

FpcLib

A Footprint Library Manager for *FreePCB*

User Guide Ver 1.30

```
Cmd> list bob *
```

```
File: bob.fpl, 6 entries
```

```
1 "LED-3PIN"  
2 "LED-T1"  
3 "LED-T1.75"  
4 "P500"  
5 "P503"  
6 "P507"
```

```
Cmd> dir
```

1 bob.fpl	Sat Sep 01 14:44:28 2007	3766 A
2 bob1.fpl	Sat Sep 01 14:22:23 2007	3766 A
3 led.fpl	Sat Sep 01 16:28:11 2007	14871 A
4 led_new.fpl	Sun Aug 26 14:18:41 2007	3360 A
5 QFN_Pkgs.fpl	Mon Dec 04 07:01:30 2006	10709 A
6 SMD_Test.fpl	Sun Sep 02 13:02:17 2007	1312 A
7 SMD.fpl	Sun Sep 02 13:01:57 2007	1312 A
8 user_created.fpl	Sun Sep 02 07:27:57 2007	741165 A
9 Xilinx_BGAs.fpl	Sun Sep 02 07:39:58 2007	597818 A

```
Cmd> list user_created FF11*
```

```
File: user_created.fpl, 60 entries
```

```
Name: FF1152
```

```
Author: OBP
```

```
Source: Altera/Xilinx generic
```

```
Courtyard: 35.98 x 35.98 mm
```

```
Centroid: 16.50, -16.5 mm Type: Default Angle: 90
```

```
Pins: 1156 Polygons: 1 Text Blocks: 0 Glue Spots: 2
```

```
Cmd>
```

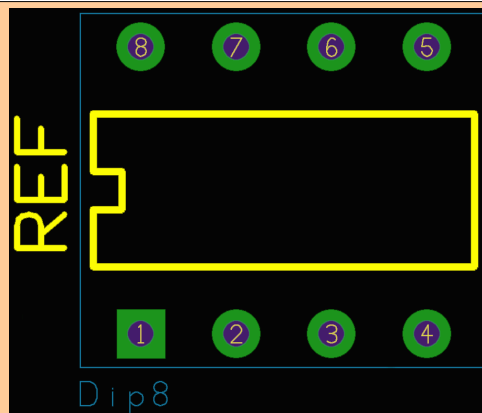


Table of Contents

FpcLib End User License Agreement (EULA).....	5
Introduction.....	7
Installation and Setup.....	8
Operation.....	9
Libraries.....	10
Command Summary.....	11
Command Details.....	12
Add - Add footprints from another library.....	12
Clean – Delete Footprints with Data Errors.....	13
Del - Delete footprint(s).....	14
Dir - Directory list.....	15
End - End Script or ForEach block.....	16
Extract – Extract the Footprints from a Project file.....	16
For - ForEach block start.....	17
Help or ? - Help.....	18
List - List footprints.....	19
Open - Open library.....	20
Plot - Plot footprints to Gerber.....	20
Quit - Quit FpcLib.....	24
Ren - Rename a footprint.....	24
Rotate – Rotate Footprints.....	25
Save - Library Save / Save As.....	27
Sort - Sort footprints by name.....	28
Test – Test a Library for Errors.....	29
Command Scripts.....	30
Temporary Files.....	30
Appendix A – Errors and Warnings.....	31
Fatal errors.....	31
Library data error and warning messages.....	32
Appendix B – Gerber Files and Viewers.....	35
Revision History.....	36

Note:

Normal text in this document uses the Times New Roman font.
Console I/O examples use the Courier New font
with user input highlighted in dark red.
This document is formatted for two-sided printing.

FpcLib End User License Agreement (EULA)

General

FpcLib Software is distributed as Freeware. You may use the Software on any number of computers for as long as you like. The Software is NOT Public Domain software. You are allowed to freely distribute the Software, but Bruce Parham retains ownership and copyright of the Software in its entirety. You may use and/or distribute the Software only subject to the following conditions:

1. You may not modify the program or documentation files in any way.
2. You must include all the files that were in the original distribution.
3. You may not reverse engineer, decompile, or otherwise reduce the Software to a human perceivable form.
4. You may not sell the Software or charge a distribution fee, except to recover the media costs.
5. You understand and agree with this license and with the Disclaimer of Warranty and the Limitation of Liability printed below.
6. The Software may be bundled and distributed together with other software products provided the same conditions apply.

Agreement to this license

By using, copying, transmitting, distributing or installing the Software, you agree to all terms of this license. If you do not agree to all of the terms of this License, then do not use, copy, transmit, distribute or install the Software and immediately remove the Software from your storage device(s).

Warranty disclaimers and liability limitations

The Software and related documentation are provided "as is", without warranty of any kind. None whatsoever. Bruce Parham disclaims all warranties, express or implied, including, but not limited to, warranties of design, merchantability, or fitness for a particular purpose or any warranty of title or non-infringement. Bruce Parham does not warrant that the functions contained in the Software or documentation will meet your requirements, or that the operation of the Software will be error-free, complete, or that defects in the Software or documentation will be corrected.

Limitation of Liability

Under no circumstances shall Bruce Parham be liable for any lost revenue or profits or any incidental, indirect, special, or consequential damages that result from the use or inability to use the Software or related products or documentation, no matter what legal theory it is based on.

Copyright and trademarks

FpcLib is Copyright © 2007-2009 by Bruce Parham. All Rights Reserved.

