Contents

Overview ................................................................................................................................. 17

Introduction ...................................................................................................................................... 17
Software features out of the box ........................................................................................................ 17
User Extendibility ................................................................................................................................ 19
One eye on the future .......................................................................................................................... 19
Our Commitment ............................................................................................................................ 20

Part 1 - Getting Started ........................................................................................................ 23

Introduction .................................................................................................................................. 24
Screen Layout .................................................................................................................................. 26
Common functionality ......................................................................................................................... 27
Pan and Zoom .................................................................................................................................. 27
Names and Numbers .......................................................................................................................... 28
Selection ......................................................................................................................................... 30
Context Menus ................................................................................................................................. 31
Commands ....................................................................................................................................... 31
Starting a New Design ................................................................................................................. 32
Overview of PCB design tasks .......................................................................................................... 32
Data Requirements ............................................................................................................................ 35
The New-Board Wizard ..................................................................................................................... 37
Starting Without the New-Board Wizard ............................................................................................. 41
Results of Creating a New Design ..................................................................................................... 41
Default Settings ................................................................................................................................. 42
PCB123 Directories ............................................................................................................................ 43

Part 2 - Schematic User Guide ............................................................................................... 45

The Schematic/PCB Contract ........................................................................................................ 46
A little background ............................................................................................................................. 46
The Design-by-contract solution ........................................................................................................ 47
Properties ......................................................................................................................................... 49
Symbols and Parts ........................................................................................................ 51

Parts .................................................................................................................................... 53
Symbols ................................................................................................................................. 53
  Symbol text .......................................................................................................................... 54
  Symbol shapes ..................................................................................................................... 55
  Symbol pins .......................................................................................................................... 55
  Symbol properties ............................................................................................................... 56

The Part Taxonomy .............................................................................................................. 61

Hierarchy and Inheritance ................................................................................................. 62
The Part Manager .................................................................................................................. 65
Acquiring Data ...................................................................................................................... 67

The Symbol Editor ................................................................................................................ 70

Inside/Outside ...................................................................................................................... 70
Adding pins ............................................................................................................................ 71
Adding Shapes ...................................................................................................................... 73
Adding Arcs ........................................................................................................................... 75
Adding Text ............................................................................................................................ 76
Adding properties ............................................................................................................... 77
Saving symbols ...................................................................................................................... 78
Sections .................................................................................................................................. 78
Binding to specific symbols ............................................................................................... 79
Mapping pin names to pin numbers ................................................................................... 79
Hidden pins ............................................................................................................................ 80
Saving parts ............................................................................................................................ 81

Finding Parts ....................................................................................................................... 82

Part search ............................................................................................................................. 82
Define New Part ..................................................................................................................... 83

Ports ...................................................................................................................................... 89

Ports and net names .............................................................................................................. 89

Schematic Pages .................................................................................................................. 91

Adding Schematic Pages ..................................................................................................... 92
Synchronization pages ......................................................................................................... 93
The Title Block ........................................................................................................................ 93
Page size ................................................................................................................................ 93

**The Schematic Editor** ........................................................................................................... 95

- Adding parts to a page ............................................................................................................. 95
- Adding ports to a page ............................................................................................................ 95
- Adding wires ........................................................................................................................ 96
- Adding nomenclature .............................................................................................................. 96
- Properties ............................................................................................................................... 97
- Cut/copy/paste ........................................................................................................................ 97
- Moving parts ........................................................................................................................... 97

**Part 3 - Layout User Guide** .................................................................................................. 99

**Layout Views and Panels** .................................................................................................. 100

- Display Panel ....................................................................................................................... 100
- Design View .......................................................................................................................... 103
- 3D View ............................................................................................................................... 103
- Footprint Editor ..................................................................................................................... 104
- DRC Panel ............................................................................................................................ 104
- Project Panel .......................................................................................................................... 104

**PCB Design Tasks** ............................................................................................................. 106

**Create missing footprints** .................................................................................................. 107

- Creating footprints from scratch .......................................................................................... 108
- Modify and existing footprint from a library .......................................................................... 108
- Use an existing component as a starting point ..................................................................... 108
- Adjusting the board outline .................................................................................................. 108

**Place components** ............................................................................................................ 109

- Fixed component locations ................................................................................................... 109
- Autoplace components .......................................................................................................... 109
- Manual place components .................................................................................................... 110

**Power, ground, and critical routing** .................................................................................. 110

- Power and ground plane layers ............................................................................................ 110
- Critical routing ....................................................................................................................... 112
Fanout and autorouting ......................................................................................................... 113
Manual routing ...................................................................................................................... 113

**DRC and Visual Inspection** ................................................................................................. 113

Design Rule Checking ............................................................................................................... 114
Visual inspection ..................................................................................................................... 114

**Order Boards** ................................................................................................................... 115

The Order process .................................................................................................................. 115
Before you route ................................................................................................................... 119
Routing the board .................................................................................................................. 119
Adding copper regions .......................................................................................................... 121
Design Changes ...................................................................................................................... 123
Design Rule Checking .......................................................................................................... 123

**Object Editing** .................................................................................................................. 124

Setting the grid ....................................................................................................................... 124
Changing layers ....................................................................................................................... 124
Changing line width ............................................................................................................... 125
Add a component to the board ............................................................................................... 125
Move a component .................................................................................................................. 125
Move a component to the other side of the board ............................................................... 125
Rotate a component ............................................................................................................... 125
Edit a component ................................................................................................................... 126
Add a pin to the board ............................................................................................................ 126
Move a pin ............................................................................................................................... 126
Change a pin ............................................................................................................................ 126
Add a polygon or polyline ........................................................................................................ 127
Add a circle ............................................................................................................................... 127
Add an arc ................................................................................................................................. 127
Move a polygon ....................................................................................................................... 128
Move polygon corners ............................................................................................................ 128
Delete polygon corners ......................................................................................................... 128
Insert corners in a polygon ..................................................................................................... 128
Add a connection between two pins ...................................................................................... 128
Remove a pin from a net ......................................................................................................... 128
Start routing from a pin ......................................................................................................... 129
Start routing from another route ............................................................................................. 129
Add a via between layers ....................................................................................................... 129
Insert corners in a route ....................................................................................................... 129
Delete route corners .............................................................................................................. 129
Move a route corner .............................................................................................................. 130
Move a route segment ........................................................................................................... 130
Change the width of a route segment ..................................................................................... 130
Add text to the board ............................................................................................................ 130
Rename a component ............................................................................................................ 131
Designing a board without a netlist ......................................................................................... 131
About loading a Netlist ........................................................................................................ 131

Part 4 - Schematic Reference ......................................................................................... 135

Main Menu ........................................................................................................................... 136

File menu ................................................................................................................................ 136

New .......................................................................................................................................... 136
Open......................................................................................................................................... 136
Close......................................................................................................................................... 137
Save.......................................................................................................................................... 137
Save As.................................................................................................................................... 137
Backup Now............................................................................................................................. 137
Print.......................................................................................................................................... 137
Print Setup............................................................................................................................... 138
Exit........................................................................................................................................... 138

Edit menu ................................................................................................................................ 138

Undo......................................................................................................................................... 138
Redo.......................................................................................................................................... 139
Cut........................................................................................................................................... 139
Copy......................................................................................................................................... 139
Paste......................................................................................................................................... 139
Delete....................................................................................................................................... 139
Properties................................................................................................................................. 140
### Insert menu

- Add Page .......................................................................................................................... 140
- Add Part ........................................................................................................................... 141
- Add Analog Ground Port ............................................................................................... 141
- Add Digital Ground Port ............................................................................................... 141
- Add Voltage Port ............................................................................................................ 141
- Add Global Port ............................................................................................................. 141
- Add Off-Page Port ......................................................................................................... 142
- Add Wire .......................................................................................................................... 142
- Add Line ......................................................................................................................... 142
- Add Arc ........................................................................................................................... 142
- Add Polyline ................................................................................................................... 142
- Add Rectangle ............................................................................................................... 143
- Add Polygon ................................................................................................................... 143
- Add Text .......................................................................................................................... 143

### View menu

- View Standard Toolbar ............................................................................................... 144
- View Add Object Toolbar ............................................................................................. 144
- View Zoom Toolbar ....................................................................................................... 144
- Zoom Submenu ............................................................................................................. 144
- Zoom In ......................................................................................................................... 145
- Zoom Out ....................................................................................................................... 145
- Zoom All ......................................................................................................................... 145
- Pan Up ............................................................................................................................... 145
- Pan Left ............................................................................................................................ 145
- Pan Right .......................................................................................................................... 145
- Pan Down .......................................................................................................................... 145
- Redraw ............................................................................................................................ 145

### Tools menu

- Symbol Editor ................................................................................................................. 146
- Manage Parts .................................................................................................................... 146
- Electrical Rules Check ..................................................................................................... 146
- Generate Netlist from Schematic .................................................................................. 146
Part 5 - Layout Reference

Main Menu

File menu

New command
Open... command
Close command
Save command
Save As... command
Save As Template... command
Export BOM command
Export Netlist command
Backup Now... command
Print... command
Print Setup... command
Print Artwork... command
Exit command

Edit menu

Undo command
Redo command
Cut command
Copy command
Paste command
Delete command
Load Selection… command ..................................................................................................... 155
Save Selection… command ..................................................................................................... 155
Find… command .................................................................................................................... 155
Bring to Cursor… command .................................................................................................... 155
Line Width command ............................................................................................................. 156
Grid command ...................................................................................................................... 156
Select Pins command ............................................................................................................. 156
Select Polygons command ...................................................................................................... 156
Select Text command ............................................................................................................. 156
Select Components command ................................................................................................. 156
Select Tracks command ......................................................................................................... 156
Select Anything command .................................................................................................... 157
Select Nothing command ....................................................................................................... 157
Properties command .............................................................................................................. 157

Insert menu ............................................................................................................................ 157

Select command .................................................................................................................... 158
Add Pin command ................................................................................................................. 158
Add Freestyle Route command ............................................................................................... 158
Add Polygon command .......................................................................................................... 158
Add Arc command ................................................................................................................. 159
Add Circle command ............................................................................................................. 159
Add Text command ............................................................................................................... 159
Add Component command ..................................................................................................... 160
Add Connection command ..................................................................................................... 160

View menu ............................................................................................................................ 161

Refresh All ............................................................................................................................ 162
Go-To First Error .................................................................................................................... 162
Go-To Next Error ................................................................................................................... 162
User Preferences .................................................................................................................... 162
Toolbars/Edit Panel ................................................................................................................ 162
Toolbars/Standard Toolbar ..................................................................................................... 162
Toolbars/Add Object Toolbar ................................................................................................. 162
Toolbars/Layers Toolbar ......................................................................................................... 162
Toolbars/Selection Toolbar ................................................................. 162
Toolbars/Zoom Toolbar ........................................................................ 163
Zoom/Zoom In ..................................................................................... 163
Zoom/Zoom Out .................................................................................. 163
Zoom/Zoom All ................................................................................. 163
Zoom/Pan Up ....................................................................................... 163
Zoom/Pan Left .................................................................................... 163
Zoom/Pan Right .................................................................................. 163
Zoom/Pan Down ................................................................................. 163
Zoom/Redraw .................................................................................... 164
Layers/Top ......................................................................................... 164
Layers/Bottom .................................................................................. 164
Layers/Inner1 .................................................................................... 164
Layers/Inner2 .................................................................................... 164
Layers/Inner3 .................................................................................... 164
Layers/Inner4 .................................................................................... 164
Layers/Silkscreen Top ........................................................................ 165
Layers/Silkscreen Bottom ................................................................. 165
Layers/Soldermask Top ...................................................................... 165
Layers/Soldermask Bottom ............................................................... 165

**Design menu** .................................................................................. 166
Board Configuration ........................................................................... 166
New Footprint .................................................................................... 166
Edit Footprint ..................................................................................... 166
Manage Library .................................................................................. 167
Optimize Net Lengths ........................................................................ 167
Autoplace ........................................................................................... 167
Fanout SMT Pads ................................................................................ 167
Autoroute ............................................................................................ 168
Gloss ................................................................................................... 168
Clear DRC Errors ................................................................................ 168
Run DRC ............................................................................................. 169

**PCB123 menu** ............................................................................... 170
MY Account........................................................................................................................ 170
PCB123 Website...................................................................................................................... 170
Report a Bug.......................................................................................................................... 171

**Window menu** .................................................................................................................. 171

Cascade ................................................................................................................................. 171
Tile .................................................................................................................................... 171
Arrange Icons ......................................................................................................................... 171

**Toolbars** .......................................................................................................................... 171

Main toolbar .......................................................................................................................... 171
Add Object toolbar ................................................................................................................ 172
Layer toolbar ........................................................................................................................ 172
Selection toolbar .................................................................................................................... 172
Zoom toolbar ........................................................................................................................ 172
3-D View toolbar ................................................................................................................... 173
Wireframe ............................................................................................................................. 173
Show Silkscreen ..................................................................................................................... 173

**Panels** .................................................................................................................................. 173

Display panel .......................................................................................................................... 173
Project panel .......................................................................................................................... 174
DRC/ERC panel ....................................................................................................................... 174

**Dialog Boxes** ..................................................................................................................... 175

About... Dialog ....................................................................................................................... 175
Add Text Dialog ...................................................................................................................... 175
Board Configuration Dialog ................................................................................................. 176
Change Grid Dialog ............................................................................................................... 177
Component Property Page ..................................................................................................... 178
Create New Library Dialog ................................................................................................. 179
ECO Report Dialog ............................................................................................................... 180
Find Object Dialog ............................................................................................................... 181
Key Assignments Dialog ....................................................................................................... 181
Line Width Dialog .................................................................................................................. 182
Load Part Dialog (simple mode) ............................................................................................ 183
Load Part Dialog (advanced mode) ......................................................................................... 184
Load Selection Dialog ............................................................................................................. 185
Manage Library Dialog ........................................................................................................... 186
Merge Nets Dialog .................................................................................................................. 188
Net Property Page .................................................................................................................. 189
Package Print Dialog ............................................................................................................. 190
Pin Property Page ................................................................................................................... 191
Polygon Property Page .......................................................................................................... 192
Print Artwork Dialog .............................................................................................................. 193
Report Dialog ........................................................................................................................ 194
Save Footprint Dialog ............................................................................................................. 195
Save Template Dialog ............................................................................................................ 195
Select Color Popup ................................................................................................................ 196
Text Property Page ................................................................................................................. 196
Track Property Page ............................................................................................................... 197
Update Components Dialog ................................................................................................... 198
User Preferences General Page ............................................................................................... 198
User Preferences Display Page ............................................................................................... 199
User Preferences Layer Colors Page ..................................................................................... 201
User Preferences Routing and Spacing Rules Page ............................................................... 202
Zoom To Coordinate Dialog ................................................................................................. 204

Shortcuts ................................................................................................................................. 206

Key Assignments ..................................................................................................................... 208
Overview

Introduction

Thank you for your interest in PCB123. With Version 3, Sunstone has fulfilled its commitment to create a PCB design system that delivers superior productivity against other systems regardless of price. In one program, it delivers all the tools needed to complete a printed circuit board from concept to manufacturing.

Version 3 is a culmination of a 1 ½ year effort to streamline the entire PCB design process and has yielded a system that is quite unlike any other. PCB layout and schematics not only reside in the same application, but they are considered two isomorphic views of the same database. A netlist is no longer the interface between the two applications. As you create a schematic, you are creating a PCB. If you don't want to bother creating a schematic, then just create your board and let PCB123 create the schematic for you. Data edits are bidirectional with either side updating the other. Most of the synchronizing is done transparently but some edits required that special facilities be created to ensure both the logical (schematics) and physical (Layout) views stay isomorphic. In fact, loading a pre-version-3 PCB123 file into version 3 will result in the automatic generation of a companion schematic. Likewise, starting a new design either using a netlist from another system or simply creating a PCB from scratch by manually placing new components and drawing the traces will result in a schematic that is automatically created and is continuously updated as the PCB is updated.

We are extremely pleased you are reading this guide because it is jam packed with tips and techniques to maximize both your productivity and the quality of your PCB.

Software features out of the box

With the PCB123 V3 software you installed, you will be able to create a multi-page schematic by placing symbols that represent complete parts or sections of a part such as a gate, and then hooking them up with wires and special port symbols such as power and ground symbols.

Locating parts has never been easier. The part database is structured as a taxonomy, which can be navigated in a tree or can be searched by typing in search criteria. Parts can contain a rich set of properties, all of which are searchable. This means you can search for a part by number (distributor or manufacturer) or by some characteristic such as “4.99K, 1%”.

Placing a part so its pins touch the pins of a different part will automatically generate wires between the pins as long as they are allowed by the rules. For instance, a power pin cannot be tied to a ground pin, and an NC pin (no connect) cannot be tied to anything. Wires can also be added by clicking on a pin or an existing wire (adding a junction) and then simply moving the mouse to the other pin/wire/junction and clicking again. This will automatically draw a wire that obeys the expected aesthetics. The auto-wiring feature is invoked when an already-connected symbol is moved or rotated. Even high-pin-count symbols are neatly rewired as they are moved around. Even though auto-wiring is always in effect, you can also digitize corners exactly where you want them.
At any point, you may flip over to the PCB view and you will see the new parts and their wires located at the system origin. You can select AutoPlace to rapidly position parts on the board. AutoPlace will attempt to minimize the overall interconnect distance between components while also locating the components in a reasonably aesthetic pattern. You can then examine and adjust the placement to perform local optimizations and enforce any special placement requirements that you may have.

At this point, you may choose to autoroute the design. The autorouter will first escape surface mount pins to a via (a plated hole) to allow the router unrestricted access to all routing layers and or to connect to any internal plane layers. After fanout, the main autorouter will engage which attempts to add traces between objects that must be connected together. In many cases, the autorouter may not complete 100% of the connections or it may do so by using a topology that is unacceptable to you. In these cases, or in the event you just want to route the board completely by hand, there are very efficient editing tools to complete the task.

While manual routing or performing other PCB editing tasks by hand, a Design Rule Check (DRC) is performed in real-time and you will see DRC error markers come and go as you create and fix DRC violations. Whether autorouted or manually routed, you can choose at any stage to run the Gloss command to remove redundant corners, segments, and vias.

You not only have the ability to designate internal plane layers, but you can also choose to view them as positive or negative images. In addition, you may wish to fill in unoccupied areas on the routing layers of the board with copper planes. These copper planes are typically assigned to a net, such as GND. Pins assigned to the same net, which appear inside the region, will be connected to the copper plane. It does so not with a direct hit between the plane and the pin, but through thermal reliefs to aid soldering.

Nomenclature may be added to the board as copper or on silkscreen layers using text strings or drawn polygonal features.

A full DRC (Design Rule Check) can be run on the board when it is believed to be complete. A full DRC will check for violations of net rules such as adherence to minimum spacing requirements, shorts, and opens. It will also check for possible manufacturing problems such as drill excursions into SMD features, silkscreen in holes or on SMD features, and even isolated pins in copper pour regions.

It is recommended that a final visual check of the PCB be done with the 3D viewer. The photorealistic rendering system displays the board in a perspective projection and allows the board to be viewed from any angle in 3D space. It is good for identifying problems that the DRC engine might otherwise deem valid. These may include problems with drill sizes, holes where there should be none, missing holes, plated/unplated hole problems, and unreadable silkscreens.

At any time, either the schematic or the layout may undergo component or net additions, deletions, or changes and the other will be synchronized. This makes tasks such as adjusting the number of decoupling capacitors based on available real estate or routing density a snap.

Finally, when the design is complete you can press the Order button to begin the process of getting your circuit boards manufactured.
User Extendibility

In addition to the core system, PCB123 Version 3 now includes a Plugin interface that allows point tools to be added at anytime. Sunstone is making available, free of charge, a Plugin Development System for anyone to download. The Plugin SDK exposes the complete PCB object database for random access and allows for the processing of virtually all PCB123 software events. In addition to editing PCB objects, developers will be able to call upon code libraries such as the 2D geometry library (over 800 analytical operations on 2D objects) and the spatial database and DRC engine.

The SDK includes a 120-page manual, code libraries and the header files that describe the various PCB123 objects and resources. There is also a Microsoft Visual Studio Application Wizard that builds a working Plugin complete with TODO comments and example code for many scenarios. It also includes the full source code for a couple of the actual Plugins that are shipped with Version 3. These include a connection density analyzer which plots estimated route density as a function of color in a separate window and a Plugin that minimizes the angular deflection of all route corners and therefore also shortens route length. It does so without causing DRC violations. We hope that over time many creative and productive Plugins will be developed by you, the user, to share or sell to the PCB123 community.

One eye on the future

First, a quick note on some capabilities that didn't quite make it into this initial release of the new schematics facility but are certainly expected in an upcoming release. The two main capabilities are bussing and hierarchical designs. The two features share the common requirement of a bus pin/port object. The hierarchical capabilities will be implemented as sub-circuits which get flattened immediately for logical/physical synchronization. You will be able to place one or more instances of a pre-existing schematic page from any design into another schematic page and it will be represented as a symbol with pins that correspond to special ports placed in the hierarchical page.

We would like to point out some of the less obvious benefits that the full integration of schematics and layout make possible. Aside from the fact that a netlist is a static snapshot of a schematic at a given time, a netlist also strips all the information from the schematic other than the net graphs. We can point to one example where some of the semantic information from the schematic has been confidently and successfully hoisted into V3 layout and that is with the automatic assignment of large net widths for nets that are attached to voltage and ground port symbols. This indicates there may be convenient, logical paradigms for visually documenting and conveying special net and component rules right in the schematic that layout will honor.

Proximity information is another example of information that is lost in a netlist. It is possible to bias the autoplacer with symbol proximity data from the schematic. Along those lines, if a schematic sub-circuit was placed and even routed in layout then that sub-circuit could then be replicated via copy and paste or through hierarchical instantiation, and the new circuit’s physical layout would be exactly relative to its original source circuit including any routing.

The benefits can work the other way too. If a SPICE simulator is hypothetically bolted on to PCB123, the simulator would have access to the exact characteristics of the routing topology of a net by simply asking layout and getting an instantaneous response. Signal integrity problems could not only be identified much
faster than using separate tools but the iterative process of simulate-analyze-tweak could actually be replaced by driving special tools in layout with the simulation results that take corrective action on the routing topology. This hysteretic optimizing process could be run until it converges on correctness or raises a red flag with a high degree of confidence that a real problem exists.

Our Commitment

Our business model has worked quite well and you can see with this release our commitment to delivering state-of-the-art productive tools to you. We are really excited about the technology and see the possibilities limited only by yours and our imaginations.

With the release of the Plugin SDK we have taken another giant step toward openness and look forward to the prospect that you, the user, may now directly contribute to the product and help guide it into niches that may otherwise be under-served.

This especially includes the development of translators to and from the various other CAD systems that you use and know well. We understand you may have a substantial investment in some of these systems and we are not looking to replace them, only augment your company's design capacity and flexibility in the most economical way. We will always provide this ever-improving software for free to equip all engineers and designers with the tools they need to at least try their ideas at work or at home without starting in red ink.

Our revenue is realized only when your ideas are realized.

Happy engineering, from Sunstone's PCB123 division!
Part 1 - Getting Started

If this design were music, it would be a Bach Canon
Introduction

Who are you and why did you download this software? Are you an electrical engineer with a new idea for a product? Are you perhaps an expert PCB designer who was just handed a netlist, board outline, and a set of datasheets and requirements? Maybe you are a hobbyist with a soldering iron and a burning need to bring to life that cool circuit you found in a magazine.

Your software requirements and your domain knowledge may vary greatly but one thing in common is the need for designing and ordering circuit boards that work the first time.

Rather than penalize the experienced engineer or designer with a tool aimed at the lowest common denominator, and rather than traumatize the novice with endless data prep requirements and rigid methodologies PCB123 provides a great deal of flexibility to get started and many powerful features to get you to the finish line.

Starting out is easy but reading this guide will allow you to unlock the full power of the software and greatly increase your productivity. It not only explains the operation of the software but delves into process, ergonomics, and tricks of the trade.

There are many automatic features that are always present and available but they are designed so as not to be intrusive or restrictive. For instance, when routing a trace you will instantly get a visual clue if you create a short to another object or violate minimum spacing requirements. You will not be prevented from continuing but a small marker is left behind to remind you that there is a problem at that location. The marker is removed as soon as the violation is cleared.

The software does not second guess your intent. The system performs an extensive design audit before you submit the design for fabrication but you never have to meet some threshold of system-determined correctness to order boards. It will not however allow you to specify or design with impossible-to-meet criteria such as routes that are less than 0.005" thick.

One of the most important reasons for reading this guide before you start an actual design is to prevent bad habits from forming early by being aware of all the different ways in which you can perform common editing tasks. For instance, though you can change to a new layer by clicking on a screen element you will find that it is much more efficient to type the layer number instead. There are many time-saving shortcuts that are less than obvious by just looking at the screen. Once these shortcuts are learned, you will find that you can operate at a continuous fast pace with one hand on the keyboard and the other
clicking the mouse buttons and spinning that mouse wheel. If your mouse doesn't have a wheel or something thereof, we recommend you obtain one to operate at maximum speed. In no time, you will have memorized the few keyboard accelerators and mouse techniques that comprise the vast majority of editing operations. After that, editing and other operations such as panning and zooming become so automatic that you will not even be aware of doing them. In fact, if you use other CAD systems you may become acutely aware of the extra click or key-press that the other system may require.

The vast majority of editing operations are done with just a few keys and mouse actions. However, there are also wealth of less common operations that are available by clicking the right mouse button to invoke context-based menus.

Before we get into specifics, you may find it useful to have PCB123 running to so you can orient this guide against the actual software. If you run it, you should close the splash screen and the New Board Wizard that appear on startup.

Just to make sure we are starting from the same point, you should see a screen that looks similar to the image on the left.

Click the File menu and select New, or click on this toolbar button.

This will invoke the Board Configuration dialog which will be discussed later. For now, just press OK to accept the defaults.

If your screen looks similar to the picture on the left then you are ready to explore the design space.
Screen Layout

More than one project at a time can be loaded into PCB123 and data can be copied between them. Each project loaded will appear as a tab at the top of the main window. Only one project can be active at any time and if more that one is loaded, the active project can be changed by clicking on its tab. Tabs from a 2-project session are shown left.

Each project will have a minimum of four different views that can be displayed in the main window. They are Layout, 3D View, BOM (Bill Of Materials), and at least one schematic page. You can navigate between the views by clicking on the appropriate tab at the bottom of the main window. The strip of tabs is shown left.

As you change views, different toolbars will come and go as different views are activated.

To the left of the main window is the Edit Panel. It too contains a strip of tabs on the bottom as shown to the left. The Display tab is quite dynamic, changing its contents not only when the main window is switched between schematics and layout, but when certain types of objects are being edited, context panes will be added and removed that provide various options for the selected object or tool.

The Project tab contains several report generators and a top-level navigator in the form of a tree that allows you to drill down into the schematic and layout data in minute detail. It also contains many top-level settings and defaults such as title block parameters that can be set for a project.

The DRC/ERC tab contains a live listing of all the current design violations in the project.

As you can see, tabs are used extensively to display applicable tools and information and hide irrelevant tools and information for a given task.
Common functionality

Common functionality really refers to the common operations between the schematic view and the layout view. The 3D view and BOM view are highly specialized views that demand very little face time. By far, the design views will receive most of your attention so we will focus on the common functionality between them.

A good deal of your time in the software will be spent moving the mouse around, poking the keyboard, and clicking on things. We hope we have created a fairly intuitive user interface but even so, we should cover some basic behavior.

Pan and Zoom

Because a PCB or a schematic can have a large number of objects packed into a small area it is next to impossible to do any work while viewing the entire extents of the data. Instead, you will want to be zoomed into and focused on a small area. Consequently, you will be doing a lot of panning and zooming. These operations are so common they become reflexive actions - you do not think about it, you just do it.

First, under the View menu a Zoom submenu lists all the commands pertaining to zoom. The first four commands pertain to the view scale or zoom factor.

Zoom In and Zoom Out magnify and... um... un-magnify the view. Using the menu commands or their keyboard accelerators, PgUp and PgDn, scale the zoom factor based on the ubiquitous Golden Ratio beloved since eternity. Using the mouse wheel scales the zoom factor by smaller increments.

Zoom All adjusts the zoom factor and view center so the entire board outline fits within the viewport.

The Zoom Actual Size command, which is not accelerated, relies on your display hardware to accurately report the number of pixels per inch of display and uses that factor to scale the view accordingly.

The next four menu items pan, or scroll the view north, east, south, and west. The pan distance using these menu commands or their keyboard accelerators is the width or height of the window minus a comfortable overlap figure. Besides the obvious reasons for the overlap, it is interesting to note that panning by a full window can actually produce a “lost” sensation at times.

The Redraw command will simply refresh the screen. It also rebuilds certain internal lookup tables and in rare situations, when you add or modify objects so their new extents are double or triple the size of the board you may find that you cannot pick them until a Redraw is performed or any number of other commands that trigger a regeneration of the tables is performed.
Drilling into submenus is probably not the most efficient way to navigate around a design but Microsoft recommends standard practices and we try to oblige. The menu also shows the keyboard accelerators for the zoom commands.

There is also a Zoom toolbar on the right edge of the application. It is lacking buttons for pan but there are stock scrollbars for that. The button labeled ‘G’ is not actually a zoom command at all. It is a way to invoke the Grid dialog using the mouse. As with most toolbars, you may hover your mouse over a button to pop up a tooltip for a brief description of the button’s operation. The Zoom Toolbar contains some of the lesser used commands such as Refresh Screen, Zoom All, and Zoom Actual Size. The Zoom Toolbar is for new users who don’t read the manual. You, however, deserve better.

Most likely you will find that using the keyboard to perform zoom operations is superior to the menu and toolbar. The numeric keypad is dedicated to the zoom functions. The chart on the left shows the keypad assignments for the different zoom commands.

By far, the best way to pan and zoom around the design canvas is to use the wheel on your mouse if it has one. The wheel can accomplish both pan and zoom. To zoom in, roll the wheel forward and to zoom out, roll it back toward you. The pan behavior is a byproduct of how Zoom In and Out work with the wheel. Wherever the mouse is pointing in the design window becomes the new viewport center. Zooming in and out actually does a pan so that whatever you were pointing at is now in the middle of the screen. The closer the mouse already is to the center, the finer the pan behavior. You point the cursor in the direction you want to pan and spin the wheel. It becomes very natural in just a short time.

There is one additional behavior to mention. If you hold down the Shift key while moving the mouse in the main window then the window will smoothly pan in real-time.

### Names and Numbers

Objects such as nets and components require unique names. Other objects may be optionally named or, in the case of PCB component pins, if they are named the names must be unique just within the component. In general, most objects that require a name will in fact have a name auto-generated during interactive design. Object names will be upper case and they currently will be restricted to the ANSI character set. There are no restrictions on name lengths.

In general, if you name an object do not include any spaces or other non-printable characters in the name. Whitespace, ‘,’ ‘&’, ‘[’, and ‘]’ are reserved characters and will be replaced with an underscore ‘_’.
Unless absolutely necessary, names should not begin with numerical digits. There is nothing inherent in PCB123 for this, but names that begin with numbers may generate XML validation errors in 3rd party Plugins or future PCB123 facilities.

The basic unit of measurement internal to PCB123 is 1/10,000,000\textsuperscript{th} of an inch (one ten-millionth). This ensures that no conversion routines are required when switching between metric and imperial user units as the round off is insignificant. It does however mean that the extent of the world coordinate system is limited to +/- 200 inches as coordinates are stored as 32-bit integers. A 400-inch domain is more than sufficient for PCB boards and schematics but be careful if you intend to use the software for other purposes.

When you supply values for physical parameters and coordinates, the values you enter are assumed to be in whatever the current system units are in. Additionally, four suffixes may be used when entering numeric data that allow an absolute unit basis to be specified. The table below lists these suffixes and shows examples of their use.

