open/Job* menu.
- Now click *Process Job*. EAGLE automatically generates six files which you have to pass on to the board manufacturer. Five Gerber plot files and one common aperture table.

- In the first step an aperture table *%N.whl* is generated. Two messages appear, which you confirm with OK.





➢ ***Messages in gerber.cam***

The left message is generated by the entry in the *Prompt* field, and reminds you to delete the temporary file *%N.$$$* created when generating the aperture table after the job is done.
The message on the right advises you that more than one signal layer are active at the same time. Normally only one signal layer is active while output is generated. However, when generating the wheel, all the layers need to be active at the same time to form a common aperture table for the Gerber output.

The following files will be subsequently output:

%N.whl	***Aperture file (Wheel)***
%N.cmp	***Component side***
%N.sol	***Solder side***
%N.plc	***Silk screen component side***
%N.stc	***Solder stop mask component side***
%N.sts	***Solder stop mask solder side***
%N.$$$	***Temporary file, please delete it***
%N.gpi	***Info file, not needed here***

9.4 Which Files do I Need for my Board?

The previous part of this chapter told you a lot about the basics of data generation and how to use pre-defined job files for default two layer boards. In this section you will find a summary of files usually generated for board manufacturing.

Files List

The output files of the CAM jobs differ in their file extensions. You are, of course, free to use unequivocal names of your own.

The CAM Processor allows the use of some placeholders for the generation of output file names. Usually the output file name consists of the name of the board file plus a special file extension. For the board file name without extension we use the placeholder *%N*. Write, for example, in the *Output File* field: *%N.cmp*. This will be expanded with the name of the layout file that is loaded plus the extension (here: *boardname.cmp*).

In the following table *%N* also stands for the name of the currently loaded board file that is used to generate manufacturing data from.

	File name	***Selected layers***	***Description***
Signal layers			
❑	*%N.**cmp***	***1 Top, 17 Pads, 18 Vias***	***Component side (top)***
❑	*%N.**sol***	***16 Bottom, 17 Pads, 18Vias***	***Solder side (bottom)***
Inner layers			
❑	*%N.**ly2***	***2 Route2, 17 Pads, 18 Vias***	***Inner layer 2***
❑	*%N.**ly3***	***3 Route3, 17 Pads, 18 Vias***	***Inner layer 3***
	.....	***.....***	***.....***
❑	*%N.**l15***	***15 Route15, 17 Pads, 18 Vias***	***Inner layer 15***
Special case: Inner layer as Supply layer (e.g.: Layer2 $GND, Layer15 $VCC)			
❑	*%N.**ly2***	***2 $GND***	***Supply layer $GND***
	.....	***.....***	***....***
❑	*%N.**l15***	***15 $VCC***	***Supply layer $VCC***
Silk screen			

❑	*%N.plc*	*21 tPlace, 25 tNames, possibly 20 Dimension(*)*	*Silk screen component side*
❑	*%N.pls*	*22 bPlace, 26 bNames possibly 20 Dimension(*)*	*Silk screen solder side*
Solder stop mask			
❑	*%N.stc*	*29 tStop*	*Solder stop component side*
❑	*%N.sts*	*30 bStop*	*Solder stop solder side*
Cream frame (for SMT devices)			
❑	%N.crc	*31 tCream*	*Cream frame component side*
❑	%N.crs	*32 bCream*	*Cream frame solder side*
Milling contours for openings, oblong holes...			
❑	%N.mill	*46 Milling (**)*	*Plated milling contours*
❑	%N.dim	*20 Dimension (**)*	*Non-plated milling cont.*
Finishing mask (e.g. gold coating)			
❑	*%N.fic*	*33 tFinish*	*Finishing component side*
❑	*%N.fis*	*34 bFinish*	*Finishing solder side*
Glue mask (for larger SMT devices)			
❑	*%N.glc*	*35 tGlue*	*Glue mask component side*
❑	*%N.gls*	*36 bGlue*	*Glue mask solder side*
Drill data			
❑	*%N.drd*	*44 Drills, 45 Holes*	*All drillings*

Distinguishing plated from non-plated drillings			
❑	*%N.**drd***	***44 Drills***	***Plated drillings***
❑	*%N.**hol***	***45 Holes***	***Non-plated drillings***

(*) Please check with your board manufacturer whether you have to output the board contour in layer 20 in a separate file or you are allowed to combine it with those layers.
(**) If there are additional milled edges in the board, you should contact your board manufacturer and ask them which layers they prefer for milling contours.

Please note that there are differences between normal inner layers and supply layers:
Normal inner layers (possibly with polygons) are treated the same as Top or Bottom layer; you have to output them with Pads and Vias together. Supply layers, however, are not combined with any further layer!

Placeholders for Output File Name Generation

%D{xxx}	***xxx stands for a string that is inserted only into the data file name***
%E	***file extension of the loaded file, without the '.'***
%H	***home directory of the user***
%I{xxx}	***xxx stands for a string that is inserted only into the Info file name***
%L	***layer range for blind&buried vias***
%N	***name of the loaded file without path and extension***
%P	***directory path of the loaded Board or Schematic file***
%%	***the character '%'***

These placeholders must be written in upper case letters!

Hints Concerning File Extensions:

cmp stands for component side, the upper side of the board, *sol* for the solder (bottom) side. It makes sense to choose the first two letters according the active layers. The third one can be *c* or *s* for belonging to component or solder side.

Of course you are free in naming your files in any manner you wish!

> *Please ensure when defining a job that the extensions of the output files are unique and therefore distinguishable.*

9.5 Peculiarities of Multilayer Boards

In case of boards with inner layers one has to know how these layers are defined in order to generate proper manufacturing data. Is it an inner layer that contains tracks and polygons, as it is in Top or Bottom layer? Or is it a supply layer that can be identified by the $ character in front of the layer name?

Simple Inner Layers

The normal inner layers are treated the same as the outer signal layers. Together with the signal layer, the layers *Pads* and *Vias* are active.

> *If the Layer Setup allows Blind and Buried vias, the combination of one signal layer and the Vias layer outputs only those vias that belong to this signal layer.*
>
> *If there is only the Vias layer active (no signal layer), the CAM Processor will output all vias of the board!*

Supply Layer

When generating manufacturing data for supply layers, for example $GND, you must not activate the layers *Pads* and *Vias*.
Isolation rings, called Annulus symbols, and thermal reliefs (Thermals) are generated automatically in a Supply layer. They have nothing in common with pads or vias of normal signal layers. A Supply layer is displayed and output inverted.

> *Supply layers are **never** output with Pads and Vias!*

Drill Data for Multilayer Boards With Blind and Buried Vias

The CAM Processor generates one drill data file for each via length for a layout that uses Blind and Buried vias.

The drill data file extension *.drd* is expanded by the via length specification. If there are, for example, vias from layer 1 to 2, the output file extension will be *.drd.0102*.

The layer specification can be moved to another position with the help of the wildcard *%L*. Writing, for example, in the *File* box of the CAM Processor *%N. %L.drd* results in an output file named *boardname.0102.drd*.

Pads and trough-hole vias will be written into an output file with extension *.drd.0116*. If you placed holes (HOLE command) in the layout and the *Holes* layer is active for output, the CAM Processor writes this data also into the file with extension *.drd.0116*.

Pass on all these files to your board manufacturer.

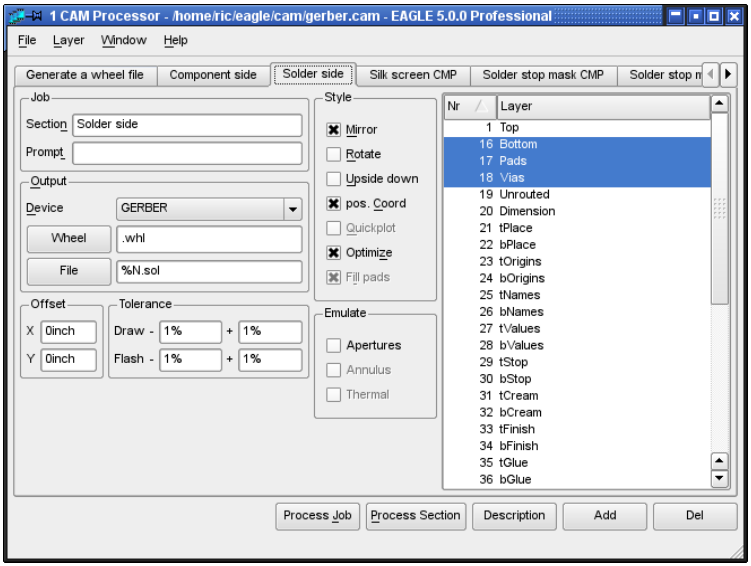
Provided you did not use the EXCELLON device which combines drill table and drill coordinates in a common file, your board house additionally needs the rack file *name.drl* which is generated by *drillcfg.ulp*.

9.6 Set Output Parameters

This section describes the setting of the parameters in the CAM Processor for the output of a board or a schematic.

Load the schematic or board file from the CAM Processor's *File/Open* menu, and set the parameters.

The CAM Processor window is divided into sections (*Output*, *Job*, *Style*, *Layer*, *Offset* and so on). Some sections, like *Emulate*, *Tolerance*, *Pen* or *Page*, are used by and therefore displayed with certain devices only.



➢ ***CAM Processor: Solder side section of the gerber.cam job***

Output:

- Select the driver for the desired output device or output format in the *Device* combo box.
- Enter the output path and file name in the *File* field or simply click onto the *File* button and use its dialog.
  If you want to output a file on a particular drive, place the drive identifier or, if appropriate, the path in front of the file name ending. For instance, under Windows, *d:\%N.cmp* would place the file *boardname.cmp* in the root directory of drive D. This also applies to the Linux version, so that, for example, */dev/hdc2/%N.cmp*, would place the file on drive *hdc2*.
  %H can be used as wildcard for the Home directory, *%P* for the loaded file's directory path.
  If output is to go directly to a plotter, enter the name of the print queue that is connected to the corresponding computer interface in UNC notation, e.g. *\\Servername\Plottername*.
- Depending on the chosen device it may be asked for *Wheel* (aperture table) or *Rack* (drill table). Select path and file by clicking the button.

Layer selection:

- Select the layers that should be output into a common file by clicking the appropriate layer number.
  Click the menu *Layer/Deselect all* to uncheck all the layers first. *Layer/show selected* displays only the currently selected layers.

- Some devices (like HPGL or certain plotter devices) allow you to select a color or pen number in an additional column.

Style:

- *Mirror*: Mirrors the output.
  It can be useful to mirror all outputs that refer to the bottom side of the board.
- *Rotate*: Rotates the output by 90 degrees.
- *Upside down*: Rotates the output by 180 degrees.
  When combined with *Rotate*, the drawing is rotated by a total of 270 degrees.
- *Pos. Coords.*: Avoids negative coordinate values for the output.
  The drawing will be moved near the coordinate's axis, even if it is already in the positive coordinates range. Negative values can lead to errors with a lot of devices! This option should be set on always by default. Switching it off, transfers the coordinate values from the Layout Editor unchanged.
- *Quickplot*: Draft output which shows only the outlines of objects. This option is available for certain devices, like HPGL and various plotters.
- *Optimize*: Activates the optimization of the drawing sequence for plotters. Should be set on by default.
- *Fill Pads*: This option is always set on. Only the devices *PS* and *EPS* allow you to switch off this option. The drill holes for pads will be visible on the output (as it is with the PRINT command).