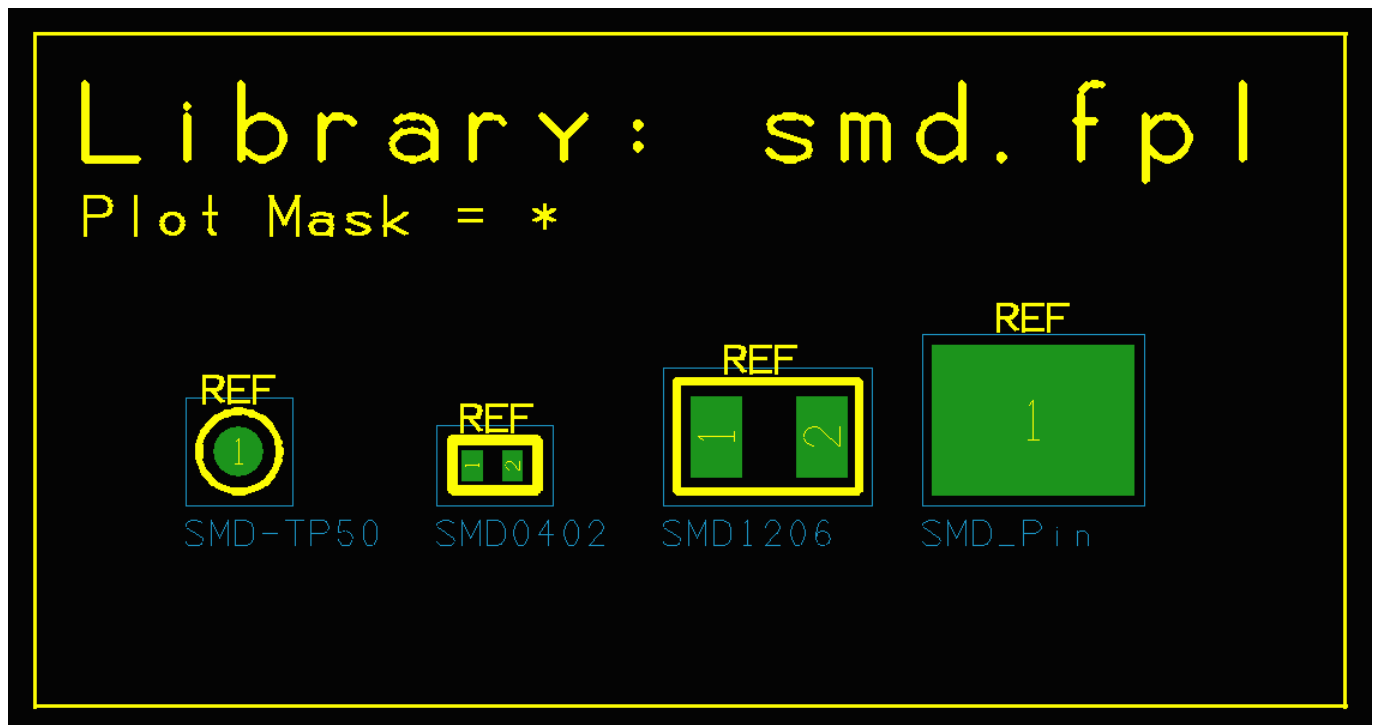
Introduction

FpcLib is a simple Win2K/XP console application that provides basic library management for **FreePCB** footprint libraries.

The major **FpcLib** functions include:

- Sort library footprints by name.
- Add footprints from another library.
- Delete footprints.
- Rename footprints.
- Create rotated footprints.
- Test libraries for valid data.
- Remove invalid footprints.
- Plot footprints to a set of RS-274X Gerber files.
- Backup and save the edited library or save as a new library.

Once started, **FpcLib** is controlled through a command interpreter. **FpcLib** can be also be started with a command script file. Commands may be entered as complete words or just one or two characters, as needed, to uniquely specify the command. Additional facilities are provided to allow a group of commands to be applied repetitively to a list of files.



Installation and Setup

FpcLib is distributed as a set of files in a zip archive; no install utility is used. To install **FpcLib**, extract the FpcLib.exe and font.xml files to any handy directory such as C:\Program Files\FpcLib\ or C:\Program Files\FreePCB\bin\.

If the path to FpcLib.exe is not already included in the **PATH** variable, it must be added. Additionally, if **FPCfab** has not been previously installed, a new environment variable, **FPCfabFont**, must be created and set to point to the newly installed font.xml file.

For those unfamiliar with setting environment variables, assume for example that FpcLib.exe and font.xml is installed in the C:\Program Files\FreePCB\bin directory:

- Open the **System Properties** form by either
 - Open the **Control Panel**
 - Open the **System** tool
- or
 - Right click **My Computer**
 - Click on **Properties**
- Click the **Advanced** tab and open the **Environment Variables**
- Under the **System Variables** click **New**
- Set the variable name to **FPCfabFont**
- The variable value to **C:\Program Files\FreePCB\bin**
- Click on **OK** to add the new variable
- Scroll the variable list down to and highlight the **path** entry and click **EDIT**
- Click in the value field to clear the highlight
- Scroll to the end of the field and add **;C:\Program Files\FreePCB\bin** to the existing value
- Click **OK** to exit the edit window
- Click **OK** to save and exit the Environment Variables window

Operation

To start *FpcLib*, open a DOS shell (Start→Run→Cmd) and make the directory where your libraries are located the current working directory. At the command prompt type `fpclib libname` where *libname* is any valid *FreePCB* library name. The *.fpl* file extension need not be entered; it will be added if no extension is present. *FpcLib* **must** be started with a valid library but, if no library name is entered, FpcLib will attempt to open the default "user_created.fpl".

Ex:

```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\BPs Box> cd \program files\freepcb\lib
C:\Program Files\FreePCB\lib> fpclib user_created
```

When *FpcLib* is started, it prints a sign on message, loads, tests and lists the specified library and starts it's own command interpreter. All subsequent commands that modify the library apply only to the currently loaded file.

Ex:

```
FpcLib - a Librarian for FreePCB
Ver 1.11R (c)2007 Bruce Parham

File: user_created.fpl, 60 entries

  1 "1mm_fiducial"
  2 "2sidedthing"
  3 "3x10_50mil"
  .
  .
  .
 57 "SMD_Pin"
 58 "Thru-Pin"
 59 "untitled"
 60 "Xilinx-TQ144"

Cmd>
```

At the **Cmd>** prompt, commands can be entered, with one or more parameters as needed. Only the first one or two characters of a command are significant and commands are not case sensitive. Commands and parameters must be separated by one or more space characters. Parameters that include spaces must be enclosed in double quotes.

Ex:

```
Cmd> Add Xilinx_BGAs FF1152      ← add two footprints from the Xilinx_BGAs library
Cmd> a Xilinx_BGAs "FF740 Rev2" ← second name is quoted because it includes a space
```

Many of the parameters can make use of wild card '*' and '?' characters to specify multiple footprints or files per DOS convention.

Ex:

```
Cmd> del FF9*      ← delete all footprints that begin with "FF9"
Cmd> Del DIP?0?    ← delete "DIP-08" and "DIP_06"
Cmd> Dir *.*       ← list all file in the current directory
```

Important Note: Footprint names are case sensitive both in *FreePCB* and *FpcLib*.

Libraries

A *FreePCB* library is a plain text collection of footprint data objects separated by one or more blank lines. Each footprint contains a number of required data records, in a more or less prescribed order beginning with the *name:* record, and may contain additional optional records. Each record type generally has a unique format but all are insensitive to leading spaces and extra spaces between parameters.

In theory, a footprint may describe any combination of *text*, polyline *graphics* and *pins* but the current version of the footprint editor (1.338) fails to properly dimension the outline of a text only footprint. So, for now, footprints must contain graphics and/or pins and may optionally contain text.