<table>
<thead>
<tr>
<th>Suffix</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>in</td>
<td>Inches</td>
<td>0.109in, -0.00002in</td>
</tr>
<tr>
<td>mil</td>
<td>Mils (1,000\textsuperscript{th} of an inch)</td>
<td>109mil, -0.02mil</td>
</tr>
<tr>
<td>cm</td>
<td>Centimeters</td>
<td>0.27686cm, -0.0000508cm</td>
</tr>
<tr>
<td>mm</td>
<td>Millimeters</td>
<td>2.7686mm, 0.000508mm</td>
</tr>
</tbody>
</table>

Anywhere the system asks for a physical value such as the Grid dialog box or a pad size form, you can perform local overrides of the current system units by supplying the appropriate suffix with the value.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>abs(n)</td>
<td>Absolute value of n</td>
</tr>
<tr>
<td>acos(n)</td>
<td>Arccosine (in degrees) of n</td>
</tr>
<tr>
<td>asin(n)</td>
<td>Arcsine (in degrees) of n</td>
</tr>
<tr>
<td>atan(n)</td>
<td>Arctangent (in degrees) of n</td>
</tr>
<tr>
<td>sin(deg)</td>
<td>sine of (degrees) deg</td>
</tr>
<tr>
<td>cos(deg)</td>
<td>cosine of (degrees) deg</td>
</tr>
<tr>
<td>tan(deg)</td>
<td>tangent of (degrees) deg</td>
</tr>
<tr>
<td>log(n)</td>
<td>Natural log of n</td>
</tr>
<tr>
<td>log10(n)</td>
<td>Base-10 log of n</td>
</tr>
<tr>
<td>log2(n)</td>
<td>Base-2 log of n</td>
</tr>
<tr>
<td>sqrt(n)</td>
<td>Square root of n</td>
</tr>
</tbody>
</table>

Any physical value can be entered as an expression so long as the terms are all in the same units. Expressions obey the C/C++ (normal) rules of operator precedence and associatively. Besides the arithmetic operators +, -, *, /, \%(mod) the table to the left lists the functions that can be used in an expression.

The ability to use expressions can come in real handy when creating footprints since most mechanical drawings of parts use relative dimensioning.

The following table shows several examples for converting between polar coordinates and Cartesian coordinates.

<table>
<thead>
<tr>
<th>Objective problem</th>
<th>What to type (in red)</th>
</tr>
</thead>
</table>
| To locate an object at 15 degrees on a circle that is centered at 0,0 and has a radius of 2 inches. | X = 2in*cos(15)  
 |                                                                                   | Y = 2in*sin(15)            |
To locate an object at 15 degrees on a circle that is centered at 4 inches, 2 inches and has a radius of 2 inches:

\[
X = (2 \text{in} \cdot \cos(15)) + 4.0 \\
Y = (2 \text{in} \cdot \sin(15)) + 2.0
\]

**Selection**

**Important:** PCB123 differs from many applications in how it treats a mouse click, where a click means pressing and releasing the left mouse button. In many popular applications, clicking on an object selects that object and waits for some operation to be performed on it.

In PCB123 a click almost universally selects the object and immediately assumes a move operation on it – it is essentially tied to the cursor until another click releases it. If you hold down the Ctrl key when clicking on an object then it behaves like other applications and simply performs a select on it. In fact, multiple objects can be clicked and selected (or deselected) by holding down the Ctrl key while clicking objects.

The way other applications accommodate an immediate move operation on an object is with a mouse drag, which involves pressing the left mouse button and moving the mouse while the button remains depressed. In most cases, PCB123 will accommodate that behavior too.

The reason for the departure from standard click behavior is for sheer efficiency. A schematic document and especially a PCB document may contain many objects – tens of thousands in some cases. For this reason, any paradigm that shaves a click off an operation translates into a large cumulative gain in productivity. The details of the efficiency gains are covered under Object Editing later in this document.

As you move the cursor over objects in the design windows you will notice they become temporarily highlighted. This is just a visual acknowledgement of which object will be selected if you were to click on it. If you let the cursor hover over an object you will get a tooltip displaying various properties of the object. The tooltip is translucent so as not to fully obscure objects beneath it. The image to the left shows a highlighted wire and its properties in a tooltip.

Holding the Ctrl key down while clicking on objects performs a disjoined selection. A continuous selection can be performed by sweeping out a rectangular area which involves pressing the left mouse button down on a spot that contains no objects, dragging until the rectangle encompasses the objects you wish to select, and then releasing the left mouse button. Hold down the Ctrl key while doing so toggles the selection state of the objects.
Objects are not automatically selectable. Objects that are set to invisible, either by object type or layer visibility will not highlight, nor are they selectable.

Additionally, the Layout view has a toolbar, shown at right, which allows the selection of PCB object types to be toggled on or off. This can be very useful to prevent the accidental selection of some object types while performing certain tasks. For instance, while routing traces it can be a nuisance to miss picking a track by just a little and picking a component instead because it will begin moving the component off of any existing routes to the component pins. Of course you can always cancel the operation.

**Context Menus**

Pressing the Right Mouse Button (RMB) will always display a context menu populated with commands that are applicable to the current selection. If nothing was selected then an implied selection is carried out on the object the cursor was pointing to when the RMB occurred.

Like the tooltip, the context menu (shown left) is translucent to prevent objects underneath the menu from being obscured.

If several different types of objects are selected, the context menu may contain submenus for commands that operate on the different object types. Submenus are indicated by a right-pointing, right-justified arrow and will be displayed when the menu item is highlighted.

In addition to the mouse, the arrow keys may be used to select different menu items.

While a menu is active, all other activity in PCB123 is suspended until a menu selection is made or the menu cancelled.

**Commands**

PCB123 supports most of the common Windows editing commands such as Cut, Copy, and Paste. Currently, these commands are specialized to perform data interchange only between PCB123 documents.

Undo and Redo is fully supported in PCB123 with one caveat: you cannot undo or redo a schematic edit while in Layout view, and vice versa. Undo and redo are unlimited as long as you remain in either view but as soon as you flip to the other view then undo and redo are flushed.
Starting a New Design

As previously mentioned, PCB123 attempts to accommodate a wide audience of which everyone, regardless of experience, needs to design a PCB. Your project may be using the most common off-the-shelf components and may need to fit in a standard housing but you still need to design that one custom part - the PCB. Because it is custom, the burden falls largely on you to ensure its correct design and operation. Good design software and good design practices will greatly aid you in this effort.

PCB123 allows for the specification of rules that can ensure manufacturability and design continuity. Some of these rules can even be used to guarantee adherence to certain standards bodies’ rules such as the UL requirements for minimum spacing between nets of certain voltage potentials.

There are issues associated with PCB design that can be extremely complex and are beyond the scope of this guide. As a rule of thumb, if you need to incorporate the electrical or temporal effects that the PCB has on nets as a function of the routing topology (transmission lines, crosstalk, emissions) then you may require more sophisticated (and very expensive) layout tools than PCB123 unless you are at an expert level in such matters. We do not have a button that ensures FCC compliance for your design but we do have a button that ensures everything is hooked up where it should be and nothing is shorted together. We also have a facility so you can add a button to ensure FCC compliance of a design by writing a Plugin that operates on the PCB data in any way you desire.

PCB123 will automatically detect and report many of the problems that may arise during the design process. Even so, there are many places where the software cannot second-guess intent. How much checking the software will do, and even how much time the software can save you by performing some tasks automatically can be a direct result of how much information you supply PCB123 at the start of a design. First, let us look at the gross tasks involved in designing a board in PCB123.

Overview of PCB design tasks

Here is a quick list of the different tasks involved in designing a PCB. For now, it ignores how the design data is acquired and input into the system. It assumes you have already designed a schematic in PCB123 or input a netlist from another tool. This is covered in detail in the next section.

The tasks are generally in chronological order but your design or your favorite techniques may differ slightly. The basic tasks are:

- Using either the **New Board Wizard** or the **Board Configuration Dialog**, specify any manufacturing requirements for your design. These include the number of layers in the design, whether you desire soldermask and/or silkscreen on your boards, and certain non-standard requirements such as board thickness or copper weight. Some features are mutually exclusive and will be clearly indicated as such. All the configuration options can be changed at any time and are only finalized when ordering. However, some options can greatly impact the design rules which, if going from a less to a more restrictive set of rules, can be much more difficult to deal with in the later stages of a design. Examples of this would be specifying less layers than actually routed on, or changing from standard 1oz. copper to 2.5oz copper which requires greater spacing.
and feature sizes to properly etch the thicker copper clad. Incidentally, if you are new to this, then the concept of “adding layers” or “deleting layers” may be confusing to you. A PCB may have more than a front layer and a back layer. Two or more circuit boards can be independently etched and then carefully aligned and glued together, one on top of the other with an insulator between them. These boards are then drilled as one and the holes are then plated. The plated holes allow copper from one layer to be connected to copper on different layers. PCB123 handles 2, 4, and 6-layer boards as a standard process. For boards with more layers than that, you are requested to send the design to Sunstone for quick evaluation. You will then receive back the same design with the desired number of layers enabled and an adjusted price quote.

• After the basic board configuration has been specified, ensure the size and shape of your board outline is correct. Closely associated with this is to correctly define and position any mounting holes or other hardware and to position any fixed electrical devices such as connectors, LEDs, and switches. In short, make sure the board will fit in a chassis or enclosure, that it can be secured in place if needed, and that any lights or dials that mount on the PCB are visible or accessible through openings in the enclosure.

• Identify the electronic components that your design will use and either map them to “footprints” from our supplied libraries or create any custom footprints as needed. A footprint is a reusable stencil that defines the pattern of copper pads, drill holes, and silkscreen information such as the component outline and a reference designator (name) for the component.

• Place all the components on the board so they don’t overlap or violate any special criteria that you may have such as height restrictions or proximity requirements such as locating filters near connectors or other sources. Once the critical mechanical and electrical requirements have been considered, the remaining components typically want to be placed in a way that minimizes the overall interconnect distance between components. Some components may require additional room around them to ensure there all the pins can be connected with routes. It is also desirable to place the components on as coarse a grid as possible. If all the pads are on a regular grid, then routing the board is much faster and easier because you will not have to make as many small bends around pads.

• Distribute power and ground. Ensure the supply pins on semiconductors are properly decoupled by placing decoupling capacitors in close proximity to the supply pins. Bulk decoupling or filtering may want to happen close to the connector pins or devices supplying power to the entire board. If the board will have dedicated copper plane layers for power and ground then those nets should be assigned to the proper plane layers. If two or more nets are to share a plane layer but still be isolated from each other (called a split plane) then the isolation needs to be manually drawn on that plane layer. Surface mounted devices may need to have their power and ground pins routed out to a plated-through-hole that ties the surface pad to a plane layer. If the power and ground is to be distributed with routes and not plane layers, then they should be routed using traces that are wide enough to handle the required amount of power. Redundant routing in the form of a matrix may also be desired at the board extremities or at some regular interval to keep voltage levels uniform across all devices.
If any signal nets have critical requirements such as complimentary routing or matched/critical length, or they require special attention such as pin-to-pin route sequencing, then these nets should be routed before other nets.

Route all the remaining interconnections. This can be the most time consuming task in PCB design and there are many strategies that can be used. One strategy that is almost always a winner is to assign a direction to a routing layer. There can be a lot of flexibility in this, but by routing mostly vertical on one layer, and horizontal on a different layer, you will be much less likely to create blockages for layer routing. The layers will have a “flow” to them and changing between layers by using vias as the routing direction changes can typically utilize the layers much more efficiently. If your design is moderately to very dense, and a sizable percentage of it uses surface mounted devices, then you can expect that the surface layer(s) will be consumed by short routes that escape a surface pin to a via (a plated drill hole) to effectively allow that pin to be routed on any layer. This process is known as Fanout routing and should be done before bulk routing. It is not only a time saver for bulk routing but can be an early indicator of problems. If you cannot fanout a device then you most likely will not be able to completely route that device without making more room around it, adjusting its location, or adding more routing layers. It is much better to discover these kinds of problems earlier than later. Moving a component after it has been routed hurts more than moving it before it was routed. Additionally, you may have struggled to route most of a dense design only to find that you need two more routing layers to complete the rest of it. Knowing this up front would have made the routing easier right from the beginning.

When all routing is finished, you may wish to improve power and ground distribution or provide ground shielding by flooding regions of the board with a copper plane that is tied to a power or ground net. Any object in the region that does not belong to a net, or has a net other than that of the copper region will be automatically isolated from the copper region. These regions are collectively known as copper pour.

The next task will be to perform a Design Rule Check (DRC) and address any problems that it may have uncovered. The DRC will report on shorts, opens, and violations of manufacturing rules such as minimum spacing between objects of different nets, and pad size vs. drill size violations. Some reported violations such as problems with the silkscreen can be ignored at this stage but certainly DRC violations on routing layers and plane layers should be addressed at this stage.

If your board is to have silkscreen on it, then you will want to place the component reference designators so they can be seen after the board is populated with components. This usually involves moving the reference designators outside the component outlines but close enough to the correct component to prevent an ambiguous reading of which name belongs to which component. You may also wish to add other nomenclature to your design such as the name of the board, a company logo, jumper configurations, or other wording or drawings. Nomenclature can be added to copper layers and silkscreen layers. If text is added to the bottom routing layer or bottom silkscreen then that text should be “mirrored” so it will read correctly when the finished board is flipped over in your hand to observe the back. After adding your nomenclature, you will want to run an additional DRC to make sure you didn’t short anything together or create silkscreen on any pads.
• As a final check, you may wish to view your design in 3-D. PCB123’s photorealistic 3-D viewer may allow you to visually spot problems that DRC assumes are correct. Because it displays the board in perspective and from any viewing angle, certain errors such as wrong/missing drill sizes or unreadable silkscreen text tend to really stand out compared to looking only at the orthogonal 2-D view. Once you are satisfied that it looks correct and DRC is reporting no errors then you can press the Order button which guides you through the order process. Depending on the board complexity and the time of day you placed the order, you will receive your boards anywhere from 2-5 business days later.

Data Requirements

PCB123 provides a host of tools to complete all the different tasks but many of them are highly dependent on how you specify or acquire the design data. This can be broken into two methods: with a schematic, or without a schematic. To be even more specific, your design has been “captured” inside a schematic program, or your design resides on a napkin, in a picture, or in your head.

It takes effort to capture a design with schematic software but this effort usually pays great dividends. Unless your design is extremely simple and involves only a few parts, it is strongly recommended that you create an electronic schematic of the design before attempting layout. PCB123 provides a schematic capture facility that works in lock-step with its layout facility. In addition, PCB123 also is capable of reading and importing netlists that have been generated by other schematic capture programs.

In case you are unfamiliar, a netlist is a file that lists all the components used in a design and defines how those components are connected together. A component, or part, is usually specified in a netlist as a name/type pair. A component name, or reference designator, typically consists of an alpha prefix and a numerical suffix such as R1 or U15. Reference designators must be unique in a design. The component type, or part type, might appear as anything from a proprietary description such as RES_33-OHM_1/4W to a manufacturer’s part number such as SN74LS00N. In some cases, the component may also be augmented with additional data such as the name of a standard package or footprint for the part. After all, a 33-ohm, ¼ watt resistor can come in many physical sizes and shapes. The connectivity information in a netlist is usually specified at the “net” level. A net has a name such as +5V, GND, or RESET, and has a list of one or more “nodes” or pins in the form of R1.2, which reads “pin 2 of component R1.” A net with one node is legal but unusual. The vast majority of nets consist of two or more nodes. A net with two or more nodes inside PCB123 will be displayed as pin-to-pin wires that must be connected by a continuous copper path or plane.

Until PCB123 V3, a netlist was the traditional data transport from a schematic program to a layout program. For the first time, PCB123 eliminates the netlist as a data transport as long as the schematic was designed in, or converted to a PCB123 V3 schematic. A schematic designed in PCB123 V3 has a more intimate relationship with the layout than does a netlist from another program. Using a netlist from a different schematic program has the following disadvantages:

• A netlist is a static snapshot of a schematic at a particular moment in time. If changes are made to the schematic then a new netlist must be generated that reflects the changes. If the PCB layout is already in progress, then the PCB must be synchronized with the new netlist. Luckily
there are facilities for this but if the schematic was created in PCB123 to begin with, any change to it is instantly reflected in the layout and vice-versa.

- A netlist is not certain to contain part references that are known to PCB123. These unknown references must be resolved. This can actually be done interactively while reading the netlist because whenever an unknown part type is encountered, you will be presented with a browser that allows you to find and graphically verify a footprint from a library to use for the part type. A native PCB123 schematic, on the other hand, will not even allow a schematic symbol to be placed on a page unless it has already defined the PCB footprint it uses, and verifies that the footprint can be found in the library.

- Very little context from the schematic is retained in a netlist. That a net is connected to a power or ground symbol is not reflected in the netlist. If it were, then that net could automatically be made to use fatter routing widths. A PCB123 schematic automatically uses that kind of contextual information to adjust the layout rules as needed. A netlist is also missing proximity information or any data that indicates how the schematic was organized by page or through a hierarchy. This information is expected to be widely used in future releases of PCB123 for such things as grouping, replication, and as hints to the autoplacer.

- Changes initiated from the PCB design, such as adding a decoupling capacitor, must be manually performed over in the other schematic program. If this is not done then there will be a mismatch when comparing the two databases and performing another netlist comparison will run the risk of undoing the changes, in this case deleting the newly added decoupling capacitor. As mentioned before, PCB123 always synchronizes changes from either side to the other. There is nothing to remember or relay to someone else.

Despite the disadvantages listed above, using a netlist from another system is far superior than designing a PCB without one. Most schematic programs have some facility for performing an Electrical Rules Check. This can flush out logic errors very early by detecting such things as outputs connected to outputs or anything connected to a pin designated as N/C (not connected). Many schematic programs also host simulators that can verify the correct operation and timing of circuits. Additionally, capturing your design in a PCB123 schematic or in a netlist allows automated tools in layout such as the autoplacer and autorouter to be used. These tools can save hours or days of labor or they can also be used for early decision-making and early identification of problems.

Though PCB123 allows you to create your PCB design using piecewise primitives such as individual pads and line segments, it is not recommended except in the most trivial designs. These designs miss the benefits of context checking and many powerful automatic tools.
The New-Board Wizard

When you first run PCB123, you will first see either a “Latest Information” window or a “Checking for Updates” window, both that assume a connection to the internet. If the computer is not connected to the internet, then a message indicating that fact is displayed.

The “Latest Information” window is a small web page that is periodically updated to reflect new services, promotions or important notices.

Every so often, the software checks if a new version is available for download. If a new version is available, a summary of new features and bug fixes is presented along with the option of downloading and installing the new version. If you continue to run with older software, you will be asked to update about every tenth time you run the software. You can always check for, and install new updates yourself from the PCB123 menu.

After dismissing the “Latest Information” or “Checking for Updates” window, you will be presented with the first step of the “New Board Wizard”. If you want to load an existing design or just start fiddling with the software, you can click the Cancel button or press the Esc key to dismiss the wizard.

The New Board Wizard is primarily for new users but many find it useful to begin a new design no matter how long they have used the system. The Wizard has been carried forward from version 2 to version 3 virtually untouched, which makes it a highly board-centric feature. It never even mentions the word ‘Schematic’ once and PCB123 reserves the right to change the Wizard in the future to make it more Project-centric as other Project-level behavior is defined or requested.

The New Board Wizard has 4 pages of which the first one is shown below:

The Board Name field is essentially the title of your new design and can be anything as long as it is not empty.

The Netlist File field is optional. If you have captured your design in a schematic program other than PCB123 V3 then you can either type the name of the netlist file or, more likely, click on the Browse button to invoke a standard Windows file dialog box that allows you to navigate to the file. If you do click the browse button, the ‘Files of Type’ will list all the netlist formats currently recognized by PCB123. The netlist will not be read until the wizard is finished.
Clicking on the Next button brings you to step 2 as shown below:

![New Board Wizard](image)

**STEP 2**

**What size board do you require?**

Define your board size

- **Width** [ ]
- **Height** [ ]

Or use a board template

- (none)
- null
- Full Ega
- ManualRouting
- ManualRouting
- ManualRouting
- Vellum
- manRoute
- one_square

You have a choice of defining your board size by entering a rectangular width and height for your board or selecting an existing board template supplied by Sunstone, a third party, or created previously by you. When selecting templates, you will get a postage-stamp preview of the template.

Templates most likely will contain a board outline but may also contain tooling holes, dimension lines, title blocks, and even components such as connectors. Because a netlist may define a component and a template may define a component there exists the possibility for a conflict. If they both attempt to load in a component with the same name, then the component in the template wins and will be used in place of the one in the netlist. A warning will be issued to that effect. The template component wins because it contains a more complete specification by including location, rotation, and board-side data.

Step three is where you specify the number of layers in your design and how those layers are used. It appears below:
In the **Number of Layers** section, you can select two, four, or six layers for your design. The default is two layers. Selecting more than two layers will enable the **Inner Layer Type** options. You may need more than two layers because of a mechanical requirements such as high component and interconnect density and unfavorable interconnect topology, or for electrical reasons such as cleaner power distribution and transient return paths, or to form transmission lines on critical nets.

You will have to specify which nets are assigned to the plane layers. Any pin belonging to the specified nets will tie to the plane at a plated hole.

This plane connection does not occur against a solid plane, but instead the plated hole has an isolation gap around it with “spokes” that bridge the gap at certain places. This isolation prevents the plane layer from sinking too much heat during soldering. It focuses the heat on the pin and allows for a clean solder bond.
Step four allows you specify additional board fabrication options as shown below:

Depending on the number of layers you selected in step three, you may or may not be able to check and uncheck the Soldermask button. A six layer board always has a solder mask.

Checking soldermask will enable the Silkscreen button. Checking silkscreen will not increase the price of your board unless you also choose to have silkscreen on the back of the board. This can be specified in the Board Configuration Dialog.

The PCB Thickness selection allows you to choose between two thicknesses under certain circumstances. By default your board will be 0.062” thick. That is a highly typical board thickness. If your board is only two layers, has soldermask, and is using standard 1oz. copper weight then you have the option of specifying a half-thickness board of 0.031”. You most likely only want to check this if your board is small in area because the thinner board material is susceptible to warping and curling, especially if there is an imbalance of copper clad from one side of the board to the other. You most likely will choose a thin board when weight is of primary concern or if a small amount of board flex is required.
The **Copper Weight** selection allows you to choose between two different copper clad weights, but only under specific circumstances. By default your board will have standard 1oz. copper weight but if your design has two layers, soldermask, and is using 0.062” thick material you will have the option of specifying a heavier 2.5oz weight copper clad. The unit measurement of weight represents the number of ounces of copper per some square area. The heavier weight is due to a thicker clad. The thicker clad can carry more current but it comes at the price of increased feature size. Trace width, pad annular size, and spacing requirements all increase by nearly double. The nominal copper line thickness increases from 0.007” to 0.013”.

Checking the **Lead Free** checkbox specifies that your design is to use the RoHS fabrication process. This option is free. If your design includes surface mounted components that have high pin counts, such as a BGA, then it is recommended that you check this box because the RoHS process results in superior pad planarity (flat surface).

When you click the **Finish** button, any specified board templates and netlists are loaded. You may be prompted to select which footprint to use for components whose part type is not found in any library or alias list.

A good deal of checking is performed while loading a netlist and if errors are found they will be reported but they will not prevent a design from being created. This may result in a partial load of the netlist so the error log should be checked carefully. It is possible that a netlist format emitted by some host system has changed slightly or does not conform to previously encountered formats. The netlist readers are designed to recover from encounters with unknown data and will attempt to re-sync on known data. Because of this, a netlist may load with errors but the errors are just benign data that the reader did not expect. In any case, check everything carefully when errors are encountered and we encourage you to send us netlists that may have tripped up the system so we can update the reader for that format.

---

### Starting Without the New-Board Wizard

You can skip the New-Board Wizard and just select File-New or click the icon in the main toolbar. This will present you with the Board Configuration Dialog which is a compact version of the New-Board Wizard. It too allows a netlist to be specified when invoked in response to File/New. The netlist field will subsequently be disabled if this dialog is invoked after a new design has been created. Only the board configuration options will be enabled and can be modified at any time.

### Results of Creating a New Design

When the Wizard completes, or the Board Configuration Dialog is dismissed, the system will have created a new design document with the requested number of routing and plane layers, the specified rectangular board or selected board template, and any components and nets specified by either the board template or the netlist.

If a netlist was specified and called in components and nets, then, one or more special schematic pages will have been created that contain an auto-generated symbol for every component on the board. The pins of these symbols may be tied to global ports with the same name as the nets on the board. You will essentially have a matching PCB123 schematic that you can use going forward, though you may also
continue using the original schematic and synchronizing any changes with a netlist. The last task that is performed after loading a netlist will be to run the autoplacer for an initial component spread. It is possible the autoplacer will warn you about the parts not fitting on the board if it is too small to contain all the parts as specified.

If no netlist or board template was specified when creating the new design document, then the only database objects created will have been a board outline and a single schematic page named “Untitled Page”. In this case, you can immediately begin designing in one of two ways: smart and safe, or quick and dirty.

- The smart and safe method has you switching to the new schematic page to begin your schematic design.
- The quick and dirty way has you manually instantiating components and routing them with the Freestyle routing tool.

Using either method will result in the complimentary view being updated as changes are made, but the information model is not completely symmetric between the two design methods. A maximum information model is created in both the physical (layout) and logical (schematic) views when objects are created from inside the schematic and not from inside the layout. This is obvious in the case of adding a component to the design. If done from the schematic, the component is most likely defined by a symbol or symbols (in the event of gates) that symbol may be richly attributed with component metadata such as value, tolerance, manufacturer's part number etc. This data will be reflected in such items as the Bill of Materials (BOM). The symbol will also have a reference to the specific PCB footprint to use.

On the other hand, a component that is instantiated inside layout can only auto-generate a corresponding schematic symbol. That symbol will appear as a rectangular box with pins sticking out of it. In the worst case it will only be endowed with a reference designator and a footprint attribute such as SOIC-14. You may instantiate two different resistors inside layout and correctly use them without regard to their possibly different values. This may result in a nearly useless BOM because the two resistors will appear on the same line item without any indication as to their values. The logical and physical views may be isomorphic in regards to electrical continuity, but they are quite different in visual appeal and accurate part metadata.

**Default Settings**

Whenever, and however, you create a new design, there are a number of software settings that obtain a default value. Most, if not all of these settings are stored directly in a design file when you save a design but how they get their initial values and how the values are subsequently changed is worth mentioning. These settings include display colors and visibility for layers and objects, the current working grid, angle snap grid, full-screen cursor display and a host of other settings. Though these setting are per-document settings, you can specify the values to be used as the default values when a new design is created. This is done through the User Preferences dialog.

This dialog contains several pages of controls to adjust various system settings. In addition, it contains buttons at the bottom that allow the entire collection of settings to be saved as a Preference File or a Preference File to be loaded. It also allows a new set of defaults to be established for new designs.
PCB123 Directories

When you install the software, whatever directory you specified for the installation is only where the program files were installed.

In addition, a directory tree will be created under Documents and Settings\All Users\Application Data\PCB123 on the system drive.

Depending on your system settings, this directory may be hidden from you. However, the PCB123 Installer has created a link file in your My Documents folder named PCB123 Files. Double-clicking on this file will open a Windows Explorer set to the above directory. You should see a list of files and folders similar to that show on the left.

Here is a brief description of the folders:

- **BackupFiles** - This folder is where the interval backups are stored. While you design with PCB123, a backup of your work is done every 10 minutes. The backup filename if the design name plus _Backup#, where # is 1 through 5. A revolving set of the last 5 backups are maintained, with #1 being the most recent. Additionally, if the software crashes, your design will be saved with _Crash appended to the design name. Incidentally, Sunstone has never had a report of lost work or a corrupted design file. Many safeguards are in place for just such an occurrence.

- **Lib** - This is where the system libraries are installed. Inside this folder, there is a folder named Custom, which is where any libraries that you create will be stored.

- **Designs** - This folder has been created for your convenience. You are not required to use it.

- **Preferences** and **BoardTemplates** - These directories store saved User Preference sets and Board Templates. There are features in the software that deal with these directories.
The Schematic/PCB Contract

A little background

PCB123’s purpose for being is to create a win-win situation for all parties. Customers download the free software and use it to design their products with a guarantee that, so long as they stay within the system parameters, the PCB’s will work as designed.

Because the software is developed and maintained on site, Sunstone can continually assess and react both to the customer’s needs and to its own internal needs. The business model has allowed a somewhat non-traditional approach to CAD software development. Because revenue derives from the use of the software and not the sale of the software, less emphasis is placed on software features and more on software throughput. We do not have to beat some competitor by providing a larger bullet list of esoteric features than the other vendor just to make a sale, but we do have to ensure that everyone from a novice to an expert can get through their designs to press the Order button. Because we know who will be processing and manufacturing the design (us) we can make certain assumptions where it might simplify the design process for PCB123 users.

This is all wonderful but there comes a point at which we (Sunstone) have to decide how far to take it. The software cannot outpace our manufacturing capabilities, and because customer support is free, we try to refrain from adding low-use features that clutter the software for novices or complex features that are deemed likely to generate a high volume of calls for instructions. There are both good and bad aspects to this method. An example is the new autoplacer tool just added; the only interface to it is the ‘Run Autoplace’ menu item. If you can click on a menu then you have met the threshold required to use it. This is simple for you and results in predictably less support needed by us. We could expose many of the internal settings that would allow the precisely control the autoplacer, but we know the performance gain that you may realize is too small to justify all the support calls that a dialog box with a bunch of strange settings would generate. Instead, we carefully pick the best values for the most typical boards. Either running the tool will help you, or you select Undo. Unless we can provide a set of options that will affect some major change in strategy, then it would probably not be fair to expose you to something that may burn your time trying to get some last magic adjustment to what is supposed to be a tool for “roughing in” an initial component placement. Win-win is the goal.

Generally, software that we develop will have wide appeal, general use, or transparent behavior. The new Plug-in development environment was, in a way, developed as the ultimate general use feature and a way forward for developing those esoteric tools that you may desire. You can write tools yourself, and it will probably be just a matter of time before a tool development community is forms around the new SDK.

This does not mean we aim at the lowest common denominator. In fact, it is just the opposite. Because our goals are different from those of a traditional CAD vendor, we sometimes arrive at solutions to problems that are quite different from the “traditional” way.

PCB123 Version 3 is a direct result of looking at the problem from a different angle. We know why customers will use the schematic; they will use it with the goal of creating PCB’s. Nothing was done to preclude the addition of simulators later on, but by admitting that we were creating a schematic capture
tool as a front-end for PCB design, we have allowed ourselves to color outside the lines of tradition and see if maybe there is a better way to do the task at hand.

Until now, PCB123 shipped a schematic capture application that was a third-party OEM product not directly developed or maintained by Sunstone. With every new release of PCB123, the schematic application fell behind in capabilities compared to the layout software. It also became a proportionally larger source of customer support issues. Serious consideration was given to just scrapping the schematic product, directing customers to freeware schematic applications and merely writing netlist importers for the layout software. The problem with that was our usage statistics did not support abandoning a schematic point tool - not everyone used PCB123 Schematic, but enough people did that we couldn't just drop the functionality. The clincher was this: we could find no free schematic program that we were comfortable recommending.

And so it was decided that a new schematic program should be created in-house and that we should aim it at being highly PCB-centric and, keeping with the model, should take advantage wherever it could to increase throughput for customers.

We like win-win situations and it seemed to make sense that we create the schematic application right inside the PCB123 application. We would be able to leverage the existing software framework and users would not have to be flipping between applications.

This led to some deep questions about the traditional interface between a schematic and a PCB. With the two applications residing in the same framework, using a netlist for transferring data seemed to be a contrived solution. Why not have the PCB and schematic synchronized at all times? If both applications were tightly integrated, they could share so much more than a netlist.

For it to work, a couple of rules had to be established. They are simple, require no extra steps by you, the customer, and are largely automatic.

The Design-by-contract solution

Design-by-contract, as applied by PCB123, simply outlines a set of requirements that must be met before some operation can be completed. For instance, one requirement for placing a symbol on a schematic page is that the symbol must identify the PCB footprint that it uses. Otherwise, the PCB would not be able to synchronize with the schematic without complaining. Why allow a symbol that is missing critical data to be used? Why not provide the means for supplying the data either automatically or though low-pain facilities in the program?

Binding a footprint to a schematic symbol should perform the necessary checking to ensure the symbol pin numbers match the PCB footprint pin numbers.

Another rule for placing a symbol is that it must have a design-wide unique reference designator (component name). That rule is a perfect candidate for automating.