Job:

- If you are on the way to define a CAM job that consists of several sections it is useful to name them. In the *Section* line, it is possible to enter a section name which will be shown also in the tab above.
  If, for example, you assign the section name *Wheel: Generate Aperture File*, only the title *Wheel* is visible as a tab name. The additional description can be read in the section line. The colon ends the title in the tab.
- In case you wish to display a message box on the screen before executing this section, enter special message text in the *Prompt* field. For example: *Please insert a new sheet of paper!* The output does not continue until confirmation of this message.

Offset:

- Define an offset in x and y direction.
  The values can be given in Inches or Millimetres, for example, `15mm` or `0.5inch`.

Tolerance:

- Tolerances for *Draw* and *Flash* apertures are necessary for devices that use an aperture file, like *GERBER* or *GERBER_23*. Usually one allows a tolerance of 1% in all fields.
  This is necessary to compensate small rounding errors that can arise during the conversion from mm to inch values and vice versa as the aperture table is generated.
- Devices for drill data generation that use a separate drill table (rack file) invoke a *Drill* entry. A tolerance of ± 2.5% is enough to compensate for rounding errors that possibly can arise during the conversion from mm to inch.

Emulate:

- If an aperture with the exact value is not available in the aperture file, you can allow *Aperture* emulation. The CAM Processor is allowed to use smaller drawing apertures for emulation. Plotting time and costs will increase, and therefore you should try to avoid aperture emulation.
- *Thermal* and *Annulus* apertures drawn in Supply layers will be emulated with *Draw* apertures, if the options *Thermals* and *Annulus* are checked.
- Arcs with flat endings (CHANGE CAP FLAT) are <u>always</u> emulated for Gerber output, which means that they are drawn with small lines. Arcs with round endings (CHANGE CAP ROUND), however, not.

> *If your layout contains objects that are rotated in any angle, you have to activate aperture emulation. Certain pad shapes must be drawn with a smaller round aperture.*

Page:

- Define the *Height* and *Width* of the sheet you want to plot on. Values are in Inches by default. Values can also be given in Millimetres, like `297mm`.

Pen:

- The *Diameter* of the plotting pen is given in here. The value has to be in Millimetres.
- For plotters that support adjustable pen *Velocity* you can define a value given in cm/s (centimetres per second). No value here results in a default value given by the plotter.

Sheet:

- Select the sheet of the schematic you want to output.

9.7 Automating the Output with CAM Processor Jobs

Defining a New CAM Job

A Job consists of one or more sections that allow you to generate a full set of manufacturing data with only a few mouse clicks. A section is a group of settings, as described above in the *Set Output Parameters* chapter, which defines the output of one file.

Define a job as follows:

- Start the CAM Processor.
  No job is loaded at first, unless there is a file called *eagle.cam* in the cam directory or an existing job is called up automatically by an EAGLE project file.
- If has not already been done, load the board file which you want to define the job for (also possible for schematics).
- It is best that you load an existing job, for example *gerb274x.cam*, through the CAM Processor's *File/Open/Job* menu.
- Save this job under a new name with *File/Save job...* So the original job file remains unchanged.
- Click the *Add* button.
  The currently active section with all its parameter settings is copied now.
- Enter a new descriptive text in the *Section* line.
- If you wish to get a message shown before executing this section, enter a specific text prompt in the *Prompt* line.
- Set all parameters now:
  Device, the layers for the output, the output *File,* a *Scale* factor, if needed, the *Style* options (*Mirror*, *Rotate*, *Upside down*...).
- Define further sections in the same way using different names.
  Very important: First use *Add* to create a new section, then set the parameters.
- Delete a section, if needed, by clicking the *Del* button.
- Save all the sections of your new job as a job file under a name of your choice with *File/Save job...*

All the sections of the job will be executed automatically one after another if you click the *Process job* button. One specific (the currently shown) section will be executed if you click the *Process section* button.

The *Description* button allows a descriptive text of the CAM job that will be displayed in the Control Panel.

Extending gerber.cam Job for Multilayer Boards

The *gerber.cam* job can be used as the basis of the job for multilayer boards. It must simply be extended for the additional inner layers.

Example:
You want to output the files for a board with SMD components on the top and bottom sides, a supply layer $*GND* in Layer 2, and another inner layer with a polygon *VCC* in Layer 15 (which is renamed to VCC).
You need silkscreen prints for the upper and lower sides, solder stop masks, and masks for the solder cream for both sides.

Before you start to change the CAM job you should save the job under a new name through the *File/Save job as..* menu.

Proceed as described in the previous section. The CAM job then contains the following sections:

New	Output file	Selected layers	Description
☑*	*%N.$$$*	*1, 2, 15, 16, 17, 18, 20, 21, 22, 25, 26, 29, 30, 31, 32*	*Generate wheel file .whl* ** modified layer selection*
	%N.cmp	*1 Top, 17 Pads, 18 Vias*	*Component side*
☑	*%N.ly2*	*2 $GND*	*Supply layer*
☑	*%N.l15*	*15 VCC, 17 Pads, 18 Vias*	*Inner layer 15*
	%N.sol	*16 Bottom, 17 Pads, 18Vias*	*Solder side*
	%N.plc	*21 tPlace, 25 tNames, 20 Dimension*	*Silkscreen component side*
☑	*%N.pls*	*22 bPlace, 26 bNames 20 Dimension*	*Silkscreen solder side*
	%N.stc	*29 tStop*	*Solderstop mask comp. side*
	%N.sts	*30 bStop*	*Solderstop mask solder side*
☑	*%N.crc*	*31 tCream*	*Creamframe comp. side*
☑	*%N.crs*	*32 bCream*	*Creamframe solder side*

Check once more whether all the necessary layers for the creation of the aperture table are active in the first section. The output file generated in the first section cannot be used. For this reason, the file *boardname.$$$* should be deleted.

> *Activate only the signal layer for the output of Supply layers (here $GND)! No Pads and no Vias layer!*
>
> *In case of other inner layers that may contain polygons, you have to activate the layers Pads and Vias!*

Error Message: Apertures Missing

If the error message *APERTURES MISSING – NO PLOTFILE HAS BEEN PRODUCED* appears after starting the modified CAM Job, the Gerber info file *boardname.gpi* contains information about the reason of the problem.
The error could possibly be not all apertures were defined or some could not be found in the table due to missing tolerance settings.
Ensure all used layers were activated in the *Generate a Wheel file* section?

It could also be the case that there are parts placed in any angle with pads that don't have a round shape in the layout. You have to activate the aperture emulation then to have a successful file creation.

Gerber Info Files

The CAM Processor generates for each Gerber plot file created an additional Gerber info file with the extension *.gpi*. This text file informs you about used apertures, the data format of the Gerber device, about possible aperture emulations or tolerances, and so on.

If you define a job for Gerber output with data files that use the extensions proposed in the table above, the info file will be overwritten with each succeeding section.

If your board manufacturer or your plot service wants to have these additional information files (they are typically not necessary), use the *%I{xxx}* placeholder as follows:

For *Output File* enter

%N.xxx%I{.info}

Here *%N* stands for the board name, *xxx* stands for any characters for the file extension, *%I{.info}* generates an additional extension *.info* for the Gerber info files.

The output files generated will look like this:

Gerber file:	*boardname.xxx*
Info file:	*boardname.xxx.info*

Example:
The board ***myboard.brd*** is loaded. The *File* field contains ***%N.cmp%I{.info}***. The output file is called ***myboard.cmp***, and the info file is named ***myboard.cmp.info***.

The files will be written into the same directory as the board file is.

Drill Data Generation with Separate Rack File

If you prefer drill data in another data format, for example Sieb&Meyer 1000 or 3000 (*SM1000*, *SM3000*) or the Excellon format with a separate rack file (*EXCELLON_RACK*), a drill table is required first.

Define a Drill Configuration (Rack) File

This file is usually created by *drillcfg.ulp* (RUN command) started directly in the Layout Editor window and will be named *boardname.drl*.

It is of course also possible to define a drill table with the aid of a text editor. For data generation enter the table's path and file name in the CAM Processor's *Rack* field.

Example of a drill configuration file:

```
T01 0.010
T02 0.016
T03 0.032
T04 0.040
T05 0.050
T06 0.070
```

All dimensions are given here in inches. It is also possible to enter the values with their unit, e.g. 0.010in or 0.8mm. Comments in drill configuration files may be used and are identified by a semicolon, which may stand at the start of a line or be preceded by a space.

Define Job for Drill Data Output

- Start the CAM Processor
- Load the predefined job *excellon.cam*, e.g. with *File/Open/Job*.
- Change the output *Device* to *EXCELLON_RACK*, *SM1000* or *SM3000* and check the parameters. Layers 44, *Drills*, and 45, *Holes*, must be selected **only**. No other layers! Use the *Layers/show selected* menu to get a summary of the active layers.

  ☞ If you want to have separate files for plated and non-plated drillings, select here Layer 44, *Drills*, only and insert a further section that outputs layer 45, *Holes*, in a separate file with the *Add* button after the following item.
  Check with your board manufacturer to see if this is the method they prefer.
- After changing the output device the CAM Processor requires a *Rack* file as a tool guide. Type in *.drl* or click the *Rack* button and choose the path to your rack file in the file dialog.

☞ In the event you wish to distinguish plated and non-plated holes: *Add* a new section which outputs only layer 45, Holes, into a second drill file with modified file name. Possibly *.hol* for holes.

- Save the job file via *File/Save job* with a new name.

> *A tolerance of ± 2.5% should be allowed for drill diameter selection in order to compensate rounding errors that possibly can arise during the conversion from mm to inch and vice versa in the drill table.*

Drill Info File

The file *name.dri* is generated with each drill data output. It contains the list of used tools and further information about the data format.

If the error message *DRILLS MISSING – NO PLOTFILE HAS BEEN PRODUCED* appears, the info file contains information which drill diameter could not be found in the drill table. Add the missing diameter in the drill table with a text editor or check the given values for tolerances.

The info file is written into the same directory as the output file. You may send this file to your PCB manufacturer, if requested.

9.8 Device Driver Definition in eagle.def

Output device drivers are defined in the *eagle.def* text file. Here you will find all the information that is needed for the creation of your own device driver. The best way is to copy the block for an output device of the same general category, and then alter the parameters where necessary.

The file eagle.def can be found in the eagle/bin directory.

Creating Your Own Device Driver

Please use a text editor that does not introduce any control codes into the file.