The following example illustrates a minimal footprint with a single SMT pin:

```
name: "1mm_fiducial"
units: NM
sel_rect: -754000 -754000 754000 754000
ref_text: 2540000 2540000 5080000 0 254000
n_pins: 1
pin: "1" 0 0 0 0
top_pad: 1 1000000 0 0 0
```

And here's one with text, graphics and an optional *author:* record. Any combination of *author:*, *source:*, or *description:* records may be present.

```
name: "Big_X"
author: "B. Parham"
units: MIL
sel_rect: -315 -315 315 315
ref_text: 20 -30 230 0 1
text: "Big X" 100 140 0 0 10 0 4
outline_polyline: 10 -300 300
next_corner: 300 -300 0
outline_polyline: 10 300 300
next_corner: -300 -300 0
outline_polyline: 10 0 300
next_corner: -300 0 2
next_corner: 0 -300 2
next_corner: 300 0 2
close_polyline: 2
n_pins: 0
```

When *FpcLib* loads a library, each footprint's data is checked for parameter counts, valid parameter values and the presence of required records. When an error is found, the footprint is flagged as containing errors, further loading of that footprint is abandoned and the load process continues with the next footprint. For this reason, any footprint containing errors is considered incomplete and unusable. The save command is disabled as long as the loaded library contains footprints with errors.

During loading, most values are also given a sanity check; if a value or size is found to be unreasonable, a warning is issued. Warning messages are only intended to be a "heads up" that a footprint may cause later problems; the footprints are complete and the library may be saved.

When the load is completed, the library is checked for name duplications. A name duplication by itself, although an error, does not prevent the library from being used because the data is complete and the name can be easily changed.

Command Summary

Add <name> <pname>	Add footprint(s) <pname> from lib <name>
Clean [<pname>]	Delete footprint(s) <pname> with data errors (default = * (all))
Del <pname> [n]	Delete footprint(s) <pname> from database
Dir [<fname>]	Directory list (default: "*.fpl")
End	End script file processing or ForEach block
Extract <ename>	Extract and load the footprint cache from project file <ename> (default extension = .fpc)
For	ForEach block begin. One pass per Dir list item
Help or ?	Help (this list)
List	List library footprints
List <pname>	List footprint(s) <pname> details
List <name> <pname>	List footprint(s) <pname> details of file <name>
Open <name>	Open library <name>
Plot [/sw] [<pname>] ...	Plot footprints to Gerber (default pname = * (all)) (Plot /? for switch details)
Quit	Quit FpcLib
Ren <n1> <n2> [n]	Rename the Nth footprint named <n1> to <n2> (default n=1)
Rot <pname> <angle>	Rotate footprint <pname> <angle> degrees and save as as new footprint named "\"name_ROT_<angle>\""
Sort [-]	Sort footprints by name in alpha order (A->Z) Reverse sort (Z->A) if - parameter entered
Save	Save library file. Source renamed <name>.fpl.bak
Save <name>	Save As library to <name>
Test <name>	Test library file <name>

- Cmd characters are not case sensitive. Long cmds OK, only 1 or 2 chars used
- <name> = footprint or file name, default file extension = .FPL
- <fname> = file name with wildcard matching. default file extension = .FPL
- <pname> = footprint name with wildcard matching. no defaults.
- Within ForEach blocks:
 - "\$" is replaced by the current dir list file name.
 - "@" is replaced by the current dir list file name without extension.
 - "#" is replaced by the current dir list item number.

Command Details

Add - Add footprints from another library

Add one or more footprints from another library. If a new footprint name matches an existing footprint, you are given the option whether or not to replace the existing footprint. Footprints with data errors will not be added. Two parameters are required:

Param1 Specifies the source library file. Wildcard characters may not be used but the default file extension *.fpl* will be appended if no extension is present. File names are not case sensitive.

Param2 Specifies the footprint or footprints to add. Wildcard characters '*' and '?' can be used to select multiple footprints. Footprint names *are* case sensitive.

Ex:

```
Cmd> Add user_created DIP-08      ← Add footprint "DIP-08" from user_created.fpl
```

```
Adding DIP-08
1 footprints added/updated from user_created.fpl
```

```
Cmd> A leds LED*                  ← Add all footprints that begin with "LED"
```

```
(Y/N) LED-3PIN exists, replace? Y  ← Skip (n) or replace (y) existing footprint
```

```
Replacing LED-3PIN
Adding LED-T1
Adding LED-T1.75
3 footprints added/updated from leds.fpl
```

```
Cmd> add test_fpclib EIA*         ← Add all footprints that begin with "EIA" from a library with errors
```

Errors:

```
** (test_fpclib.fpl - line 6483) "feed_thru" sel_rect record missing 1 parameter **
** (test_fpclib.fpl - line 6502) "SMT Feed-Thru small" invalid bottom SMT pad shape: none **
** (test_fpclib.fpl - line 7069) "EIA-0603" degenerate Polyline record **
```

```
Adding EIA-0805
Adding EIA-0402
test_fpclib.fpl "EIA-0603" has errors, skipped  ← Footprints with data errors are not added
Adding EIA-1210
3 footprints added/updated from test_fpclib.fpl
```

Clean – Delete Footprints with Data Errors

Clean up the loaded library by deleting footprints that contain data errors. Only those footprints flagged as having data errors are removed. One optional parameter, which may contain wildcard characters, can be entered to select which footprints are deleted. If no parameter is entered, "*" is used.

```
Cmd> open test_fpclib                               ← Load a library with errors

File: test_fpclib.fpl, 192 entries

  1 "SMT Feed-Thru small" ** Error: Duplicate name **
  2 "SMT Feed-Thru small" ** Error: Duplicate name **
  3 "PAD-.250 rect_6-32"
  .
  .
154 "SW-mhps2273"
155 "PAD_TH .1 x .028"
156 "feed_thru"   ** Footprint data error(s) **
157 "SMT Feed-Thru small" ** Error: Duplicate name ** ** Footprint data error(s) **
158 "LQH32C"
  .
  .
179 "TP1-Small"
180 "EIA-0603"   ** Footprint data error(s) **
181 "TANT_A+"
  .
  .
192 "EDGE-8X2"

** 3 duplicate names found **
** 3 Footprints have data error(s) **

** (test_fpclib.fpl - line 6483) "feed_thru" sel_rect record missing 1 parameter **
** (test_fpclib.fpl - line 6502) "SMT Feed-Thru small" invalid bottom SMT pad shape: none **
** (test_fpclib.fpl - line 7069) "EIA-0603" degenerate Polyline record **

Cmd> clean                                           ← Delete all with error (same as "Clean *")

(Y/N) Remove ALL footprints with data errors? y     ← Query for all ("*") mask

feed thru removed                                  ← Listed as removed
SMT Feed-Thru small removed
EIA-0603 removed
3 footprints removed

Cmd>
```

Del - Delete footprint(s)

This command deletes footprints from the loaded library. One parameter is required to specify which footprint or footprints to remove. The footprint mask parameter may contain wildcard characters to select multiple footprints.

Param1 Specifies the footprint or footprints to remove. Wildcard characters may be used to specify multiple footprints.

Param2 This parameter is optional and only used if Param1 contains no wildcard characters. It specifies which copy of a duplicated footprint to remove. This parameter defaults to 1 if not entered.

Ex:

```
Cmd> de LED*                ← Remove all footprints starting with LED

LED-3PIN removed
LED-T1 removed
LED-T1.75 removed
3 footprints removed

Cmd> 1

File: bob.fpl, 3 entries

1 "P500"
2 "P503"
3 "P507"
```

The special, two parameter, form of this command exists to handle corrupted libraries with duplicate footprints¹. The first parameter, the name mask, must specify the exact name without any wildcard characters. The second parameter specifies which copy to remove.