Notice above we only mention restricting the placement of a symbol instance with an incomplete or invalid symbol, we say nothing about the creation of one! So why allow a symbol to be created unless it fulfills a requirements contract? The answer to that is simple: some schematic symbols, such as a resistor, live up to their names and are completely symbolic. The same resistor symbol can be shared by
many thousands of different parts that physically have various shapes and sizes. It would be ridiculous to have to create the same schematic resistor symbol for every different size resistor. Conversely, there are common chip packages that are shared by thousands of different chips, but each one may require a unique schematic symbol. An elegant solution to these problems was worked out so that the fulfillment of the symbol contract requirements happens even while providing the flexibility for symbol reuse. This will be covered in detail under Symbols and Parts.

All this talk of contracts and requirements may sound restrictive, but that is not the case. Most contract requirements, such as the unique reference designator requirement mentioned above, are automatically fulfilled by the software. The only time a requirement might bubble up as an action for the user are a couple of situations where the user would have to eventually resolve the problem anyway.

So far, we have only been talking about changes in the schematic initiating changes to the layout but the reverse must be accommodated too. Changes must be allowed to happen from both the schematic and the layout tool and synchronization should be bidirectional. It is reasonable to expect a component such as a decoupling capacitor to be added from within the layout tool. This operation requires that a schematic symbol representing the PCB component be added to the schematic in order for the two sides to match. The problem is, which symbol should it use? To expect that a PCB footprint would contain a reference to a schematic symbol for it to use does not even make sense. To accommodate this, a couple of services had to be created in the software.

The first service is a generic-symbol generator. At a minimum, it creates a rectangular symbol with a reference designator and one symbol pin for every package pin found in the footprint.

The other service needed by the software is the generation of special schematic pages called Sync pages. Parts added from Layout are automatically placed on sync pages. These are just like normal pages except they maintain internal bookkeeping for where to place the next auto-generated part. In addition to parts, unattached (board-level) pins that may have been added to a net to provide test points will be added to sync pages to maintain a one-to-one node relationship in the net graphs. Connections between pins (edges in the graphs) are not mapped, nor need to be mapped between the PCB and the schematic but the nodes (pins) do. The main reason Sync pages are differentiated from regular pages is they are considered under system control. You should not manually add parts on these pages. Nothing dangerous should happen, it’s just that any part you add may no be seen and a future auto-generated part may be placed on top of your part.

If a new connection is added to a pin in the PCB, then a Global port is added and attached to the pin in the schematic. This will happen on whatever page the pin is on, and is not limited to a sync page. Global ports are identical in operation to power and ground port symbols in that they expose a net name for global use. A picture of an automatically-generated symbol wired with global ports is shown to the left.

Besides detecting when new objects are added to the design, the software must also accommodate object deletions and modifications. Deletions are very straightforward, as nothing has to be synthesized
or de-referenced. There is not necessarily a one-to-one correspondence between deleted objects from both sides. Deleting a component from Layout may result in several gates being deleted from the schematic. Conversely, deleting a gate from the schematic may not delete a component in the PCB if there are other gates from the same part still in use. The pins used by that gate will certainly be disconnected from any nets in the PCB.

There is a use case that you may have noticed and are worried about: If you want to perform a simple cosmetic operation such as cutting a circuit from one page and pasting it into another page in the schematic then this operation as describe above would do catastrophic damage to the existing PCB layout. The moment the circuit was cut from the original page, all the components and routing in that circuit would be removed from the PCB. Then, when you added it back in to the destination page all those parts and their connections would be added to the PCB at the system origin. Fortunately, PCB123 deals with this situation just fine. The only restriction is that the operation must be performed during a single session in the software. You cannot cut, then exit the software, or even close the design and expect it to paste successfully in a later session.

Special note: Some object modifications are not directly allowed. Luckily, these are rarely required. One example is you cannot simply change the footprint reference in an already-placed schematic symbol and expect a new footprint to be swapped into use for the corresponding PCB component. To perform this operation, you are forced to delete the old symbol and add it back in with a symbol that specifies the new footprint reference. This ensures that the contract requirements are satisfied in a standard order. We expect these limitations will be removed in the near future.

Properties

A property is a named piece of data. It is the most primitive type of object in the system and every type of schematic object in PCB123 can act as a container for an arbitrary amount of data in the form of properties. A property is either a user-defined property or a system-defined property. There is no difference between the two types of properties other than system-defined properties will be actively interpreted by the software.

The only restriction placed on a property name is that it must begin with a letter or underscore ‘_’, but property names that begin with an underscore are reserved for system-defined properties and should not be used for user-defined properties. Property names will always be converted to upper case.

The value of a property is arbitrary. It may represent a color, a size, a picture, or anything else as long as it can be stored as a string of characters either directly or by some conversion. The character string can be any length except zero. Setting a property to empty has the effect of removing that property from the object that it is attached to.

Schematic objects have specific fields that are intrinsic to the object such as a location or rotation field in a text object. These fields are present in every instance of a text object. Objects may be specified with additional fields that are not intrinsic - they are essentially optional fields. It would be wasteful to store
the font name and font size inside every text object that is displayed. It would be bothersome to change the font on every piece of text that stored it intrinsically. This is the reason system-defined properties exist. A system-defined property such as TEXTFONT can be attached to a text object to specifically set the font to use when displaying that piece of text. If the property is not present, then some default is used. Let's look at how that default is chosen.

Properties exhibit ‘bubbling’ behavior. What this means is that if a property value is requested from an object and that object does not contain the requested property, then it will forward that request ‘up’ to its parent and so on. Let's look at a simplified diagram of the parent-child relationship of a part placed on a schematic page.

The picture to the right shows that a top-level object called the Schematic Container is the parent of schematic pages, which in turn contain children such as parts. A part can be a container for shapes and pins (among other things).

Attached to the schematic container is a default value for every system property. They act as a sink for property value requests that are not honored by child items. When the system attempts to draw some shape that is owned by a part, such as the resistor shape on the left, then a request for the LINECOLOR property is made of the shape. If the shape contains a LINECOLOR property that is set to green, then the search ends and the line will be drawn in green. If the shape didn't have a LINECOLOR property, the request is forwarded up to the parent of the shape which is a part. If the part had a LINECOLOR property set to purple, then that becomes the default line color used to draw all shapes owned by the part that do not specify a LINECOLOR property. When objects are created, they will use the default values for many of the various system-defined properties by simply not defining the property for the object. This means you can change the value of a single property such as TEXTFONT in the top-level schematic object and all text will use that font when being drawn unless specifically overridden in an object.

There are numerous system-defined properties and a full list can be found in the Schematic Reference Guide. Most are accessible through the Properties Dialog by clicking on this icon from the various context panes it appears on. In most cases you do not have to deal directly with the property names but rather their descriptions and many have context sensitive controls such as a color popup if the property value is a color.

Some properties are read only. In these cases, they will be displayed in gray and cannot be changed. What use is a read-only property? The answer to that is you can add Text objects that display the value of a property instead of a static text string. That means a text object can be created that displays the value of the NUMPAGES property even though that property is a system-managed value. This will be covered in detail later on.
Symbols and Parts

This section describes the relationship between symbols, sections, parts, footprints, and components.

Wow! That was a mouthful. Isn’t this supposed to be a simpler system? The answer is yes, but allowances have been made to accommodate a high degree of structure, and that in turn can aid efficiency and increase productivity.

Designing an electronic circuit has traditionally been graphical in nature where the electronic components used in the circuit were represented by symbols and their interconnections by lines, or “wires”, drawn between the symbols. Today, many circuits are designed in text editors using structured languages such as Verilog, VHDL, and C/C++ but that’s if the circuit being designed resides in a chip. Using that chip in a product still entails hooking it up to the rest of the circuit using a graphical schematic.

When drawn on paper, a schematic can play fast and loose with the level of detail required to convey the circuit’s intent. Most symbols didn’t need specific pin numbers or pin names and many didn’t even indicate reference designators but only a value or part number.

An electronic version of a schematic is a different story. The reason they are drawn using special schematic capture software is to corral a great deal of information for use by other software programs such as digital simulators, SPICE simulators, and PCB design systems. The schematic software also generates BOM (Bill of Materials) reports, various statistical reports, and even performs simple semantic checking on the circuit.

Half the information contained in (defined by?) a schematic is the interconnection between components by drawn wires. They define their own data by where they are drawn, or more specifically where they end. They describe the interconnection by conductors (edges) between component pins (nodes). In graph theory, a set of connected nodes is called a network but the ECAD industry has shortened it to just net. So there it is, half the information in a schematic is the interconnection of components by connecting networks of pins with wires. If the other half of the information in a schematic were so simple, you would only have one more paragraph to read.

Unlike wires, the other half of the information doesn’t describe itself. Luckily, the great majority of the information is contained in the reusable symbols and parts which are stored in a library. The rest of the information is design-specific and most of that is auto-generated.

So what is the other half of the information? The simple answer is part data. A part is more than just a symbol plopped down on a schematic page.

A part is actually assembled from various entities that reside in libraries. The relationship between these libraries is as follows:
The part taxonomy is a searchable database of parts organized in a flexible hierarchy. It accommodates the reuse of both schematic symbols and PCB footprints by allowing one library object to participate in many different parts. It is described in detail below.

Schematic symbols are created in the Symbol Editor or the Part Wizard and are saved into either the standard symbol library named std.sym or into a custom library that you created.

PCB Footprints are created in the Footprint Editor and are saved into either the standard footprint library named Std.slb or into a custom footprint library that you created.

The single best reason for creating custom symbol and footprint libraries is because the standard libraries may get overridden during software updates. Sunstone reserves the right to grow the standard libraries from release to release.

Data is copied in its entirety from the libraries and into a design. Once an object is loaded into a design, it is fully encapsulated and travels with the design file and does not require that the library file be present to work with the design.

If you wish to copy libraries from one machine to another, then you need to know where they are. The root directory for all library files will be Documents and Settings\All Users\Application Data\PCB123\Lib on the system drive. This directory may be hidden from you but the PCB123 installer should have installed a link named PCB123 Files in your My Documents folder. The link points to the Documents and Settings\All Users\Application Data\PCB123 directory. Clicking on it will browse to that directory. From there, you will see the Lib subdirectory. Any custom library that you create will be created in the Lib\Custom subdirectory.
Parts

A part is typically an electronic component that is purchased and mounted to a PCB. There is a logical manifestation of a part and a physical manifestation of a part. The physical manifestation is a single instance of a PCB footprint that contains the pad geometries and/or drill hole locations for mounting and soldering the part on the PCB. The logical manifestation of a part consists of one or more graphic symbols instantiated on a schematic page or pages. The reason a part may have several schematic symbols is due to some logical or functional partitioning of the part. PCB123 has adopted the term Section for each partition but you may also think of them as gates.

A part has a name called a reference designator that is unique throughout an entire design. The reference designator consists of an alpha prefix and a numerical value. In addition, each section of a multi-section part will have an alpha suffix that is unique to each section of the part. You have total freedom to specify the alpha prefix but the numerical value and possible suffix are managed by the PCB123 software.

<table>
<thead>
<tr>
<th>Class</th>
<th>Prefix</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Battery</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>Capacitor</td>
<td>C, or CR</td>
<td></td>
</tr>
<tr>
<td>Diode</td>
<td>D, or CR</td>
<td></td>
</tr>
<tr>
<td>Fuse</td>
<td>F</td>
<td></td>
</tr>
<tr>
<td>Connector</td>
<td>J, or P</td>
<td></td>
</tr>
<tr>
<td>Relay</td>
<td>K</td>
<td></td>
</tr>
<tr>
<td>Inductor</td>
<td>L, or FB</td>
<td></td>
</tr>
<tr>
<td>Transistor</td>
<td>Q, or VR</td>
<td></td>
</tr>
<tr>
<td>Resistor</td>
<td>R, or RN</td>
<td></td>
</tr>
<tr>
<td>Switch</td>
<td>S, or SW</td>
<td></td>
</tr>
<tr>
<td>Transformer</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td>IC or Chip</td>
<td>U</td>
<td></td>
</tr>
<tr>
<td>Crystal</td>
<td>Y, XTAL, OSC</td>
<td></td>
</tr>
<tr>
<td>Zener Diode</td>
<td>Z, or D, or CR</td>
<td></td>
</tr>
</tbody>
</table>

There are standard prefixes for different part classes such as resistors and capacitors. The table on the left lists the most common part classes along with the prefixes and a typical schematic symbol for that class.

You have the ability to assign a prefix to a symbol. All parts that use that symbol will have that reference designator prefix.

Besides a reference designator, one or more symbols, and a PCB footprint specification, parts may contain other information such as a value, tolerance, manufacturer's part number or distributor number such as a Digikey catalog number. In reality, a part may hold any information you desire. How this extra information is acquired will be described shortly.

For now, let's move on to the role a symbol plays in a part description.

Symbols

A schematic symbol is a collection of primitive objects such as pins, 2-D shapes, and text that are treated as a single entity and may represent a complete part or one section of a part. Symbols reside in libraries where they may be referenced by parts.

Some symbols, such as resistors and capacitors, are highly generic and can a single symbol may be suitable to represent thousands of different parts. Other symbols, such as an MCU may be created that
are unique to just one part. There are even some symbols that are never intended to represent a part at all but only sections of a part, such as a NAND gate.

There are no hard rules, but in general the more specific a symbol is to a particular device, the more information you want to bind to it. Conversely, a symbol that will be used for a large assortment of parts will want to stay reasonably generic in the information it specifies.

Symbols can be drawn by hand using the Symbol Editor or they can be created automatically with the Part Wizard. Symbols created automatically with the Part Wizard are limited to a rectangular symbol with pins that can be described in spreadsheet fashion.

Symbol text

Symbols may contain both static and dynamic text objects. Intrinsic to a text object is a string to display. How that string is interpreted depends on the text type. If it is static text, then the string is displayed verbatim and does not change unless a new string is supplied. Dynamic text will interpret the string as the name of a property (system-defined or user-defined) and display the bubbled value for the property name. The symbol on the left is a highly-generic symbol that has two dynamic strings. When this symbol is incorporated into a part and that part is placed on a page, then the _REFDES will be replaced with the actual reference designator for the part and _PARTNUMMFG will be replaced with the manufacturer's part number if supplied to the part.

Other intrinsic text fields besides the string include:

- **Text Location** - The X,Y coordinate for the text origin. The Y origin is always at the top of the character cell but the X origin depends on the text alignment described below.
- **Text Rotation** - Either 0 degrees (horizontal text) or 90 degrees for vertical text.
- **Text Alignment** - Specifies either Left, Center, or Right aligned text. The X origin for the text will coincide with the left edge, center, or right edge of the text respectively.

The following system-defined properties may be used to affect the display of text objects:

- **_TEXTFONT** - Specifies the font name to use when drawing the text.
- **_TEXTSIZE** - Specifies the text height in points. There are 72 points in one inch.
- **_TEXTCOLOR** - Specifies the color of the text. This is usually set with a color picker but in the event you gain textual access to this property, you can encode the Red, Green, and Blue values using the following formula: color = (Red*65536)+(Green*256)+Blue, where each color may range between 0 and 255. Additionally, you may set a color to invisible by setting it to (or adding) 2,147,483,648.
Symbol shapes

You can add lines, arcs, and regions to symbols. Virtually every display characteristic for shapes are controlled by attaching system-defined properties to the shape or, to specify them all at once, attaching the property to the symbol. The system-defined properties that affect the display of shapes are:

- **_LINEWIDTH** - Specifies the width of the line. This is typically selected from a choice but numerically it is in 100ths of an inch (10 1/100ths of an inch = .1 inches)
- **_LINECOLOR** - Specifies the color of the line or outline.
- **_LINESTYLE** - 0 = solid, 1 = dashed, 2 = dotted, 3 = dot-dashed.
- **_FILLCOLOR1** - Color to fill the interior of closed regions.
- **_FILLSTYLE** - 0 = hollow, 1 = solid, 2 = horizontal hatched, 3 = diagonal hatched, 4 = cross hatched, 5 = diamond hatched.

Symbol pins

Probably the most important primitive that is included in a symbol is a pin. Pins map a logical port on the symbol to a physical pin in the footprint. Pins have a hotspot (designated by a small, solid circle) that provides a tie point for wires.

Graphically, A symbol contains an inside and an outside. The smallest the inside can be is the rectangle that bounds all drawn shapes in the symbol. Pins will be constrained to the edge of this rectangle and will always face outward. This provides uniform alignment of the pins and ensures the automatic wire router has a clean escape from the pins without bumping into shapes.

Intrinsic to pins are:

- **Internal pin name** - A label for the pin. A pin name must be supplied for a pin. It is recommended that the names be unique within the symbol but it is not required. This relaxed rule is to accommodate multiple names such as GND or VCC in a symbol or part. If a pin name ends with 2 hyphens, or minus signs (‘—’), then the hyphens will not be displayed, but rather a bar will be drawn over the pin name. This is a visual indicator for “NOT”, or inverted.
- **External pin number** - The pin number of the footprint this pin represents. When creating a generic symbol that may be used by different footprints, it is not necessary to define a pin number, as shown in the symbol above. Generic symbols are expected to be assembled into a part using the Part Wizard. By the time a symbol is used in a part, it must have acquired a unique pin number for every pin and the pin number must match a pin in the footprint. The double-hyphen mnemonic does not apply to pin numbers because they have to match up with the name of a footprint pin. They are called pin numbers but may in fact contain other characters than digits.
- **Pin shape** - There are several graphical styles that a pin can assume. They are:

  - Zero Pin
  - Normal Pin
  - Dot Pin

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Zero Pin" /></td>
<td>Zero-length pin. Not typically used.</td>
</tr>
<tr>
<td><img src="image" alt="Normal Pin" /></td>
<td>No special meaning attributed to pin.</td>
</tr>
<tr>
<td><img src="image" alt="Dot Pin" /></td>
<td>Signifies and inverted state.</td>
</tr>
</tbody>
</table>
Clock Pin
Indicates pin is used to clock the device.

Dot-Clock Pin
Inverted clock.

- **Pin Type** - Classifies the pin’s function. Assigning a pin type to a pin allows that pin to participate in certain rules checking such as the Electrical Rules Check. This check will catch wiring combinations that are suspect such as a GND pin tied to a VCC pin. Possible values are:
  - **N/C** - Pin is never connected.
  - **VCC** - Pin is tied to voltage source.
  - **GND** - Pin is tied to ground.
  - **PASSIVE** - Default pin type.
  - **INPUT** - Pin is an input to an active device.
  - **OUTPUT** - Pin is an output from an active device.
  - **BIDIRECTIONAL** - Active input or output pin.
  - **TRISTATE** - Pin may assume high impedance state.
  - **ECL** - Emitter-coupled logic.

Pins will always snap to an immutable pin grid. This, coupled with a symbol rotation restriction of 90-degree increments ensures that pins will always stay on a neat grid.

The only active system-defined properties for pins are `_PINNAMEVIS` and `_PINNUMVIS`. When these properties are set to zero, the pin name and pin number respectively will not be displayed.

In the future, pins may contain properties that both drive and probe a simulator.

Symbol properties

Recall that in the discussion of symbol text we used as an example a system-defined property named `_PARTNUMMFG`. In that discussion we showed how a dynamic text whose string value was set to `_PARTNUMMFG` would display the manufacturers part number if it found one. So where would it find this value? If we wanted, we could add a property called `_PARTNUMMFG` to the symbol itself and set its value to SN74AHCT00N just as an example. When it was time to display the text object, it would bubble up a request for the value of a property named `_PARTNUMMFG` and would get back SN74AHCT00N almost right away. The fastest if could get back an answer would be if it contained the named property itself. Because the text object didn’t possess that particular property, it forwarded the request up to its parent, which is the symbol itself. In this example symbol (the NAND gate pictured above) it wouldn’t make much sense to bind such explicit information to the symbol. After all, we want to use that symbol wherever we can to avoid duplicating effort. So the symbol is not a good place to store device-specific properties if the symbol is intended for reuse. That doesn’t mean it can’t contain other useful properties.

One such property is the reference designator prefix, which in this case would always be ‘U’. Now you wouldn’t have to specify it using a property because there is a dedicated place to fill in that information but behind the scenes, the symbol editor is actually binding the reference designator prefix to the symbol in the form of a property. Even on a symbol that just represents a gate and not a part, this property will find a home because when a part object is assembled from one or more symbols, the part will suck up any properties that were specified by the symbol.
In fact, if we specified just one more property we could automatically create a part object that could be placed on a schematic page. Take a look at the symbol on the left. It is a single-sourced relay that is available in only one package size.

Because this is most likely a one-off symbol, it makes sense to specify device-specific properties right in the symbol. There are a pair of system-defined properties named _PARTFOOTPRINT and _PARTFOOTPRINTLIB. By specifying these properties right in the symbol, PCB123 will have all the information it needs to instantiate the symbol directly on a schematic page.

In fact, this is not what happens. PCB123 actually creates a part object and assembles all the necessary information automatically. Again, you do not have to specify the properties directly because the symbol editor has a very handy graphical footprint browser and search tool. You can just click on a footprint to use and it will automatically create and bind the footprint properties. In addition, when you save the symbol to a library, it will perform full validation on the pin number mappings and only save the symbol if it completes the contract terms. It will also add an entry into the part taxonomy so the symbol can be searched in the normal way. It adds an entry under the “Generic Symbols” family of the “Other” part class in the taxonomy. You can move it to a more orderly place if you desire.

So, we have seen how we can quickly flash up some pins and shapes in the symbol editor and then graphically browse for the footprint the symbol will use. After saving the symbol there will now be a part in the taxonomy that is immediately available for use.

We should back up just a minute and address the other scenario that we glossed over and that is how best to create generic symbols and then use them in parts with part-specific properties.

The picture to the left shows a pin assignment diagram for a Texas Instruments 7400 Quad NAND Gate. There is an inner rectangle showing the 4 NAND gates inside the chip and the small outer rectangles represent the physical pins on the part. The bold numbers are the pin numbers of the package and the small labels are the internal pin names. In this case, there is a pattern where the two inputs of each gate are named A and B, and the output is named Y.

Let’s look again at our NAND gate. This symbol seems to have all the things we need if we want to place the NAND gates individually. It has the 3 pins and their names are labeled correctly. It has placeholders for values we want to see on the schematic page. So, let’s just see what part numbers this gate can be used in. We know it can be used in a 7400 part but that’s just a generic name for the part. We would have a difficult time...
ordering a part number 7400 from Texas Instruments. Here is just a sampling of the actual 7400 devices from various logic families.

SN7400N, SN74S00N, SN74LS00N, SN74HC00N, SN74HC00E, SN74HCT00N, SN74HCT00E, SN74AS00N, SN74ALS00AN, SN74AHCO0N, SN74AHCT00N, SN74AC00N...

There are similar series for the 7401, 7403, 7424, 7426, 37, 38,... 74804, 748003, all using the NAND gate and that's just from Texas Instruments. Many of the parts where we would like to use the NAND symbol have widely varying configurations. Many have the same pin assignments but use different packages such as through-hole or surface mount packages, and still other parts have different pin configurations or even differ in the number of gates in a part.

With a little bit of organizing, we will be able to reuse the NAND symbol in hundreds of parts and decorate the parts with a rich information model.

The first step is to review the candidates and organize them by the number of gates in a part and the pin assignments for each gate in the part. A quick check of the TI data book shows that in addition to the 7400, the 7403, 7426, 7437, and 7438 series devices all share the same configuration.

Now that we've identified the gate and pin assignments that are needed, we run the Part Wizard which is shown at left.

There are 3 fields we are concerned with. The first is to set the number of sections to 4, one for each gate. Then make sure the Same Symbol for All Sections is checked. Lastly, we want to use the NAND gate symbol and not the blocky default symbol. We do this by clicking on the Load From Library... button and then select NAND gate symbol. Notice that the Number of Pins field is now set to 3. The preview looks similar to our gate but it has prefixed the pin names with a different number for each section. Also, it has automatically supplied default pin numbers for the pins. These will need to be fixed.
One last point worth mentioning; we still have not supplied a footprint at this stage, because we are still specifying common logical data. We are making a template.

Clicking on the Next button takes us to step 2 of the Part Wizard as shown at left.

The sections of the part and the pins in each section are displayed in a grid.

All the data except the pin number column appears to be complete. We want to fill in the pin numbers by looking back at the pin assignment diagram and matching the pin names with the right numbers.

We could double-click on the PIN_NUM cell for pin 1A of section A and simply enter a ‘1’ there but we can do one better. The first two gates, A and B, are in order. We would end up typing 1 through 6 in each consecutive PIN_NUM cell and there is an easy way to do this.

Starting in the first cell, we type \([1:6]\) and press Enter. This is a shorthand way of entering the digits from 1 to 6 in consecutive cells. The next two sections do not lend themselves to this shorthand because the numbers do not fall in sequence. Luckily there is another trick to use. If we double-click on the PIN_NUM cell for Section C, Pin 3A we can type \([9,10,8,12,13,11]\) to enter those numbers in consecutive cells.

At this point we have all the information filled out for each section but we still have one problem. Looking at the pin assignment diagram we see that we have not provided a way to power the chip. Because there are 4 sections, how do we decide which one gets the power and ground pins, and for that matter, will we have to make another symbol to show the pins?

The answer is no. The pins will not be added to any section but rather to the part itself. These will be hidden pins that will never appear on the schematic. Instead, we will attribute them with pin types of GND and VCC. When a new part is instantiated in a schematic, any pins that it owns that are hidden power and ground pins will make an implied connection to the nets assigned to GND and VCC. If no net is associated with these pin types, then you will be prompted to supply a net name for these types. Why doesn’t the system just create a net named GND and VCC? Because in this day and age of different supply voltages for the different families the software cannot make that assumption. VCC may be 5V or it
may be 3.3V or something else. This way we can still create generic part templates which allow for
different supply voltages.

After clicking on the Add Pin button twice and filling in the
two new rows our grid appears as follows:

We save the part into our library using the generic name
QUAD_NAND and select Finish.

Remember, all we have done is create a template to be used by
a large number of different parts. Nowhere in this exercise did we
specify an orderable part number or even bind the template to any
physical footprint.

There is another tool for that and it will be discussed in the next
section, The Part Taxonomy.
The Part Taxonomy

Sunstone examined various schematic capture programs before and during the development of the new schematics facility in PCB123. We looked not so much at the features of these systems, but how they managed design processes. One thing that stood out was how dissimilar the various systems were in how they dealt with libraries and managed parts. Surprisingly, the places where they did agree we didn’t necessarily think were the best solutions to the problem. Sunstone walked away from the evaluations more with a list of what not to do than of features to emulate. This is not a knock on those systems. After all, they paved the way to make the comparisons possible.

One thing that we wanted to move away from was the idea of file-centric libraries. Most of the systems reviewed had management facilities where library files could be activated and deactivated. Additionally, how libraries were partitioned into files was a subjective thing. Some systems partitioned the library by part class and others by part manufacturer. Neither way was wrong but that begged a review of the file-centric partitioning itself – it felt artificial. PCB123 will allow such file-centric partitioning to take place, but we deemphasize it in favor of a different mechanism that will be explained shortly.

Searching for a part was another issue. Most of the systems we evaluated based their search criteria on the name of a part. Systems that provided capabilities beyond that typically did so by drilling down through additional dialog boxes that offered up different search criteria. None of the systems we evaluated allowed an arbitrary search term, such as a resistor value, to be entered in a general search that successfully found a part. We were of the opinion that the name of a part was almost irrelevant unless maybe it was a manufacturer’s part number or something that was universally known. With some parts, it does not even make sense that you would key off the name when selecting it. For instance, if you need a resistor then you should be able to type 4.99k, 1% and get a list of parts that included those criteria. You could guide the search more specifically by entering 4.99k, 1%, 1/8W, Rohm, or by entering a specific manufacturer’s part number.

Something that is always fun to play when developing a new product is the “wouldn’t it be neat if...” game, such as “wouldn’t it be neat if we included an entire catalog of parts, such as the DigiKey catalog, with the system?” Well yes, it would, and if we could create 1000 parts a day, and there were approximately 500,000 parts in the catalog, then it would take 500 days to complete. This clearly precludes the creation of a catalog library from the initial release. Even so, we must be sure the system is capable of handling such a large library for future releases that just might contain such a catalog. A library containing half a million or a million parts would be massive. Each part would include not only a catalog number and manufacturer’s information, but also any number of properties that could be useful as searchable criteria or as parameters for tools such as simulators.

Another problem with such a large library is creating it! We went to great pains to describe how the symbol editor and the part wizard may be used to create generic symbols and part templates. If the symbol library only contained a single resistor symbol, and the footprint library only contained a handful of resistor footprints such as the standard 0402, 0603, 0805, and 1206... sizes, we could still create parts for about 20,000 manufacturer or catalog parts numbers. Each part would proclaim, “I am such and such part. I am manufactured by company X as part number Y, and I can be purchased from this catalog as catalog number Z. I use this schematic symbol and this PCB footprint. Some of my important characteristics are...."
Something like that could almost be done in a spreadsheet, as shown below:

<table>
<thead>
<tr>
<th>Mfg.</th>
<th>Mfg. #</th>
<th>Cat. #</th>
<th>Symbol</th>
<th>Footprint</th>
<th>Value</th>
<th>Tol.</th>
<th>Power</th>
<th>V Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rohm</td>
<td>MCR10EZHF17.8K</td>
<td>RHM17.8KCCCT-ND</td>
<td>RESISTOR</td>
<td>RC0805L</td>
<td>17.8K</td>
<td>1%</td>
<td>1/8W</td>
<td>150V</td>
</tr>
<tr>
<td>Rohm</td>
<td>MCR10EZHF18.0K</td>
<td>RHM18.0KCCCT-ND</td>
<td>RESISTOR</td>
<td>RC0805L</td>
<td>18.0K</td>
<td>1%</td>
<td>1/8W</td>
<td>150V</td>
</tr>
<tr>
<td>Rohm</td>
<td>MCR10EZHF18.2K</td>
<td>RHM18.2KCCCT-ND</td>
<td>RESISTOR</td>
<td>RC0805L</td>
<td>18.2K</td>
<td>1%</td>
<td>1/8W</td>
<td>150V</td>
</tr>
<tr>
<td>Rohm</td>
<td>MCR10EZHF18.7K</td>
<td>RHM18.7KCCCT-ND</td>
<td>RESISTOR</td>
<td>RC0805L</td>
<td>18.7K</td>
<td>1%</td>
<td>1/8W</td>
<td>150V</td>
</tr>
<tr>
<td>Rohm</td>
<td>MCR10EZHF19.1K</td>
<td>RHM19.1KCCCT-ND</td>
<td>RESISTOR</td>
<td>RC0805L</td>
<td>19.1K</td>
<td>1%</td>
<td>1/8W</td>
<td>150V</td>
</tr>
<tr>
<td>Rohm</td>
<td>MCR10EZHF19.6K</td>
<td>RHM19.6KCCCT-ND</td>
<td>RESISTOR</td>
<td>RC0805L</td>
<td>19.6K</td>
<td>1%</td>
<td>1/8W</td>
<td>150V</td>
</tr>
<tr>
<td>Rohm</td>
<td>MCR10EZHF20.0K</td>
<td>RHM20.0KCCCT-ND</td>
<td>RESISTOR</td>
<td>RC0805L</td>
<td>20.0K</td>
<td>1%</td>
<td>1/8W</td>
<td>150V</td>
</tr>
<tr>
<td>Rohm</td>
<td>MCR10EZHF20.5K</td>
<td>RHM20.5KCCCT-ND</td>
<td>RESISTOR</td>
<td>RC0805L</td>
<td>20.5K</td>
<td>1%</td>
<td>1/8W</td>
<td>150V</td>
</tr>
</tbody>
</table>

That would be great because spreadsheets lend themselves beautifully to automation or tabular processing. The only problem is that some of the columns shown in the table above will not work for any part types other than a resistor, and even then, it might be missing columns that are important to you such as temperature stability or RoHS compliance.

There is something interesting to observe with this table; each row defines a part but nowhere does it specify a name for the part. What is its name? The answer is, who cares? If each row specifies all the necessary information for a part, including how to purchase one, then we don't care what it is called, as long as we can find it in a search. That is what we mean by part names being contrived entities and why we did not want to rely on them for searches.

You could point PCB123 to any one row and it could assemble a part for placement on a schematic page. In fact, every column in the table could be removed except the Symbol and Footprint columns and PCB123 could assemble a part for placement. With all the other columns, PCB123 creates a property that uses the column name for the property name and the cell value for the property value.