Example 1: Gerber(auto) device, Millimetre

```
[GERBER_MM33]

Type = PhotoPlotter
Long = "Gerber photoplotter"
Init = "G01*\nX000000Y000000D02*\n"
Reset = "X000000Y000000D02*\nM02*\n"
ResX = 25400
ResY = 25400
Wheel = ""
Move = "X%06dY%06dD02*\n" ; (x, y)
Draw = "X%06dY%06dD01*\n" ; (x, y)
Flash = "X%06dY%06dD03*\n" ; (x, y)
Units = mm
Decimals = 4
Aperture = "%s*\n" ; (Aperture code)
```

```
Info = "Plotfile Info: \n" \
"\n" \
"Coordinate Format : 3.3 \n" \
"Coordinate Units : 1/1000mm \n" \
"Data Mode : Absolute \n" \
"Zero Suppression : None \n" \
"End Of Block : * \n" \
"\n"


[GERBERAUTO_MM33]

@GERBER_MM33
Long = "With automatic wheel file generation"
Wheel = "" ; avoids message!
AutoAperture = "D%d"; (Aperture number)
FirstAperture = 10
MaxApertureSize = 2.0
```

Example 2: EXCELLON Device, Output with Leading Zeros

```
[EXCELLON-LZ]

Type     = DrillStation
Long     = "Excellon drill station"
Init     = "%%\nM48\nM72\n"
Reset    = "M30\n"
ResX     = 10000
ResY     = 10000
;Rack     = ""
DrillSize  = "%sC%0.4f\n" ; (Tool code, tool size)
AutoDrill  = "T%02d"      ; (Tool number)
FirstDrill = 1
BeginData  = "%%\n"
Units    = Inch
Decimals = 0
Select   = "%s\n"         ; (Drill code)
Drill    = "X%06.0fY%06.0f\n"      ; (x, y)
Info     = "Drill File Info:\n"\
           "\n"\
           " Data Mode         : Absolute\n"\
           " Units             : 1/10000 Inch\n"\
           "\n"
```

Units in the Aperture and Drill Table

When automatically generated with the *GERBERAUTO* driver, the aperture table contains values in inches.
This is also the case for the drill table which is automatically written into the drill data file with the output device *EXCELLON*.
If your PCB manufacturer insists on mm units for aperture sizes and drill diameters, you can achieve this by altering the *GERBER* or *GERBERAUTO* respectively for the *EXCELLON* driver.

Use a text editor that does not introduce any control codes to edit the *eagle.def* file, look for the line

```
[GERBER]
```

or

```
[GERBERAUTO]
```

and add/edit in this section the lines

```
Units = mm
Decimals = 4
```

In order to change the drill table units look for the line

```
[EXCELLON]
```

and change:

```
Units = Inch
```

to

```
Units = mm
```

9.9 Gerber Files for Photoplotters with Fixed Aperture Wheels

This section goes into more detail on the definition of the aperture table. Some board manufacturers may perhaps still be using a Gerber plotter that works with a fixed aperture wheel. In such a case it is necessary to adapt the aperture table to the restricted facilities of the Gerber plotter. Files for Gerber photoplotters with fixed aperture wheels are generated with the GERBER driver. It is essential to confer with your photoplot service ahead of time, so as to adjust EAGLE to the available apertures. The aperture table has to be defined manually.

There are various types of apertures. They differ in size and shape. The most common are circle, octagon, square, Thermal and Annulus symbols. The drawing aperture (*Draw*) used for tracks is normally a round aperture.

You must specify the aperture configuration before you can generate files for a fixed aperture wheel photoplotter. To do this, enter the configuration file for apertures *name.whl* e.g. with the EAGLE Text Editor, and then load this file into the CAM Processor by clicking the *Wheel* button after selecting the GERBER device driver (see *Set Output Parameters* beginning with page 280).

Defining the Aperture Table (Wheel)

The CAM Processor distinguishes *Draw* apertures from *Flash* apertures. The first type is used to draw objects (e.g. tracks). The second type is used to generate symbols (e.g. pads) by a light flash. Only if draw apertures are defined can the plotter draw lines. Therefore, if the plotter doesn't distinguish between draw and flash apertures, you must additionally define round or octagonal apertures as draw apertures.

The following apertures are available:

```
Name            Dimension

Draw            diameter
Round           diameter
Square  length
Octagon         diameter
Rectangle       length-X x width-Y
Oval            diameter-X x diameter-Y
Annulus         outside diameter x inside diam.
Thermal         outside diameter x inside diam.
```

Use of aperture shapes in the CAM Processor:

```
Draw            draws wires and emulates apertures
Round           draws round pads and vias
Square  draws square pads, SMDs and vias
Octagon         draws octagonal pads and vias with
                the same X- and Y-dimensions
Rectangle       draws rectangles and SMDs
Oval            draws pads with different
                X- and Y-dimensions
Annulus         draws isolation rings in a supply
                layer
Thermal         draws connections in a supply
                layer
```

Aperture configuration file example:

```
D001    Annulus         0.004 x 0.000
D002    Annulus         0.005 x 0.000
D017    Annulus         0.063 x 0.000
D020    round           0.004
D033    round           0.059
D040    square  0.004
D052    square  0.059
D054    thermal         0.090 x 0.060
D057    thermal         0.120 x 0.080
D105    oval            0.090 x 0.030
D100    rectangle       0.060 x 0.075
D101    rectangle       0.075 x 0.060
D110    draw            0.004
D111    draw            0.005
```

The D code determines the tool number, then follows the aperture shape after at least one blank character, then the dimensions are defined.

All values default to inches, unless a unit is added, for example `0.010in` or `0.8mm`.

Comments are marked with semicolons at the beginning of a line, or with a semicolon following a blank character.

Aperture Emulation

If objects exist in a drawing which is not compatible with the available aperture sizes, it is possible to emulate the desired dimensions by selecting the *Emulate Apertures* option. The CAM Processor then selects smaller apertures to emulate dimensions which are not matched by aperture sizes. Emulation results in longer plot times and higher film costs, so it should be avoided whenever possible.
Thermal or Annulus apertures in Supply layers are only emulated with *Draw* apertures if the *Emulate Thermal* or *Emulate Annulus* options, respectively, is activated.

The file *name.gpi* indicates which apertures are emulated.

Chapter 10
Appendix

10.1 Layers and their Usage

In Layout and Package Editor

```
 1 Top  Tracks, top side
 2 Route2      Inner layer (signal or supply)
 3 Route3      Inner layer (signal or supply)
 4 Route4      Inner layer (signal or supply)
 5 Route5      Inner layer (signal or supply)
 6 Route6      Inner layer (signal or supply)
 7 Route7      Inner layer (signal or supply)
 8 Route8      Inner layer (signal or supply)
 9 Route9      Inner layer (signal or supply)
10 Route10     Inner layer (signal or supply)
11 Route11     Inner layer (signal or supply)
12 Route12     Inner layer (signal or supply)
13 Route13     Inner layer (signal or supply)
14 Route14     Inner layer (signal or supply)
15 Route15     Inner layer (signal or supply)
16 Bottom      Tracks, bottom side
17 Pads        Pads (through-hole)
18 Vias        Vias (through all layers)
19 Unrouted    Airlines (rubber bands)
20 Dimension   Board outlines (circles for holes) *)
21 tPlace      Silk screen, top side
22 bPlace      Silk screen, bottom side
23 tOrigins    Origins, top side (generated autom.)
24 bOrigins    Origins, bottom side (generated autom.)
25 tNames      Service print, top side (component NAME)
26 bNames      Service print, bottom s. (component NAME)
27 tValues     Component VALUE, top side
28 bValues     Component VALUE, bottom side
29 tStop       Solder stop mask, top side (gen. autom.)
30 bStop       Solder stop mask, bottom side (gen. Autom.)
31 tCream      Solder cream, top side
32 bCream      Solder cream, bottom side
33 tFinish     Finish, top side
34 bFinish     Finish, bottom side
35 tGlue       Glue mask, top side
36 bGlue       Glue mask, bottom side
37 tTest       Test and adjustment information, top side
38 bTest       Test and adjustment inf., bottom side
39 tKeepout    Restricted areas for components, top side
```

```
40 bKeepout    Restricted areas for components, bottom s.
41 tRestrict   Restricted areas for copper, top side
42 bRestrict   Restricted areas for copper, bottom side
43 vRestrict   Restricted areas for vias
44 Drills      Conducting through-holes
45 Holes       Non-conducting holes
46 Milling     Milling
47 Measures    Measures
48 Document    Documentation
49 Reference   Reference marks
51 tDocu       Detailed top screen print
52 bDocu       Detailed bottom screen print
```

In Schematic, Symbol, and Device Editor

```
91 Nets        Nets
92 Busses      Busses
93 Pins        Connection points for component symbols
               with additional information
94 Symbols     Shapes of component s
95 Names       Names of component symbols
96 Values      Values/component types
97 Info        Additional information/hints
98 Guide       Guiding lines for symbol alignment

*) Holes generate circles with their diameter in this
   layer. They are used to place restrictions on the
   Autorouter.
```

Layers can be used with their names or their numbers. Names can be changed with the LAYER command or in the DISPLAY menu. The functions of the special layers remain.

If you want to create your own layers, please use layer numbers above 100. Use the DISPLAY menu to create new layers (*New* button) or type the LAYER command on the command line. If you want to create, for example, layer 200, *Remarks*, type in:

```
LAYER 200 Remarks
```

To set up color and fill style of this layer use the DISPLAY command.

10.2 EAGLE Files

EAGLE uses the following file types:

Name	**Type of file**
*.brd	Layout
*.sch	Schematic
*.lbr	Library
*.ulp	User Language Program
*.scr	Script file
*.txt	Text file (also other suffixes)
*.dru	Design Rules
*.ctl	Control parameter for the Autorouter
*.pro	Autorouter protocol file

```
*.job          Autorouter job
*.b$$          Backup file of brd after finishing
               the Autorouter
*.cam          CAM Processor job

*.b#x          Backup file of BRD (x = 1..9)
*.s#x          Backup file of SCH (x = 1..9)
*.l#x          Backup file of LBR (x = 1..9)
*.b##          Automatic backup file of BRD
*.s##          Automatic backup file of SCH
*.l##          Automatic backup file of LBR
```

EAGLE for Linux only creates and recognizes lower case characters in file endings!

10.3 EAGLE Options at a Glance

In order to output manufacturing data, for instance, with the CAM Processor, EAGLE can be started directly from a terminal window under Linux and Mac, or from a console window under MS Windows.

Since Windows programs give up their connection to the console they have been started from, you can use the file *eaglecon.exe* (located in the *eagle\bin* subdirectory of your installation) if you want to run the CAM Processor from a batch file.
This version of EAGLE is exactly the same as the *eagle.exe*, except that it doesn't disconnect from the console.
Type *eaglecon -?* for a list of CAM Processor options.

The following options are permitted:

```
-C     execute a given EAGLE Command
-Dxxx  Draw tolerance (0.1 = 10 %)
-Exxx  Drill tolerance (0.1 = 10 %)
-Fxxx  Flash tolerance (0.1 = 10 %)
-N+    Suppress message prompts
-O+    Optimize pen movement
-Pxxx  Plotter pen (layer=pen)
-Rxxx  Drill rack file
-Sxxx  Script file
-Wxxx  Aperture wheel file
-X-    Execute CAM Processor
-a-    Emulate annulus
-c+    positive coordinates
-dxxx  Device (-d? for list)
-e-    Emulate apertures
-f+    Fill pads
-hxxx  Page height (inch)
-m-    Mirror output
-oxxx  Output filename/channel
-pxxx  Pen diameter (mm)
-q-    Quick plot
-r-    Rotate output 90 degrees
-sxxx  Scale factor
```

```
-t-     emulate Thermal
-vxxx   Pen velocity
-u-     rotate output 180 degrees
-wxxx   Page width (inch)
-xxxx   Offset X (inch)
-yxxx   Offset Y (inch)
```

Where:

```
xxx     stands for further data, e.g. file name as with
        -W or a decimal number as with -s.
      Examples:
        -W /home/user/eagle/project/aperture.whl
        -s 1.25
- Default for option is off
+ Default for option is on
      Example:
        -e Aperture Emulation on
        -e+ ditto
        -e- Aperture Emulation off

Flag options (e.g. -e) can be used without repeating
the '-' character:

        -eatm  Aperture emulation on, annulus and thermal
               emulation on, mirror output
        -ea-t+ Aperture emulation on, NO annulus
               emulation, thermal emulation on

Defining tolerance values:
If there is no sign, the value applies to either direction,

+       signifies a positive tolerance,
-       a negative tolerance.

-D0.10                adjusts the draw tolerance to 10 %
-D+0.1 -D-0.05        adjusts the draw tolerance to
                      +10 % and -5 %
```

Notes on the individual options:

-C **Execute a command**
After loading an EAGLE file the given command will be executed in the Editor window's command line.
See also help function, *Command Line Options*.