Ex:

```
Cmd> so                      ← Sorting reveals two copies of "P500"

File: bob.fpl, 7 entries

1 "LED-3PIN"
2 "LED-T1.75"
3 "P500" ** Error: Duplicate name **
4 "P500" ** Error: Duplicate name **
5 "P503"
6 "P507" ** Warning: Name match, dif by case only **
7 "p507" ** Warning: Name match, dif by case only **

Cmd> del P500 2              ← Delete the second copy

P500 removed
1 footprint removed

Cmd> so                      ← Resorting updates dup status

File: bob.fpl, 6 entries

1 "LED-3PIN"
2 "LED-T1.75"
3 "P500"
4 "P503"
5 "P507" ** Warning: Name match, dif by case only **
6 "p507" ** Warning: Name match, dif by case only **
```

¹ Editing libraries with a text editor can result in duplicated footprints. *FreePCB* and its *Footprint Editor* do not handle this well.

Dir - Directory list

Display and save a directory listing of selected files. One parameter can be used to specify the file name mask. If no parameter is entered, **.fpl* will be used. If a mask is entered without a file extension, *.fpl* will be appended. Relative or absolute paths can be used to display the contents of other directories. The path is saved along with the file names allowing the dir list to be used by subsequent ForEach commands.

The displayed list shows the file name, modification time/date, file size in bytes and file attributes. In addition, the names of all normal files (files without ReadOnly, Hidden, System or Directory attributes set) are saved to a file list that may be used by subsequent ForEach commands.

Ex:

Cmd> **di** ← Directory list using default "*.fpl" mask

1	bob.fpl	Mon Sep 03 12:41:12 2007	3766	A
2	bob1.fpl	Mon Sep 03 12:41:12 2007	3766	A
3	led.fpl	Mon Sep 03 12:41:12 2007	14871	A
4	led_new.fpl	Mon Sep 03 12:41:12 2007	3770	A
5	QFN_Pkgs.fpl	Mon Sep 03 12:41:12 2007	11105	A
6	SMD_Test.fpl	Sun Sep 02 13:02:17 2007	1312	A
7	SMD.fpl	Mon Sep 03 12:41:12 2007	1312	A
8	test1.fpl	Mon Sep 03 16:10:38 2007	14871	A
9	user_created.fpl	Mon Sep 03 12:41:13 2007	741165	A
10	Xilinx_BGAs.fpl	Mon Sep 03 12:41:13 2007	597818	A

Cmd> **dir *.*** ← List everything

1	.	Mon Sep 03 17:48:00 2007	0	Dir
2	..	Mon Sep 03 17:48:21 2007	0	Dir
3	bob.fpl	Mon Sep 03 12:41:12 2007	3766	A
4	bob1.fpl	Mon Sep 03 12:41:12 2007	3766	A
5	FpcLib.exe	Mon Sep 03 16:09:23 2007	73728	A
6	FpcLibDirList.txt	Mon Sep 03 17:48:21 2007	0	A
7	led.fpl	Mon Sep 03 12:41:12 2007	14871	A
8	led_new.fpl	Mon Sep 03 12:41:12 2007	3770	A
9	New Text Document.txt	Mon Sep 03 06:29:51 2007	0	A H Ro
10	QFN_Pkgs.fpl	Mon Sep 03 12:41:12 2007	11105	A
11	QFN_Pkgs.fpl.bak	Mon Sep 03 12:40:16 2007	11105	A
12	SMD_Test.fpl	Sun Sep 02 13:02:17 2007	1312	A
13	SMD.fpl	Mon Sep 03 12:41:12 2007	1312	A
14	test_script.flc	Sun Sep 02 10:23:18 2007	75	A
15	test1.fpl	Mon Sep 03 16:10:38 2007	14871	A
16	user_created.fpl	Mon Sep 03 12:41:13 2007	741165	A
17	Xilinx_BGAs.fpl	Mon Sep 03 12:41:13 2007	597818	A
18	Xilinx_BGAs.fpl.bak	Mon Sep 03 12:40:17 2007	597818	A

Cmd> **dir smd*** ← Just the smd libs

1	SMD_Test.fpl	Mon Sep 03 21:08:58 2007	1312	A
2	SMD.fpl	Mon Sep 03 21:08:58 2007	1312	A

Cmd> **di ???_*** ← Fourth character is an underscore

1	led_new.fpl	Mon Sep 03 21:08:58 2007	3770	A
2	QFN_Pkgs.fpl	Mon Sep 03 21:08:58 2007	11105	A

End - End Script or ForEach block

Terminate processing of a script file or end a **ForEach** command block entry.

Scripts normally terminate when a Quit command is executed, ending *FpcLib*, or when the script is exhausted, returning control to the command interpreter. The **End** command can be used to stop a script and return control to the command interpreter prior to reaching the End-of-File.

ForEach block entry stores each command until an **End** is entered.

Extract – Extract the Footprints from a Project file

Extract and load the footprint cache of a FreePCB project (.FPC) file. Once loaded, the library may be manipulated by any valid command, **Sort**, **Plot**, **List**, etc, but may only be saved with the **Save As** command. Extract uses one parameter to specify the project file for extraction; the default file extension *.FPC* will be appended if no extension is present.

Ex:

```
Cmd> extract demo_1sided ← Extract the footprint cache from "demo_1sided.fpc"
```

```
File: "- Extracted Cache -", 15 entries
```

```
1 "CHIP_B"
2 "CHIP_D"
3 "14DIP300"
4 "Logo1"
5 "1X2HDR-100"
6 "C1206"
7 "HOLE_125_SQR_250"
8 "5X2HDR-100"
9 "28DIP300_A5"
10 "C0805"
11 "C-Right07"
12 "Hole-100mil"
13 "TO-220"
14 "8DIP300"
15 "3X2HDR-100"
```

```
Cmd> save ← Simple save fails because no parent library exists
```

```
Rename failed!
```

```
Cmd> save demo_1S_extracted ← Use Save As to create a new library
```

```
save complete
```


For - ForEach block start

ForEach allows a block of commands to be defined. Once the command block is defined and confirmed, it is executed once for each file found in the last directory listing.

During each pass through the command block, the meta character **\$** is replaced with the current file name from the list. This allows a set of files to be loaded, modified and then saved or saved to a new file by adding prefix characters to the name, i.e. **Save New_\$**. The parameter containing the **\$** character will automatically be quoted if the current file name includes any spaces.

Two additional meta characters **@** and **#** exist. The **@** sign will expand into the current file name without the extension or period: "QFN Pkgs", and **#** is replaced by the directory listing line number.

While a **ForEach** block is executing, the **Yes/No** query mechanism is bypassed and execution continues as if a **Yes** were entered. The two commands **Dir** and **For** are not allowed within a ForEach block.