This is where the payoff happens by keeping symbol definitions generic if there is any hint that they can be reused. A search for a part can be performed using any criteria and all the matches (read choices) will be presented to you. You can then select your choice and the software will bring in the referenced schematic symbol and the referenced footprint, perform all the necessary validation, and if everything is fine, it can then create a part instance, give it a reference designator, and add all the other columns of data to the part in the form of properties. Later, when you want to generate a bill of materials, it may include such items as a catalog number and quantity for each part type which you can then send to the part distributor. Even better, it puts PCB123 on the cusp of automating the purchase of your PC boards, the part kits, and even assembly services just by pressing the Order button. It's too bad we didn't have a Sell Product button too!

Getting back to the table shown above, careful observation shows that only three out of the nine columns have varying data. That is a lot of redundant data. In addition, we still have not addressed the fact that the table is only good for specifying resistors, and that is not even indicated anywhere. Luckily, there is a neat way to solve both problems.

**Hierarchy and Inheritance**

We call our part database the taxonomy because it separates parts by species and subspecies, with part definitions being progressively refined with each subspecies level. For our purposes, taxonomy is just
another word for hierarchy. In fact, we don’t use taxonomy terms such as species, order, phylum, or genus, we simply use part class and family.

A part class is the first level of the taxonomy. The number of part classes and the name of each class are fixed by the system – they are the only absolutes in the taxonomy structure.

Part classes may contain any number of uniquely named child items called Families. A family is an organizational unit to which properties can be attached. They too may contain any number of uniquely named child families and they may also contain parts.

A Part, in the taxonomy, is just a collection of properties. They are the leaves in the hierarchical tree structure of the taxonomy and may contain no children.

Besides the list of properties that a part may explicitly define, it also inherits all properties that its parent family and any ancestor thereof may define. This is how the redundant data from the table above is removed. In fact, this data “folding” is a natural way to define a hierarchy for a part. To do this, we set two goals: the first is that all parts in a family have the same number of properties (columns). The second is there should be no column whose every value is the same. If we find a column with invariant data, then it should become a property of the parent and the column removed by deleting the property from all the child instances. As an example, we will convert the table above to a hierarchy using these two goals.

If we were starting from scratch, all we would have is the fixed part classes as shown on the left. We know the table represents resistors so the Resistor part class is where we will start defining or discovering the hierarchy for the table.

First, we check if there are any invariant columns to eliminate by adding a property to the parent, in this case the Resistor part class. The closest candidate would be the Symbol column with its value RESISTOR but that might asking too much of the Resistor part class because there are other resistor symbols such as a variable resistor. This indicates the need for a subspecies to differentiate between at least fixed resistors and variable resistors. We will create two families called **Fixed** and **Variable**, but we will only be concerned with the **Fixed** family for our purposes.

To the **Fixed** family, we will assign a property named Symbol the value RESISTOR, as show to the right. This will assign the RESISTOR symbol to any child family and part under the **Resistor\Fixed** family and we can remove the Symbol column from the table.

With our active node being the **Fixed** family, we again look at the table and determine there are no more invariant columns that we can safely eliminate by adding additional properties to the **Fixed** node. A good check for this is by asking the question “are all...?”, as in “are all fixed resistors made by Rohm?”, or “are all fixed resistors 1/8W?” If the answer is no, then the column must be eliminated with further refinement.
Families do not have to be limited to just the invariant columns in the table. In fact, we will add another family named **Thick Film Chip** under the **Fixed** family, but we will not add any properties to it that we know of yet.

The only way we can eliminate another invariant column is to further refine the hierarchy with another family. This is a subjective process, but in this case, it seems to make sense to refine by manufacturer, so we will create a family named **Rohm** and assign to it a property named Mfg who value is also Rohm. This is shown at left.

We can then eliminate the Mfg column from the table and proceed in the same fashion for the additional invariant columns.

The image below shows what the family hierarchy looks like after all invariant columns have been eliminated. The table now contains only 3 columns, all of which have variable data.

<table>
<thead>
<tr>
<th>Mfg. #</th>
<th>Cat. #</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCR10EZHF17.8K</td>
<td>RHM17.8KCCT-ND</td>
<td>17.8K</td>
</tr>
<tr>
<td>MCR10EZHF18.0K</td>
<td>RHM18.0KCCT-ND</td>
<td>18.0K</td>
</tr>
<tr>
<td>MCR10EZHF18.2K</td>
<td>RHM18.2KCCT-ND</td>
<td>18.2K</td>
</tr>
<tr>
<td>MCR10EZHF18.7K</td>
<td>RHM18.7KCCT-ND</td>
<td>18.7K</td>
</tr>
<tr>
<td>MCR10EZHF19.1K</td>
<td>RHM19.1KCCT-ND</td>
<td>19.1K</td>
</tr>
<tr>
<td>MCR10EZHF19.6K</td>
<td>RHM19.6KCCT-ND</td>
<td>19.6K</td>
</tr>
<tr>
<td>MCR10EZHF20.0K</td>
<td>RHM20.0KCCT-ND</td>
<td>20.0K</td>
</tr>
<tr>
<td>MCR10EZHF20.5K</td>
<td>RHM20.5KCCT-ND</td>
<td>20.5K</td>
</tr>
</tbody>
</table>

These are our parts and they will be added to the **Resistor\Fixed\Thin Film Chip\Rohm\1% \ 1/8W** family, and each will contain a Mfg #, Cat #, and Value property as shown below.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Property Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mfg #</td>
<td>MCR10EZHF17.8K</td>
</tr>
<tr>
<td>Cat #</td>
<td>RHM17.8KCCT-ND</td>
</tr>
<tr>
<td>Value</td>
<td>17.8K</td>
</tr>
</tbody>
</table>

Those three properties are what is stored for each of the potentially thousands and thousands of fixed resistor parts. However, because of inheritance, when a query for a part is made the set of properties returned is usually more than just what is bound to a specific part. The returned list is called the effective properties for the part.

Here is an example of the list of effective properties returned due to inheritance:

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Property Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mfg #</td>
<td>MCR10EZHF17.8K</td>
</tr>
<tr>
<td>Cat #</td>
<td>RHM17.8KCCT-ND</td>
</tr>
</tbody>
</table>
You can see we pick up a substantial amount of nearly free data including the all-important symbol and footprint bindings.

Before we proceed, a small clarification needs to be made. What is shown as the **Symbol** property and the **Footprint** property are simplifications for clarity. The primary tool used to edit the taxonomy, which is called the Part Manager, actually provides graphical browsers to select the symbol and footprint for these properties. In reality, the browsers add two properties each. **Symbol** is actually two properties named SymName and SymLib. **Footprint** is actually two properties named FpName and FpLib. The library names are needed to fully resolve the location of a symbol or footprint.

Additionally, the Part Manager’s interface includes several built-in property names to encourage their use. We have been used Mfg #, Cat #, Value, Tol, and Mfg in the above example, but in reality these property names are system-defined property names that map to:

<table>
<thead>
<tr>
<th>Example Name</th>
<th>System-defined Property</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mfg #</td>
<td>PARTNUMMFG</td>
</tr>
<tr>
<td>Cat #</td>
<td>PARTNUMDIGIKEY</td>
</tr>
<tr>
<td>Value</td>
<td>PARTVALUE</td>
</tr>
<tr>
<td>Tol</td>
<td>PARTTOL</td>
</tr>
<tr>
<td>Mfg</td>
<td>PARTMFG</td>
</tr>
</tbody>
</table>

Typically, you will not have to deal with these names because there is a user interface to deal with them. However, there is an alternative way to edit the taxonomy besides the Part Manager. The taxonomy file itself is an XML document and it is possible you have a way of acquiring part data en-mass in XML form and might find it more efficient to convert that data with an XML editor in place of the Part Manager. For XML fans, the taxonomy schema is listed in the appendix. We just wanted to make you aware that not all the columns are as general as we have led on in the spirit of clarity. The above properties do in fact want to map to their specific system-defined property names because PCB123 will make great use of them.

### The Part Manager

If you create a generic symbol – a symbol without a footprint binding, no default entry will be created in the part taxonomy because it is considered “incomplete” for use in a schematic. There is no rule against adding an incomplete part to the taxonomy but it is assumed that generic symbols will never be intentionally used as is. It is assumed you will create specific bindings to footprints using the Part Manager.

While in the schematic editor, the Part Manager may be invoked at any time from the Display Panel by pressing the
Manage Parts button. The Part Manager dialog is split into a left side and a right side as pictured below. In this example, we have drilled down into the Resistor part class, and through several Part Families to select an individual part in the tree on the left. The effective properties for the part are displayed in the grid on the right side. The properties displayed in red are inherited from the ancestors of the part, and the properties displayed in black are assigned directly to the part.

When you select and item this way, you may add, delete, or modify the properties of that item. If you click on a different item after changing the properties, the changes will be applied to the selected item before changing to the new selection.

If you right-click on either Part Class items or Family items, you will be presented with a popup menu that allows you to perform several commands. Create New Family allows you to add structure to the taxonomy as you see fit. Families may be added under the root part classes and under other families. You may select Edit Family Name, or just click on a family label two times to engage in-place editing of a family name. Delete will delete the selected item and any child items it may contain.

Create New Parts Here... will enable the Part Name field and the Create Parts button on the right side. Depending on what you had previously selected, you will have the names of at least two properties in the grid filled in, the SymName and FpName properties. The Value of these properties can only be
filled in by clicking on the ellipsis (...) button in the Value column. This will invoke the Symbol and Footprint browsers which hide the implementation details of these properties.

By clicking on the down-arrow in a new property row, you will be presented with a menu containing several system-defined properties that you may choose from for the new property. You are not limited to these properties, they are just the list of system-defined properties that are recommended to use for part specifications where applicable. You may also enter a custom property name if you wish.

Entering part information one part at a time is not very much fun. If you only needed to specify one part then you may as well have bound a footprint right to the symbol when creating it with the Symbol Editor of Part Wizard. Where the Part Manager really becomes useful is when you have to create a number of parts.

**Acquiring Data**

Below is a sample of one part series from the Digikey catalog. It is a family of oscillators made by Citizen.

**CMX309HWC Series Oscillators**

Using shorthand, we can create each part in the series all at once. The first thing we want to do is create a place in the taxonomy for them. Under the Crystal part class, we will create the following families:

**Oscillators**

**Surface Mount**

**Citizen**

**CMX309HWC Series**

After creating the families and sub-families, click on the Citizen family item and on the 3rd row of the grid, under the Name column, click the arrow to present the list of system-defined property names. Select the...
**Part Manufacturer** list item. You will notice when this property is selected a down-arrow is displayed in the value column. Clicking on this arrow will display a list of known company names. If Citizen is in the list, then select it. Otherwise, ignore the list and just type Citizen for the value. This will be added to the list of companies for future use. The reason a list is maintained is to avoid duplicating company names where the spelling may deviate just a little.

Now, in the taxonomy tree, select the family you just created named **CMX309HWC Series**. We will be able to bind a good deal of information to this item.

The information above shows a single package being used for all the parts in the series. We can click on the Value ellipsis for SymName and FpName to select the symbol and footprint this series will use. On the next empty row in the grid, select Part Tolerance from the list of system-defined properties and enter +/-100ppm for its value.

Next, add the following custom properties to the grid, each one on a blank row:

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUPPLY_VOLTAGE</td>
<td>5V</td>
</tr>
<tr>
<td>SUPPLY_CURRENT</td>
<td>35mA</td>
</tr>
<tr>
<td>NOMINAL_DUTY</td>
<td>50%</td>
</tr>
</tbody>
</table>

We are now ready to create the parts under the selected family. We are going “screen scrape” the information for the parts. Since many part specifications are PDF files, we will use the Adobe Acrobat Reader as the example program where we are viewing part data. Acrobat Reader has a tool called the Select tool whose icon is show to the left.

Clicking this icon put Reader in a mode that allows a block of text in the PDF to be selected and copied to the clipboard. This tool also has a secret mode that allows for columns of data to be selected. If you click on the beginning of the column you wish to copy and, while holding the Alt key down, drag to the end of the column, you can release the mouse button and you should see just that column highlighted. Press Ctrl+C or select Edit/Copy from the menu in Reader to copy that column to the clipboard. In our case, using the table above, we will column-select the values from the Freq.(Mhz) column taking care not to include the column title itself, and then press Ctrl+C to copy the values to the clipboard.

Back in PCB123, right-click on the **CMX309HWC Series** item in the tree and choose Create New Parts Here. This should have enabled the Part Name field just above the property grid. In this field, enter CMX309HWC- and then press Ctrl-V or right-click and select Paste.

Before pasting, the Part Manager inspects the clipboard looking for columnar data. If it finds it, then it will convert the clipboard contents to the shorthand notation that PCB123 uses elsewhere, namely [item1,item2,item3,...].

Next, on a blank row in the grid, select Part Value from the list of system-defined properties presented when clicking on the down-arrow of the blank row. Click in the Value cell to the right, press Ctrl + V, or right-click, and select Paste. This will have added a long string of characters that end with a square bracket ‘]’. Immediately after the square bracket, type Mhz.
Back in Acrobat Reader, select the Digikey Part No. column and paste it into the clipboard. On the next empty row of the grid in PCB123, select DigiKey Part No. from the list of system-defined properties and for its value, paste the contents of the clipboard. Do the same thing again for the Citizen Part No. column in the PDF and paste it in as a system-defined property named Mfg Part No.

We have defined all the properties we want so now click the Create Parts button in the upper-right of Part Manager. The Part Manager will examine the values and if it encounters any shorthand in the fields, it will create a new part for every expansion of the shorthand.

This will Below is a picture of the newly created parts. The properties shown are for the selected new part. When you press Finish, the taxonomy is refreshed in the schematic editor.
The Symbol Editor

The symbol editor in PCB123 is a dialog box with a work area and a few toolbars, as shown below. The actual symbol editor is usually larger than the picture; it was compressed to fit on a page. The symbol editor is invoked from the Tools menu while in the schematic editor. Because it is a dialog box, no other work may proceed on a design until the symbol editor is dismissed.

The symbol editor does not interact with any design; it only interacts with schematic symbol libraries. When it is invoked, a new symbol containing a placeholder for a reference designator is created. You may either begin adding objects to this symbol or load a different symbol from a library into the editor.

The editor is purely for graphic symbols. It will not allow multi-section parts to be created; the Part Wizard is used for that. However, the symbol editor has facilities to bind a symbol to a footprint and to specify a reference designator prefix. If this information is supplied in a symbol, then the system can turn the symbol into a part for immediate placement on a schematic page. When a symbol with this information is saved to a library, a generic entry is added to the taxonomy that allows the symbol to be searched and used as though it were a part.

Inside/Outside
You may have noticed the crosshatched rectangle in the work area in the above picture. This rectangle represents the interior portion of the symbol. Symbols are divided into an interior and exterior. Shapes will always reside in the interior of a symbol and pins always on the exterior, as shown to the left.

When adding pins, they will always be confined to the perimeter of the rectangle with the hotspot always pointing outwards.

If you draw a shape that extends outside the rectangle, the rectangle will be enlarged to encompass the shape. Any pins present on the sides that enlarge will move with the new rectangle. This is all done automatic for you.

You may also change the size of the rectangle yourself. With the Select tool active, you can move the cursor to any corner or edge of the rectangle to display drag handles, as show at left. Clicking and dragging one of these handles allows you to adjust the size of the rectangle. The rectangle cannot shrink smaller than the shapes it contains.

Besides being constrained to the rectangle edges, pins will always be located on a fixed grid, called the unit grid. The size of this grid happens to be 0.1” and is immutable for pin placement. This restriction, coupled with symbol rotations restricted to 90° steps, ensures that all pins land on a uniform grid on a schematic page, which in turn ensures wires are drawn on a uniform grid.

The Wire Router also takes advantage of the inside/outside separation by not routing into the interior of any symbol, which in turn means it will not cross any lines of a symbol.

Text is under no such restrictions. It may be placed inside or outside the rectangle. The Wire Router currently ignores text and may route through it.

The working grid in the symbol editor may be adjusted either in half by pressing the Minus key (-) or doubled by pressing the Plus key (+). This is only effective when drawing shapes or placing text. When placing pins the grid is always set to the unit grid.

**Adding pins**

Pictured below is the symbol editor as it appears with the Pin tool selected. It has been shrunk to view all the salient features in one picture. As soon as the Pin tool is selected, a new pin will be created that will track the cursor while staying constrained to the interior rectangle. As the pin is moved, it will snap to the unit grid and will automatically change orientation as it is moved to different edges of the rectangle.
While a pin is being moved, the Pin Attributes pane will be active on the left panel. It is here that you may fill out the information about the pin.

To release the pin at the desired location, just click the left mouse button. Adding pins is modal. When you left-click and place a pin, another one is created whose pin name and number are either auto-generated based on the last pin name or number, or they are obtained from a macro expansion. You must cancel out of the Add Pin mode by pressing the Esc key, selecting Cancel from the context menu, or selecting a different tool.

After creating two pins, a “stride” has been established that successive pins will add to the last pin location to derive a default location for the next pin. Both the pin and the cursor are snapped to this location allowing a whole row or column of pins to be quickly added by simply left-clicking without moving the mouse.

By default, the auto-generated pin name and number take the first sequence of digits from the previous pin name/pin number, and increment them by one. If no digits were present, it will change to the next letter in the alphabet.

At any time you may use bracketed shorthand notation in either the pin name field or the pin number field to establish a continuous sequence or a disjoint sequence of pin names and pin numbers to use for consecutive pins. The lifetime of the shorthand sequence is the duration of the pin tool being active. Once you cancel the pin tool, any remaining names in a shorthand sequence will be dropped.
A pin name must be supplied for a pin but pin numbers may be left blank. The only time you should do this is if the symbol is a generic gate or section and will be assembled into a part with the Part Wizard. It is perfectly ok to include pin numbers anyway in that situation.

If you are tempted to set a pin name or number to empty because you do not want it displayed, supply the name or number anyway and click on the eye icon to the right of the field so it shows this icon. This sets the visibility of the pin name or number to invisible.

If you bind a footprint to a symbol in the symbol editor, the pin numbers in the symbol will be checked against the footprint to ensure they match up before the symbol is saved to a library.

In the above picture, an example of each pin shape is shown. The pin shapes themselves carry no meaning in the software; they are merely added as a convenience so you do not have to draw the dot and clock symbolisms yourself.

Pin types do carry meaning in the software. The list of pin types forms a matrix of rules that specify which pin types are allowed to connect to other pin types. This is used during the automatic creation of wires when unconnected pins from two devices are placed on top of each other and during an Electrical Rules Check. By default, a pin is assigned the Passive pin type that is allowed to connect to any other pin type except N/C pins. It is recommended that you supply a pin type for symbol pins if you have the correct information at hand.

Finally, the pin properties button presents the standard Properties Dialog where you may specify any custom properties that you may wish to attach to a pin.

Adding Shapes

Pictured below is the symbol editor with the Shape tool selected. It also shows an outline shape and a filled shape that have already been created.
When the shape tool is selected, the symbol editor is placed in a mode where it is waiting for the first corner to be created by clicking anywhere in the work area. Once a corner has been created, a rubber band is drawn from the last corner to the current gridded cursor position. Clicking the left mouse button creates additional corners.

You can continuously “erase” the last segment by drawing back over the segment and click on the previous corner.

Finishing the shape can be done in two ways; the first way is to cancel or escape out of the mode and is the method to use when creating open polylines of one or more segments, or you can close the shape by creating the last corner at the same location as the first corner. This assumes there are at least 3 corners not in a line. Once finished, you are free to begin drawing another shape. Pressing escape or cancelling once again will put you back into the Select tool.

You may change the width or thickness of a shape outline by selecting one of the pre-defined line widths from the drop list in the Shapes attribute pane. These widths are provided for consistency. If a different width is required, you can specifically add a _LINEWIDTH property to the shape and setting it to the desired value.
Another outline attribute you may change is the Line Style. By default, shapes use a solid line style. Other styles are dotted, dashed, and dot-dashed lines. Only the solid line style will display outlines in true thickness.

The interior of a shape is controlled by the Fill Style attribute. By default, the fill style of a shape is hollow. The picture to the left shows the different fill styles available in the drop list. They are, in order from the top, Hollow, Solid, Horizontal Hatched, Vertical Hatched, Cross Hatched, and Diagonal Hatched. Selecting any fill style other than hollow will automatically close the shape if not already so.

The color of the outline and the shape interior may be independently set in the Shape Attributes pane.

Shapes may be assigned an ID. This can be any text string. The ID will play a roll in upcoming releases as a means to programmatically identify shapes.

Finally, the Shape properties button presents the standard Properties Dialog where you may specify any custom properties that you may wish to attach to a shape.

**Adding Arcs**

Pictured below is the symbol editor with the Arc tool selected. It shows an arc in the process of being created. It also shows a circle that was created with two arcs.
Creating an arc requires three clicks of the mouse. When the Arc tool is selected, the symbol editor is placed in a mode where it is waiting for the first click, which represents one end of the arc. The next click will be the other end, and the last click will be the arc apex. Once the first corner has been created, a rubber band is drawn showing the arc progress and a perpendicular guide is drawn showing the line upon which the apex may land.

The arc is automatically finished after the apex has been defined, and the symbol editor will be ready to create another one.

At any point during the arc creation, you may cancel or press the Esc key. Pressing escape or cancelling once again will put you back into the Select tool.

You may change the width or thickness of an arc by selecting one of the pre-defined line widths from the drop list in the Shapes attribute pane. These widths are provided for consistency. If a different width is required, you can specifically add a _LINEWIDTH property to the shape and setting it to the desired value.

Another outline attribute you may change is the Line Style. By default, arcs use a solid line style. Other styles are dotted, dashed, and dot-dashed lines. Only the solid line style will display outlines in true thickness.

Currently, arcs have no interior. They are outline objects only. The color of an arc outline may be set in the Shape Attributes pane.

Arcs may be assigned an ID. This can be any text string. The ID will play a role in upcoming releases as a means to programmatically identify shapes and arcs.

Finally, the Shape properties button applies to arcs and presents the standard Properties Dialog where you may specify any custom properties that you may wish to attach to an arc.

**Adding Text**

When you select the Insert Text tool, you will immediately be prompted for a text string to add using the dialog box shown below. If you have defined a user property with the same name as the string you enter here, the text string will be substituted with the value of the property. You may also expand the drop list to select the name of a system-defined property that will be substituted.

Once the string is defined, a text object will be attached to the cursor until you cancel or release the text string by left clicking.

After you place the text object, you will be prompted for another string. To exit either the dialog box or the Add Text Mode, press the Esc key or select cancel for a context menu.

The picture below shows the Symbol Editor while in Add Text mode.
Text attributes can be changed as you define text strings. Attributes include changing: font size, rotation, text color, background color, and alignment.

**Adding properties**

To add a property to a symbol component, right mouse-click on the component and select Properties... from the context menu. The picture below shows the Properties dialog for a pin in the symbol editor.
Users can select from the list of predefined properties, or create user-defined custom properties. Clicking OK stores the property on the property list for the symbol component.

**Saving symbols**

Click the SAVE icon in the Symbol Editor to save the symbol to the taxonomy. The picture below shows the Save Symbol dialog box. The Save Symbol dialog box contains three fields you must specify to complete save process.

- The **Name of the Symbol** will be the identifying name the symbol will use. Be descriptive.
- The **Description** is a text field in which you can describe the symbol in more detail.
- **Parts Library, customer\generics.sym** by default, is the library location to which the part symbol will be saved. There is also a New Library... button to allow you the ability to create a new custom library on-the-fly for your newly designed symbol.

**Sections**

You can schematically represent a single layout footprint with a symbol comprised of several sections. This is often done to represent the various gates of a particular component. To add additional sections select the “Add New” button. If you want to add a new section with an existing symbol select the “Add Existing” button. If you need to remove a section select
the “Remove” button. You can change the letter shown under section to view and edit any of the sections you have added or created.

**Binding to specific symbols**

The process of binding together a schematic symbol (or set of sections) and a layout footprint happens in the Define Part dialog (shown in the figure below)

![Define Part Dialog](image)

If you have already defined, or located, suitable symbols for schematic and footprint, you can quickly bind them together here.

- **Schematic Symbol** – click on Select Symbol… navigate to the schematic symbol, and select it. The schematic symbol appears in the schematic symbol preview window.

- **Footprint** – click on Select Footprint… navigate to the footprint and select it. The footprint appears in the footprint preview window.

Specify the part meta-data, part manufacturer, manufacturer part number, DigiKey Part number and a description, then select Create New Part.

If you wish to update the properties or symbols to an existing part, you can do that too. Just navigate the taxonomy to your part, then select it to load it into the Define Part dialog. Make the necessary changes, then select Apply Changes (instead of Create New Part) and submit your updates to the part definition.

**Mapping pin names to pin numbers**
Using the Symbol Editor in PCB123 V3.3 and later, it’s easy to map pin names to pin numbers. In this figure (below) the assignment of pin names and numbers can easily be changed on the fly. Just click on the appropriate text box for the pin number of the pin name, and replace the existing value with the new value.

Pin numbers can be in non-sequential order, pin names can be functional. Each pin number must be unique to the part, but pin names can be duplicated as needed.

**Hidden pins**

Hidden pins are often used to provide power and ground connections to components, but not clutter the schematic views with all the power related routing.
By clicking on the Pins tab at the bottom of the Symbol Editor pane, you can access the Pin properties table, as shown in the figure above. The Shape column allows users to specify a number of pin configurations, including Hidden P/G.

Once a pin is hidden, the pin structure for that pin disappears from the symbol graphics. The pin, however, is still present in the part definition.

A hidden pin is required to be identified as either a power or a ground pin. You will not be allowed to save your component until each hidden pin is so designated. To specify the type for the hidden pins, select the appropriate setting from the choices in the Type column pull down menu.

**Saving parts**

When you are ready to save your part to a library, select the SAVE function from the toolbar.

The SAVE function gives you access to the Save Symbol dialog, as shown in the figure above. Enter your name for the new symbol, an optional description, then the symbol (.sym) library to which the part will be added. If you need to create a new library, use the New Library... button on the dialog box.
Finding Parts

In PCB123 V3, locating schematic parts for use in your design falls into two primary methodologies: *Searching* and *Browsing.*

Searching: You enter search criteria; PCB123 filters the entire domain of available parts to match your search criteria. The PCB123 search mechanism is quite flexible, using a generally free-form text search that scans properties, property values, and part names for text string matches.

Browsing: allows you to click-navigate the taxonomy to find appropriate parts.

Further details for both methods of finding schematic parts can be highly effective, depending upon what you need at the time.

Part search

Searching for parts by search criteria is highly effective when you know one or more key parameters for the part you require.

Just type the parameters into the Search Criteria field, and the Select Part dialog filters the parts list down to the existing parts containing that criteria.
When browsing for parts, the taxonomy representation comes into play. As shown in the figure below, users can browse through the hierarchical folder structure interactively until locating the desired part.

If, in either case, the user cannot locate a predefined part to suit the specific design needs, the next step may be to build a new part from scratch.

**Define New Part**

The Define New part button at the top of the Select Part dialog starts the new part definition process. Clicking the Define New Part button takes you to the Define Part dialog, shown below. The Define Part dialog is quite powerful and flexible in how it approaches parts definitions. The key parts of the Define Part dialog include:

- The Parts Taxonomy pane
- The Properties definition table
- The Simple Tab for component specification
- The Spreadsheet Tab for advanced component specification
- The symbol pane, showing the assigned symbol graphics
- The footprint pane, showing the assigned layout graphics.
Whether you’re putting together a collection of pre-existing symbols to configure a new part, or whether you’re building everything from scratch, you’ll probably want to start by locating the taxonomy destination for the part before you do anything else.

In the figure below, the user is creating a subfolder in the taxonomy to hold the soon-to-be-designed component definitions. The context menu in this figure comes from right-clicking the mouse on an entry in the taxonomy window. Right-Clicking in this manner provides access to a number of commands for editing, organizing and modifying the taxonomy and its components.
Once the taxonomy location has been identified, the next step is to fill in the Manufacturer information, manufacturer part number, and Digi-Key part number, if known. User can also provide a free-text description of the component in the Description field. Notice that, as the part number information is filled out, a property entry for that information is automatically added to the Properties table in the lower left corner of the Define Part dialog.

Next up, is binding symbols and footprints to the part definition. Starting with the schematic symbol, the user has two choices: Select a Symbol, or Generate a Symbol.

Selecting a symbol presents the user with a search dialog, as shown below. Locate the symbol of your choice, using search parameters.
Once located, click on the Assign Symbol button to bind the schematic symbol to the part definition, resulting in a Define Part dialog that shows the newly bound schematic symbol.

Now, bind the footprint. Like selecting the schematic symbol, users have two choices: Select a footprint, or Generate a footprint.

Selecting a footprint presents the user with this Select Footprint dialog.

Following a usage model similar to the schematic selection process, In step one, users can search for a component by parameter, or browse the footprint libraries using the taxonomy-like view in the lower left corner of the dialog box. In step two, you select the footprint that matches the component you have
under definition. For step three, click the Select button in the center left. The figure below shows a Select Footprint dialog box ready for the binding process.

At this point, we've created a fully defined part. The figure below shows how the Define Part dialog box might look after binding both the schematic symbol and the footprint.

Click Create New Part, and reconfirm the location in the taxonomy. Once the part has been created, it appears in the taxonomy pane:
When you click OK on the Define Part dialog, you will be returned to the Select Part dialog (where this all started). Now, you will notice that your newly created part is available for use in the Select Part taxonomy, as shown below.
Select your part, and click the OK button. Your part is now ready to be placed in your design’s schematic. The figure below shows the part ready for placement.

Ports

A port is used to add a net label to pin, wire or group of connections on the schematic. Ports can be used to make connections between pins without connecting the pins with a wire. Ports can also be used to make connections between schematic pages.

PCB123 provides you with an AGND, DGND, Voltage, Global and Off-Page port. The labels for the AGND and DGND ports cannot be changed. You can select a port from the menu bar by selecting Insert > Add Port. A port tool bar is also included on the far right side of screen.

Ports and net names

After selecting either the Voltage, Global or Off-Page port you can change the net name assigned to it. You should see the Edit Panel in the on the left side of the screen. If you do not see the edit panel select View > Toolbars > Edit Panel. Start by selecting the desired port. Next, go to the top section of the edit panel. It should now show the label Net Name(s). Type in the desired net label in the Net
Name(s) field then place the port and connect it to a pin or wire. If you have already placed a port, and wish change the assigned net, then left-click on the port to select it. You will now see the Net Name(s) section in the Edit Panel. Update the text in the Net Name(s) field and replace the port. When you place the port you will see the message “Do you want to rename all ports ‘Net Name’ to ‘NewNetName’?” If you select Yes, all ports of net name NetName will be changed to NewNetName. If you select No, only the selected port will change to the new net name.
Schematic Pages

PCB123 can support multi-page schematic designs, allowing you to create logical sub-designs to manage the complexity of the schematic design.
Adding Schematic Pages

To add a new, blank sheet to your design project, click the Add Page item on the Toolbar (highlighted below) and a new, blank schematic sheet will be added to your design. The default name for the new sheet will be “Schem Page x”. A Schematic title block will also be attached to the new sheet.
Synchronization pages

A Synchronization page is a special type of schematic page used to capture ad hoc symbols created for footprints not defined on the schematic side.

The Title Block

The Project tab contains several report generators and a top-level navigator in the form of a tree that allows you to drill down into the schematic and layout data in minute detail. It also contains many top-level settings and defaults such as title block parameters that can be set for a project.

Page size

To set the page size for a schematic sheet, right click in the schematic pane. Select the Set Page Size... command from the Context commands, below.
The page size is specifiable by individual sheets. Your multi-page schematic design can accommodate multiple page sizes at the same time.
The Schematic Editor

PCB123 allows you to design your schematic and simultaneously populate your layout with the parts in your design. By default, each PCB123 design file contains a minimum of one associated schematic page, labeled as Main Schematic. In the Main Schematic tab you will find all the tools to locate, create, place and connect all the components that will comprise your board.

Adding parts to a page

Upon opening the Main Schematic tab you should be able to view the Part Taxonomy tab within the Edit Panel. (If you cannot view the Edit Panel, choose View>Toolbars>Edit Panel). Here in the Part Taxonomy you are able to drill down into one of the folders that organize parts, such as Generics>Chips>Dips in order to find all of the available generic symbols for dual inline packages. You can simply click on the part you wish to place then click on the schematic to place it. This provides a quick way to sort through available symbols by category.