-D **Draw Tolerance (0.1 = 10 %):**
Default: 0

-E **Drill Tolerance (0.1 = 10 %):**
Default: 0

-F **Flash Tolerance (0.1 = 10 %):**
Default: 0

-N **Suppress messages:**
This option suppresses warnings or other information in the console window (DOS box, Linux console). Thus CAM jobs run without interruption. Default: off

-O **Route-Optimizing**:
With this option the route-optimizing for the plotter can be turned on and off. Default: on

-P **Plotter Pen (layer=pen):**
If you use a color pen plotter, you can determine which layer is to be drawn in which color. Example: -P1=0 -P15=1

-R **Drill Rack File**:
With this option you define the path to a file with the drill configuration table.

-S **Script File:**
When opening the editor window, EAGLE executes the *eagle.scr* file. This option allows a different name or directory to be selected for the script file. The script file is not read by the CAM Processor.

-W **Aperture Wheel File**:
This option defines the path to the wheel file which should be used.

-X **Calls command line version of the CAM Processor**

-a **Annulus Symbol Emulation:**
Default: off

-c **Positive Coordinates:**
If this option is set the CAM Processor creates data without negative coordinates. The drawing is moved to the zero-coordinates. This option can be turned off with the option -c-.
Please be careful with this option, especially if you use mirrored and rotated drawings, because negative coordinates normally cause problems. Default: on

-d **Device:**
This option determines the output driver.
`eagle -d?` displays a list of the available drivers

-e **Emulate Apertures:**
If this option is selected, apertures that do not exist are emulated with smaller apertures. If this option is turned off, no apertures are emulated. Thermal or Annulus apertures are not emulated either. Default: off

-f **Fill Pads:**
This option can only work with generic devices like Postscript. Default: on for all devices

-h **Page Height (inch):**
Printable region in the y-direction (in inches). The Y direction is the direction in which the paper is transported. See also the -w parameter.

-m **Mirror Output:**
Default: off.

-o **Output File Name**

-p **Pen Diameter [mm]:**
EAGLE uses the Pen-diameter measurement to calculate the number of lines required when areas are to be filled. Default: 0

-q **Quick Plot:**
Generates a draft or fast output, which only prints the frames of the objects. Default: off

-r **Rotate Output:**
Rotates the output by 90 degrees. Default: off

-s **Scale Factor:**
Those devices which cannot change their scale-factor (in the menu of the CAM Processor), have a scale factor of 1. Default: 1

-t **Emulate Thermals:**
Works only in combination with -e+. Default: off

-u **Rotate Output by 180 degrees:**
In combination with -r+ one can rotate by 270 degrees. Default: off

-v **Pen Velocity in cm/s:**
This option is for pen plotters supporting different speeds. To select a plotter's default speed, use a value of 0. Default: 0

-w **Page Width (inch):**
Printable area in x direction. See also -h.

-x **Offset X (Inch):**
This option can be used to move the origin of the drawing. Default: 0

-y **Offset Y (Inch):**
Default: 0

Example for starting eaglecon.exe:

```
eaglecon -X -dgerber -Waperture.whl -oboard.sol -D-0.1
name.brd pad via bottom
```

This generates a Gerber file board.sol which contains data of the layers Pads, Vias, and Bottom. Draw apertures may have a negative tolerance up to 10 %.

All parameters have to be written in a common line!

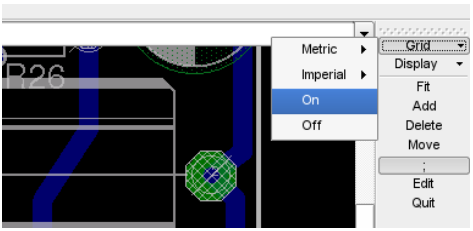
Paths that include space characters, must be set into single quotes!

10.4 Configuration of the Text Menu

With the help of a script file (e.g. *menu.scr*) you can configure your own text menu.

```
  # Command Menu Setup
  #
  # This is an example that shows how to set up a complex
  # command menu, including submenus and command aliases.

MENU 'Grid { \
               Metric { \
                       Fine : Grid mm 0.1; | \
                       Coarse : Grid mm 1; \
                       } |  \
               Imperial { \
                       Fine : Grid inch 0.001; | \
                       Coarse : Grid inch 0.1; \
                       } |  \
               On : Grid On; |  \
               Off : Grid Off; \
        }'  \

        'Display { \
               Top : Display None Top Pads Vias Dim; | \
               Bottom : Display None Bot Pads Vias Dim; | \
               Placeplan { \
                       Top : Display None tPlace Dim; | \
                       Bottom : Display None bPlace Dim; \
                       } \
               }' \
'---' \
'Fit : Window Fit;' \
Add Delete Move ';' Edit Quit \
;
```

The backslash \ at the end of a line shows that a command continues in the next line. Here the MENU command runs from the first line after the comment to the last line.
The pipe sign | has to be used if a command within braces { } is followed by another command.



➢ ***Text menu from menu.scr in the Layout Editor***

10.5 Text Variables

Text variable	Meaning
>NAME	Component name (eventually + gate name) 1)
>VALUE	Component value/type 1)
>PART	Component name 2)
>GATE	Gate name 2)
>SHEET	Sheet number of a circuit diagram in the form of, for example 1/3 3)
>SHEETS	Total nu>SHEETNRmber of sheets 3)
>SHEETNR	Current sheet number 3)
>DRAWING_NAME	Drawing name
>LAST_DATE_TIME	Time of the last modification
>PLOT_DATE_TIME	Time of the plot creation

1) Only for package and symbol
2) Only for symbol
3) Only for symbol or circuit diagram

All texts starting with the character >, will be interpreted as placeholder texts for attributes. See ATTRIBUTE command.

10.6 Options for Experts in eaglerc

The user-specific file eaglerc.usr for Windows and .eaglerc for Linux and Mac stores various settings defined during the work with EAGLE. Among them you find some expert settings that can be adjusted in this file directly. The most important of them are listed here.

Since version 5.2 it is possible to change these parameters with the help of the SET command in the commandline. Please see the help function about the SET command for details.

Change Component Value Warning

Some users don't want the warning message about a part not having a user definable value, so this warning can be disabled by appending the line

```
Warning.PartHasNoUserDefinableValue = "0"
```

to the file.

Consistency Check

In order to handle Board/Schematic pairs that have only minor inconsistencies, the user can enable a dialog that allows him to force the editor to perform Forward&Back Annotation, even if the ERC detects that the files are inconsistent. This can be done by appending the line:

```
Erc.AllowUserOverrideConsistencyCheck = "1"
```

PLEASE NOTE THAT YOU ARE DOING THIS AT YOUR OWN RISK!!!
If the files get corrupted in the process, there may be nothing anybody can do to recover them. After all, the ERC did state that the files were inconsistent!

Delete Wire Joints

If you absolutely insist on having the DELETE command delete wire joints without pressing the Ctrl key, you can append the line

```
Cmd.Delete.WireJointsWithoutCtrl = "1"
```

to the file.

Device Name as Value for all Components

Some users always want to use the device name as part value, even if the part needs a user supplied value. Those who want this can append the line

```
Sch.Cmd.Add.AlwaysUseDeviceNameAsValue = "1"
```

to the file.

Disable Ctrl for Radius Mode

If you don't like the special mode in wire drawing commands that allows for the definition of an arc radius by pressing the Ctrl key when placing the wire, you can add the line

```
Cmd.Wire.IgnoreCtrlForRadiusMode = "1"
```

to the file. This will turn this feature off for all commands that draw wires.

Group Selection

Since the context menu function on the right mouse button interferes with the selection of groups, a group is now selected with Ctrl plus right mouse button. If you want to have the old method of selecting groups back, you can add the line

```
Option.ToggleCtrlForGroupSelectionAndContextMenu = "1"
```

to the file. This will allow selecting groups with the right mouse button only and require Ctrl plus right mouse button for context menus.

Open Project

The automatic opening of the project folder at program start (or when activating a project by clicking onto its gray button) can be disabled by appending the line

```
ControlPanel.View.AutoOpenProjectFolder = "0"
```

to the file.

Panning Drawing Window

Panning can be done with the Ctrl button (as in previous versions) by writing

```
Interface.UseCtrlForPanning = "1"
```

into the file. Note, though, that the Ctrl key is now used for special functions in some commands, so when using these special functions (like selecting an object at its origin in MOVE) with this parameter enabled you may inadvertently pan your drawing window.

Polygon Edges as Continuous Lines

If you don't like the way unprocessed polygons display their edges (as dotted lines), you can add the line

```
Option.DrawUnprocessedPolygonEdgesContinuous = "1"
```

The edges of polygons will be displayed as continuous lines then.

Reposition of the Mouse Cursor

Normally EAGLE does not automatically position the mouse cursor. However, if you prefer the cursor to be repositioned to the point where it has been before a context menu in the drawing editor was opened, add the line:

```
Option.RepositionMouseCursorAfterContextMenu = "1"
```

Units in Dialogs

The automatic unit determination in dialog input fields can be controlled by appending the line

```
Interface.PreferredUnit = "x"
```

to the file, where "x" can be

"0" for automatic unit determination (default)
"1" for imperial units
"2" for metric units.

10.7 Error Message

When Loading a File

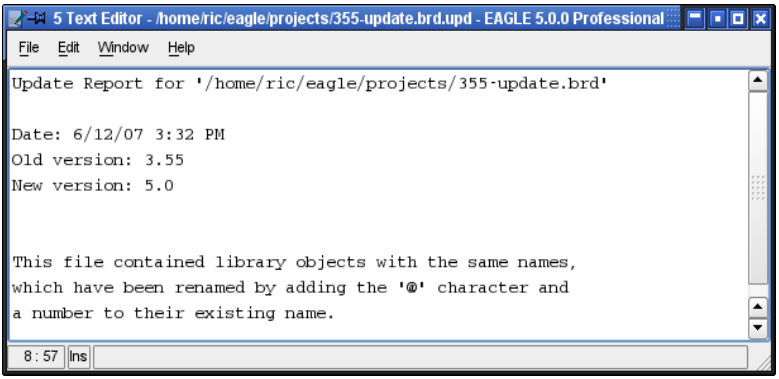
Restring smaller than in older version

➢ *Pad diameter changed*

In EAGLE version prior 4.0 the pad diameter has been fixed in the Package definition. Due to the given values in the Design Rules the pad diameters have changed.