Ex - sort all libraries:

```
Cmd> dir ← Step 1 – Get a directory list for all .fpl files
 1 bob.fpl Mon Sep 03 12:41:12 2007 3766 A
 2 bob 1.fpl Mon Sep 03 12:41:12 2007 3766 A
 3 led.fpl Mon Sep 03 12:41:12 2007 14871 A
 4 led_new.fpl Mon Sep 03 12:41:12 2007 3770 A
 5 QFN_Pkgs.fpl Mon Sep 03 12:41:12 2007 11105 A
 6 SMD_Test.fpl Sun Sep 02 13:02:17 2007 1312 A
 7 SMD.fpl Mon Sep 03 12:41:12 2007 1312 A
 8 test1.fpl Mon Sep 03 16:10:38 2007 14871 A
 9 user_created.fpl Mon Sep 03 12:41:13 2007 741165 A
10 Xilinx_BGAs.fpl Mon Sep 03 12:41:13 2007 597818 A
```

```
Cmd> f ← Step 2 – Define the command sequence
```

```
Build> o $ ← Replace $ with file name and open the file
```

```
Build> so
```

```
Build> sa
```

```
Build> en
```

```
(Y/N) Run? Y ← Step 3 – Execute the sequence
```

```
For-> o bob.fpl ← Pass 1: Open file ($ replaced with bob.fpl)
```

```
File: bob.fpl, 6 entries
```

```
1 "P507"
2 "P503"
3 "P500"
4 "LED-T1.75"
5 "LED-T1"
6 "LED-3PIN"
```

```
For-> so ← Sort
```

```
File: bob.fpl, 6 entries
```

```
1 "LED-3PIN"
2 "LED-T1"
3 "LED-T1.75"
4 "P500"
5 "P503"
6 "P507"
```

```
For-> sa ← Backup and save bob.fpl
```

For-> o "bob 1.fpl" ← Pass 2: Open file (\$ replaced with "bob 1.fpl")

File: bob 1.fpl, 6 entries

```
1 "P507"
2 "P503"
3 "P500"
4 "LED-T1.75"
5 "LED-T1"
6 "LED-3PIN"
```

For-> so ← Sort

File: bob 1.fpl, 6 entries

```
1 "LED-3PIN"
2 "LED-T1"
3 "LED-T1.75"
4 "P500"
5 "P503"
6 "P507"
```

For-> sa ← Backup and save bob 1.fpl

... and so on until all ten passes are completed. Another use would be to consolidate all footprints of a certain type into one new library:

Cmd> del * ← Empty the current library

(Y/N) Remove all? Y

Cmd> f ← Get all forms of DIP* from all libraries

Build> a \$ DIP*

Build> a \$ Dip*

Build> a \$ dip*

Build> en

(Y/N) Run? Y

.
.
.

Cmd> sa new_dips ← Save to a new file not in the dir list

Cmd> f ← Remove from all other files

Build> o \$ ← Open each file

Build> del DIP* ← Delete

Build> del Dip*

Build> del dip*

Build> save mod_\$ ← Save to a new library by adding a prefix

Build> en

(Y/N) Run? y

.
.
.

Help or ? - Help

Display the command summary as shown on page 11.

List - List footprints

This command lists the footprints in either the current library or an external file. The displayed order reflects the current footprint order in the library.

The command has three forms as determined by the number of parameters present.

Form 1 uses no parameters and produces a short, name only, listing of the currently loaded library.

Ex:

```
Cmd> 1
```

```
File: Xilinx_BGAs.fpl, 7 entries      ← Current library and footprint count
```

```
1 "FF1152"      ← Footprint names
2 "FF1513"
3 "FF1600"
4 "FF1600-1"
5 "FF1760"
6 "FF64"
7 "FF961"
```

Form 2 uses one parameter, with possible wildcard characters, as a footprint name mask to produce a more detailed listing of some or all of the currently loaded footprints.

Ex:

```
Cmd> 1 P*      ← List all footprints whose name starts with "P"
```

```
File: led.fpl, 8 entries
```

```
Name: P500      ← Footprint name
  Author: Ivex   ← Non blank optional text fields
  Source: DIGIKEY CATALOG NO.941, PG. 121
  Courtyard: 96.0 x 155.0 mil      ← Bounding box size and footprint units
  Pins: 2 Polygons: 5 Text Blocks: 0      ← Footprint objects summary

Name: P503
  Author: Ivex
  Source: DIGIKEY CATALOG NO.941, PG. 121
  Courtyard: 104.0 x 197.0 mil
  Pins: 2 Polygons: 5 Text Blocks: 0
```

Form 3 lists the contents of another file, in the more detailed format of form 2, without loading it. Two parameters are used: the first parameter, without wildcards, is the file name and the second is the footprint name mask.

Ex:

```
Cmd> 1 smd ???0*      ← List any footprint whose name has the fourth character = "0" in smd.fpl
```

```
File: smd.fpl, 4 entries
```

```
Name: SMD0402
  Courtyard: 115.0 x 80.0 mil
  Pins: 2 Polygons: 1 Text Blocks: 0
```

Open - Open library

This command loads a new library into memory. If the currently loaded library has been modified, the user is asked to verify the load operation. One parameter, without wildcards, is used to specify the new library. The new library name may contain a relative or absolute path.

Ex:

```
Cmd> o bob                               ← Open bob.fpl
      (Y/N) Library changed, replace? Y   ← Overwrite query (if the currently loaded lib has been modified)
File: bob.fpl, 6 entries                  ← Display the newly load library
1 "LED-3PIN"
2 "LED-T1"
3 "LED-T1.75"
4 "P500"
5 "P503"
6 "P507"
```

Plot - Plot footprints to Gerber

This command will generate a set of RS-274X Gerber files to graphically depict the selected footprints. One or more parameters may be used, with wildcards, to specify which footprints are plotted. If no parameters are entered, * is used and all footprints are plotted. Optional switches, *not case sensitive*, can be used to control some plot parameters. Any switches must be first, before any footprint name masks.

Switches: /D Allow duplicate footprint plots
 /H or /? Help
 /Wn.nnn Change the plot width to n.nnn inches

Note that using multiple wildcard parameters can allow a footprint to be selected more than once, ex: `plot QFP* PQFP* P*`. Normally, *FpcLib* will plot a footprint only once, on the first match, and skip any additional matches to avoid duplicate outputs. This action can be overridden, and duplicates plotted, by entering the /D switch: `plot /D QFP* PQFP* P*`.

The plot is built by adding footprints, from left to right, until the row width limit is reached or all footprints are plotted. When the row limit is reached, a new row is started below the previous and the process continues until all of the selected footprints are added. The default row width limit is 8.0 inches but this value can be changed with the /W option switch: `plot /W10.5 QFP*`. Note that, if a footprint is wider than the row limit, it will still be plotted in an oversize row by itself.