You can also right click on any folder in the taxonomy list and choose Find to search for particular part numbers or types of parts. You could search for “amp” in order to find all parts that have been identified as amplifiers. Once the first matching item is found you can press F3 or right click and choose Find Next to select the next part that matches your criteria. This allows a more thorough search by specific criteria.

Once highlighted, a part is ready to place as soon as the cursor is moved back into the Main Schematic page.

To search for parts with more detail, one can use the Add Parts window. The button to open this window is represented by the NAND gate symbol in the Standard Toolbar. Once open users are first presented with the Search tab view. Here one can enter search criteria and click GO. Unlike the Part Taxonomy view, every part found will be visible at once in the list displayed on the right. This allows one to quickly cycle between each located part and view them in greater detail.

Pressing OK will close the Add Parts window and move the focus back to the Main Schematic tab. Here you can simply click to place the selected part.

Adding ports to a page

You are able to add ports to your schematic from two locations. The first location is on the menu bar. Select Insert > Add Port to select a AGND, DGND, Voltage, Global or Off-Page Port. Your second option for placing port is located on the far right side of your screen on the Ports toolbar.

Once you have selected the desired port you can place the port on your schematic with a left click. If you have selected a Voltage, Global or Off Page port you can modify the Net Name. With the port selected look to the top section of the Edit Panel and type in the desired net label in the Net Name(s) field. You cannot change the net name assigned to a AGND or DGND port.
To add a port to the schematic page currently under edit, select the port type you require from the Insert>Add Port menu. Port types available from this menu include:

- Analog Ground (AGND)
- Digital Ground (DGND)
- Voltage
- Global Net
- Off-Page Port

Select the type of port you require, and the port is assigned to the cursor for placement in the schematic. Place the port as many times as your require, then press the ESC key to unbind the port.

In the case of a voltage port, a global port and an off-page port, selecting the port will also present you with a pane in the edit panel. This pane can be used to set the specific voltage value or net name for the port to be placed.

**Adding wires**

To add a wire between pins on a schematic sheet, simply left-click on the first pin, then mouse over to the next pin and left-click the second pin. PCB123 will interactively draw a wire net that follows your mouse cursor. If you find that you prefer to route the net on a specific path, you can left-click on the schematic sheet to specify an interim location for the net. This function works much like digitizing a layout trace.

Wires can also be initiated and terminated anywhere along an existing wire. Doing so creates a connection to that wire, represented by a connector dot at that location (as shown at right).

Wires can be assigned properties. Wire properties can be accessed by placing the mouse cursor on the wire, then right-clicking to access the context menu, and selecting Properties...

**Adding nomenclature**

In the schematic, nomenclature is any text, unattached from a part, port or net, that has been added to the schematic for documentation purposes. You are free to use Text, Poly lines, Arcs, Circles and Polygons to add nomenclature to annotate your schematic. You can add these objects to your schematic by selecting Insert from the menu bar. You may also use the Sch Objects toolbar to access these tools. This tool bar is located across the top of your screen and is shown here.
Adding Text - select the function Insert > Add Text. You will be presented with a dialog box into which you can type the text you require. Clicking OK attaches that text box to the mouse cursor and allows you to place the text where needed on your schematic sheet. Press the ESC key to exit text placement mode. Right click the text to access display properties to change such features as font, text size, color, etc.

Editing the Title Block - this function is documented elsewhere in this manual. Please refer to: “The Title Block”, page 93.)

**Properties**

For all objects that support properties, properties can be accessed by placing the mouse cursor on the object, then right-clicking to access the context menu, and selecting Properties...

A full discussion of properties can be found elsewhere in this manual. Please refer to “Properties”, page 49.

**Cut/ copy/ paste**

The **Edit-Cut** command deletes the currently selected set of objects and places them into the paste buffer. There is only one paste buffer so only the last cut operation is available for paste.

The **Edit-Copy** command copies the currently selected set of objects into the paste buffer. There is only one paste buffer so only the last copy operation is available for paste.

The **Edit-Paste** command makes a copy of the objects in the paste buffer and inserts them into the currently active design.

First, any components in the paste buffer are added to the design by assigning them a new reference designator that does not conflict with one already in the design. Their alpha-prefix is retained.

Next, any complete routes or connections that start and end at a pins that are also in the paste buffer will be added to the design. They will retain their original net name with a unique numerical suffix.

Lastly, all text and non-net-attributed polygons will be copied without change. Net-attributed polygons will either inherit a cloned net name or retain their original net names depending on whether their original net names were cloned during the paste operation.

After all objects are created, they will be in a selected state attached to the cursor ready to be placed.

**Moving parts**

Depending on your experience, there may be a tendency to point at a component, hold down the LMB and drag that component. Though this is allowed, a much easier way is to point at the component and LMB click and release. This attaches the component to your cursor and allows you to do such things as use the RMB for context commands. You click once to select it and begin moving and click again to release it where you want.
Part 3 - Layout User Guide
Layout Views and Panels

Display Panel

One of the interesting things that PCB123 technical support notices when processing orders is the incredible diversity of color schemes and display settings that different users prefer. Some schemes are quite creative and pleasing to the eye while others seem to offend the senses. The point is that what works for some people, does not always work for others. This becomes especially important on very dense designs or designs that have a complex topology. These designs can lead to the subjective phenomenon of information overload. This can have a negative effect on productivity and can be a major contributor to mental fatigue. Beginners tend to be adversely affected more than experienced users but everyone has a different threshold of overload and fatigue and not everyone agrees on the best way to reduce it. It is for this reason that PCB123 employs a variety of global display settings.

Take a look at the design on the left.

This is a 6-layer design with everything visible, rendered in true width, and with color blending on. Even though the zoom factor is reasonable, the density of objects and the large angular fills make this screen difficult to look at. In this case, there is even confirming evidence of information saturation in that some features have turned white because so many objects with different colors occupy the same screen location.
This next picture is with the silkscreen layer set to invisible and the zoom factor has been increased to a greater magnification.

This does not hurt the eyes nearly as much but it is zoomed in so far that it is difficult to move anything without bumping into a window edge.

We will zoom back out to the original scale and try changing the display properties of the objects as opposed to the number of objects displayed.

The first thing we will try is turning off color blend.

As you can see, the data separation is much better but the opaque copper regions make it impossible to see the layers underneath. If we had to insert a via, we would be shooting blind and would have to keep changing layers to determine if the via would interfere with the obscured layers. Turning color blend off works best when there are no large fill areas.

Next we will try changing the Wireframe setting.

Wireframe does not fill any object interiors. It simply displays either the outline or the center line of an object
Now this is probably a reasonable screen to look at while finding room to insert a via. It provides visibility and selectability to all layers.

If we had to route an entire trace through the area we might try to use the Hi-Contrast mode in conjunction with Wireframe.

Hi-Contrast always displays the current layer in the contrast color and all other layers are displayed using the same dim color. This mode works great for cycling through layers looking for a path across the window.
Design View

The Design view, also identified as the Layout view is (along with the schematic view) one of the most-accessed views during the board design process.

3D View

The 3D viewer is a photorealistic rendering system that displays the board in a perspective projection. The board can be viewed from any angle in 3D space. 3D View is good for identifying problems that the DRC engine might otherwise treat as valid. These may include problems with drill sizes, holes where there should be none, missing holes, plated/unplated hole problems, and unreadable silkscreens.
Footprint Editor

Designing a new footprint requires the use of the footprint editor. The footprint editor can be used as a stacked-interrupt edit function. In other words, as you're laying out your board, you may discover that the part you wish to use is not yet represented with a footprint in your current libraries. By selecting Design -> New Footprint… you enter the footprint editor window. Note that your layout editor work is maintained in a edit pane underneath the footprint editor.

You can learn more about creating footprints in the section titled Create Missing Footprints (page 107).

DRC Panel

The DRC/ERC tab contains a live listing of all the current design violations in the project.

The DRC will report on shorts, opens, and violations of manufacturing rules such as minimum spacing between objects of different nets, and pad size vs. drill size violations.

Locations of errors reported by the DRC are marked with a grey circle-? icon. By hovering the mouse pointer over the error, users can access a summary explanation for the cause of the error being reported.

The DRC/ERC tab also allows users to access violations by category or by layer. At the top of the DRC/ERC tab are the ‘stepper’ functions, allowing users to move from error to error so as to view/fix each one interactively.

Project Panel

The Project tab contains several report generators and a top-level navigator in the form of a tree that allows you to drill down into the schematic and layout data in minute detail. It also contains many top-level settings and defaults such as title block parameters that can be set for a project.
The Statistics Report gives you a summary report on various board metrics for the currently active design, including: board area; routing layers; plane layers; number of components; pins per sq/inch; number of nets; etc.

The Netlist Report generates a (WHICH!!) format netlist to a file location of your choice. You can also opt to view the netlist output in a notepad window.

The BOM Report generates a text file bill of materials in the old-fashioned CAD system part-list format.

The Board Configuration button presents the board configuration wizard dialogs, comparable to the Design -> Board Configuration menu item.

In the Schematic Properties pane, you can edit/update the default settings for schematic sheet properties.

In the nested properties view, near the top of the Project tab, you can navigate down through attributes for virtually every design element in either the schematic or layout side. This includes symbols, footprints, nets, and their individual attributes and identifiers.
### PCB Design Tasks

You can view this section as a recipe for creating a PCB. The tasks are listed in roughly chronological order to begin and complete a PCB design. It assumes you are starting the layout from a state listed in *Starting a New Design*. Some tasks may not apply to your design.

1. **Create Missing Footprints**
   - Creating footprints from scratch
   - Modify a library footprint
   - Create a footprint from a component

2. **Prepare Mechanical Features**
   - Adjusting the board outline
   - Add mounting hardware
   - Mechanical restrictions

3. **Place Components**
   - Fixed component locations
   - Autoplace components
   - Manually placing components

4. **Power/Ground Critical Routing**
   - Power/ground plane layers
   - Routed power and ground
   - Critical net routing

5. **General Routing**
   - Fanout and autorouting
   - Manual routing
   - Routing techniques

6. **Fills, and Nomenclature**
   - Active copper fill regions
   - Nomenclature
   - Silkscreens and soldermask

7. **DRC and Visual Inspection**
   - Design Rule Checking
   - Visual inspection
   - Best practices

8. **Engineering Changes**
   - Forward annotating changes
   - Back annotating changes
   - Netlist comparison

9. **Order Boards**
   - The order process
   - What you will receive
   - Other services
Create missing footprints

PCB123 supplies a library of footprints. Even so, it is inevitable that you will have to create a custom footprint or modify an existing one at some time. Footprints are created or modified in the Footprint Editor, which is very similar to the board editor but does not allow the net or route tools to be invoked.

Additionally, layers are treated generically. There is no concept of a specific layer number, only layer types such as Top, Inner, Plane. In this way, footprints are created board-agnostic. When a footprint is instantiated as a component, features in the footprint are mapped to the physical layers of the board based on the layer type. Some layers do not differentiate between top and bottom such as Silkscreen, Assembly, and Soldermask. This is because any feature placed on these layers will be mapped to the proper layer based on the side of the board any instantiating component is placed. This is a dynamic behavior. As you move the component between top and bottom (by using the Opposite command), the features on the above footprint layers appear on the correct board layers.

Very rarely will you create footprint features on the Plane or Soldermask layers. Any feature created on the Plane or Soldermask layers will result in copper or soldermask being voided, not added to those layers. These layers are traditionally treated as “negative” images. They are a continuous plane of material voided only where objects such as pads have been defined.

PCB123 does not have a traditional padstack object. A pad for a pin is defined simply with a size, shape, and drill. If the drill size is zero then the pad is a surface mount pad and will only appear on the top or bottom routing layer depending on which side of the board the component is placed. If the pin’s drill size is not zero, then the pin is considered a thru-hole pin and a pad will appear on all layers. You are free from the burden of specifying plane layer and soldermask pad features. Plane-layer clearance pads are parametrically derived from the drill size and soldermask clearance pads are parametrically derived from the pad size on the surface routing layer. This not only relieves you of the burden of defining the data but it allows Sunstone to optimize certain processes over time.

Footprints are defined in their own local coordinate system. When a component is placed on a board, its features will be relative to the origin of the footprint. It is recommended that a pin such as pin 1 should be placed at 0, 0 in the footprint. This helps to align component pins during component placement and makes life easier when routing. If you right-click on a pin in the Footprint Editor then one of the context commands is Use Pin as Origin. This will shift everything so that the pin is the new origin.

You can locate a pad at a specific coordinate by invoking the Pin Property dialog. There you can enter the specific X, Y coordinates for the pin using the full power of expressions. This can be quite useful when the mechanical specs locate pins relative to other pins.

If a mechanical drawing is referenced from the bottom view of a part, go ahead, create the footprint in the same way, and then from the main menu select Tools/Mirror All Geometry. This will flip everything back to a topside view, which is how a footprint should be defined.
When you need to create a custom footprint, you can create one from scratch, modify an existing footprint from a library, or create a footprint from a component on a board.

### Creating footprints from scratch

With a document open, you can select Design/Create Footprint from the menu. This will open an untitled Footprint Editor window whose only contents are a reference designator placeholder and the origin marker. When you are finished creating the footprint and you want to save it to a library you will have to name it and choose a library to save it in.

### Modify and existing footprint from a library

If you can identify an existing footprint in a library that closely matches the footprint you need then it may be advantageous to start with the close match. Select Design/Edit Footprint and browse the library to select which footprint to edit. This will open a new Footprint Editor Window fully populated with the selected footprint. After making the edits you want, choose the File/Save Footprint As command. This will show the Save Footprint Dialog with all the original information retained such as the footprint name and the library it came from. It is important that you do not overwrite the original footprint; you must choose either another name or a different library to save the modified footprint into. If by accident you overwrite the original, undo all your changes to the footprint and save again. Then you can redo all the changes and save correctly this time.

### Use an existing component as a starting point

If you have a component that is close to what you need or you just want to update one or more similar components on a design then you can point at a component in the Design View, right-click, and select Edit Footprint from the context menu. This will open a new Footprint Editor Window fully populated from the contents of the component. To save the footprint to a library then choose the File/Save Footprint As command. Whether you save the footprint or not, closing the window will prompt you with the Update Components Dialog.

### Adjusting the board outline

When you first create a new design, you acquire a board outline in one of five ways:

- use the default board outline in which case you have a 5" x 3" rectangle,
- specify some other rectangle,
- load a board template that already has a board outline,
- use a gutted previous design for the board outline.
- Import and IDF outline from your mechanical CAD tools via the PCB123 -> IDF Import/Export plugin.

In any event, you may need to adjust the board outline manually.
Place components

If you did not start with a netlist then you are probably inserting components manually and placing them roughly where you think they should go as best you can without the aid of a netlist.

If you did start with a netlist then all the components are spread out beneath the board outline showing the ratsnest of connections to be routed.

Fixed component locations

The most logical way to begin placing the board is to identify all the components with fixed locations and place them first. You can right-click on a component to display its properties and from there enter the precise location for that component. Even if the locations are not precisely fixed starting you placement with the I/O circuitry is usually a smart strategy.

Before you start placing the bulk of the components, you should select a reasonable placement grid for the components. The larger the grid the easier it is to align and uniformly distribute components. On very dense boards, it may become impossible to place every device on the same grid. Having the bulk of the pins on a uniform grid can pay big dividends while routing.

After placing fixed, I/O, or generally critical components there are a couple of strategies to employ for placing the remainder of the components:

- If you have the schematic handy then you can press Ctrl+F to find and select other components and snap them to the cursor for placement.
- If you started with a netlist, you can also use the resulting ratsnest as a placement guide. By roughly placing the devices with the larger pin counts, you try to minimize the overall connection length by placing components that heavily interconnected close together.

These are just some rough guidelines. You design may have some restrictions such as critical nets that dictate a less-than-optimal placement for overall routing, or there may be height restrictions on some devices.

Autoplaces components

PCB123 V3 introduces an Autoplacer function as well. While you create the schematic for your design, PCB123 tracks both the schematic symbol in the schematic sheet, and the associated footprint for that schematic symbol. As you place symbols, the footprints and ratsnest connection data are queueing up on the layout view, ready for placement and routing.

To get a quick start on the layout process, you may choose to use the menu item, Design -> AutoPlace. Using this function, PCB123 will make its best effort to pre-arrange component footprints in a way to minimize the amount of routing required.
Smart Autoplacement is a complex algorithm that often relies heavily on heuristics and ‘best practices.’ The AutoPlace function in PCB123, while useful, is not intended to be a full-blown autoplacer in the spirit of some other enterprise-level tools.

Manual place components

Pragmatically speaking, whether you’ve used AutoPlace or not, you’ll probably want to do some amount of manual placement of components.

If the component footprints are already placed on your layout view, you can manually adjust the component placement. Click on the selection tool. In selection mode, hover the mouse pointer over a component until the mouse cursor turns into a chip package icon. Then left-click the component to enter move mode. The component will ghost interactively on the screen; place the component in its new location. Ratsnest data will follow; previously routed connections will ratsnest from any broken connections to the component in its new location.

Power, ground, and critical routing

Power and ground plane layers

In this section, we’ll show you how to add a Plane to the routing layer, and name the plane ‘+12v’

In this case we have selected a trace on the +12V net. Clicking on the Unroute Net menu item will remove+12V trace routing from all layers

Set the edit layer to layer Bottom, using the edit pane on the left side of PCB123.
Click on the add polygon icon, draw a polygon around the board to cover the area under all +12V pins from any layer. It helps to check bottom layer routing and adjust any traces that will cut across the plane.

The plane you draw will be just an outline with default line width. Right mouse over the new polygon you drew and select Toggle Filled/Unfilled.

This is your filled polygon. It is named 12VPLANE, but not attached to a net yet.

Right mouse over the plane area again, select the net name (+12V) and ok the form.
Now, Route all +12V pins out to a breakout via to finish your connections.

Reroute your pre-existing traces, and cleanup.

**Critical routing**

Some nets may have special needs such as a minimal routing length or routed in a particular sequence from pin to pin. Often it is easier to deal with these nets at the beginning then later on in the design when things may get dense.
Fanout and autorouting

If your board has a considerable percentage of surface-mounted devices then you may want to perform a Fanout on your design, especially if your board is dense or has more than two routing layers. The idea behind this is that you have to route out of the SMD pads on the layer they are on so you may as well make sure you can at least escape the pad to an immediate via. Once this is done, you have the option of routing the remainder of the connection on any suitable layer. Besides automatically fanning out a board you can do it manually by routing out of a pad and then pressing the ‘V’ key to terminate the route at a via. You can always remove unneeded vias after the board is complete.

On dense boards, running the autorouter can be beneficial in determining trouble spots on the board. The autorouter settings can be modified in the User Preferences Dialog. You can always undo the results of an autoroute. Autorouting may be an acceptable solution on sparsely to moderately dense designs or designs that have very good connection ‘flow’ to them.

Manual routing

Using the Freestyle tool you can freely insert routes from pins and other routes. You can even define new connectivity this way. There is no auto-finish because there is no end-point defined. The route is terminated by pressing the escape key (leaving a dangling end) or by routing to another pin or route.

DRC and Visual Inspection

The DRC engine is a key component to the PCB123 design process. PCB123 captures the Sunstone manufacturing parameters as DRC rules. If the DRC engine returns a clean report, then there are no known design issues that would affect the manufacturability or yields for your design.

Not all design rule violations, however, would be considered a factor for manufacturing. In some instances, what is strictly a violation of the manufacturing tolerances may, in context, be considered allowable. As a designer, you have the ability to submit a design to Sunstone even with design violations. You do, however, take responsibility for any manufacturability issues that might arise from violations left in the design file.

If you have questions regarding whether a design violation would impact your design’s manufacture, you can always contact Sunstone’s PCB123 Customer Service team for clarification.
Design Rule Checking

The DRC/ERC engine provides three styles of reporting: interactive, batched and at-order.

Interactive: The DRC/ERC engine will check for violations as you perform the layout. Place a route in violation of space/trace parameters, and PCB123 will immediately flag the location of the violation.

Batched: The DRC/ERC checks can be run as a batch as well. This choice is at the user's discretion for timing, and can be initiated with the menu item Design -> Run DRC.

At-order: When you initiate an order, PCB123 will first run a batch version DRC/ERC check and report the status of the check to you. We expect that users will review any DRC/ERC errors reported at this time and choose to either correct the violation(s) or ignore the violations. Please note that proceeding with the order process from this point indicates that the user is choosing to ignore any violations still present in the design file.

Visual inspection

Prior to placing an order, it is recommended that you use the 3D View tab to inspect a 3D model of your board design. In this way, you can identify design issues that may pass the scrutiny of the DRC rules, but not be according to your specified needs. This might include incorrectly sized drill holes, or hole plating mismatches, among other issues.

To use the 3D viewer, just click on the 3D View tab at the bottom of the PCB123 application. If this is your first time in the 3D viewer, you may be presented with a notice dialog indicating that 3D view is beta functionality.

In the 3D view, you can rotate, pan and zoom the view to inspect your board.
Order Boards

PCB123 is unique in that the tool tracks your design against a fully documented set of manufacturing process parameters. In fact, because of the close connection between the tool and manufacturing capabilities at Sunstone Circuits, PCB123 can even provide real-time quoting for board fabrication as you design.

The in-tool quote-and-order functionality provides you with two key benefits:

1. You can design to your fabrication budget, and get immediate notification if a design change affects your pricing.
2. Thanks to the web interface ordering process, you can efficiently submit your design to Sunstone for manufacture, with a clear understanding of the manufacturability (thanks to DRCs) and pricing for your order.

The order process itself is quite easy On the following pages, we’ll walk you through the ordering process.

The Order process

1. Pre-Order DRC check

![PCB123 DRC check dialog]

There are 4 errors and 80 warnings still being reported.

IF you choose ignore these errors/warnings and place the order as is, PCB123 does not assume any liability for the finished product.

Do you wish to proceed?

Yes  No

2. Sunstone Account Log-in

![Sunstone Account Login]

Email Address:  nohsonesunstone.com
Password:  ********
Remember me on this computer
OK  Cancel
3. Board Configuration Review Page. You can review the settings used for the board being ordered. On this page, at this point in the process, you can modify the board quantity, and the ROHS manufacturing options. Any other configuration options will require quitting the order process and accessing the board configuration inside the editor environment.

4. Add On Services Page. Additional services can be selected here. As of this printing, PCB123 provides the following cost-adder services:

- Flying Probe Electrical Testing
- Assembly services (via a partner company)
- Expedited manufacturing

5. File Transfer. Click the transfer button on this page, and your PCB123 design file will be transferred via your internet connection to Sunstone Circuits. Along with the design file we will receive the board configuration and basic manufacturing parameters you specified in step 2. Please note, your order is not yet complete. But, the only remaining steps are the payment details.
6. Order Summary. After the transfer is complete, order control transfers to a web browser page similar to the example below. Here you can either enter credit card or billing information or, if you have credit card information already on file in your Sunstone account, you can choose the pre-loaded information. This page confirms the cost-adder services you may have selected, and also shows you the estimated ship date. Please be aware that, for some payment options like Phone/FAX, your order will not be released to manufacturing until payment information is received by Sunstone Circuits. As such, the ship date is an estimate until payment is confirmed.

Click next on this page, and you proceed to order confirmation

7. Order Confirmation: Final confirmation of the order, services, ship date, options, etc. At this point, all payment and manufacturing information is complete. Clicking the SUBMIT ORDER button places the order and initiates the process to charge the credit card account.
And that completes the order process. Your order appears in the PCB123 order queue at Sunstone Circuits and immediately moves onto the manufacturing floor for tooling, imaging and manufacture.

You can check on the status of your order (including where it is in manufacturing) at any time by logging into your account at www.sunstone.com.
Before you route

After you have all the components placed on the board, you may need to setup special rules for some nets. These include:

- **Plane Layer Nets:** If you have plane layers, you will need to establish which nets are tied to the planes. You can navigate to the nets in the Object Hierarchy or graphically select the nets and use RMB Properties to get to the Net Properties Dialog. Assuming you have already identified at least one inner layer as a plane layer you can click on the plane layer checkboxes in the net properties dialog. This tells the system that as soon as a pin connects to a plane layer either because it is a thru-hole pin, it is routed to another thru-hole pin or via, then it is considered connected. If you have multiple nets assigned to the same plane layer you will have to isolate the pins from each net by adding a polyline that encompasses just the pins of one of the plane nets. This has to be done manually.

- **Special Net Widths:** Depending on the copper weight of the board, the default width of nets will be either .007” or .012”. This may not be adequate for certain nets, especially those that may be carrying higher current. You can set the line width for these nets in Net Properties Dialog.

- **Special Net Spacing:** Depending on the copper weight of the board, the default spacing for nets will be either .007” or .012”. This may not be adequate for certain nets, especially those that may be carrying a high potential and need to meet regulatory requirements or nets that are edge sensitive and need to ensure minimal coupling. You can set the net spacing in the Net Properties Dialog.

You can also set Per-Layer spacing rules in the User Preferences Dialog.

Routing the board

After you have all the components placed on the board, your next task is to route all the nets. There are many factors to consider and there are many tools to help. Generally, you want each layer to have a preferred routing direction or ‘flow’. Too much meandering on a layer most likely will cut off routing channels before all the routing is complete.

As for a routing strategy, here is a sequence to consider:

1. **Fanout:** If your board has a considerable percentage of surface-mounted devices then you may want to perform a Fanout on your design, especially if your board is dense or has more than two routing layers. The idea behind this is that you have to route out of the SMD pads on the layer they are on so you may as well make sure you can at least escape the pad to an immediate via. Once this is done, you have the option of routing the remainder of the connection on any suitable layer. Besides automatically fanning out a board you can do it manually by routing out of a pad and then pressing the ‘V’ key to terminate the route at a via. You can always remove unneeded vias after the board is complete.
2. **Critical Routing:** Some nets may have special needs such as a minimal routing length or routed in a particular sequence from pin to pin. Often it is easier to deal with these nets at the beginning then later on in the design when things may get dense.

3. **Short connections:** Short connections are very easy to route at the start of a design but because of limited options can be extremely difficult to route on a dense board if saved until the end.

4. **Wide Traces:** for the same reasons as mentioned above.

For general routing, you have several options available to you:

- **Autoroute:** This may be an acceptable solution on sparse to moderately dense designs or designs that have very good connection ‘flow’ to them. On dense boards, running the autorouter can be beneficial in determining trouble spots on the board. The autorouter settings can be modified in the User Preferences Dialog. You can always undo the results of an autoroute.

- **Partial Autoroute:** You can select a connection or group of connections, right-click the mouse, and select Autoroute from the context menu. This will engage the autorouter for just those connections.

- **Pin-To-Pin Manual Routing:** If you started your design with a netlist then you can just click on a connection (ratline) and start digitizing corners. Because there is knowledge of where the connection terminates you can double click at anytime and the system will automatically finish routing the connection using a straight line or an orthogonal pair of lines. This type of auto-finish does not avoid obstacles and may produce DRC markers.

- **Freestyle Manual Routing:** Using the Freestyle tool you can freely insert routes from pins and other routes. You can even define new connectivity this way. There is no auto-finish because there is no end-point defined. The route is terminated by pressing the escape key (leaving a dangling end) or by routing to another pin or route.

For manual routing, there are two broad strategies. One of them is intuitive and the other is less intuitive but can be a very powerful technique on dense designs.

**Complete each Connection, one at a time:** this is an obvious technique that involves selecting or inserting connections, one at a time, and then routing them to completion. It may require a lot of panning around and can possibly lead to a large amount of rerouting toward the end of dense designs.
**Sweep route one window at a time:** take a look at the picture below. It shows one completely routed window of a design. The idea is to zoom in to a comfortable magnification and to route everything in that window. If a connection passes through the window on a diagonal, then drag it to a corner and ‘tack’ it there because it has no business in the current window. If you did not turn off angle snap in User Preferences then you may have to override the snap behavior by holding down the Ctrl key while routing. For connections that pass orthogonally through the window use the same tactic but find an open channel in the window and route it through the window. Once the window is complete, pan to the next window and route that one. You should start in the densest region of the board and pan against the long axis of the board, i.e. if the board is mostly horizontal then pan vertically as your primary pan direction. This very powerful technique results in less rerouting and can identify serious trouble quicker than any other method.

**Adding copper regions**

If you need to add shielding around nets or generally make other nets as wide as possible a simple method to do this is to add copper pour regions to your design. Simply draw a closed polygon that has been assigned to a net. Objects that are contained within the polygon region that do not belong to the polygon’s net will be isolated from the copper region. Those that are assigned to the same net will be tied to the copper region.

In the picture below, NET1 is surrounded by a copper pour that has been assigned to the GND net. Copper pour does not flood over anything that has a different net property than itself. Even pins that do have the same net property as the copper are cleared out and tied to the copper through ‘spokes’ to help soldering by focusing the heat on the pin.
The default isolation gap between copper pour and other objects is 0.008" for standard 1oz copper clad and 0.013" for 2.5oz copper clad boards. This clearance may be increased on certain objects whose net spacing rule dictate a greater clearance, including the copper pour net itself. By specifying worst case spacing on grounds and voltages, copper pour can be safely used while guaranteeing adherence to standards and regulation bodies requirements such as Underwriter Labs (UL).

Care must be taken when adding copper pour to congested areas so that pins are not isolated. Pins that belong to the same net have connection lines, or a ratsnest between them. The connection lines disappear when:

- A continuous routed path is completed between the pins.
- The pins’ net is assigned to a plane layer and the pins hit the plane layer through a plated hole.
- The pins are contained in a copper pour region assigned to the same net.

Because the display of the ratsnest is calculated on the fly, connections made with copper pour are assumed to be complete. However, this may not always be true. It is possible to isolate pins by carving up the copper pour region with too many other-net objects.

Running a full DRC check on the board will detect possible isolations and the resulting error markers should be investigated carefully. The DRC check for isolations is pessimistic. First, it rasterizes the copper region at some minimum feature size so a pin might be flagged as isolated even though it may be connected by one or more slivers smaller than 0.004". Second, the isolation check is performed on a per-copper pour basis without regard to the rest of the net topology. This means that two pins inside a copper pour region of the same net may be reported as isolated even though they are connected by a different means such as another copper pour region on a different layer or even a plane layer. This may be corrected in the future but is currently considered beyond the scope of the software.
An example of isolated pins and the corresponding DRC error marker is shown to the left. Note that only one pin is reported as isolated as the other is considered connected to the copper. Only the smallest number of isolated pins is reported by the DRC. The largest connected set of pins is not reported as isolated to reduce the number of error markers generated.

You can also add solid regions of copper that do not clear anything out by simply leaving the net property blank.

If you add a polygon to a plane layer or a soldermask layer, you are actually clearing copper or soldermask from that region because those layers are assumed to be continuous planes and anything on them is considered a break in the plane.

**Design Changes**

Design changes are an inevitable unpleasant fact so PCB123 has provided some tools to make incorporating design changes as simple and painless as possible. There are two methods for performing design changes. They are:

**Manual Changes:** some types of changes are obvious. Adding a new component is as simple as clicking on the Add Component toolbar button and adding a connection is as simple as clicking on the Add Connection toolbar button. To delete a component just point at it and press the Del key on your keyboard. To remove a pin from a net use the connection tool, right click on a pin, and select Disconnect Pin from the Context Menu. You can change the footprint for a component in Component Properties Dialog.

**Automatic ECO:** if your design is captured in a schematic then a very powerful tool available to you is the Automatic ECO feature. It will compare a design against a netlist and perform all the necessary changes to the design to make it match the netlist. It will do this in a highly optimal manner, disturbing the existing design as little as it can. It will even re-bind component footprints so if nothing changed on your schematic except a part type change then it will detect that in perform the update.

**Design Rule Checking**

One of the last tasks you want to perform on your design is Run DRC. It is important enough that it is automatically run before you place an order. During the course of design, certain checks are done dynamically - mainly checks for shorts. Running a full DRC not only detects shorts but it also will detect opens and a range of fabrication rules such as pad annular ring size and drills too close together. Warnings may also be generated such as silkscreen on exposed copper areas such as pads.

After DRC has been run you can cycle through the errors and resolve them. Pressing the ‘N’ key on your keyboard will pan you to the next error. When you have reached the end of the error list, you will be prompted to run another DRC to check if you cleared them all.
Object Editing

This section describes the operation of the commands used to edit objects in the PCB123 tool environment.

Setting the grid

The quickest way to change the grid is to press the ‘G’ key on your keyboard. You can also select Grid from the Edit menu.

The working (snap) grid and the display grid are not independent.