Please check and, if required, change the Restring settings. Run the Design Rule Check in any rate to recognize possible clearance errors.

Library objects with the same names



➢ *Update report: Objects with the same name*

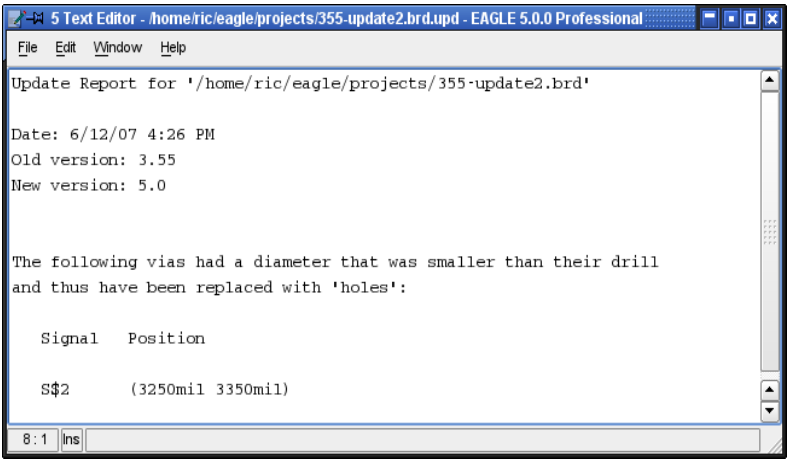
The Text Editor shows this message if you attempt to load an older file (BRD or SCH) that contains different versions of a library element. In this case it added @1, @2, @3... to the names of the Devices so that they can be identified.

This message can also appear if you use CUT and PASTE commands.

Pad, Via Replaced with a Hole

In older versions of EAGLE it was possible to define pads in which the hole diameter was larger than the pad diameter. This is no longer permitted.

If you attempt to load a library file that was created with an earlier version and that contains such a pad, the following message appears:

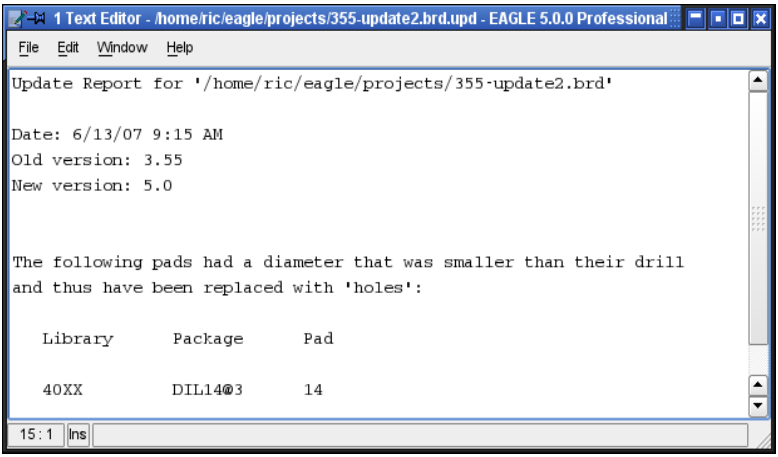


➢ ***Update report: Via replaced with hole***

The pad or via is automatically converted into a hole, provided it is not connected by CONNECT to a pin in one of the library's Devices.

If there is pad that has a connection to a pin (it is defined in the library), the following message appears:

In that case the Library file must be manually edited in order to correct the pad. Then you can update the board file with the new library definition.



➢ *Update report: Pad replaced with a hole*

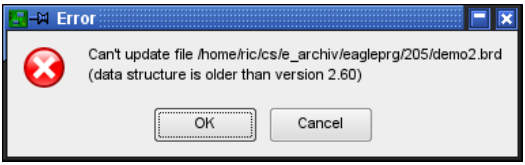
Skipped unsuitable objects

If this message is shown, while you are loading a file or copying objects with CUT and PASTE from one file into another, the data structure contains objects that do not belong to the current file type and can't be displayed. For example, a text or rectangle that has a non-orthogonal angle and is placed in a user-defined layer (above 100) in the Layout editor which should be pasted into a schematic. The Schematic editor doesn't allow non-orthognal angles and therefore can't display such an object.

This message could be prompted as well, if the file's origin is one of the first EAGLE versions. The file can be used without problems nevertheless. The data structure is cleaned up automatically while loading it.

Can't Update File

If this message appears when loading an EAGLE file that was made with a version earlier than 2.60 it is necessary first to convert the file.



➢ *Update error: File older than version 2.6*

The program *update26.exe*, which is located in the *eagle/bin* directory, is used for this purpose.

Copy the file that is to be converted into the directory containing both *update26.exe* and the file *layers.new*. Then open a DOS window under Windows, and change into this directory. Type the command:

```
update26 dateiname.ext
```

The file is converted, after which it can be read by the new version of EAGLE. If the conversion is successful, the message in the DOS box is: *ok...*

If the message *Please define replacement for layer xxx in layers.new* should appear, it means that you have defined your own layers in layout/schematic/library.

Because of the new layer structure used since version 2.6, a new layer number (greater than 100) must be assigned.

This requires you to edit the file *layers.new* using a simple text editor, adding, for example, a new layer number as the last line of the file.

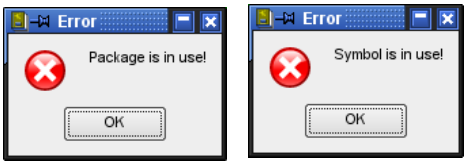
If, for instance, you have used layer 55, and want to give it number 105, enter:

```
55 105
```

In a Library

Package/Symbol is in use

If a Package or Symbol is already used in a Device, no pads or pins which are already referenced to a pin or pad with the help of the CONNECT comand, may be deleted . In such a case EAGLE shows the following messages:



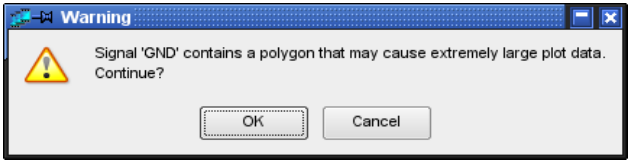
➢ ***Error while editing Package or Symbol***

But it is allowed to CHANGE or NAME such pins or pads. It's also possible to add further pins/pads with the PIN or PAD/SMD command and you are allowed to DELETE pins/pads which are not referenced via the CONNECT command.

This message also appears, if you try to remove the whole Package/Symbol from the library with the REMOVE command. You have to delete the whole Device or the Package variant or symbol in the Device before.

In the CAM Processor

Polygon may cause extremely large plot data



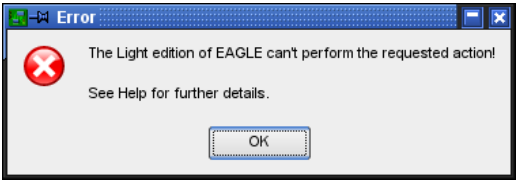
➢ *Polygons with width 0*

This message appears, if you selected a layer in the CAM Processor which contains a signal polygon in the layout whose line thickness is less than the resolution of the selected output driver (*Device*).

In order to avoid unnecessary large plot files you should assign a higher value to the polygon's line width (CHANGE width).

In the Light, Freemium or Standard Edition

Can't perform the requested action



➢ *Light/Standard limits*

This message is shown if the limits of the Light, Freemium or Standard Edition are exceeded. This can be the case, for example, if you want to place a part outside the Layout size limits, if you want to start the Autorouter, or set parameters for the Follow-me router, although there are parts outside the Layout limits, or you want to define a not allowed inner layer.

10.8 EAGLE License Agreement

This license agreement ('License') is a legal agreement between you ('Licensee' or 'you') and CadSoft Computer GmbH of Pleidolfweg 15, Pleiskirchen, D-84568, Germany ('Licensor' or 'we') for this EAGLE software product ('Software'), which includes computer software and any data supplied with it (such as electronic manuals or tutorials) ('Documentation').

BY CLICKING ON THE 'YES' OR 'AGREE' BUTTON BELOW YOU AGREE TO THE TERMS OF THIS LICENSE (INCLUDING THE RELEVANT SCHEDULES TO IT) WHICH WILL BIND YOU AND ANYONE WHO USES THE COPY OF EAGLE LICENSED TO YOU. IF YOU DO NOT AGREE TO THE TERMS OF THIS LICENSE, WE ARE UNWILLING TO LICENSE THE SOFTWARE TO YOU. YOU SHOULD NOT IN THIS EVENT CLICK THE 'YES' OR 'AGREE' BUTTON AND THE INSTALLATION PROCESS WILL NOT CONTINUE. .

1. GRANT AND SCOPE OF LICENSE

1.1 In consideration of your agreeing to abide by the terms of this License, the Licensor hereby grants to you a non-exclusive, non-transferable license to use the Software and the Documentation on the terms of this License.

1.2 The terms on which you are licensed to use the Software vary according to the edition licensed to you. Please see the Schedules to this License for any terms applicable to the Software edition licensed to you which are in addition to or amend those set out below. The Schedule relevant to the edition of EAGLE licensed to you (in each case the 'Applicable Schedule') will form part of this License. In the event of any conflict between conditions 1 to 14 of this License and the provisions of the Applicable Schedule, the provisions of the Applicable Schedule will prevail.

1.3 You may:

(a) use the Software for your purposes only, either:

(i) on one computer if the License is a single-user license or the Software is for single use; or

(ii) if the License is a multi-user license, by the number of concurrent users agreed between you and us

provided that, in either case and save where prohibited in the Applicable Schedule, the Software may be installed on more than one computer or file server simultaneously provided that it is not used concurrently on more than the number of computers for which use is licensed in accordance with (i) or (ii) above, whichever is appropriate.

(b) transfer the Software from one computer to another provided it is used only on the number of computers for which use is licensed in accordance with condition 1.3 (a);

(c) make up one copy of the Software for back-up purposes only, provided that this is necessary for the activities permitted under condition 2.1;

(d) receive and use any free supplementary software code or update of the Software incorporating "patches" and corrections of errors as may be provided by us from time to time;

(e) use any Documentation in support of the use permitted under condition 1.2..

2. LICENSEE'S UNDERTAKINGS

2.1 Except as expressly set out in this License or as permitted by any local law, you undertake:

(a) not to copy the Software or Documentation except where such copying is incidental to normal use of the Software, or where it is necessary for the purpose of back-up;

(b) not to rent, lease, sub-license, loan, translate, merge, adapt, vary or modify the Software or Documentation;

(c) not to make alterations to, or modifications of, the whole or any part of the Software, nor permit the Software or any part of it to be combined with, or become incorporated in, any other programs;

(d) not to disassemble, decompile, reverse engineer or create derivative works based on the whole or any part of the Software nor attempt to do any such thing except to the extent that such actions cannot be prohibited because they are essential for the purpose of achieving inter-operability of the Software with another software program, and provided that the information obtained by you during such activities:

(i) is used only for the purpose of achieving inter-operability of the Software with another software program; and

(ii) is not unnecessarily disclosed or communicated without the Licensor's prior written consent to any third party; and

(iii) is not used to create any software which is substantially similar to the Software;

(e) to keep all copies of the Software secure and to maintain accurate and up-to-date records of the number and locations of all copies of the Software;

(f) to supervise and control use of the Software and ensure that the Software is only used in accordance with the terms of this License;

(g) to include the copyright notice of the Licensor on all copies (entire or partial) you make of the Software on any medium;

(h) to replace the current version of the Software with any updated or upgraded version or any new release provided by the Licensor under the terms of this License immediately on receipt; and

(i) not to provide or otherwise make available the Software in whole or in part (including but not limited to program listings, object and source program listings, object code and source code), in any form to any person (other than, in the case of a multi-user license only, such number of your employees or authorised representatives as are permitted to use the Software at any time under the terms of this License) without prior written consent from the Licensor.