The generated files are named by appending a short descriptor to the library base name. Assume, for example, *led.fpl* is currently loaded and contains only smt parts. The plot files created will be named **led_Silk.gbr** and **led_Top.gbr**.

As a minimum, for smd parts, the two files `<libname>_Silk.gbr` and `<libname>_Top.gbr` are created to handle footprint pads, pin numbers, text and graphics. Additional files are created, as needed, to handle inner and bottom pads, holes and bottom layer smt pad numbers. The bottom layer pin number text, in `<libname>_BSilk.gbr`, is mirrored to allow normal reading in Gerber viewer "Bottom View" mode.

Generated files and contents:

<libname>_Silk.gbr	Footprint ref text, text, polylines, pin numbers, plot outline and titleblock (ldrawn). Centroid icons and glue spots (flashed).
<libname>_Holes.gbr	Footprint pin drill holes (flashed).
<libname>_Top.gbr	Top layer pads (flashed), footprint bounding box and footprint name text (drawn).
<libname>_Inner.gbr	Inner layer pads (flashed).
<libname>_Bottom.gbr	Bottom layer pads (flashed).
<libname>_BSilk.gbr	Bottom layer smd pad pin numbers (drawn).

For proper display, the files should be loaded and displayed in a Gerber viewer in the above order with flashed and drawn item assigned different colors.

Ex: (with th_dip.fpl loaded)

Cmd> plot /w8.5

← plot everything 8.5 inches wide using the default "*" mask

Width = 8.500 In

← width changed from default

th_dip_Top.gbr created
th_dip_Inner.gbr created
th_dip_Bottom.gbr created
th_dip_Silk.gbr created
th_dip_Holes.gbr created

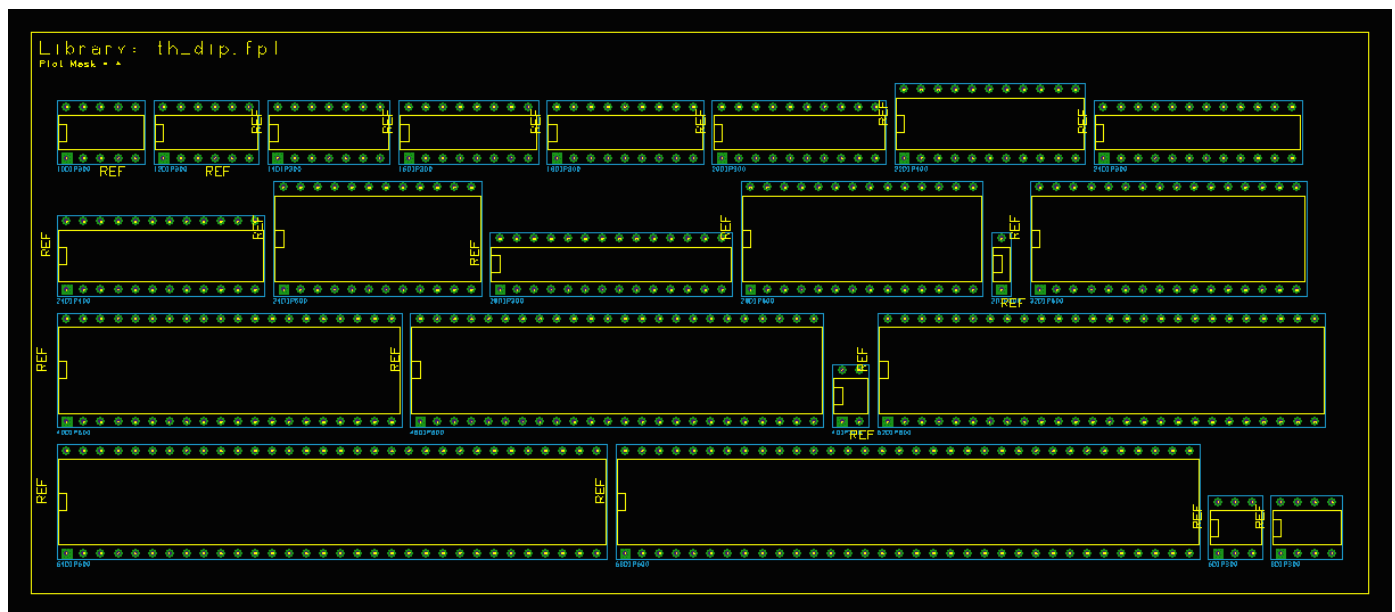
← files created for the plot

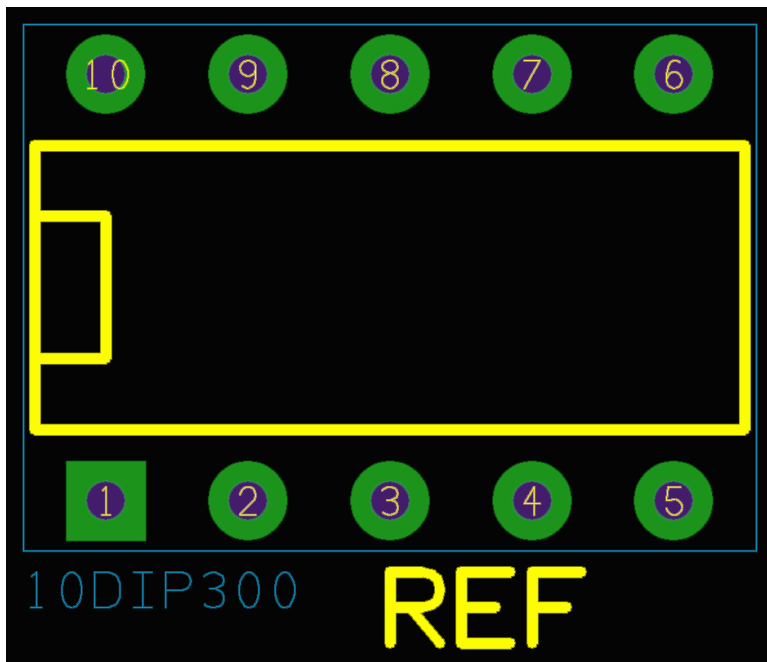
10DIP300 plotted
12DIP300 plotted
14DIP300 plotted

← footprints listed as they are added to the plot

.
.
.

6DIP300 plotted
8DIP300 plotted





A detail view of the first footprint in Fig 1.

Fig 2. Detail view

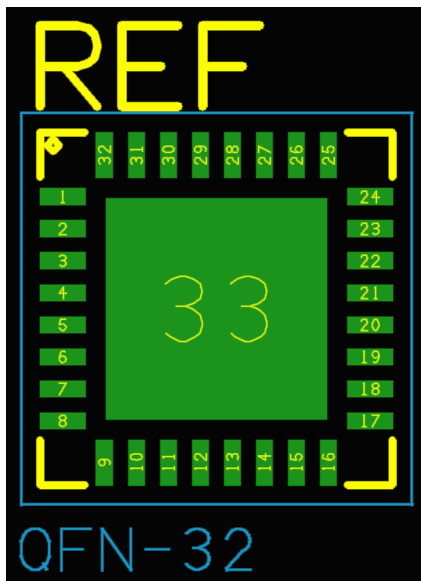


Fig 3. QFN-32 in GC-Prevue

The same smd footprint shown in two popular Gerber viewers **GC-Prevue** and **ViewMate**.