See the Grid Command for more information.

See the Grid Dialog for more information.

Changing layers

There are many ways to change the current layer but the easiest way is to type the layer number on the keyboard. The layer numbers are:

- 1 - Top layer
- 2 - Bottom layer
- 3-6 - Inner layers (if present)
- Ctrl+1 - Top soldermask (if present)
- Ctrl+2 - Bottom soldermask (if present)
- Shift+1 - Top silkscreen (if present)
- Shift+2 - Bottom silkscreen (if present)

In addition to typing layer numbers, you can select View/Layer from the menu. You can also LMB click a layer in the Layer Display panel shown below:
Lastly, you can use the Layer toolbar (shown above) to click among the active layers.

Changing layers is allowed while editing many objects. For instance, while you are routing you can change the layer to insert a via at the last corner.

**Changing line width**

The simplest way to change the width of a line while drawing is to press the ‘W’ key. This will invoke the [Line Width Dialog](#). You can also RMB to get the context menu, which has a Change Line Width item.

**Add a component to the board**

If you want to manually add a component to your design (as opposed to automatically with a netlist) then the simplest way to do this is by clicking on the Add Component Button  on the Insert Object Toolbar. You can also choose Insert/Component from the menu bar. Either way you will be presented with the [Add Component Dialog](#) that allows you to graphically browse for a footprint that the component will use.

**Move a component**

Depending on your experience, there may be a tendency to point at a component, hold down the LMB and drag that component. Though this is allowed, a much easier way is to point at the component and LMB click and release. This attaches the component to your cursor and allows you to do such things as use the RMB for context commands. You click once to select it and begin moving and click again to release it where you want.

**Move a component to the other side of the board**

Once you have a component selected the easiest way to move it from one side of the board to the other is by pressing the ‘O’ (for opposite) key on your keyboard. Alternatively, with the component selected you can RMB to display the context menu and select the Opposite command.

A component placed on the bottom of the board is automatically mirrored for you.

**Rotate a component**

Once you have a component selected on your cursor the easiest way to rotate it is to press the ‘R’ key on your keyboard. Alternatively, while you have the component on your cursor you can RMB to display the context menu and then select the Rotate command.

Component rotation is always performed counter-clockwise.
By default, component rotations are performed in 90-degree increments but you can change this default in the User Preferences - General dialog.

**Edit a component**

There are two ‘levels’ of component editing: you can edit information about the component and you can edit the contents of the component.

To edit information about the component and you have one or more components selected you can press Alt+Enter on your keyboard or alternatively RMB to display the context menu and select the Properties command to invoke the Component Properties dialog.

To edit the contents of a component, select a component, RMB to display the context dialog, and select the Edit Footprint command. This will invoke the Footprint Editor, which allows you to graphically modify the component contents such as pins and outline data.

**Add a pin to the board**

If you want to manually add a pin to your design, then the simplest way to do this is by clicking on the Add Pin Button on the Insert Object Toolbar. You can also choose Insert/Pin from the menu bar.

Once you have selected this command, a default pin will be attached to your cursor and there will be a Pin Context Pane in the left panel that allows you to edit some of the more common pin properties.

**Move a pin**

Only board-level pins can be moved in a design. To move component pins, you must edit the component in the Footprint Editor.

Depending on your experience, there may be a tendency to point at a pin, hold down the LMB and drag that pin. Though this is allowed a much easier way is to point at the pin and click it. This attaches the pin to your cursor and allows you to do such things as use the RMB for context commands. You click once to select it and begin moving and click again to release it where you want. Try it, it feels good.

**Change a pin**

If you have one or more pins selected you can press Alt+Enter on your keyboard or alternatively RMB to display the context menu and select the Properties command to invoke the Pin Properties Dialog which allows you to change any pin property.
Add a polygon or polyline

If you want to add a polyline or filled polygon to your design then the simplest way to do this is by clicking on the Add Polygon Button 🗺 on the Insert Object Toolbar. You can also choose Insert/Polygon from the menu bar.

Once you have selected this command, this symbol 🗺 will appear next to the cursor informing you that you are in Add Polygon mode and the system is waiting for you to click the first corner. After clicking the first corner, a line segment will then be drawn from the last corner clicked to the current cursor location. To stop digitizing the polygon you can either press the Esc key on your keyboard or just double-click the last corner.

If you double back over a line segment, the overlapping portion of the segment will be removed. You can keep doubling back to the previous corner to incrementally undo the polygon.

Once you have selected this command there will be a Polygon Context Pane in the left panel that allows you to edit some of the more common polygon properties.

You can also perform many of the global commands such as Change Grid while creating polygons.

Add a circle

If you want to add a hollow circle to your design then the simplest way to do this is by clicking on the Add Circle Button 🗺 on the Insert Object Toolbar. You can also choose Insert/Circle from the menu bar.

Once you have selected this command, this symbol 🗺 will appear next to the cursor informing you that you are in Add Circle mode and the system is waiting for you to click the circle center. After clicking the circle center moving the mouse simply adjusts the circle radius. Click again at the desired radius to accept.

You can also perform many of the global commands such as Change Grid while creating circles.

Add an arc

If you want to add an arc to your design then the simplest way to do this is by clicking on the Add Arc Button 🗺 on the Insert Object Toolbar. You can also choose Insert/Arc from the menu bar.

Once you have selected this command, this symbol 🗺 will appear next to the cursor informing you that you are in Add Arc mode and the system is waiting for you to click one end of the arc segment. After clicking the arc start, you will be rubber banding a line. Click again to establish the arc end then any mouse movement will adjust the radius. After you have the desired radius click again to accept the arc.

You can also perform many of the global commands such as Change Grid while creating arcs.
Move a polygon

To move an entire polygon just left click on it and drag, releasing the mouse button when you have position the polygon where you want.

Move polygon corners

Moving a polygon corner is simple: just point at a polygon corner, left-click, and release. Now you are moving a polygon corner. Just click again to release the corner where you want.

Delete polygon corners

There is no explicit Delete command for a polygon corner because that would require a mode change or some command that has to executed by pressing a key or clicking a button. Instead, deleting a corner can be accomplished with the mouse simply by selecting a corner and dropping it on the previous or next corner in the polygon. It is called corner reduction. It is very efficient, and can all be done with the mouse.

Insert corners in a polygon

To insert a corner in a polygon simply click on the middle third of a segment or at least .050” away from a corner. You will then be in digitizing mode that allows you to add corners until you stop by pressing the Esc key, double clicking, or inserting another corner at the same location as the previous one.

Add a connection between two pins

To add a connection between two pins click on the Connection Tool to place the system in Add Connection Mode. Once in this mode simply click on the first pin you want to add the connection between. After clicking on the first pin, you will be rubber-banding a line while the system waits for you to click on the other pin that completes the connection.

If neither pin is tied to a net then a new, unique net name will be created for the connection. If one of the pins is tied to a net, then the connection and the other pin will inherit that net name. If both pins are tied to different nets then you will be prompted to merge the nets together.

Remove a pin from a net

To remove or disconnect a pin from a net first click on the Connection Tool and then RMB over the pin and select Disconnect Pin from the context menu.
Alternatively, you can navigate to the pin in the Object Hierarchy tree and select Properties from the context menu. This will invoke the **Pin Properties Dialog** where you can either disconnect the pin from any net or assign it to a different net.

---

### Start routing from a pin

If there is already a ratline (connection) tied to a pin simply click on it to begin routing. If the pin is a thru-hole pin then you will be routing on the current layer. If the pin is an SMD pin then the route will begin on whatever surface layer the pin is on. Once you have begun routing you can continue to digitize new corners until the route is completed or you press the Esc key to stop routing.

Alternatively, you can click on the FreeStyle Routing Tool and then click on a pad to start routing. The same features as above apply in this mode with the only difference being if the pin was not previously tied to a net then a new, unique net name will be created with the pin and the route tied to it.

---

### Start routing from another route

You can start a new route from a previous on by clicking on the FreeStyle Routing Tool and then clicking on an existing route. The new route will inherit the net from the route you clicked on and you will be routing on the same layer as that route.

---

### Add a via between layers

Adding a via is simple: while you are routing you can press the layer number on your keyboard or use any of the other mechanisms for changing layers. This will move the current segment you are creating to the layer you just changed to by adding a via from the previous corner in the route.

Alternatively, you can press the ‘Z’ key to toggle between complimentary pairs of layers.

---

### Insert corners in a route

To insert a corner in a route simply click on the middle third of a segment, or at least .050" away from a corner and you will by in a digitizing mode that allows you to add corners until you stop by either pressing the Esc key or double clicking to auto-complete the segment. It is easy to tell if you are in Insert Corners mode because the segment you click on is essentially “ripped up” and will show a ratline from the cursor to the next corner.

---

### Delete route corners

There is no explicit Delete command for a route corner because that would require a mode change or some command that has to executed by pressing a key or clicking a button. Instead, deleting a corner
can be accomplished with the mouse simply by selecting a corner and dropping it on the previous or next corner in the polygon. It's called corner reduction. It is very efficient and can all be done with the mouse.

It can also be accomplished by getting into Insert Corners mode and effectively routing “backwards” as doubling back over existing corners has the effect of removing them.

---

**Move a route corner**

Moving a route corner is simple: just point at a corner, left-click, and release. Now you are moving the corner. Just click again to release the corner where you want.

Releasing the corner over the prior or next corner has the effect of deleting the prior or next segment.

---

**Move a route segment**

If you point at a route segment and hold down the left mouse button while dragging, the entire route segment will move with the cursor.

---

**Change the width of a route segment**

You can only change the width of an individual route segment while you are routing by pressing the ‘W’ key on your keyboard or RMB and selecting Width… from the context menu. To change the width of an existing route segment you must first click on that segment to get into Insert Corner mode and then press the ‘W’ key.

---

**Add text to the board**

If you want to add text to your design then the simplest way to do this is by clicking on the Add Text Button T on the Insert Object Toolbar. You can also choose Insert/Text from the menu bar.

Once you have selected this command you will immediately be prompted for the string to add with the **Add Text Dialog**. After entering a string and dismissing the dialog a piece of text will be attached to your cursor. While it is attached you can press the ‘R’ key to rotate it, the ‘W’ key to change its width, or a layer number to change its layer. When changing its layer interactively it will automatically toggle its mirrored state between the top and bottom layer.

Whenever you have a piece of text attached to your cursor you can modify many of its properties then and there in the Text Pane of the Edit Panel. You can always select Properties on the text to display the **Text Properties Dialog**.

You can also perform many of the global commands such as **Change Grid** while creating text.
Rename a component

If you are new to PCB123 renaming a component may appear counter-intuitive at first because you may just point at a reference designator text object and attempt to change its name only to find that you can’t. You must select the component object and change its name in the Component Properties Dialog. The text object is just a manifestation of a macro placeholder (&RefDes) that inherits its value from the component reference designator. When you change the component name the text will automatically reflect the new name.

When you rename a component for the first time a special attribute record is generated internally that remembers the original name. The original name will appear in a BOM Report so you can back-annotate your schematic if you so desire. This is also to facilitate the automatic back-annotation of schematics in the future.

Designing a board without a netlist

Depending upon the nature or complexity of the design, you may opt to skip a schematic altogether and start right in on the physical layout. In this case, you probably will not use or need a netlist or a schematic editor. PCB123 can support you on these very simple designs as well. Just dive right in with physical layout.

About loading a Netlist

If you have captured your design in a schematic tool then you should be able to create a netlist that PCB123 can import. If you use the PCB123 schematics package to capture your design then any device you use from the standard libraries will be mapped to a footprint in the layout software.

If you are using a third party schematics package it should be able to emit a Tango or Protel formatted netlist that can be read into the PCB123 layout software. The devices you use in the third-party schematics package most likely will not specify an explicit footprint that matches the PCB123 library. Not to fear: when you load a netlist that specifies footprints that cannot be matched in the library, or a netlist that only specifies part types then during the netlist load process you will be prompted to browse for a footprint with a very flexible tool called the Part Browser. This tools lets you perform a search against various criteria and graphically preview the footprints. Once you have selected a footprint to use you will be given the opportunity to let the system forever remember the mapping between your part type and the footprint you chose for that part. It will not ask for a footprint again when it encounters that part.

Starting the layout with a netlist opens a large array of benefits that you may or may not take advantage of. They include:

- **Autoplace** - You can run the Autoplace tool. This tool can be real handy just to gauge board density and get a feel for the overall flow of routing.

- **Placement Optimization** - You can easily see what is connected to what as you place components on the board. In addition, the length of nets is continuously optimized as you move part around.
• **Easier Rules Specification** - With all nets specified at once you can specify which ones need special width or spacing considerations and establish which nets are tied to any plane layers.

• **Fanout and Autorouting** - With a complete netlist loaded, you can run automatic tools that fanout surface mount devices and autoroute your board. Even if you don’t normally use an autorouter running it can identify trouble spots in your design.

• **Automatic ECO** - If you make a change to your schematic after your layout has been started you can compare the new netlist against the design and have the system automatically make the necessary changes to the design so that it matches the new netlist.
Part 4 - Schematic Reference
Main Menu

When in schematic view, the following menu bar is displayed.

If you do not have a design open, only the File, View, PCB123, and Help menu items are available.

File menu

File

New
Open...
Close
Save
Save As...
Backup now...
Print...
Print Setup...
Exit

New

The File-New command is one method of creating a new PCB design. It will invoke the Board Configuration Dialog to allow you to set board parameters or load a netlist unless you have checked the Do Not Show button in the Board Configuration dialog in which case a blank design is created.

Open...

The File-Open command invokes a standard Windows file dialog that allows you to browse to and select one or more PCB123 design files. You can select multiple designs to open by holding down the Shift or Ctrl key while clicking on files. If you select multiple design files, each one will be opened in its own window.
Close

The File-Close command closes the currently active design. If the design has been modified, you will first be prompted to save your changes.

Save

The File-Save command writes a copy of the currently active design to disk using the same filename the design was opened with. If it is a new design that hasn’t been previously saved, then you will be prompted for a filename to save as (see below).

Save As...

The File-Save As command prompts you for a filename using the standard Window file dialog. This will write a copy of the currently active design to disk under the name supplied. If the file exists, you will be prompted to overwrite the existing file.

The entire state of your design is saved to disk with the exception of DRC markers and the local footprint cache. All display settings are saved in the file so the next time the file is opened it will appear the same as when it was saved.

Backup Now

The File-Backup Now command initiates an immediate backup of the currently active design.

Normally, in the course of design, PCB123 automatically saves a backup of all open designs once every ten minutes. It saves them to the Backup subdirectory using the name of the design and appending “_BACKUPn” where n is from 1 to 5, 1 being the most recent backup.

Print

The File-Print command invokes a schematic print dialog box.

From the Schematic print command, you can print the current page or all schematic pages in your design. You can specify who to fit or scale your output, and you can specify color printing options.
**Print Setup**

The File-Print Setup command invokes the standard Windows print setup dialog and is used to set print defaults for the above print command.

---

**Exit**

The File-Exit command exits the PCB123 program. You will be prompted to save any modified documents that you may have open.

---

**Edit menu**

<table>
<thead>
<tr>
<th>Command</th>
<th>Shortcut</th>
</tr>
</thead>
<tbody>
<tr>
<td>Undo</td>
<td>Ctrl+Z</td>
</tr>
<tr>
<td>Redo</td>
<td>Ctrl+Y</td>
</tr>
<tr>
<td>Cut</td>
<td>Ctrl+X</td>
</tr>
<tr>
<td>Copy</td>
<td>Ctrl+C</td>
</tr>
<tr>
<td>Paste</td>
<td>Ctrl+V</td>
</tr>
<tr>
<td>Delete</td>
<td>Del</td>
</tr>
<tr>
<td>Find</td>
<td>F</td>
</tr>
<tr>
<td>Properties</td>
<td>P</td>
</tr>
</tbody>
</table>

---

**Undo**

The Edit-Undo command undoes the last change to currently active design. Virtually all changes to the design are recorded and can be undone in reverse order they were made. There is no limit to the size of the undo buffer other than memory limitations.

What constitutes a change for purposes of undo is operation dependent. In many cases an entire group of changes will be undone with a single undo command. For instance, selecting undo after an autoroute will undo the entire autoroute results, not just the last connection the autorouter completed.

When a change is undone, that change will be added to the Redo buffer in case you change your mind and want to redo it again. See below.
**Redo**

The Edit-Redo command redoes the last undo operation. Repeated undo operations are stored to the redo buffer so you can redo a whole series of undo operations.

**Cut**

The Edit-Cut command deletes the currently selected set of objects and places them into the paste buffer. There is only one paste buffer so only the last cut operation is available for paste.

See the Paste command for more information.

**Copy**

The Edit-Copy command copies the currently selected set of objects into the paste buffer. There is only one paste buffer so only the last copy operation is available for paste.

See the Paste command below for more information.

**Paste**

The Edit-Paste command makes a copy of the objects in the paste buffer and inserts them into the currently active design.

First, any components in the paste buffer are added to the design by assigning them a new reference designator that does not conflict with one already in the design. Their alpha-prefix is retained.

Next, any complete routes or connections that start and end at a pins that are also in the paste buffer will be added to the design. They will retail their original net name with a unique numerical suffix.

Lastly, all text and non-net-attributed polygons will be copied without change. Net-attributed polygons will either inherit a cloned net name or retain their original net names depending on whether their original net names were cloned during the paste operation.

After all objects are created, they will be in a selected state attached to the cursor ready to be placed.

**Delete**

The Edit-Delete command deletes all selected objects from the currently active design.

The delete operation can later be undone.
Properties

The Edit-Properties command is context sensitive. If you have nothing selected it invokes the Board Configuration Dialog. If you have one or more objects selected, then you will get a property page for each class of object selected.

See Object Property Pages for more information.

Insert menu

Top-level Insert menu

Add Page
Add Part
Add Port
Add Wire
Add Line
Add Arc
Add Polyline
Add Rectangle
Add Polygon
Add Text
Add Image

Insert Port submenu

AGND
DGND
Voltage
Global Net
Off-Page Port

Add Page

The Insert-Add Page command inserts a new, blank schematic page for use in your design.

Often, designers will find it helpful to break the schematic into multiple pages organized by function. The network connections between these pages are accomplished using port connectors. The entire schematic network is then made available to the layout page for physical design.
Add Part

The Insert-Add Part command displays the Select Part dialog box, allowing you to search for, preview and select the part you need from the schematic parts taxonomy.

Add Analog Ground Port

The Insert-Add Port – AGND command provides you quick and easy access to a predefined analog ground symbol.

By selecting the Analog Ground Port, your cursor moves into parts placement mode with an analog ground pin ready to be placed.

Add Digital Ground Port

The Insert-Add Port – DGND command provides you quick and easy access to a predefined analog ground symbol.

By selecting the Digital Ground Port, your cursor moves into parts placement mode with a digital ground pin ready to be placed.

Add Voltage Port

The Insert-Add Port – Voltage command provides you quick and easy access to a predefined analog ground symbol.

By selecting the Voltage Port, your cursor moves into parts placement mode with a voltage port ready to be placed.

Add Global Port

The Insert-Add Port – Global Net command provides you quick and easy access to a predefined analog ground symbol.

By selecting the Global Net, your cursor moves into parts placement mode with a voltage port ready to be placed.

Global nets serve to connect to the net with the same net name across the entire set of schematic pages.
Add Off-Page Port

The Insert-Add Port - Off-page Port command provides you quick and easy access to a predefined analog ground symbol.

By selecting the Off-page Port, your cursor moves into parts placement mode with an off-page connector ready to be placed.

Off-Page connectors are similar in function to global nets, but more restricted in their connection. Off-Page connectors work in pairs. An off-page connector will only connect to the (only) other correspondingly-named off-page connector on a schematic sheet set.

Add Wire

The Insert-Add Wire command allows you to digitize new network connections in your schematic. You must start the new wire from a pin, port or another net wire. If starting from another net wire, the new wire will inherit the starting wires’ layer and net properties.

Once you have started routing you can digitize new corners by left-clicking the mouse. To terminate a net you can press the escape key, click on another pin, another port or click on another net.

Add Line

The Insert-Add Line command allows you to add line objects to the currently active design schematic by first defining one endpoint, then the other.

Add Arc

The Insert-Add Arc command allows you to add arc objects to the currently active design or footprint by first defining one endpoint, then the other, and then adjusting the radius.

Arcs have definite width and are always chords of a perfect circle. They cannot be elliptical.

Add Polyline

The Insert-Add Polyline command allows you to add a poly-line object to the currently active design.

A polyline is a continuous series of line segments. They have a physical width and may be net attributed. In most cases polylines are not net attributed as they are widely used for nomenclature such as component outlines and title blocks.

Once you enter add polyline mode, a series of straight line segments will be connected between each successive left-mouse click. Double click on the last point to terminate the polyline definition process.
**Add Rectangle**

The Insert-Add Rectangle command allows you to begin digitizing corners of rectangular polygons.

As you add a rectangle, you will be prompted to digitize two opposite corner points. The rectangle will determine its width and length from these two points.

**Add Polygon**

The Insert-Add Polygon command allows you to begin digitizing corners of polygons and polylines.

A polyline is a continuous series of line segments. They have a physical width and may be net attributed. In most cases polylines are not net attributed as they are widely used for nomenclature such as component outlines and title blocks.

A polygon is a closed 2-D surface. They do not have width. What you draw is the absolute edge of the polygon. They can be convex or concave but should not self-intersect.

If a polygon has no net attribute, then the entire polygon is filled.

If a polygon does have a net attribute, then it becomes a copper pour and any object inside the polygon that does not have the same net attribute is “scratched” out of the polygon so as not to create a short. If a pin resides inside a copper pour polygon, and that pin has the same net attribute then the pin is cleared for thermal isolation but “spokes” are added between the pin and the copper pour to tie them together electrically. Determining where a thermal spoke is legal can be a fairly intensive calculation. Therefore what you see on the screen is only an approximation of the actual spokes. You must perform a Print Artwork to get a true representation of the thermal spokes.

See the Polygon Context Pane and the Polygon Property Page for more information.

**Add Text**

The Insert-Add Text command allows you to add rendered text “strings” to the currently active design or footprint. When you first select this command you will be prompted for the string to add with the Add Text Dialog.

Currently, only the ANSI character set is supported. Extended ASCII characters are not rendered but are stored in the string.

There is only one font available for text strings because the characters must be stroked and there is no reliable way to convert a TrueType font into a series of strokes that can be added to the board. This may change in the future.
View menu

**View Standard Toolbar**

The Toolbars/Standard Toolbar command toggles the visibility of the Standard Toolbar.

**View Add Object Toolbar**

The Toolbars/Add Object Toolbar command toggles the visibility of the Add Object Toolbar.

**View Zoom Toolbar**

The Toolbars/Zoom Toolbar command toggles the visibility of the Zoom Toolbar.

**Zoom Submenu**

The Toolbars/Zoom Toolbar command toggles the visibility of the Zoom Toolbar.
**Zoom In**

The Zoom/Zoom In command increases the magnification of the display. If this command was invoked through the menu then the view center remains unchanged. If this command was invoked with the PgUp key then the view center will be where the cursor was when the command was invoked.

**Zoom Out**

The Zoom/Zoom Out command decreases the magnification of the display. If this command was invoked through the menu then the view center remains unchanged. If this command was invoked with the PgDn key then the view center will be where the cursor was when the command was invoked.

**Zoom All**

The Zoom/Zoom All command sets the screen magnification such that all objects fit on, and are centered on the screen.

**Pan Up**

The Zoom/Pan Up command moves the view center up nearly one full screen height.

**Pan Left**

The Zoom/Pan Left command moves the view center to the left nearly one full screen width.

**Pan Right**

The Zoom/Pan Right command moves the view center to the right nearly one full screen width.

**Pan Down**

The Zoom/Pan Down command moves the view center down nearly one full screen height.

**Redraw**

The Zoom/Redraw command repaints the screen.
Tools menu

Symbol Editor…

The Tools – Symbol Editor… command will present the user with the Symbol Editor dialog box, ready to create a new symbol from scratch. The function of the symbol editor is discussed elsewhere in this manual.

Manage Parts…

The Tools – Manage Parts… command will present the user with the Define Part dialog box. From here, the user can create parts, review or edit the contents of parts, and delete parts as needed.

Electrical Rules Check…

Generate Netlist from Schematic…

The Tools – Generate Netlist from Schematic command will create a netlist file of your schematics’ electrical network. The user will be presented with a Windows standard file save dialog box, allowing the user to specify the location for the netlist file to be output.

PCB123 menu

Top Level PCB123 menu
**MY Account...**

The PCB123 - My Account command will invoke a browser session to your account status page at Sunstone.com. Once you've logged in, you can view your personal account settings including: your default user preferences (shipping address, carrier, payment method, etc.); personal profile information (name, company, email, etc.); and recent order activity including current status for any orders in process.

**PCB123 Website...**

The PCB123 - PCB123 Website command will invoke a browser session at [www.sunstone.com](http://www.sunstone.com)

**Report a Bug...**

The PCB123 - Report a Bug command will invoke a browser session and take you to the Contact Us page for technical support on the [www.sunstone.com](http://www.sunstone.com) website. From here you can report the bug details. Customer Service will acknowledge receipt of your bug report, and follow up with you as necessary.

**Window menu**

**Cascade**

The Window - Cascade command will rearrange open design windows into a cascading pattern.

**Tile**

The Window - Tile command will rearrange open design windows into a tiled pattern.

**Arrange Icons**

The Window - Arrange Icons command will move any iconized design files into an orderly row across the bottom of the PCB123 editing pane background.
Part 5 - Layout Reference
Main Menu

When you have a design open and Layout view is active, then the main menu bar will appear as follows:

If you do not have a design open, only the File, View, PCB123, and Help menu items are available.

File menu

Top-level File menu

New \(\text{Ctrl}+\text{N}\)
Open… \(\text{Ctrl}+\text{O}\)
Open Template…
Close
Save \(\text{Ctrl}+\text{S}\)
Save As…
Save As Template…
Import          Export
Backup now…
Print… \(\text{Ctrl}+\text{P}\)
Print Setup…
Print Artwork…
Exit

Import submenu
Gerber Files…
PCB123 XML…

Export submenu
Simple Netlist
BOM
XML Data…

New command

The File-New command is one method of creating a new PCB design. It will invoke the Board Configuration Dialog to allow you to set board parameters or load a netlist unless you have checked the Do Not Show button in the Board Configuration dialog in which case a blank design is created.
**Open... command**

The File-Open command invokes a standard Windows file dialog that allows you to browse to and select one or more PCB123 design files. You can select multiple designs to open by holding down the Shift or Ctrl key while clicking on files. If you select multiple design files, each one will be opened in its own window.

**Close command**

The File-Close command closes the currently active design. If the design has been modified, you will first be prompted to save your changes.

**Save command**

The File-Save command writes a copy of the currently active design to disk using the same filename the design was opened with. If it is a new design that hasn’t been previously saved, then you will be prompted for a filename to save as (see below).

**Save As... command**

The File-Save As command prompts you for a filename using the standard Window file dialog. This will write a copy of the currently active design to disk under the name supplied. If the file exists, you will be prompted to overwrite the existing file.

The entire state of your design is saved to disk with the exception of DRC markers and the local footprint cache. All display settings are saved in the file so the next time the file is opened it will appear the same as when it was saved.

**Save As Template... command**

The File-Save As Template command saves the currently active design as a template for design reuse. Only a limited set of objects are saved as a template - specifically, board outlines, components, holes, text and polygons. Net information and routing is not saved as a template, though this may change in the future.

You will be prompted for a descriptive name for the template and it will automatically be placed in the Templates subdirectory. The saved template is stored as a Sunstone NLF (Neutral Library Format) file. This is a fully encapsulated format. There are no external references. All data is represented in the file.
Export BOM command

The Export BOM (Bill Of Materials) command outputs a BOM report for the currently active design. You will first be prompted for a report filename (default is design name with a .bom extension) and you will then be prompted to view the report.

The BOM report is a simple linear list of components alphabetically sorted by reference designator. In addition, each entry lists the components part type, footprint name, and the original component name if the component has been subsequently renamed.

Export Netlist command

The Export Netlist command outputs a native PCB123 netlist report for the currently active design. You will first be prompted for a netlist filename (default is design name with a .net extension) and you will then be prompted to view the netlist.

There are two sections in the netlist report. The first section is a linear list of components alphabetically sorted by reference designator. The second section is a linear list of nets alphabetically sorted by net name. Inside each net there will be a list of zero or more component pins.

Backup Now… command

The File-Backup Now command initiates an immediate backup of the currently active design.

Normally, in the course of design, PCB123 automatically saves a backup of all open designs once every ten minutes. It saves them to the Backup subdirectory using the name of the design and appending “_BACKUPn” where n is from 1 to 5, 1 being the most recent backup.

Print… command

The File-Print command invokes the standard Windows print dialog and will print whatever is visible on the screen. It is meant for quick screen shots when you want a hard copy of some isolated section of the board or a reference print that is not scale dependent.

If you need scaled prints of the entire design use the Print Artwork command.

Print Setup… command

The File - Print Setup command invokes the standard Windows dialog for configuring the print settings. Along with specifying the printer to use, this command can also be used to set paper sizes and orientation for print jobs.
Print Artwork... command

The File-Print Artwork command invokes the Print Artwork dialog that allows you to do a detailed print that will match what you see on the boards that will be built. In fact, the Print Artwork command executes the same code that Sunstone generates Gerber files from with the exception of non-plated holes which are a special case to Print Artwork and will show up in the prints.

See the Print Artwork Dialog for more details.

Exit command

The File-Exit command exits the PCB123 program. You will be prompted to save any modified documents that you may have open.

Edit menu

Top-level Edit Menu

<table>
<thead>
<tr>
<th>Selection submenu</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select Pins</td>
</tr>
<tr>
<td>Select Polygons</td>
</tr>
<tr>
<td>Select Text</td>
</tr>
<tr>
<td>Select Components</td>
</tr>
<tr>
<td>Select Tracks</td>
</tr>
<tr>
<td>Select anything</td>
</tr>
<tr>
<td>Select nothing</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Properties</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
Undo command

The Edit-Undo command undoes the last change to currently active design. Virtually all changes to the design are recorded and can be undone in reverse order they were made. There is no limit to the size of the undo buffer other than memory limitations.

What constitutes a change for purposes of undo is operation dependent. In many cases an entire group of changes will be undone with a single undo command. For instance, selecting undo after an autoroute will undo the entire autoroute results, not just the last connection the autorouter completed.

When a change is undone, that change will be added to the Redo buffer in case you change your mind and want to redo it again. See below.

Redo command

The Edit-Redo command redoes the last undo operation. Repeated undo operations are stored to the redo buffer so you can redo a whole series of undo operations.

Cut command

The Edit-Cut command deletes the currently selected set of objects and places them into the paste buffer. There is only one paste buffer so only the last cut operation is available for paste.

See the Paste command for more information.

Copy command

The Edit-Copy command copies the currently selected set of objects into the paste buffer. There is only one paste buffer so only the last copy operation is available for paste.

See the Paste command below for more information.

Paste command

The Edit-Paste command makes a copy of the objects in the paste buffer and inserts them into the currently active design.

First, any components in the paste buffer are added to the design by assigning them a new reference designator that does not conflict with one already in the design. Their alpha-prefix is retained.

Next, any complete routes or connections that start and end at a pins that are also in the paste buffer will be added to the design. They will retail their original net name with a unique numerical suffix.
Lastly, all text and non-net-attributed polygons will be copied without change. Net-attributed polygons will either inherit a cloned net name or retain their original net names depending on whether their original net names were cloned during the paste operation.

After all objects are created, they will be in a selected state attached to the cursor ready to be placed.

**Delete command**

The Edit-Delete command deletes all selected objects from the currently active design.

The delete operation can later be undone.

**Load Selection... command**

The Edit-Load Selection command is identical to the Paste operation except the pasted objects are from a previously saved set of objects or from a design template as opposed from the paste buffer. You will be prompted for a selection to load.

Currently this operation excludes any net information. A saved selection can contain everything except nets. See the Save Selection command for more information.

**Save Selection... command**

The Edit-Save Selection command saves the selected objects from the currently active design to a disk file for later reuse. No net information is currently saved though this may change in the future.

This command is only available when you have an object or objects selected. Once invoked, you will be prompted for a description of the selection set.

**Find... command**

The Edit-Find command prompts you for the name of an object in the currently active design and then centers the viewport on that object.