2.2 The Licensor may at any time request you to provide a certificate of confirmation of your compliance with the terms of this License and you will provide such certificate on request.

3. SUPPORT

As any support provided for the Software is provided free of charge, we provide only such support for it as we see fit and that support may vary according to the edition of the Software licensed to you. The extent of support provided (if any) is detailed on our website or those of our associated companies. We reserve the right to change the support provided or the means by which it may be accessed or remove the support altogether at any time without notice.

4. INTELLECTUAL PROPERTY RIGHTS

4.1 You acknowledge that all intellectual property rights in the Software and the Documentation anywhere in the world belong to the Licensor, that rights in the Software are licensed (not sold) to you, and that you have no rights in, or to, the Software or the Documentation other than the right to use them in accordance with the terms of this License.

4.2 You acknowledge that you have no right to have access to the Software in source code form or in unlocked coding or with comments.

4.3 The integrity of this Software is protected by technical protection measures (TPM) so that the intellectual property rights, including copyright, in the Software of the Licensor are not misappropriated. You must not attempt in any way to remove or circumvent any such TPM, nor apply or manufacture for sale or hire, import, distribute, sell or let for hire, offer or expose for sale or hire, advertise for sale or hire

or have in your possession for private or commercial purposes any means the sole intended purpose of which is to facilitate the unauthorised removal or circumvention of such TPM.

5. LIMITED WARRANTY

5.1 Further details as to the extent of any warranty applicable to the edition of the Software licensed to you are as set out in the Schedule relevant to that edition.

5.2 Save as may be set out in the Applicable Schedule, we make no representation or warranty as to the suitability or fitness for purpose of the Software, whether your purpose or any other and regardless of whether that purpose has been disclosed to us and save as above, we make no warranties or representations as to the Software or the Documentation or as to the performance, accuracy, quality, condition, completeness, compliance with statement or description of either of them and all such warranties and representations are excluded to the fullest extent permitted by law.

5.3 You acknowledge that:

(a) the Software has not been developed to meet your individual requirements, and that it is therefore your responsibility to ensure that the facilities and functions of the Software as described in the Documentation meet your requirements; and

(b) that the Software may not be free of bugs or errors, and you agree that the existence of minor errors shall not constitute a breach of this License.

5.4 It is your responsibility to ensure that you make regular and frequent back-ups of all designs, files and other work produced with Software. As set out below, we can accept no liability for any loss of designs, files or other data.

6. LICENSOR'S LIABILITY

6.1 Nothing in this License shall limit or exclude the liability of either party for death or personal injury resulting from negligence, fraud or fraudulent misrepresentation.

6.2 Subject to condition 6.1, the Licensor's liability for losses suffered by you arising out of or in connection with this License (including any liability for the acts or omissions of its employees, agents and subcontractors), however such losses may arise, whether in contract, tort (including negligence), misrepresentation or otherwise, shall be limited as set out in the Applicable Schedule and shall in any event exclude any liability for:

(a) loss of income;

(b) loss of business profits or contracts;

(c) business interruption;

(d) loss of the use of money or anticipated savings;

(e) loss of information, data or designs;

(f) loss of opportunity, goodwill or reputation;

(g) loss of, damage to or corruption of data; or

(h) any indirect or consequential loss or damage of any kind

even if such losses, claims or damages result from the Licensor's deliberate personal repudiatory breach of this License and all and any such liability is hereby expressly excluded to the fullest extent permitted by law.

6.3 This License sets out the full extent of the Licensor's obligations and liabilities in respect of the supply of the Software and Documentation. In particular, there are no conditions, warranties, representations or other terms, express or implied, that are binding on the Licensor except as specifically stated in this License. Any condition, warranty, representation or other term concerning the supply of the Software and Documentation which might otherwise be implied into, or incorporated in, this License,

or any collateral contract, whether by statute, common law or otherwise, is hereby excluded to the fullest extent permitted by law.

6.4 Any reference to loss or liability arising out of or in connection with this License shall be deemed to include any loss, claim, liability or damage arising out of or in connection with the Software, its availability or otherwise, its performance, operability, functionality or in connection with the loss of any designs, data or information or the accuracy or validity of any designs produced using the Software.

7. TERMINATION

7.1 The Licensor may terminate this License immediately by written notice to you if you commit a material or persistent breach of this License which you fail to remedy (if remediable) within 14 days after the service of written notice requiring you to do so.

7.2 Upon termination for any reason:

(a) all rights granted to you under this License shall cease;

(b) you must cease all activities authorised by this License; and

(c) you must immediately delete or remove the Software from all computer equipment in your possession, and immediately destroy or return to the Licensor (at the Licensor's option) all copies of the Software then in your possession, custody or control and, in the case of destruction, certify to the Licensor that you have done so.

8. TRANSFER OF RIGHTS AND OBLIGATIONS

8.1 This License is binding on you and us, and on our respective successors and assigns.

8.2 You may not transfer, assign, charge or otherwise dispose of this License, or any of your rights or obligations arising under it, without our prior written consent.

8.3 We may transfer, assign, charge, sub-contract or otherwise dispose of this License, or any of our rights or obligations arising under it, at any time during the term of the License.

9. NOTICES

All notices given by you to us must be given to CadSoft Computer GmbH at Pleidolfweg 15, Pleiskirchen, Germany. We may give notice to you in accordance with the Applicable Schedule. Notice will be deemed received and properly served immediately when posted on the Websites or any of them, 24 hours after an e-mail is sent, or three days after the date of posting of any letter. In proving the service of any notice, it will be sufficient to prove, in the case of a letter, that such letter was properly addressed, stamped and placed in the post and, in the case of an e-mail, that such e-mail was sent to the specified e-mail address of the addressee.

10. EVENTS OUTSIDE OUR CONTROL

10.1 We will not be liable or responsible for any failure to perform, or delay in performance of, any of our obligations under this License or any unavailability of the Software that is caused by events outside our reasonable control (Force Majeure Event).

10.2 A Force Majeure Event includes any act, event, non-happening, omission or accident beyond our reasonable control and includes in particular (without limitation) the following:

(a) strikes, lock-outs or other industrial action;

(b) civil commotion, riot, invasion, terrorist attack or threat of terrorist attack, war (whether declared or not) or threat or preparation for war;

(c) fire, explosion, storm, flood, earthquake, subsidence, epidemic or other natural disaster;

(d) impossibility of the use of railways, shipping, aircraft, motor transport or other means of public or private transport;

(e) impossibility of the use of public or private telecommunications networks or

(f) the acts, decrees, legislation, regulations or restrictions of any government.

11. WAIVER

11.1 If we fail, at any time during the term of this License, to insist upon strict performance of any of your obligations under this License, or if we fail to exercise any of the rights or remedies to which we are entitled under this License, this shall not constitute a waiver of such rights or remedies and shall not relieve you from compliance with such obligations.

11.2 A waiver by us of any default shall not constitute a waiver of any subsequent default.

11.3 No waiver by us of any of these terms and conditions shall be effective unless it is expressly stated to be a waiver and is communicated to you in writing.

12. SEVERABILITY

If any of the terms of this License are determined by any competent authority to be invalid, unlawful or unenforceable to any extent, such term, condition or provision will to that extent be severed from the remaining terms, conditions and provisions which will continue to be valid to the fullest extent permitted by law.

13. ENTIRE AGREEMENT

13.1 This License and any document expressly referred to in it represents the entire agreement between us in relation to the licensing of the Software and Documentation and supersedes any prior agreement, understanding or arrangement between us, whether oral or in writing.

13.2 We each acknowledge that, in entering into this License, neither of us has relied on any representation, undertaking or promise given by the other or be implied from anything said or written in negotiations between us prior to entering into this License except as expressly stated in this License.

13.3 Neither of us shall have any remedy in respect of any untrue statement made by the other, whether orally or in writing, prior to the date we entered into this License (unless such untrue statement was made fraudulently) and the other party's only remedy shall be for breach of contract as provided in these terms and conditions.

14. LAW AND JURISDICTION

This License, its subject matter or its formation (including non-contractual disputes or claims) shall be governed by and construed in accordance with German law and submitted to the non-exclusive jurisdiction of the German courts.

SCHEDULE 1

CONDITIONS APPLICABLE TO THE LIGHT, STANDARD AND PROFESSIONAL EDITIONS OF EAGLE (TOGETHER WITH ANY OTHER COMMERCIAL EDITION OF EAGLE ISSUED BY US FROM TIME TO TIME TO WHICH THE FOLLOWING SCHEDULES DO NOT APPLY)

1 Reference in this Schedule 1 to the Software is reference to the commercial edition of EAGLE licensed to you.

2 The Licensor warrants that for a period of 30 days from the date of installation ('Warranty Period') the Software will, when properly used, perform substantially in accordance with the functions described in the Documentation (provided that the Software is properly used on the computer and with the operating system for which it was designed, as specified by us) and the Documentation correctly describes the operation of the Software in all material respects.

3 If, within the Warranty Period, you notify the Licensor in writing of any defect or fault in the Software in consequence of which it fails to perform substantially in accordance with the Documentation, and such defect or fault does not result from you having amended the Software or used it in contravention of the terms of this License, the Licensor will, at its sole option, either repair or replace the Software, provided that you make available all the information that may be necessary to help the Licensor to remedy the defect or fault, including sufficient information to enable the Licensor to recreate the defect or fault.

4 Subject to condition 6.1 and 6.2 of the License above, the Licensor's maximum aggregate liability under or in connection with this License, or any collateral contract, whether in contract, tort (including negligence) or otherwise, shall be limited to a sum equal to the amount received by us by way of license fee for the Software.

5 Subject to condition 6.1 and 6.2 of the License and clause 4 of this Schedule 1 above, the Licensor's liability for infringement of third-party intellectual property rights shall be limited to breaches of rights subsisting in the USA and/or Germany.

6 Notice in connection with this License may be given by us to you at either the e-mail, fax or postal address you provided to us or our representative at the time of order of the Software or where such information is generic and not specific to you by displaying on our website or that of any of our associated companies or undertakings.

SCHEDULE 2

CONDITIONS APPLICABLE TO THE FREEMIUM EDITION ONLY

1 The Freemium edition requires access to the internet to operate. The computer on which the Freemium edition is run must have access to the internet on each occasion on which you launch EAGLE in order for the Freemium edition to open and operate. Reference in this Schedule 2 to the Software is reference to the Freemium edition of EAGLE.

2 All licenses of the Freemium edition are for single use only. The Freemium edition may be downloaded to and used only on the computer licensed to run the Software at the time you activated the Freemium edition..