Note that the bounding box and name text are on the Top layer along with the top pads. **GC-Prevue** allows pads and traces to use different colors; **ViewMate** does not.

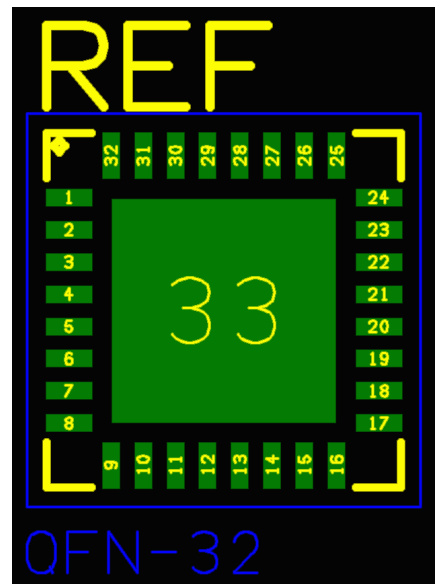


Fig 4. QFN-32 in ViewMate



Fig 5. Two sided Edge Conn Top view

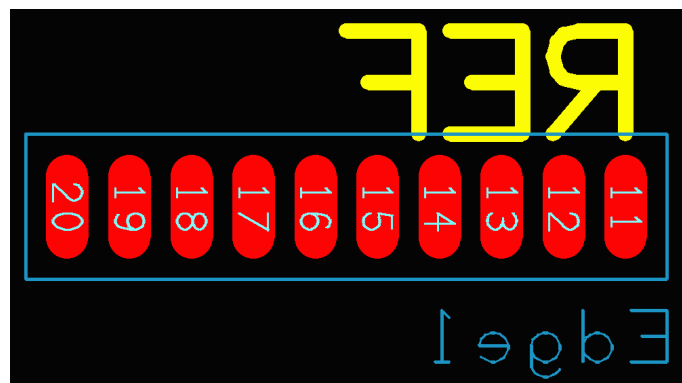


Fig 6. Two sided Edge Conn Bottom view

Bottom side SMT pads use the optional BSilk file for pin numbers.

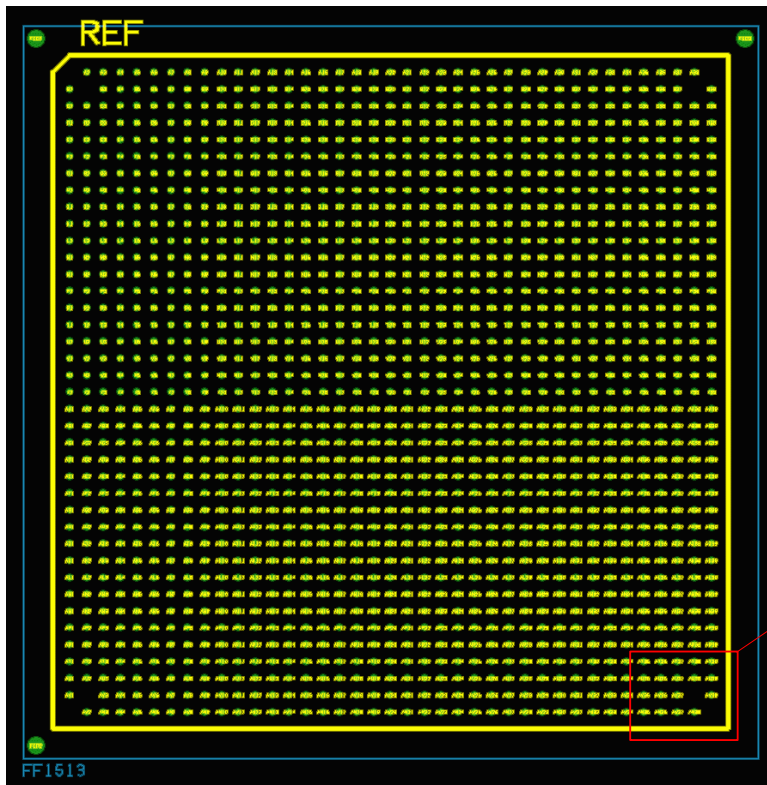


Fig 7. BGA example

BGA's can lead to large Silk files due to pin number data.



Fig 8. BGA detail

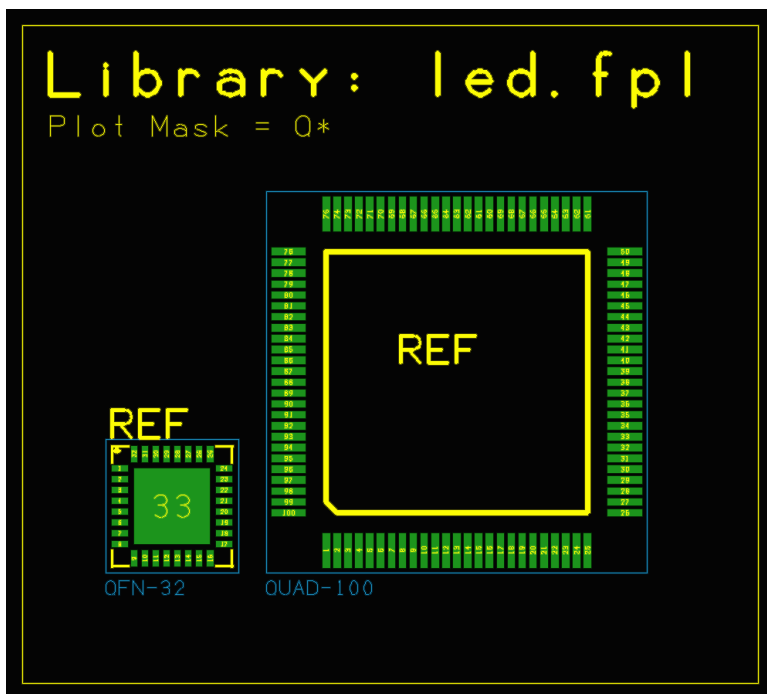


Fig 9. Selected part plot

As shown in Fig's 1 and 9, a complete library plot consists of the footprints arranged in rows with a box around the entire plot and header text that includes the library name and the footprint selection mask parameters used.

Centroids and, if available, the footprint angle icons are now included in the top silkscreen plots as are both default and user defined glue spots. The system default glue spots use a fixed 25 mil diameter icon as shown whereas the user defined spots are sized as defined.