See the Find Dialog for more details.

**Bring to Cursor... command**

The Edit-Bring to Cursor command is similar to the Find command but instead of zooming you to an object it selects the named object and snaps it to the cursor.

See the Find Dialog for more details.
Line Width command

The Edit-Line Width command is only available when you are adding/editing polylines and routes. When selected, you will be prompted to enter a line width with the Line Width Dialog.

For polylines, the new line width applies to the whole polyline.

For routes, the new line width applies to the currently active line segment and subsequent segments you add to the route.

Grid command

The Edit-Grid command sets the working (snap) grid and the display grid for the currently active design. You can set the X and Y grid independent by separating the two values with a comma.

See the Grid Dialog for more details.

Select Pins command

The Edit-Select Pins command toggles the selectability of board-level pin objects in the currently active design or component pins in Edit Footprint window.

Select Polygons command

The Edit-Select Polygons command toggles the selectability of polygon objects in the currently active design.

Select Text command

The Edit-Select Text command toggles the selectability of text objects in the currently active design.

Select Components command

The Edit-Select Components command toggles the selectability of component objects in the currently active design.

Select Tracks command

The Edit-Select Tracks command toggles the selectability of track (route) objects in the currently active design.
Select Anything command

The Edit-Select Anything command turns on the selectability of all objects in the currently active design.

Select Nothing command

The Edit-Select Nothing command turns off the selectability of all objects in the currently active design.

Properties command

The Edit-Properties command is context sensitive. If you have nothing selected it invokes the Board Configuration Dialog. If you have one or more objects selected, then you will get a property page for each class of object selected.

See Object Property Pages for more information.

Insert menu
Select command

The Insert-Select command is the default state for selecting objects. When you stop adding or modifying objects by pressing the Escape key or selecting RMB-Cancel the system enters the Select state.

Add Pin command

The Insert-Add Pin command immediately allows you to start adding pins to the currently active design or footprint.

When you first select this command, a pin will be attached to the cursor whose properties will be defaulted, usually to the same values as the last pin inserted.

By default, board-level pins do not have names, though they can be assigned names – even duplicate names. Board-level pins do not show up in a netlist report, only component pins do.

When creating a footprint, pins names start at 1 by default and increment from there. The only time a pin name can be reused in a footprint is if it has no name at all.

See the Pin Context Pane and the Pin Properties page for more information.

Add Freestyle Route command

The Insert-Add Freestyle Route command allows you to begin digitizing new route corners. You must start the new route from a pin or another route. If starting from another route, the new route will inherit the starting route’s layer and net.

If starting from a pin the new route layer will either be the pin layer in the case of an SMD pin or the current layer if it is a thru-hole pin. If the starting pin already has a net attribute then the route will inherit that pins net.

Once you have started routing you can digitize new corners by left-clicking the mouse and you can insert vias by changing layers or pressing the ‘Z’ key or change the current width by pressing the ‘W’ key.

To terminate a freestyle route you can press the escape key, click on another pin, or click on another track on the same layer as the last segment. If the new route has a temporary net then it will be ‘hardened’ by either inheriting the terminating objects net property or by creating a new net and assigning everything in the temporary net to the new net.

Add Polygon command

The Insert-Add Polygon command allows you to begin digitizing corners of polygons and polylines.

A polyline is a continuous series of line segments. They have a physical width and may be net attributed. In most cases polylines are not net attributed as they are widely used for nomenclature such as component outlines and title blocks.
A polygon is a closed 2-D surface. They do not have width. What you draw is the absolute edge of the polygon. They can be convex or concave but should not self-intersect.

If a polygon has no net attribute, then the entire polygon is filled.

If a polygon does have a net attribute, then it becomes a copper pour and any object inside the polygon that does not have the same net attribute is “scratched” out of the polygon so as not to create a short. If a pin resides inside a copper pour polygon, and that pin has the same net attribute then the pin is cleared for thermal isolation but “spokes” are added between the pin and the copper pour to tie them together electrically. Determining where a thermal spoke is legal can be a fairly intensive calculation. Therefore what you see on the screen is only an approximation of the actual spokes. You must perform a Print Artwork to get a true representation of the thermal spokes.

See the Polygon Context Pane and the Polygon Property Page for more information.

**Add Arc command**

The Insert-Add Arc command allows you to add arc objects to the currently active design or footprint by first defining one endpoint, then the other, and then adjusting the radius.

Arcs have definite width and are always chords of a perfect circle. They cannot be elliptical.

**Add Circle command**

The Insert-Add Circle command allows you to add hollow circle objects to the currently active design or footprint by first defining the circle center, and then sweeping out a radius.

Circles have definite width. They cannot be elliptical.

**Add Text command**

The Insert-Add Text command allows you to add rendered text “strings” to the currently active design or footprint. When you first select this command you will be prompted for the string to add with the Add Text Dialog.

Currently, only the ANSI character set is supported. Extended ASCII characters are not rendered but are stored in the string.

There is only one font available for text strings because the characters must be stroked and there is no reliable way to convert a TrueType font into a series of strokes that can be added to the board. This may change in the future.
**Add Component command**

The Insert-Add Component command allows you to interactively insert components into the active design.

When this command is selected the Add Component Dialog will be shown. After you select a part type or footprint with this dialog, a component will be created and snapped to the cursor ready for placement.

A unique reference designator (name) will be created for the component that uses the same alpha prefix as the last component added but with a unique numerical suffix.

**Add Connection command**

The Insert-Add Connection command is really a tool that has a couple of features that allow you to define and modify nets.

The default mode for the tool allows to specify point-to-point wiring for a net by first clicking on a component pin and then clicking on subsequent pins to define the connectivity of a net.
View menu

Toolbars submenu
- Edit Panel
- Standard Toolbar
- Add Object Toolbar
- Command Toolbar
- Layers Toolbar
- Selection Toolbar
- Zoom Toolbar

Zoom submenu
- Zoom In
- Zoom Out
- Zoom All
- Home
- Actual Size
- Pan Up
- Pan Left
- Pan Right
- Pan Down
- Redraw
- Center

Layers submenu
- Top
- Bottom
- Inner 1
- Inner 2
- Inner 3
- Inner 4
- Silkscreen Top
- Silkscreen Bottom
- Solder Mask Top
- Solder Mask Bottom
Refresh All

The Refresh All command not only repaints the screen but refreshes the Object Hierarchy and rebuilds some internal tables. Pressing the Esc key when nothing is selected performs the same operation.

Go-To First Error

The Go-To First Error command zooms to, and centers the display around the first DRC error. If there are no DRC errors then you will be prompted to run a DRC.

Go-To Next Error

The Go-To Next Error command zooms to, and centers the display around the next DRC error. If there are no more DRC errors then you will be prompted to run another DRC and go to the first error found.

User Preferences

The User Preferences command is used to invoke the User Preferences Dialog.

Toolbars/ Edit Panel

The Toolbars/Edit Panel command toggles the visibility of the Edit Panel.

Toolbars/ Standard Toolbar

The Toolbars/Standard Toolbar command toggles the visibility of the Standard Toolbar.

Toolbars/ Add Object Toolbar

The Toolbars/Add Object Toolbar command toggles the visibility of the Add Object Toolbar.

Toolbars/ Layers Toolbar

The Toolbars/Layers Toolbar command toggles the visibility of the Layers Toolbar.

Toolbars/ Selection Toolbar

The Toolbars/Selection Toolbar command toggles the visibility of the Selection Toolbar.
Toolbars/Zoom Toolbar

The Toolbars/Zoom Toolbar command toggles the visibility of the Zoom Toolbar.

Zoom/Zoom In

The Zoom/Zoom In command increases the magnification of the display. If this command was invoked through the menu then the view center remains unchanged. If this command was invoked with the PgUp key then the view center will be where the cursor was when the command was invoked.

Zoom/Zoom Out

The Zoom/Zoom Out command decreases the magnification of the display. If this command was invoked through the menu then the view center remains unchanged. If this command was invoked with the PgDn key then the view center will be where the cursor was when the command was invoked.

Zoom/Zoom All

The Zoom/Zoom All command sets the screen magnification such that all objects fit on, and are centered on the screen.

Zoom/Pan Up

The Zoom/Pan Up command moves the view center up nearly one full screen height.

Zoom/Pan Left

The Zoom/Pan Left command moves the view center to the left nearly one full screen width.

Zoom/Pan Right

The Zoom/Pan Right command moves the view center to the right nearly one full screen width.

Zoom/Pan Down

The Zoom/Pan Down command moves the view center down nearly one full screen height.
**Zoom/ Redraw**

The Zoom/Redraw command repaints the screen.

---

**Layers/ Top**

The Layers/Top command sets the current layer to the Top layer. It also repaints the screen with the Top layer being painted last.

---

**Layers/ Bottom**

The Layers/Bottom command sets the current layer to the Bottom layer. It also repaints the screen with the Bottom layer being painted last.

---

**Layers/ Inner1**

The Layers/Inner1 command sets the current layer to the Inner1 layer. It also repaints the screen with the Inner1 layer being painted last. This command will only be available if the current design has more than 2 layers.

---

**Layers/ Inner2**

The Layers/Inner2 command sets the current layer to the Inner2 layer. It also repaints the screen with the Inner2 layer being painted last. This command will only be available if the current design has more than 2 layers.

---

**Layers/ Inner3**

The Layers/Inner3 command sets the current layer to the Inner3 layer. It also repaints the screen with the Inner3 layer being painted last. This command will only be available if the current design has more than 4 layers.

---

**Layers/ Inner4**

The Layers/Inner4 command sets the current layer to the Inner4 layer. It also repaints the screen with the Inner4 layer being painted last. This command will only be available if the current design has more than 4 layers.
**Layers/ Silkscreen Top**

The Layers/Silkscreen Top command sets the current layer to the Top Silkscreen layer. It also repaints the screen with the Top Silkscreen layer being painted last.

**Layers/ Silkscreen Bottom**

The Layers/Silkscreen Bottom command sets the current layer to the Bottom Silkscreen layer. It also repaints the screen with the Bottom Silkscreen layer being painted last.

**Layers/ Soldermask Top**

The Layers/Soldermask Top command sets the current layer to the Top Soldermask layer. It also repaints the screen with the Top Soldermask layer being painted last.

**Layers/ Soldermask Bottom**

The Layers/Soldermask Bottom command sets the current layer to the Bottom Soldermask layer. It also repaints the screen with the Bottom Soldermask layer being painted last.
Design menu

Top-level Design menu

Board Configuration

New Footprint

Edit Footprint...

Manage Library...

Optimize Net Lengths

Autoplace

Fanout SMT Pads

Autoroute...

Gloss

Compare Netlist/Update Design

Clear DRC Errors

Run DRC

Tools

Tools submenu

Generate Mechanical Drawing

Center Assembly Reference Designators

Board Configuration

The Board Configuration command invokes the Board Configuration Dialog which is where you set such parameters as the number of layers, whether or not you want soldermask, etc.

New Footprint

The New Footprint command creates a new, unnamed, empty footprint and switches the view to the Footprint Editor.

Edit Footprint

The Edit Footprint command invokes the Library Browser (same as the load part dialog) so you can select a footprint to edit. Once you have selected a footprint then that footprint will be loaded into a Footprint Editor for you to modify and save.
**Manage Library**

The Manage Library command invokes the Library Manager Dialog which lets you perform such operations as create new libraries, catalog existing libraries, and cut and paste among libraries.

**Optimize Net Lengths**

The Optimize Net Lengths works only on unrouted portions of nets, not existing routing. The left side of the picture below contains a net before optimization and the right side after optimization. You can override this behavior on a per-net basis by setting the Do Not Optimize flag in the Net Properties dialog.

![Optimize Net Lengths Diagram]

**Autoplace**

Autoplace makes a best effort attempt to place components on the layout board in a manner that minimizes overall routing complexity. The Autoplace function is not designed to be exhaustively accurate, but rather as a convenience to get you started as you complete schematic design and transition to the early stages of physical layout.

**Fanout SMT Pads**

The Fanout SMT (surface mounted) Pads command automatically routes out any connection tied to a surface mount pad and then terminates at a via as pictured below.

If a connection coming out of a surface mount pin needs to be tied to a plane then fanning it out basically completes the connection to that pin and the ratline will disappear. Fanout also aids a very important routing strategy for high-density SMT boards. Because the surface layers on these boards are highly congested with pads most of the routing has to take place on inner layers. A smart strategy
to get to these inner layers is to fanout the board early in the design process - preferably before any other routing.

**Autoroute**

The Autoroute command attempts to automatically complete all routing on the board. A reasonable set of defaults are used by the router but a detailed explanation of possible settings that can control the router can be found under Autorouter User Preferences.

**Gloss**

The Gloss command automatically performs aesthetic rerouting. It typically removes redundant corners and segments, often spreads routes out, and generally fixes such minor problems as acute angles and 90-degree bends. An example is shown below.

![Gloss Example](image)

**Compare Netlist/Update Design**

The Compare Netlist/Update Design command loads a master netlist from disk, converts it to a PCB, and detects any differences between the master PCB and the current slave PCB. The difference list is then optimized to a set of changes that will have the least impact in the board. The list will then be presented to you in the form of an ECO Report Dialog. When this dialog is dismissed you will then have the option of automatically performing the changes that will make the slave match the master.

Even if there were no changes detected during the comparison all component footprints will be updated from the master.

**Clear DRC Errors**

The Clear DRC Errors command will simply delete all the DRC markers from the current design. DRC markers are generated from running the DRC command or interactively while editing your design.
Run DRC

The Run DRC (Design Rule Check) performs the following checks on all objects in the current design:

**Track-To-Track Spacing Violation** - Two tracks (routes) from different nets are less than the required minimum distance apart.

**Track-To-Pad Spacing Violation** - A track and a pad from different nets are less than the required minimum distance apart.

**Track-To-Via Spacing Violation** - A track and a via from different nets are less than the required minimum distance apart.

**Track-To-Polygon Spacing Violation** - A track and a polygon from different nets (or polygon with no net) are less than the required minimum distance apart.

**Track-To-Text Spacing Violation** - A track and a piece of text are less than the required minimum distance apart. This will result in overlapping or unreadable text.

**Pad-To-Pad Spacing Violation** - Two pads from different nets are less than the required minimum distance apart.

**Pad-To-Via Spacing Violation** - A pad and a via from different nets are less than the required minimum distance apart.

**Pad-To-Polygon Spacing Violation** - A pad and polygon from different nets (or polygon with no net) are less than the required minimum distance apart.

**Pad-To-Text Spacing Violation** - A pad and a piece of text are less than the required minimum distance apart. This will result in overlapping or unreadable text.

**Via-To-Via Spacing Violation** - Two vias from different nets are less than the required minimum distance apart.

**Via-To-Polygon Spacing Violation** - A via and a polygon from different nets (or polygon with no net) are less than the required minimum distance apart.

**Via-To-Text Spacing Violation** - A via and a piece of text are less than the required minimum distance apart. This will result in overlapping or unreadable text.

**Polygon-To-Polygon Spacing Violation** - Two polygons from different nets are less than the required minimum distance apart.

**Polygon-To-Text Spacing Violation** - A polygon and a piece of text are less than the required minimum distance apart. This will result in overlapping or unreadable text.

**Text-To-Text Spacing Violation** - Two pieces of text are less than the required minimum distance apart. This will result in overlapping or unreadable text.
Object-To-Board Outline Spacing Violation - A route, pad, via, polygon, or piece of text is too close to the board outline.

Drill-To-Drill Spacing Violation - Two holes overlap at offset locations. This condition will result in a broken drill bit.

Unfinished-Route Violation - There is a connection between two pins that still needs to be routed to complete the circuit.

Board-Too-Small Violation - The minimum dimension in either the x or y axis is ¼ inch. If you receive the Board-Too-Small violation, then the board is less than ¼ inch in either axis. We cannot manufacture a board that small.

Silkscreen-Over-Pad Warning - Silkscreen outline or text runs into a pad that is unmasked. This is a problem for the fabrication process as well as possibly making the silkscreen illegible.

Dangling-Trace Warning - A route does not end at a pad, via, or other route. Traces in this configuration creates an antenna effect. While the antenna may be intentional, PCB123’s DRCs will alert you to the condition as a precaution.

Pad-Annular Warning - The pad size minus the drill size is less than the recommended minimum.

PCB123 menu

Top Level PCB123 menu

MY Account…

The PCB123 - My Account command will invoke a browser session to your account status page at Sunstone.com. Once you've logged in, you can view your personal account settings including: your default user preferences (shipping address, carrier, payment method, etc.); personal profile information (name, company, email, etc.); and recent order activity including current status for any orders in process.

PCB123 Website…
The PCB123 – PCB123 Website command will invoke a browser session at [www.sunstone.com](http://www.sunstone.com).

**Report a Bug...**

The PCB123 – Report a Bug command will invoke a browser session and take you to the Contact Us page for technical support on the [www.sunstone.com](http://www.sunstone.com) website. From here you can report the bug details. Customer Service will acknowledge receipt of your bug report, and follow up with you as necessary.

**Window menu**

**Cascade**

The Window – Cascade command will rearrange open design windows into a cascading pattern.

**Tile**

The Window – Tile command will rearrange open design windows into a tiled pattern.

**Arrange Icons**

The Window – Arrange Icons command will move any iconized design files into an orderly row across the bottom of the PCB123 editing pane background.

**Toolbars**

**Main toolbar**

The Toolbars/Main Toolbar for the Schematic editor toggles the visibility of the Main toolbar. The main toolbar provides quick access to common file level and viewing commands such as: new design; save; open; cut; copy; paste; rotate; print.
**Add Object toolbar**

The Toolbars/Add Object Toolbar command toggles the visibility of the Add Object Toolbar. The Add Object Toolbar gives quick and easy access to commands for adding objects to the schematic, including Title Bar info; component symbols; polylines; polygons; arcs; circles and text.

**Layer toolbar**

The Toolbars/Layer Toolbar command toggles the visibility of the Layers toolbar. The Layers toolbar typically displays vertically, along the right edge of the PCB123 application window. The Layers toolbar provides one-click view filtering for all layers specified in the currently active edit panel.

**Selection toolbar**

The Toolbars/Selection Toolbar command toggles the visibility of the Selection toolbar. The Selection toolbar typically displays vertically, along the right edge of the PCB123 application window. The Selection toolbar provides one-click view filtering for selectable items in the layout editing panel.

**Zoom toolbar**

The Toolbars/Zoom Toolbar command toggles the visibility of the Zoom toolbar. The Zoom toolbar typically displays vertically, along the right edge of the PCB123 application window. The Zoom toolbar provides one-click view adjustment in the edit window.
**3-D View toolbar**

Top Level 3D Menu

![3D View toolbar](image)

**Wireframe**

The 3D – Wireframe command disables the 3D view’s solid surface rendering feature. This view often allows for close viewing of vias and traces in multi-layer designs. Wireframe can also provide a faster rendering of the 3D model on some computers.

**Show Silkscreen**

By default, the 3D view displays the silkscreen layer data. Turning the 3D – Show Silkscreen command off removes silkscreen data from the 3D model.

**Panels**

**Display panel**

The Display Panel provides easy access to layout window display parameters to assist in the design creation, review and editing process. The display panel contains three major functional regions: the Global View, the Appearance menu, and the Layer Display.

Global View provides the user with a thumbnail view of the entire design space. A ghost rectangle shows the current zoom and pan location for the edit window. As the edit view zooms in further, the ghost rectangle naturally gets smaller. By dragging/dropping a rectangle directly on top of the Global View, you can quickly reset your edit pane view to a new location without tedious panning and zooming commands.

Also in the global view are a read-only display of the current cursor position coordinates, a read-only view of the current grid space settings, and user-selectable units for reporting positions in the design space. The available units include: Inches; Mils, Centimeters and Millimeters.

The Appearance region allows you to change the rendering for grids, backgrounds, ratsnest data, layer data blending and other methods for customizing the representation of your design. These tools allow...
you some sophisticated methods for managing the context data represented around your current area of concentration. Many of these appearance features can also be edited on the User Preferences dialog. A functional description for each attribute on the appearance page can be found in the User Preferences Display Page section.

The Layer Display region allows the user to turn layers and data types on or off. Presented in tabular view with rows representing a layer and columns representing a data type, users can interactively select the data to display. This technique comes in handy when concentrating on data for a specific layer, such as a ground plane or component/connector escape routing.

**Project panel**

Unlike the Display Panel, the Project Panel stays the same for all tabbed views of the design. The Project Panel displays three buttons across the top: Statistics Report; Netlist Report; BOM Report. Each of these three buttons generates an ascii format report based on your current design.

The Board Configuration area provides an interactive tree-style hierarchical access to all the design primitives: nets, components, pins, etc. and the associated meta-data. Properties can be edited from this location so care must be used by casual users not to inadvertently change their design.

The Schematic Properties are provides an interactive area for setting and adjusting line styles, text fonts and styles, and data fields for the design title block.

**DRC/ ERC panel**

The DRC/ERC tab contains a live listing of all the current design violations in the project.

The DRC will report on shorts, opens, and violations of manufacturing rules such as minimum spacing between objects of different nets, and pad size vs. drill size violations.

Locations of errors reported by the DRC are marked with a grey circle-? icon. By hovering the mouse pointer over the error, users can access a summary explanation for the cause of the error being reported.

The DRC/ERC tab also allows users to access violations by category or by layer. At the top of the DRC/ERC tab are the ‘stepper’ functions, allowing users to move from error to error so as to view/fix each one interactively.
Dialog Boxes

About... Dialog

Besides some shameless vanity information, the Help/About dialog tells what version of PCB123 you are running and how much memory is the software is using.

Add Text Dialog

The Add Text Dialog is invoked when the Text Tool is selected in current design or footprint. It simply prompts you for the string of text to add.

The Text to Add field is where you enter the text string. See the Add Text Command for more information.
Board Configuration Dialog

The Board Configuration Dialog is invoked from the Design menu or by RMB Properties when nothing is selected. It will also be shown when you choose to bypass the New Board Wizard and select File/New in which case additional fields will be activated that allow you to load a netlist for instance.

It is primarily used to specify various board manufacturing parameters.

The **Board Name** field is simply a label for your design. There are no restrictions on it.

The **Board Size** X and Y fields specify the rectangular size of the board. These fields are only active when this dialog is invoked in response to File/New, otherwise they are grayed out and only inform you of the existing current board size.

The **Netlist File Name** field can only be used when this dialog is invoked from File/New. It is used to specify a netlist or netlists to load into the new design.

The **Layers Required** field is where you can specify the number of layers in your design. The default is two layers but if you choose more than two then other options will be enabled such as plane layer
assignments. If you invoked this dialog from Design/Board Configuration on an existing design and you decrease the layer count then objects on the deleted layers will be permanently removed from the database. Any routing on the removed layers will be unrouted and the ratsnest possibly updated.

The **Power Planes** checkboxes are only enabled when you specify more than two layers for your design. When they are enabled you can check which layers will be treated as poser planes. Nets that get assigned to power plane layers are assumed to be connected as soon as something in the net touches a plated-through-hole.

The **Solder Mask** checkbox is only enabled for two and four layer boards. Six layer boards automatically get a solder mask. Solder mask, also known as solder resist is a green coating that is uniformly applied to the board except where solderable surfaces are required such as pads. You must check Solder Mask if you want silk screen on your board. There is an additional manufacturing charge for selecting solder mask.

The **Silk Screen** radio buttons are only enabled if you have solder mask enabled. You can choose to have no silk screen, silk screen only on the top layer, silk screen only on the bottom layer, or silk screen on both top and bottom. If you choose to have silk screen on the bottom layer (or both top and bottom) then there is additional fixed cost applied.

The **Material Thickness** is almost always 0.062". There are several restrictions placed on a 0.031” board which are the board can only be a two-layer board, must have solder mask, and cannot have a 2.5oz copper weight.

The **Finished Copper Weight** option is almost always 1oz. There are several restrictions placed on a 2.5oz copper board which are the board can only be a two-layer board, must have solder mask, and cannot be 0.031” thick.

The **Quantity** drop list is where you specify how many boards you want to build. You can select between 2 and 100 boards indifferent increments. The price-per-board and the lead time is impacted by the quantity. You can freely experiment with this setting.

The **Check If You Do Not Want This Displayed** button can prevent this dialog from being displayed when you use File/New. This can always be unchecked by invoking this dialog in Design/Board Configuration.

### Change Grid Dialog

The Change Grid Dialog is nearly universal in the system and can be invoked while doing other tasks such as moving a component. The simplest way to invoke the Change Grid Dialog is to press the ‘G’ key on your keyboard but you can also choose Edit/Grid from the menu bar.
The **New Grid** field is where you specify the desired grid. You can specify independent X and Y axis grids by separating the two with a comma. Specifying only one value implies that both the X and Y grid will be the same.

### Component Property Page

The Component Property Page is invoked whenever a component is selected and Properties is chosen either by pressing Alt+Enter on your keyboard or RMB/Properties. It can also be invoked by selecting a component in the Object Hierarchy tree and selecting RMB/Properties.

If you only have components in your selection set then the only tabs that will appear in the property pages are components and pins. Otherwise other object tabs may be present too.

If you only have one component selected then all the fields in the Component Property Page will be filled in with the values from that component. If you have more than one component selected then the fields will be empty or in an indeterminate state. Any field left empty or indeterminate when the page is dismissed will be unchanged in the selected components.

The **Reference Designator** field (or component name) is the name to assign to the component. It will be forced to upper case and it must be unique in the design. If more than one component is selected then this field will be disabled. Any change to the reference designator here will also be reflected in the companion schematic.
The Component Attributes button can be pressed to invoke the Attributes Dialog.

The X Location field is where the component is located in the X-axis of the design. It will default to the current units but can be in the format of your choice. The X location is relative to the component’s local origin.

The Y Location field is where the component is located in the Y-axis of the design. It will default to the current units but can be in the format of your choice. The Y location is relative to the component’s local origin.

The Rotation field is the component rotation about its local origin. Rotations are counter-clockwise unless you enter a negative number in which case they are clockwise but will be converted to the appropriate positive rotation when this dialog is dismissed.

The Opposite button, when checked, signifies that the component is on the back side of the board. A component on the back side is automatically mirrored.

The Part Type field may or may not be filled out depending on how the component was originally loading into the design. If the component was referenced from a Device as opposed to a Footprint then the Part Type field will be the name of the device. This field is actually just a placeholder and be filled in with anything. This field will appear in the BOM report.

The Package Name field contains the name of the footprint this component uses and is not directly editable but can be filled in by the Part Browser dialog. If you select a different footprint for this component the new footprint must contain pins that will match up with the current nets used by this component. I.E. it cannot change the netlist.

Create New Library Dialog

The Create New Library dialog can be invoked either from the Library Manager or when saving a footprint in the Footprint Editor.

Why would you want to create a new library? Library management is a highly subjective topic leading to many lively debates but here are some facts:

- You don’t want to save custom footprints into any standard library that is supplied with the software because they may possibly be overwritten the next time you update to a newer revision.
- There is no penalty in creating new libraries. Part searching by default spans all libraries in and under the Lib directory.
- By placing custom footprints for a particular design into its own library, you most likely will not have any naming conflicts with other parts of the same name but different content. The
library will also have a small size that easily copied from machine to machine.

The **Library Name** field is where you define the name of the library you are creating. It will automatically be put into the Lib/Custom directory.

The **Description** field is where you can enter a brief description the library you are creating.

**ECO Report Dialog**

The ECO report dialog is automatically displayed when you perform a Design/Compare Netlist and there are differences between the netlist and the design.

![Netlist compare results](image)

The **Print** button can be used to generate a hardcopy of the change report.
Find Object Dialog

The Find Object dialog is invoked by pressing the ‘F’ key on your keyboard or by selecting Edit/Find from the menu bar. It is used to locate a named object and center it on the screen. This same dialog will be invoked by the Snap Object to Cursor command which instead of zooming to the object will actually move the object to the cursor.

![Find Object Dialog](image)

The **Name of Item to Find** field is where you specify the named item or sub-item to find. Sub-items are specified by separating the item and sub-item name with a period. For instance:

- U1 will center the screen on the origin of component U1.
- U1.3 will center the screen on pin ‘3’ of item U1.

Key Assignments Dialog

The Key Assignments Dialog can be invoked under Help/Key Assignments. By pointing at a key on the pictorial keyboard or actually typing a key on your physical keyboard you will be presented with information on what that key does by itself or when the Ctrl or Shift key is held in combination.
Line Width Dialog

The Line Width dialog can be invoked from the File/Edit menu, from various context menus, or by pressing the 'W' key. It is used to change the width of any line object being drawn. If you invoke this dialog while routing, then the new width will apply to the segment from the last corner clicked and all subsequent corners until you stop drawing or change the width again.

The Line Width field is where you specify the new width. It will default to the current units but can be in the format of your choice.
The Load Part dialog is invoked in many ways including from inside PCB123’s schematics program. It may be automatically invoked while loading a netlist when a part type cannot be found in the library. It will be invoked when you choose to add a part manually using the Add Component Tool or edit the footprint properties of an existing component.

The **Search drop list** is where you enter the part search criteria. The drop list contains the last five searches performed. The search criteria may contain the wildcards '*' and '?'. The '*' wildcard means match anything after the criteria and the '?' means match any character in that position. Example: searching for DIP* will match DIP8, DIP14, etc. Searching for DIP?0 will match DIP10, DIP20, etc. By default all libraries are scanned for a Part Type, Footprint, or a description field that matches the search criteria.

The **Begins With** and **Contains** radio buttons are used to expand or limit the scope of the search criteria. When you have the Begins With radio button checked, only items whose name begins with the search criteria are matched.

The **Go** button will initiate a search of the libraries. Any item matching the search criteria will be listed in the **Select** list and the first item highlighted and displayed in the Preview Pane.

The **Advanced Search** button expands the dialog to include additional options found in the Advanced Mode of this dialog.

The **Select** list contains all the items that match the search criteria. The name, the number of pins in the part, and the computed X and Y pitch (spacing between pins) will be displayed for each matching item. The computed pitch will be shown in inches if the pitch happens to divide evenly into .005" otherwise it will be shown in millimeters.

The **Preview** pane shows a graphical view of the currently selected item.

The text of the **Insert Selection** button may change based on the context of how this dialog was invoked but its action is to dismiss the dialog use the currently selected item.
Load Part Dialog (advanced mode)

The Advanced Mode of the Load Part dialog is the same as the Simple Mode except for the following additional items described below:

- The **Library Tree** shows the hierarchical structure of the Lib directory and all subdirectories. You may check and uncheck individual libraries or folders to include or exclude them from the search. You may also click on an individual library to list the entire contents of that library in the **Select** list.

- The **Footprints Used** list will be filled in if you have selected a Part Type in the **Select** list as opposed to a Footprint. Some libraries contain Part Types such as 74LS00 and those Part Types may list multiple footprints that can be used.

- The **Footprint Details** section lists various properties of the currently selected footprint.
Load Selection Dialog

The Load Selection Dialog can be invoked either by choosing Edit/Load Selection from the menu bar or by RBM with nothing selected and choosing Load Selection from the context menu.

This dialog will allow you to merge and existing template or a previously saved selection into your design.

The **Item** list contains all the named templates/selections you have. When you click on one it will be previewed in the **Preview Pane**. The previewed item will be merged into your design when select Ok.
Manage Library Dialog

The Manage Library dialog packs a big punch in a small area. It can be used to create and delete libraries, copy, move, and delete library contents, and even create a web-based visual catalog of library contents.

The dialog is broken into four areas with simplest being the Preview pane that simply displays the selected part type or footprint. The other three sections each contain a list and a toolbar to perform various operations on the lists.

The Libraries list contains all the individual library files in the Lib directory and subdirectories. If you have not selected any libraries then the only toolbar option available will be the Create New Library button. You may select multiple libraries by holding down the Shift or Ctrl keys while clicking on library items. When one or more libraries are selected all the library tools will be available. They are as follows:

- Import NLF Items into selected libraries. This command will prompt you for the name of a PCB123 Neutral Library Format file that may contain part types and footprints to add to the selected libraries.
- Create a new library. This command will create a new library in the Custom subdirectory using the Create New Library Dialog.
- Delete selected libraries. This command will delete the selected libraries from your disk. Use with caution.
List contents of selected libraries. This command will generate a columnar list of all items in the selected libraries as shown below.