3 The Freemium edition is licensed for use for evaluation or not for profit purposes only. 'Evaluation' allows you to use the Software in a commercial environment but purely for the purposes of evaluating the functionality and suitability for your purposes of the Software. 'Not for profit' allows you to use the Software for personal use or in the course of your studies if you are in education or for any other not-for-profit aim. In no case must the Software be used in order to generate revenue (whether directly or by way of costs savings).

4 This license of the Software is for a period of sixty (60) days from the date of its download by you. On expiry of this sixty (60) day period, this license and your right to use the Software under it shall cease and the Software shall cease to operate. At that time you should delete the Software from any computer onto which it is loaded. You may apply to download and be licensed to use the Freemium edition under the terms of a separate license between you and us at any time after the expiry of this license but we may apply restrictions as to the frequency with which such downloads may be available and may amend, revise, supplement or remove those restrictions from time

to time. We shall endeavour to publish the terms of those restrictions on our website or those of our associated companies or undertakings or any other website(s) on which we make the Software available but shall have no liability to you in the event that we do not do so for any reason and failing to do so shall not affect the efficacy of those restrictions.

5 You acknowledge that the Software is licensed to you free of charge and that accordingly we make no representation or warranty as to:

(i) the suitability or fitness for purpose of the Software, whether your purpose or any other and regardless of whether that purpose has been disclosed to us;

(ii) the performance or functionality of the Software, whether in accordance with any statements made, materials published or the Documentation or otherwise;

(iii) the availability or functionality of any server or system to allow registration for, download or communication with the Software;

(iv) whether the Software infringes the intellectual property rights of any third party anywhere in the world.

6 You further acknowledge that the Software will not operate without access to the internet, such that the Software is able to communicate with our servers, and that it is your responsibility to ensure that you have such access at all times when starting the Software; and that that access is adequate and uninterrupted throughout the start-up process.

7 Notice in connection with this License may be given by us to you at either the e-mail or postal address you provided to us or our representative when registering to receive the Software or by displaying such notice on our website or that of any of our associated companies or undertakings or any other website(s) on which we make the Software available.

SCHEDULE 3

CONDITIONS APPLICABLE TO THE FREEWARE EDITION ONLY

1 Reference in this Schedule 3 to the Software is reference to the Freeware edition of EAGLE.

2 All licenses of the Freeware edition are for single use only.

3 The Freeware edition is licensed for use for evaluation or not for profit purposes only. 'Evaluation' allows you to use the Software in a commercial environment but purely for the purposes of evaluating the functionality and suitability for your purposes of the Software. 'Not for profit' allows you to use the Software for personal use or in the course of your studies if you are in education or for any other not-for-profit aim. In no case must the Software be used in order to generate revenue (whether directly or by way of costs savings). The Freeware edition may also be used by a PCB manufacturer to determine production data necessary for the manufacture of a PCB where that PCB manufacturer has been commissioned by the third party designer of that PCB to produce a PCB to that design on his or her behalf.

4 You acknowledge that the Software is licensed to you free of charge and that accordingly we make no representation or warranty as to:

(i) the suitability or fitness for purpose of the Software, whether your purpose or any other and regardless of whether that purpose has been disclosed to us;
(ii) the performance or functionality of the Software, whether in accordance with any statements made, materials published or the Documentation or otherwise;
(iii) whether the Software infringes the intellectual property rights of any third party anywhere in the world.

6 Notice in connection with this License may be given by us to you or by displaying such notice on our websites or that of any of our associated companies or undertakings.

Index

A

Action toolbar **48, 50**

Addlevel **237**
- Always 244, 246
- Can 244
- Must 244, 245
- Next 244
- Request 244

Airwire **20**
- Calculate 70
- Display/hide 70

Alias **91**
- Deleting 93

Alpha blending **101**

Alt-X **42, 60**

Annulus symbol
- Dimension 136
- In supply layer 136

Annulus Symbol **21**

Aperture **290**

Aperture File **264**

Attributes
- Defining 58, 121, 240
- For elements 122
- Global 121

Automatic Naming **93**

Autorouter **70**
- Backup 191
- Blind vias 163
- Bus router 178
- Continue existing job 186
- Control file, ctl 193
- Control parameter 186
- Controlling 178
- Cost factors 186
- Design rules 179
- Features 177
- Hints 195
- Information 192
- Interruption 191
- Layer selection 182
- Load settings 184
- Log file 193
- Memory requirement 181
- Menu 183
- Micro via 191
- Min. distance, clearance 179
- Min. routing grid 177
- Module 32
- Multilayer board 182
- Net classes 179
- Optimization 179
- Placement grid 180
- Polygon 189
- Polygons as supply 191
- Preferred direction 182, 184
- Restarting 186
- Restricted area 183
- Ripup/Retry 190
- Routing grid 180
- Routing pass 179
- Save settings 184
- Select signals 185
- Single-sided boards 195
- Smds and supply layer 195
- Status display 192
- Supply layer 161, 182, 190
- Track width 179
- Unreachable Smd 181, 182

B

Background color **101**

Backup files **44, 295**

Beep **103**

Bill of material **268**

Blind via **21**

Blind via ratio **132**

Blind, Buried via **162**

Bmp file **96**

Board
- Arrange components 142
- Attributes, global 144
- Creating 139
- Cut-out 251
- Cutouts 141
- Design Rules 128
- Draw outline 140
- Layer setup 130
- Lock component 66
- Multilayer 158
- Multiple board 174
- Placement grid 139

Prior considerations 127
Routing manually 147

Board Manufacture 263

Bubble help 46, 48

Buried via 21

C

CAM Job
Define drill job 287
Description 284
Excellon.cam 273
Extending gerber.cam 285
gerb274x.cam 272
gerber.cam 274
Output parameter 280

CAM Processor 77
Choose pen number 282
Component side 276
Cream frame 277
Creating device driver 288
Drill data 265, 277
EPS output 267
Error: Apertures missing 286
Error: Drills missing 288
Extremely large plot data 307
File extensions 279
Fill Pads 282
Finish mask 277
Glue mask 277
HPGL 97
Inner layer 276
Job 284
Load job file 79
Milling contours 277
Plotter 281
Pos. Coords. 282
PostScript 266
Prompt 275
Save time and money 271
Section name 282
Silk screen 277
Solder stop 277
Start 50, 78
Start from batch 295
Supply layer 276, 286
Tolerance 283
UNC notation 281
Vias 279
282

Caption 172

Circle
Filled 247

Clearance 131

Cmd key 85

Color
Background 46
Settings 100

Command
Activating 81
Language 85
Line 48, 81
Parameters 49
Text menu 299
Toolbar 48

Commands
ADD 55, 66, 76, 209
ARC 57, 69
ASSIGN 59, 83, 99
ATTRIBUTE 58, 70, 77
AUTO 70
BOARD 50, 139
BUS 57, 118
CHANGE 54, 65, 76, 205
CIRCLE 57, 69, 85, 89
CLASS 59, 117
CLOSE 59
CONNECT 76, 210, 232
COPY 53, 64, 254
CUT 54, 65
DELETE 54, 65
DESCRIPTION 74, 77
DISPLAY 52, 63, 92
DRC 71, 152
EDIT 59, 72, 110
ERC 58, 71, 124
ERRORS 71, 154
EXPORT 59, 85, 95
FRAME 59, 249
GATESWAP 55, 210
GRID 50
GROUP 53, 64
HELP 49
HOLE 70, 82, 247
INFO 51, 63, 132
INVOKE 56, 112
JUNCTION 57, 114
LABEL 58, 115
LAYER 59, 294
LOCK 66
MARK 52, 63, 256
MENU 60
MENU 99
MIRROR 53, 64, 82, 144, 250
MITER 56, 67

MOVE 52, 63, 82
NAME 55, 66, 76
NET 57, 114
OPEN 60
OPTIMIZE 67
PACKAGE 60, 76, 252
PAD 73, 202
PASTE 54, 65
PIN 90, 206
PINSWAP 55, 66
POLYGON 57, 69, 149
PREFIX 76, 211
PRINT 60, 171
QUIT 60
RATSNEST 70
RECT 57, 69
REDO 51
REMOVE 60, 72, 110, 260
RENAME 73, 260
REPLACE 55, 66, 146
RIPUP 68, 148
ROTATE 53, 64, 143, 255
ROUTE 67, 147
RUN 51
SCRIPT 51, 95
SET 60, 100, 300
SHOW 52, 63
SIGNAL 70
SMASH 55, 67, 112, 144
SMD 73, 202, 221
SPLIT 56, 67
TECHNOLOGY 61, 77, 234
TEXT 57, 68
UNDO 51
UPDATE 61, 170, 260
USE 50, 213
VALUE 55, 66, 76, 211
VIA 70, 82, 168
WINDOW 51
WIRE 56, 68
WRITE 61

Component
Add from library 55
Attribute 122
Changing Technology 146
Copying by Drag&Drop 257
Create symbol 205
Creating 75, 201
Cross-reference 247
Description 205
Keepout 205
Labeling 204
Lock 66
Name 204
On both sides 144
On bottom side 202, 250
Output list 96
Package editor 204
Placement grid in board 139
Prefix 211
Replace device 146
Replace package 144
Replacing 55
Rotation 142
Separate name/value 55, 67
Update 170
Value 55, 66, 204

Configuration
Commands 99
eagle.scr 105
eaglerc, eaglerc.usr 106, 300
of EAGLE 99
User interface 99

Connector 245

Consistency check 58, 71, 99, 124

Consistency lost 175

Contact cross reference
>CONTACT_XREF 116
>XREF 247

Context menu 41, 82

Control Panel 37
Options menu 43

Control parameters 186

Coordinates
Display 48, 52, 63
Entering 88
Modifier 89
Polar 88, 256
Relative 88, 256
Select group 89

Copper plane 149

Core 21, 162

Cost factors 186

Cream mask 137

Cross reference
For contacts 116, 247
For nets 58, 115
Specify format 116

Ctrl key 85

Current units 90

Cursor appearance 46

D

Data output **78**

Date/time stamp **249**

Delete
- All signals 96
- Wire bend 65

Design Rule Check **21, 71**
- Approve errors 155
- Correcting errors 152
- Fonts 139
- Meaning of errors 155
- Restricted areas 139
- Show errors 71
- Wire styles 158

Design Rules **40, 128**
- Clearance 131
- Definition 71
- Layer setup 130
- Options 129
- Restring 132

Designlink interface **114**

Desktop Publishing **267**

Device **21**
- Assign Package 210
- Attributes 240
- Build Device Set 229
- Copying 257
- Creating 209
- Delete 72
- Description 212
- Driver 288
- Editing 72, 74
- Gate names 210
- Placeholder in name 230
- Prefix 211
- Remove from LBR 260
- Rename 73
- Replacing 55, 146
- Technologies 234
- Value on/off 76, 211

Device Set **21**

Directories **43**

Distance **131**

Documentation **268**

Documentation
- Export image 96
- Print 203

Documentation field **249**

Drag&Drop **37**

Draw lines **56**

Drawing area
- Alias 92
- Display last 51
- Panning 51

Drawing frame **110, 249**

Drawing name **249**

DRC **21**
- See Design Rule Check 152

Drill **21**
- Diameter 247
- Legend 270
- Non-plated 265
- Plan 269
- Plated 265
- Symbols 270

Drill data **265**
- Blind/buried vias 280
- Configuration file 287
- drillcfg.ulp 287
- Error: Drills missing 288
- EXCELLON 265
- EXCELLON_RACK 287
- Info file 288
- Leading zeros 289
- Multilayer boards 280
- Rack 287
- Separate rack file 287
- SM1000 265
- SM1000/SM3000 287
- SM3000 265
- Tolerance 288
- Units 289

Dxf data export **97**

E

eagle.def **288, 290**

eagle.epf **107**

eagle.scr **105**

eaglecon.exe **295**

eaglerc, eaglerc.usr **106, 300**

Edition
- Freemium 34
- Light 34
- Professional 32
- Standard 34

Electrical Rule Check **21, 58, 124**
- Approve errors 125

Electrical schematic **247**
Electrical Schematic **116**
Elongation **135**
Encapsulated PostScript **267**
Environment variable **43**
ERC **21**
Error messages
CAM Processor 307
Correcting 152
DRC - Meaning of 155
File prior version 2.60 305
In a library 306
Loading a file 303
excellon.cam **287**
Exit program **42**
Expert options **300**
Export data **94**

F

Fiducials **271**
File
Backup 295
Edit 59
New 41
Open 42, 50
Print 50
Save 50
Film Generation **266**
Fixing hole **246**
Follow-me Router **21, 196**
Font
Checking 139, 157
No vector error 157
Persistent in drawing 45
Typeface 57, 68
Vector 45
Forbidden area **146**
Forward&Back Annotation **21, 98**
Consistency lost 175
Function keys **59, 83, 99**

G

Gate **21, 230**
Gate
Hidden supply 112
Place particular 56
Gateswap **119**
Gerber
Extending gerber.cam 285
Fixed aperture wheel 290
Supply layer 279
Units 289
Gerber device
RS-274D 264
RS274X 264
Gerber output
Annulus aperture 283
Aperture configuration 290
Aperture types 290
Arc 283
Draw aperture 290
Emulate aperture 283
Emulation 292
Error: Apertures missing 286
Flash aperture 290
GERBER 274
gerber.cam 274
GERBERAUTO 274
Info files 286
Messages gerber.cam 275
Resolution 264
Supply layers 279
Wheel 290
Wheel, example file 291
GND symbol **238**
Graphic format **96**
Graphics data
Import 98
Grid **90**
Alias definition 92
Alternative grid 91
Check 139
Menu 91
Min. visible size 104
Pad placement 202
Group
Define 53, 64
Move 53
Move to sheet 53
Rotate 255

H

Help function **46, 49**
History function **82**
Hole **21**
Diameter 247
Min. diameter 132
HOME variable **43**
Hyperlinks
In descriptions 212

I

Import **94**
In-circuit tester **267**
Inner layer **159**
Installation **23**
Invalid Polygon **156**

J

Job **274, 284**
Junction
Set automatically 104

K

Keepout **156, 205**

L

Language setting **29**
Layer
Abuse 156
Alias definition 92
Available 100
Creating 59
Display/hide 52, 63
Hide unused 100
Inner 159
Power supply 159
Qty. of signal layers 130
Setup 130, 161, 163
Signal 159
Stack 22, 162
Thickness 131, 164
Usage 293
Layout Editor **31, 62**
Library
Attributes 240
Composition of your own 259
Copy elements 256
Copying by Drag&Drop 257
Create new 215
Description 213
Device creating 209
Drop 111
Important comments 20
List contents 95
Open 60
Output script file 96
Package creating 202
Package variants 231
Remove element 260
Rename element 260
Search for elements 110
Summary 38
Symbol creating 205
Update 170
Update Package 260
Updating older files 27
Use 39, 50
Library Editor **71**
License
Extending 27
Multiple Users 28
New Installation 23
License agreement **307**
Light edition **32**
Line **22**
Type 68
Logo import **98**

M

Magnetic pads **148**
Menu
Configure Text menu 299
Contents parameter menu 100
Menu bar **48**

Micro Via
Definition 22, 169
Restring, diameter 132
Set in SMD 169
Milling
Contour 251
Cutout in board 141
Prototype board 266
Milling machine 266
Mounting hole 70, 246
Mouse click 87
Right click 89
Mouse keys 61
Mouse wheel zoom 45
Multilayer boards 158
4-Layer 163
6-Layer 165
8-Layer 167
Blind, Buried vias 162
Through vias 161
Via display 163

N

Name
Automatic naming 94
Forbidden characters 94
Length 93
Net 22
Connection point 57
Cross reference 58, 115
Net classes 117
Netlist 95
Netscript 96

O

Object
Move 52, 63
Properties 51, 63
Show properties 46
Oblong holes 251
Output
Drawing 59, 60
Image 96

P

Package 22
Assigning 210
Changing 145
Copying 254, 256
Creating new variant 251
Delete 72
Delete variant 253
Description 205
Editing 72, 73
In use 306
New 73
Radial pad arrangement 256
Remove from LBR 260
Rename 73, 260
Rename variant 231
Replacing 66
Rotation 255
Update in LBR 259
Variants 231
Pad 22
Appearance in Editor 136
Aspect ratio 135
Automatic naming 93
Change shape 205
Diameter 202
Diameter in inner layer 134
Display mode 104
Display name in board 104, 217
First 135, 216
Form 135
Layer color 136
Magnetic pads 148
Oblong hole 251
Offset pad 135
Radial arrangement 256
Restring, Diameter 132
Shapes 216
Solder stop mask 137
Stop flag 217
Thermals flag 137, 217
Palette 101
Panelize boards 174
Panning 51
Parameter toolbar 48, 50
Parts list 96, 268
Paste buffer 54, 65
Path specifications 43
Pbm file 96
PDF output 173

Pgm file **96**
Photoplotters **290**
Pick-and-place data **267**
Pin **22**
Automatic naming 94
Connection point 114, 208
Direction 207
Function 206
Inverted signal 227
Labeling 207
Length 207
Name 208
Orientation 206
Properties 206
Same names 237
Superimposed 127
Swap 55, 66
Visible 207
Pin/Pad connection **210, 233**
Pin/Pad list **96**
Pinswap **119**
Placeholder
For attributes 243
>CONTACT_XREF 116, 250
>DRAWING_NAME 249
>GATE 243
>LAST_DATE_TIME 249
>NAME 208, 243
>PART 243
>PLOT_DATE_TIME 250
>SHEET 249
>SHEETNR 250
>SHEETS 250
>VALUE 208, 243
Placeholder texts **300**
Plated-through hole **70**
Png file **96**
Polar coords. **256**
Polygon
Calculation on/off 104
Invalid 152
Isolate 151
Orphans 151
Outline mode after Ratsnest 152
Pour 150
Rank 150
Spacing 151
Thermal connector width 151
Thermals 151
Width 150
Postscript **173**
PostScript **266**
Power supply **120**
Power supply layer **159**
Ppm graphic file **96**
Prefix **76**
Prepreg **22, 162**
Print out
Date/time 250
Drawing 171
Options 172
Page limit 172
PDF file 173
Printing **60**
Product information **47**
Product registration **46**
Professional edition **32**
Project
Close 42
Create new 41, 107
Directory 43
Edit Description 41
File, eagle.epf 107
Mangement 40
Open recent p. 42
Prototype Manufacture **266**

R

Rack file **22, 287**
Ratsnest **22**
Registration **46**
Relative coords. **256**
Relay **245**
Repetition points **87**
Restricted area **146, 246**
Restricted area
For components 205
Restring **22, 132**
RGB value **100**
Roundness **135**
Rubber band **20**

S

Schematic

Checking 124
Create sheet 50
Creating 109
Delete sheet 60
Draw nets 114
Drawing frame 110
Editor 31
Global attributes 121
Grid 110
Merge 127
More than one sheet 126
New sheet 110
Points to note 127
Remove sheet 50
Sheet preview 48
Sheet preview on/off 126
Sort sheets 126
Various supply voltages 120

Script files 95
Comments 95
defaultcolors.scr 103
Syntax 85

Select factor 104

Selecting objects 61

Serial number 47

Sheet
Delete 110
Max. number of 31
New 110
Sorting 48, 126

Signal 22
Display name 147

Signal layer 159

Silkscreen 203

SMD
Automatic naming 94
Cream flag 222
Define size 221
Parameter 202
Placement 221
Round shape 221
Roundness 134
Solder cream mask 137
Solder stop mask 137
Stop flag 222
Thermals flag 137, 222

Snap length 104

Solder cream mask 137

Special characters 94

Standard edition 32

Status line 48

Stop frame 137

Superimposed pins 127

Supply
Addlevel for gates 244
Autorouting supply layer 190
Invisible pins 236
Layer with polygons 159
Symbol 22, 120, 238
Various voltages 120
Voltages 236

Swaplevel 55, 119, 208

Symbol 22
Copying 256
Creating 205
Delete 72
Editing 72, 74
In use 306
Labeling 243
New 74
Power supply 228
Remove from LBR 260
Rename 73

T

Technologies 233

Technology
Changing 146

Termination
Of command 51

Text
Bar over text 94
Change size 57, 68
Editor 79
Font 57, 68
HTML text 205
In copper layer 271
Inverted in copper layer 69
Menu 60, 99, 299
Min. visible size 104
Ratio 204
Separate from component 144
Special characters 94
Spin flag 142
Upside down 143, 218
Variables 249, 300

Thermal symbol
Dimension 136
In polygon 151, 159
In supply layer 136, 161

Tif graphic file **96**
Title bar **48**
Tool tips **48**
Track
Bend mode 149
Decompose 148
Delete all 65
Min. width 132
Set width automatically 104
Smooth wire bends 149
Tree view
Update 42

U

UNC notation **29**
Undo buffer **103**
Unsmash texts **67**
Update
Check for Update 47
User guidance **46**
User interface **44**
User Language **22, 97**
User Language Program
bom.ulp 268
Calculate milling contour 266
cam2dxf.ulp 174
cam2print.ulp 174
designlink.ulp 114
dif40.ulp 268
drill-legend.ulp 270
drillcfg.ulp. 280
dxf.ulp 97
fabmaster.ulp 268
gencad.ulp 268
List of all 40
mill-outlines.ulp 266
mount.ulp 268
mountsmd.ulp 268
outlines.ulp 266
Start ULP 51
unidat.ulp 268

V

Value
Placeholder text in package 218
Placeholder text in symbol 227
V. for Device 234
V. is always Device name 301
Warning 300
Variable
$EAGLEDIR 43
$HOME 43
Variant
Creating new 251
Delete 253
Using modified one 254
Vector font **45**
Checking 139
Via **22**
Appearance in Editor 136
Blind 163
Blind via ratio 132, 169
Buried 162
Diameter display with INFO 132
Diameter in inner layer 134
Layer color 136
Length 168
Limit 138
Micro via 163, 169
Restring, Diameter 132
Shape in inner layer 135
Solder stop 138
Thermal symbol 136

W

Wheel file **22**
Wheel mouse **45**
Window
Fetch into foreground 84
Menu 46
Number 46
Store position 46
Wire **22**
Bend mode 149
Style 68

X

Xbm graphic file **96**
Xpm graphic file **96**
XREF label **115**

Z

Zoom factor limit **45**
Zoom in/out **51**