<table>
<thead>
<tr>
<th>Condensed.lib</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>XPART</td>
<td>Special Part Used With PLACE Utility</td>
</tr>
<tr>
<td>LED_T1</td>
<td>2-Fin, 1&quot; Spacing, LED</td>
</tr>
<tr>
<td>LED1</td>
<td>2-Fin, 1&quot; Spacing, LED, Digi-Key, Catalog 963, 96 pg338</td>
</tr>
<tr>
<td>LED2</td>
<td>2-Fin, 1&quot; Spacing, LED, Digi-Key, Catalog 965, 96 pg338</td>
</tr>
<tr>
<td>DIP12</td>
<td>12-Fin DIP, 3&quot; Spacing</td>
</tr>
<tr>
<td>DIP14</td>
<td>14-Fin, 3&quot; Spacing, DIP, N014, TTL, Databook 88 pg3-15</td>
</tr>
<tr>
<td>DIP14J</td>
<td>14-Fin DIP Package, 3&quot; Spacing For Connector/Reader</td>
</tr>
<tr>
<td>DIP14K</td>
<td>16-Pin DIP Package, 3&quot; Spacing, N014, TTL, Databook 88 pg3-15</td>
</tr>
</tbody>
</table>

Generate HTML catalog of selected libraries. This command generates an HTML table with statistics and images of all the items in the selected libraries as show below.

The Part Types list contains all of the part types found in all the selected libraries. The Part Type Tools will be disabled until you select one or more part types. You may select multiple part types by holding down the Shift or Ctrl keys while clicking on part type items. The Part Type Tools are as follows:

- **Delete Selected Part Types.** This command removes the selected part types from their respected libraries. This is not an undoable operation and you will be prompted for confirmation.

- **Cut Selected Part Types.** This command removes the selected part types from their respected libraries and places them in the paste buffer. You can then select a different set of libraries and paste the part types into them, effectively moving them from one library to another.
Copy Selected Part Types. This command copies the selected part types from their respected libraries and places them in the paste buffer. You can then select a different set of libraries and paste the part types into them, effectively copying them from one library to another.

Paste Part Types. This command will only be available if you have previously cut or copied part types. If part types with the same name already exist in the library you are pasting into, you will be prompted to overwrite them.

The Footprints list contains all of the footprints found in all the selected libraries. The Footprint Tools will be disabled until you select one or more footprints from the list. You may select multiple footprints by holding down the Shift or Ctrl keys while clicking on footprint items. The Footprint Tools are as follows:

Delete Selected Footprints. This command removes the selected footprints from their respected libraries. This is not an undoable operation and you will be prompted for confirmation.

Cut Selected Footprints. This command removes the selected footprints from their respected libraries and places them in the paste buffer. You can then select a different set of libraries and paste the footprints into them, effectively moving them from one library to another.

Copy Selected Footprints. This command copies the selected footprints from their respected libraries and places them in the paste buffer. You can then select a different set of libraries and paste the footprints into them, effectively copying them from one library to another.

Paste Footprints. This command will only be available if you have previously cut or copied footprints. If footprints with the same name already exist in the library you are pasting into, you will be prompted to overwrite them.

Merge Nets Dialog

The Merge Nets Dialog is only invoked when you are in the Connection Tool and you attempt to add a connection between two different nets. Because the two nets must become one, this dialog is provided as a convenience to resolve the final net name.

The two Merge into 'Netxxx' radio buttons let you choose to use one of the existing nets. 'Netxxx' is actually a placeholder. The real net names will appear in its place.

The Use New Net Name radio button, when checked, will allow you to enter a completely different net name for the resulting net. The net name you enter must not already exist in the design.
Net Property Page

The Net Property Page is invoked whenever a net or track/connection is selected and Properties is chosen either by pressing Alt+Enter on your keyboard or RMB/Properties. It can also be invoked by selecting a net in the Object Hierarchy tree and selecting RMB/Properties.

If you only have nets or tracks/connections in your selection set then the only tabs that will appear in the property pages are Nets and Tracks. Otherwise other object tabs may be present too.

If you only have one net selected then all the fields in the Net Property Page will be filled in with the values from that net. If you have more than one net selected then the fields will be empty or in an indeterminate state. Any field left empty or indeterminate when the page is dismissed will be unchanged in the selected nets.

The **Net Name** field is name to assign to the net. It will be forced to upper case and it must be unique in the design. If more than one net is selected then this field will be disabled.

The **Width** field is the default width for tracks in this net. You cannot change existing track widths by changing this field. You must use the Track Property Page for that. The width will be clamped to a range whose maximum must not exceed ½ inch. The minimum width will typically be .007” unless the board is configured to use 2.5oz copper in which case the minimum width will be .012”.

The **Spacing** field is where you can enter the minimum required spacing for this net or nets. It is typically .007” unless the board is configured to use 2.5oz copper in which case the minimum spacing will be .012”. Any physical object that does not belong to this net must meet or exceed this spacing or a DRC error will be issued.

The **Via Size** field is where you specify the default via size for any vias belonging to this net. It is typically defaulted to the smallest size allowed for the given board configuration. You should only change this for special circumstances such as you feel that a via in this net may carry a large current.

The **Drill Size** field is where you specify the default via drill size for any vias belonging to this net. It is typically defaulted to the smallest size allowed for the given board configuration. You should only change this for special circumstances such as you feel that a via in this net may carry a large current.
The **Net Color** button can be used to select the color of the ratsnest for the selected net or nets. This does not affect the color of routes, only the unrouted connections. This can be useful during the placement of components. Critical nets can be set to a different color for easy identification of the net’s topology.

The **Routing Priority** slider is only used by the autorouter. You typically do not need to set this except for unusual circumstances where you feel this net might need the most direct routing and therefore should be routed early in the process. To set the priority low really means that everything else is a high priority except this net.

The **Assigned to Plane Layers** check list will only be available if you have set up plane layers in the Board Configuration. If you have, you can signify that this net is to be assigned to one or more of the plane layers. When a net is assigned to a plane layer then any drill in the net will constitute a plane “hit” and will be tied to that plane layer through a thermal relief.

The **Do Not Optimize** checkbox, when checked, will honor the current connection topology or connection ordering. The default is off which results in a dynamic shortest-length connection ordering base on the relative locations of all the pins in the net.

### Package Print Dialog

The Package Print Dialog in invoked when you are in the Footprint Editor and choose Print from the File Menu. It is used to print a hardcopy of the currently active footprint.

The **X-Shift** and **Y-Shift** fields are used to specify an offset for the footprint. The page origin is the lower-left of the page so positive X shifts are to the right and positive Y shifts are up.

The **Rotation** drop list allows you choose an orthogonal rotation for the footprint when it is printed.

The **Scale** field is used to specify a scale factor for the footprint when it is printed. It is a scalar value and not a ratio. 2.0 is double scale, 0.5 is half scale.

The **Mirror** checkbox, when checked, will flip the footprint image as though you were looking at it from the back.
The Pin Property Page is invoked whenever a component or pin is selected and Properties is chosen either by pressing Alt+Enter on your keyboard or RMB/Properties. It can also be invoked by selecting a component or pin in the Object Hierarchy tree and selecting RMB/Properties.

If you only have components and pins in your selection set then the only tabs that will appear in the property pages are components and pins. Otherwise other object tabs may be present too.

If you only have one pin selected or one single-pin component then all the fields in the Pin Property Page will be filled in with the values from that pin. If you have more than one pin selected then the fields will be empty or in an indeterminate state. Any field left empty or indeterminate when the page is dismissed will be unchanged in the selected pins.

The Pin Property Page is the only mechanism for drilling down into a component without editing that component in the Footprint Editor.

The **Pin Name** field is where you label this pin. If you have multiple pins selected this field will be disabled. Pins can be unlabelled and net connected but they will not appear in any netlists or reports. Pins that belong to components must be unique unless unnamed. Pin names will be forced to use upper case.

The **Net Name** drop list is where you can assign or reassign a net to this pin. If this property page was invoked from the Footprint Editor then this field will be disabled.

The **Round Pad Shape** button sets the shape of this pin to round. Only the pad width needs to be filled in.

The **Rectangular Pad Shape** button sets the shape of this pin to a square or rectangle. Both the pad width and height need to be filled in.

The **Oblong Pad Shape** button sets the shape of this pin to a rectangle with rounded ends. Both the pad width and height need to be filled in and if they are the same value then the result will effectively be a circle.

The **Pad Width** field is used to specify the width of rectangular and oblong pads and the diameter of round pads.

The **Pad Height** field is only enabled for rectangular and oblong pads. It specifies their height. It must not be zero.
The **Hole Size** drop list is used to specify the finished (final) hole size for a pin. Zero is valid, indicating a surface mounted pin. The list of selectable hole sizes changes between plated and non-plated holes. The Pad Size must be larger than the hole size plus a minimum annular ring that will change depending on the copper height of the board. The typical minimum annular ring is 0.007”.

The **Non-Plated** checkbox, when checked signifies that the hole will not be plated and therefore not conduct electricity. They are typically used for mounting and tooling holes.

The **Pin Location X** and **Y** fields are used to position the pin. The coordinate system used depends on the ownership of the pin. If the pin belongs to a component then the location will be relative to the component’s origin. If the pin is a free-standing board-level pin then the location will be relative to the board origin.

The **Pin Rotation** field and spin control are used to specify the pin rotation in the pin’s own local coordinate system. Pin rotations are counter-clockwise unless you enter a negative number in which case they are clockwise but will be converted to the appropriate positive rotation when this dialog is dismissed.

---

**Polygon Property Page**

The Polygon Property Page is invoked whenever a polygon is selected and Properties is chosen either by pressing Alt+Enter on your keyboard or RMB/Properties.

If you only have polygons in your selection then the only tab that will appear in the property pages are Polygons. Otherwise other object tabs may be present too.

If you only have one polygon selected then all the fields in the Polygon Property Page will be filled in with the values from that polygon. If you have more than one polygon selected then the fields will be empty or in an indeterminate state. Any field left empty or indeterminate when the page is dismissed will be unchanged in the selected polygons.

The **Name** field is simply a way to label the polygon. There are no restrictions on polygon names and in fact they are not required.

The **Layer** drop list is where the polygon’s layer is specified. This field will be disabled if the polygon is a board outline as the board outline is assumed to be on all layers.

The **Net** drop list is where you can specify a net that this polygon belongs to. This field will be disabled for Board Outlines. If you assign a net to a copper pour then anything inside the copper pour region that
does not belong to the same net will be cleared away from the copper pour. If you assign a net to a polyline then anything too close or touching the polyline that does not belong to the same net will be flagged with a DRC marker.

The **Width** field is where you specify the line width of a polyline. It is disabled for all other types of polygons.

---

**Print Artwork Dialog**

The Print Artwork Dialog is invoked from the File/Print Artwork menu item and is used to generate detailed prints that match how the boards will be built. See Print Artwork for more details.

The **Select Layers** checklist allows you to check which layers will be combined together for the print. The layer colors will be the same as on the screen. You must have at least one layer checked for the print to happen.

The **Fit to Page** checkbox, when checked, will print the entire design on one sheet of paper (or one logical page). This is mainly used for quick visual checks as the scaling will be arbitrary. When this item is checked, the **Scale** field will be disabled.

The **Scale** field is used to specify a true physical scale of your choosing for the print. The scale can range from one tenth (0.1) to ten times (10.0). If the design does not fit on one page at the scale you choose then the print will span multiple pages that you will have to assemble as one print.

The **Mirror** checkbox, when checked, will produce a print that is mirrored (backwards) as though you flipped the design around and were looking at the back of the board.

The **Print in Full Color** choice will print using the current object color settings. Print artwork will ignore the invisible object settings and always prints the collection of objects for the selected layers in the same way that the boards will be built.

The **Print in Grayscale** choice will print in gray values based on a standard RGB to luminosity conversion.

The **Print in Black and White** choice converts every printed object to black in the print.
Report Dialog

The Board Statistics or Report Dialog is invoked by pressing the Statistics Report button in the Project Pane. This dialog gives you a summary report of various board metrics for the currently active design.

The **Board Area**, always given in square inches, is the rectangular area of the board outline and not the true polygonal area. It is basically what you are being charged for.

**Routing Layers** is the number of layers you have specified for routing. The number of routing layers plus the number of plane layers are the total layers for the design.

**Plane Layers** is the number of layers you have dedicated to continuous vss/vdd planes. The number of routing layers plus the number of planes layers are the total layers for the design.

**Number of Components** is the count of parts on the board with reference designators.

**Component Pins** is the total count of pins inside all components. This includes unnamed pins.

**Other Pins** is the count of board-level pins (not associated with components) you have added to the design.

**Pins Per Sq/Inch** is the average number of pins (both component and board-level pin) per square inch of board area. A number greater than 30 indicates a very dense board.

**Number of Nets** is the count of different networks in the design.

**Number of Connections** is the total number of all point-to-point paths in all nets that have to be routed.

**Unrouted Connections** is the total number of all point-to-point paths that remain to be routed or connected to a plane. This field will be red until all connections are completed.

**Number of Vias** is the total number of vias used by all the routing on the design.

**Total Track Length**, always given in inches, is the linear length of all routing in the design. This does not consider any plane layers, only routes you added. It also does not optimize any route sharing that may have been done – two or more routes from the same net that may have been drawn on top of each other.

**Length to be Routed**, always given in inches, is the theoretical minimal Euclidian length of etch needed to complete routing. Multiplying this number by 1.4 probably gives you a more accurate true distance needed.
**Total Number of Holes** is the grand total of all drilled component pins, board-level pins, and vias in the design.

**Number of Drill Sizes** is total count of all the different drill sizes used in the design.

### Save Footprint Dialog

The Save Package Dialog is invoked from the Footprint Editor by selecting File/Save As, selecting File/Save on an unnamed footprint, or automatically if you close a modified footprint.

You can only save a footprint into a single library. If, for some reason you need to save the footprint into multiple libraries then you can save multiple times or just use the Library Manager to clone the footprint.

The Footprint **Name** Field is where you supply the name of the footprint as it will appear in the library you save to. There are very few restrictions on footprint names except that the name must not contain the vertical bar character '|' and must not begin with an ampersand '&'. It may contain spaces but this is generally a bad idea because it can wreak havoc with report formatting, especially if that report is going to be processed by later tools.

The **Select Library** drop list is where you choose what library you want to save the footprint into. Custom\Custom.slb will be the default library to save to. It is highly recommended that you always save into this library of some other custom library that you created. If you save into one of the standard distribution libraries then you run the risk of it being overwritten during some future software update.

The **New...** button allows you to create a new library before you save the footprint. The newly created library will be selected by default.

The **Description** field is where you can supply a searchable description of the footprint. It can be a readable string or a list of keywords. There are not restrictions.

### Save Template Dialog

The Save Template Dialog can be invoked from the File menu or by RMB while you have objects selected and then choosing Save Selection from the context menu.

The Save template/Save Selection facility aids in design reuse.
The **Description** string gets inserted into the template and is displayed by the Load Template dialog. The load/save template mechanism doesn’t use the standard filename-based approach because it can become difficult to come up with descriptive filenames for templates. The supplied description is cooked into a filename that is stored in the Templates subdirectory.

### Select Color Popup

The Select Color Popup is more like a menu than a dialog and is usually invoked by left clicking on a color widget but in the case of the Layers Display Pane by right clicking on a color square as left clicking in a square toggles visibility.

There is a fixed palette of color swatches in the upper left area, a 1-dimensional grayscale gradient below them, and below that is a 2-dimensional color gradient. As you move the cursor over the various areas the sample swatch will change color. Clicking the left mouse button will select that color.

There is a Visible/Invisible button that toggles the current visibility.

### Text Property Page

The Text Property Page is invoked whenever text is selected and Properties is chosen either by pressing Alt+Enter on your keyboard or RMB/Properties.

If you only have text in your selection then the only tab that will appear in the property pages is Text. Otherwise other object tabs may be present too.

If you only have one piece of text selected then all the fields in the Text Property Page will be filled in with the values from that text. If you have more than one piece of text selected then the fields will be empty or in an indeterminate state. Any field left empty or indeterminate when the page is dismissed will be unchanged in the selected text.

The **String** field is where you specify the actual text string to be rendered. There is not limit to the string length but it will all be on one line. You can use any characters and they will be maintained in the database but the only character symbols currently rendered are the ANSI character set (ASCII codes 20-127).

The **Layer** drop list is where the text's layer is specified.

The **Text Height** field is where you specify the letter height. Letter height is in physical units and logical units like point size. The text height is basically the cell height: not all letters are of uniform height such as the difference between ‘e’ and ‘E’.
The **Line Width** field is where you specify the thickness of the individual strokes that make up a character. A good rule of thumb is the line width should be about 1/10th of the text height. If the line width is too thick as compared to the text height then the text may blur together and become unreadable.

The **X** and **Y Location** fields are used to position the text. The origin of the text is always the lower-left of the first character in the string. The coordinate system used depends on the ownership of the text. If the text belongs to a component then the location will be relative to the component’s origin. If the text is a free-standing board-level piece of text then the location will be relative to the board origin.

The **Text Rotation** field and spin control are used to specify the text rotation in the text’s own local coordinate system. Text rotations are counter-clockwise unless you enter a negative number in which case they are clockwise but will be converted to the appropriate positive rotation when this dialog is dismissed.

The **Mirrored** checkbox, when checked specifies the text string is mirrored. Text placed on the bottom of the board should be mirrored so as to be right-reading when the board is turned around.

**Track Property Page**

The Track Property Page is invoked whenever a track or connection is selected and Properties is chosen either by pressing Alt+Enter on your keyboard or RMB/Properties. It can also be invoked by selecting a track in the Object Hierarchy tree and selecting RMB/Properties.

If you only have tracks or connections in your selection set then the only tabs that will appear in the property pages are Nets and Tracks. Otherwise other object tabs may be present too.

If you only have one track selected then all the fields in the Track Property Page will be filled in with the values from that track. If you have more than one track selected then the fields will be empty or in an indeterminate state. Any field left empty or indeterminate when the page is dismissed will be unchanged in the selected tracks.

The **Routed Width** field is basically the width assigned to the first corner of the track. Other corners may have different widths but they must be done graphically while routing. There currently is no dialog for editing corner-by-corner information for tracks. The width will be clamped to a range whose maximum must not exceed ½ inch. The minimum width will typically be .007” unless the board is configured to use 2.5oz copper in which case the minimum width will be .012”.

The origin of the text is always the lower-left of the first character in the string. The coordinate system used depends on the ownership of the text. If the text belongs to a component then the location will be relative to the component’s origin. If the text is a free-standing board-level piece of text then the location will be relative to the board origin.

The **Text Rotation** field and spin control are used to specify the text rotation in the text’s own local coordinate system. Text rotations are counter-clockwise unless you enter a negative number in which case they are clockwise but will be converted to the appropriate positive rotation when this dialog is dismissed.

The **Mirrored** checkbox, when checked specifies the text string is mirrored. Text placed on the bottom of the board should be mirrored so as to be right-reading when the board is turned around.
Update Components Dialog

The Update Components Dialog will be automatically invoked when you modify a component in the footprint editor and then dismiss the Footprint Editor window. Besides asking you to save the footprint into a library, you will also be given the opportunity to immediately update any affected components with this dialog. This will only be true if you initiated the footprint editor by modifying an existing component.

The **Update Component xxx** radio button should be checked if you only want to update just the component you selected to initiate the footprint editor.

The **Update All Instances of Footprint xxx** radio button should be checked if you have multiple components that use the footprint you just changed and you want to apply the changes to all components that use the footprint. The update only applies to the currently active design that you used initiate the footprint edit. It will not span designs.

The **Do Not Update** radio button should be checked if you do not actually want the footprint changes to update any components.

User Preferences General Page

The User Preferences Dialog is invoked from the View/User Preferences menu item. The General Tab contains a somewhat miscellaneous collection of global settings. When this dialog is invoked all the settings are obtained from the currently active design. You can load a new set of preferences from disk, save the existing settings to disk, reset them factory defaults, or save the settings as the new defaults.

The **Units** radio buttons set the system-wide default units for display and for input. Changing the units does not affect the database at all. There is no internal conversion because the internal representation is independent of the external units display.

The **X and Y Grid** fields specify the current snap and dot grid for the system. They can be changed independently. See also the Change Grid Dialog.
The **Grid Color** swatch is used to set the color of the display grid. See the Color Popup for more details.

The **Dot Grid** and **Line Grid** radio buttons allow you to choose between displaying the grid as a matrix of dots or lines. Dots provide less clutter but lines afford better alignment guides.

The **Full-Screen Cursor** checkbox allows you to turn on and off the full-screen crosshairs that track the mouse.

The **Default Drawing Line Width** field is used to set the default widths of lines and freestyle routing. It does not affect the width of a new route whose net has already had a net width supplied.

The **Default Text Height** field is used to specify the default height of newly added text. If, in the course of adding new text, you change the height from the default, then that new value will override the default value for the remainder of the session.

The **Default Text Line Width** field is used to specify the default line width of newly added text. If, in the course of adding new text, you change the line width from the default, then that new value will override the default value for the remainder of the session.

The **Rotation Increment** field is used to specify the default rotation increment for objects that can be rotated. The default is 90 degrees.

The **Snap Lines to 45 Degrees** checkbox is checked on by default. When snap is on, new lines will snap to 45-degree increments from the last corner drawn. When snap is off, lines will simply snap to the cursor grid regardless of the line slope. Line snap can always be temporarily overridden by holding down the Ctrl key while drawing.

---

**User Preferences Display Page**

The User Preferences Dialog is invoked from the View/User Preferences menu item. The Display Tab contains a somewhat miscellaneous collection of global settings related to drawing styles and object colors. When this dialog is invoked all the settings are obtained from the currently active design. You can load a new set of preferences from disk, save the existing settings to disk, reset them factory defaults, or save the settings as the new defaults.

The **Blend Layers** checkbox controls how overlapping colors from different layers are combined on the screen. When checked, two or more different overlapping colors are combined to make a third color providing a see-through quality for the display. When this is not checked the colors are not blended, creating an overlay effect where one layer will obscure another where they overlap. The benefit of having blend on is
basically more information on the screen, which can also be a drawback on very busy boards with many layers. The default is on.

The **Draw Wireframe** checkbox controls the level of detail at which objects are drawn. When Draw Wireframe is checked all objects are drawn hollow, and at single pixel width. When not checked (the default) all objects are drawn filled and at true width. The benefits of drawing with Wireframe turned on are greatly reduced screen clutter and to check very small objects who are obscured by other objects.

The **Draw Hi-contrast** checkbox allows you to highlight the currently active layer in one color and everything else in a different color. As you change layers only the current layer is in the hi-contrast color, all other layers are in the lo-contrast color. If you set the Lo-Contrast color to invisible then only the current layer is drawn.

The **Show Drills** checkbox controls the display of drill holes inside pads. When checked (the default), the true finished drill size is 'drilled' out of pads and vias. When it is off, then only the pad or via is shown without any drill. Turning drills on has the additional advantage of making pin name and net name display less cluttered as they are shown inside the drill-hole. This checkbox does not affect non-plated holes as they are always displayed.

The **Draw Positive Planes** checkbox controls the polarity in which plane layers are drawn. When this is checked (the default), the interior of the board outline is drawn in the layer color while non-copper areas, such as pads that are not tied to the plane, are drawn in black. If you have a complex plane involving multiple nets that have to be split then it can be easier to draw and visualize the splits by checking this button off and displaying the plane as a negative.

The **Draw Positive Soldermask** checkbox controls the polarity in which the soldermask layers are drawn. When this is checked the interior of the board outline is drawn in the soldermask layer color while non-mask areas, such as pads are drawn in black. If you have a complex soldermask that involves custom unmasking of polygonal areas then it can be easier to draw and visualize checking this button off (the default) and displaying the soldermask as a negative.

The **Draw Hollow Copper** checkbox controls how the interior of solid copper and copper pour is displayed. When this is not checked (the default) copper items are solid filled and copper pour items display the cutouts. On boards with large copper areas or copper areas that overlap on different layers it can be advantageous to turn on hollow copper to speed up screen drawing and prevent flicker.

The **Draw Textured Copper** checkbox allows for the display of textured copper to be turned on and off. Textured copper can provide visual separation on congested copper regions.
The **Background Color** swatch is used to set the background color for the display. By default it is black.

The **Hi-Contrast Color** swatch specifies the color to draw the current layer while in Hi-Contrast mode.

The **Lo-Contrast Color** swatch is used to specify the color to draw all objects not on the current layer while in Hi-Contrast mode.

The **Ratsnest Color** swatch specifies the color to draw ratsnest lines (connections yet to be made).

The **Pin Names Color** swatch is used to specify the color to draw pin names in. It can be invisible.

The **Net Names Color** swatch is used to specify the color to draw net names in. It can be invisible. Net names only appear inside of pads and vias and then only if they fit.

The **Object Colors** swatches are not really a global setting but rather provide an easy way to set the color of a class of objects regardless of what layer they are on.

**User Preferences Layer Colors Page**

The User Preferences Dialog is invoked from the View/User Preferences menu item. The Layer Colors Tab contains per-layer color settings. When this dialog is invoked all the settings are obtained from the currently active design. You can load a new set of preferences from disk, save the existing settings to disk, reset them factory defaults, or save the settings as the new defaults.
All the **Layer Color** swatches control the color of all objects on their respective layers. Setting the color of a layer sets the color of all object types for that layer.

**User Preferences Routing and Spacing Rules Page**

The User Preferences Dialog is invoked from the View/User Preferences menu item. The Routing and Spacing Rules Tab contains settings for the autorouter and for design rule checking. When this dialog is invoked all the settings are obtained from the currently active design. You can load a new set of preferences from disk, save the existing settings to disk, reset them factory default, or save the settings as the new defaults.
The **Global Minimum Spacing - All Layers** field allows you to set the minimum spacing for all layers at once.

The **Per-Layer Minimum Spacing** fields allow you specify the minimum spacing on a per-layer basis. On high volume boards you may want to increase the spacing on the outer, solderable layers while employing less restrictive spacing on the inner layers for additional routing channels.

The **Routing Grid** field controls the granularity by which the autorouter attempts to find paths. The finer the routing grid the denser the routing will be at the expense of time and possibly area. The area restriction will most likely be removed in a future release.

The **Via Bias** drop list controls how liberally the autorouter will use vias while searching for paths. The possible values are:

- **Use Freely** - The autorouter will meander very little on a layer, enforcing a very strict horizontal/vertical layer direction. This is useful if the design has plenty of room for vias but poor ‘flow’ the layout where an excessive amount of meandering may block too many channels.
• **Neutral** - This is the default for the autorouter. It does a reasonable job of reducing via count while maintaining decent layer directionality.

• **Avoid Using** - The autorouter will attempt to meander around objects rather than move to a different layer with a via. This can prove fruitful when there is minimal room for vias.

• **No Vias** - The autorouter will not attempt to use vias at all. It will basically attempt to route the board as a series of single-sided boards.

The **Autorouter Layer Bias** drop lists allow you to set up layer preferences or even restrict the router from using some layers altogether. The possible values are:

• **Favored** - The autorouter will attempt to use this layer (and others with the same setting) before any other layer.

• **Neutral** - This is the default. The autorouter neither prefers nor avoids this layer.

• **Avoid** - The autorouter will only use this layer if it absolutely necessary to complete a path.

• **Do Not Use** - The autorouter will not attempt to use this layer at all. This does not prevent a via from passing through this layer. You can route a single layer board by setting all layers except one to this value and instructing the router not to use vias.

The **Autorouter Layer Direction** drop lists control the preferred layer direction for each layer. The possible values are:

• **Horizontal** - The autorouter will route on this layer in a horizontal direction.

• **Any** - The autorouter will meander in any direction on this layer.

• **Vertical** - The autorouter will route on this layer in a vertical direction.

**Zoom To Coordinate Dialog**

The Zoom to Coordinates dialog is invoked by pressing the ‘X’ key in the Design or Edit Footprint view. It is used to center the viewport on a specific coordinate.

The **Enter Coordinate** field is initially defaulted to the current gridded cursor location at the time of invocation. Enter the X and Y coordinates you desire in the format of your choice. The coordinates can be separated by spaces or commas.
Shortcuts

In a number of locations throughout PCB123 – especially where specifying pin or landing pad names – users can use some shortcut notations to speed up and simplify the process of adding pin names.

A Bar (line) can be placed over pin names and net names by appending two consecutive hyphens (--) to the end of the name.

Sequences of pin names, pin numbers, and net names can be generated using shorthand notation. All shorthand within a base name appears inside square brackets [], and will expand using the following rules:

- `[NAME1, NAME2, NAME3, …]` - Disjoint list. A list of names separated by commas will expand once for each name when placing consecutive objects. Example:

  INTENS_[RED,GREEN,BLUE] - expands for the next 3 items as:
  
  INTENS_RED
  INTENS_GREEN
  INTENS_BLUE

- `[#:#:(:#)]` - Numeric Range. Two numbers separated by a colon (;) denote an inclusive sequential range. If a third number is supplied, then it is the increment, or step, to be used when generating the sequence. Example:

  ADDR[3:0] - expands for the next 4 items as:
  
  ADDR3
  ADDR2
  ADDR1
  ADDR0

    [0:100:10]_DEC - expands to:
    
    0_DEC
    10_DEC
    20_DEC
Key Assignments

The Key Assignments Dialog can be invoked under Help/Key Assignments. By pointing at a key on the pictorial keyboard or actually typing a key on your physical keyboard you will be presented with information on what that key does by itself or when the Ctrl or Shift key is held in combination.
<table>
<thead>
<tr>
<th>Key</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ctrl-C</td>
<td>Copy Selection</td>
</tr>
<tr>
<td>F</td>
<td>Find an object</td>
</tr>
<tr>
<td>Ctrl-F</td>
<td>Snap object to cursor</td>
</tr>
<tr>
<td>G</td>
<td>Change working Grid</td>
</tr>
<tr>
<td>L</td>
<td>Change Layer</td>
</tr>
<tr>
<td>M</td>
<td>Measure Distance</td>
</tr>
<tr>
<td>N</td>
<td>Next DRC Error / Zoom</td>
</tr>
<tr>
<td>Ctrl-N</td>
<td>New Document</td>
</tr>
<tr>
<td>O</td>
<td>Move component to opposite side of board</td>
</tr>
<tr>
<td>Ctrl-O</td>
<td>Open saved document</td>
</tr>
<tr>
<td>P</td>
<td>View object properties</td>
</tr>
<tr>
<td>Ctrl-P</td>
<td>Print</td>
</tr>
<tr>
<td>R</td>
<td>Rotate Object</td>
</tr>
<tr>
<td>Ctrl-S</td>
<td>Save file</td>
</tr>
<tr>
<td>U</td>
<td>Unroute Object</td>
</tr>
<tr>
<td>V</td>
<td>Insert Fanout Via</td>
</tr>
<tr>
<td>Ctrl-V</td>
<td>Paste</td>
</tr>
<tr>
<td>W</td>
<td>Change line width</td>
</tr>
<tr>
<td>X</td>
<td>Zoom to Coordinate</td>
</tr>
<tr>
<td>Ctrl-X</td>
<td>Cut selection</td>
</tr>
<tr>
<td>Ctrl-Y</td>
<td>redo</td>
</tr>
<tr>
<td>Z</td>
<td>Swaps routing to complementary layer; inserts a via.</td>
</tr>
<tr>
<td>Ctrl-Z</td>
<td>Undo</td>
</tr>
<tr>
<td>F5</td>
<td>Refresh Screen</td>
</tr>
<tr>
<td>Up arrow</td>
<td>Zoom up one page</td>
</tr>
<tr>
<td>Down arrow</td>
<td>Zoom down one page</td>
</tr>
<tr>
<td>Right arrow</td>
<td>Zoom right one page</td>
</tr>
<tr>
<td>Left arrow</td>
<td>Zoom left one page</td>
</tr>
<tr>
<td>ESC</td>
<td>Cancel Current operation; terminate mode</td>
</tr>
<tr>
<td>SHIFT</td>
<td>Pans the display with mouse movement</td>
</tr>
<tr>
<td>CTRL</td>
<td>Allows any-angle routing when held during route procedure</td>
</tr>
<tr>
<td>HOME</td>
<td>Full fit screen</td>
</tr>
<tr>
<td>END</td>
<td>Refresh screen</td>
</tr>
<tr>
<td>DEL</td>
<td>Delete selected object</td>
</tr>
<tr>
<td>Page UP</td>
<td>Zoom In</td>
</tr>
<tr>
<td>Page DOWN</td>
<td>Zoom Out</td>
</tr>
</tbody>
</table>