Fig 10. Centroid

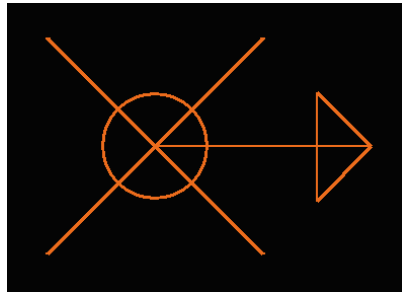


Fig 10. Centroid with Footprint Angle



Fig 11. Default Glue Spot

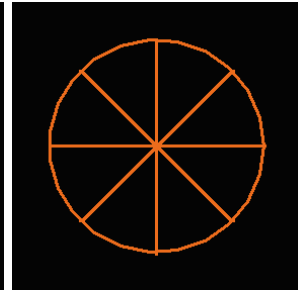


Fig 12. User Defined Glue Spot

Quit - Quit FpcLib

This command terminates *FpcLib*. If the currently loaded library has been modified and not saved, the user is asked to verify the command.

Ex:

```
Cmd> q
(Y/N) Library changed, really quit? y
```

Ren - Rename a footprint

This command changes the name of a footprint. It requires two parameters, without wildcards, to specify the old and new name. To handle corrupted libraries with duplicated names, an optional third parameter can be used to select which copy to rename. Valid copy select values must be greater than zero and, if not entered, defaults to one.

Param1 Old footprint name

Param2 New name

Param3 Optional copy select

Ex:

```
Cmd> ren LED-T1 Led_1p5          ← Rename LED-T1 to Led_1p5
1st "LED-T1" renamed to "led-1p5"

Cmd> r DIP-08 DIP-08B 2          ← Rename the second occurrence of DIP-08 to DIP-08B
2nd "DIP-08" renamed to "DIP-08B"

Cmd> ren LED-3PIN LED-T1         ← Attempt to change to an existing name
LED-T1 already exists

Cmd> r fred bob 2                ← Second fred not found
"fred" (2) not found
```


Rotate – Rotate Footprints

This command creates a new copy of an existing footprint and rotates it about its origin by the specified angle. The new footprint inherits the source's name with "_ROT_xxx" appended where xxx is the specified rotation angle. For example, if **DIP-16** is rotated 42.7°, the new footprint is named **DIP-16_ROT_42.7**. The rotation angle is specified in decimal degrees and can be entered as any value but the actual value used is normalized to a value between 0 and 360 and that normalized value is used when forming the new name. For example, rotating **BGA-1156** by -45° results in the new footprint being named **BGA-1156_ROT_315**.

Ex:

```
D:\Code\FpcLib\test>fpclib rotate

FpcLib - a Librarian for FreePCB
Ver 1.30R (c)2009 Bruce Parham

File: rotate.fpl, 4 entries

1 "BGA-144"
2 "DIP-16S"
3 "DIP_16"
4 "TQFP_44"

Cmd> rotate * 30                                ← Rotate all footprints +30°

BGA-144_ROT_30 created
DIP-16S_ROT_30 created
DIP_16_ROT_30 created
TQFP_44_ROT_30 created

4 footprints created or updated

Cmd> ro BGA-144 60                               ← Rotate a single footprint +60°

BGA-144_ROT_60 created

1 footprint created or updated

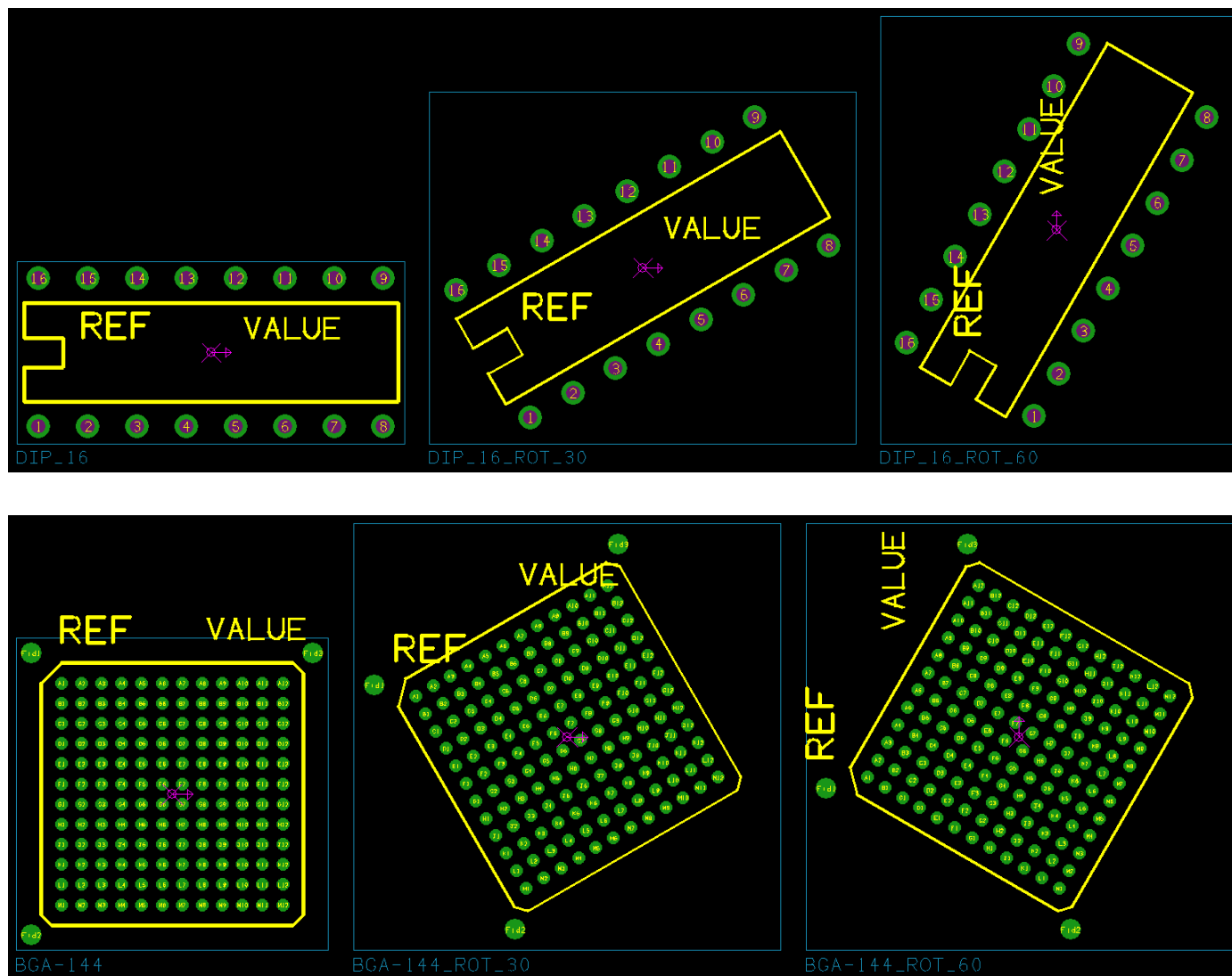
Cmd> sa                                           ← Save the modified library

save complete

Cmd> q                                           ← Done

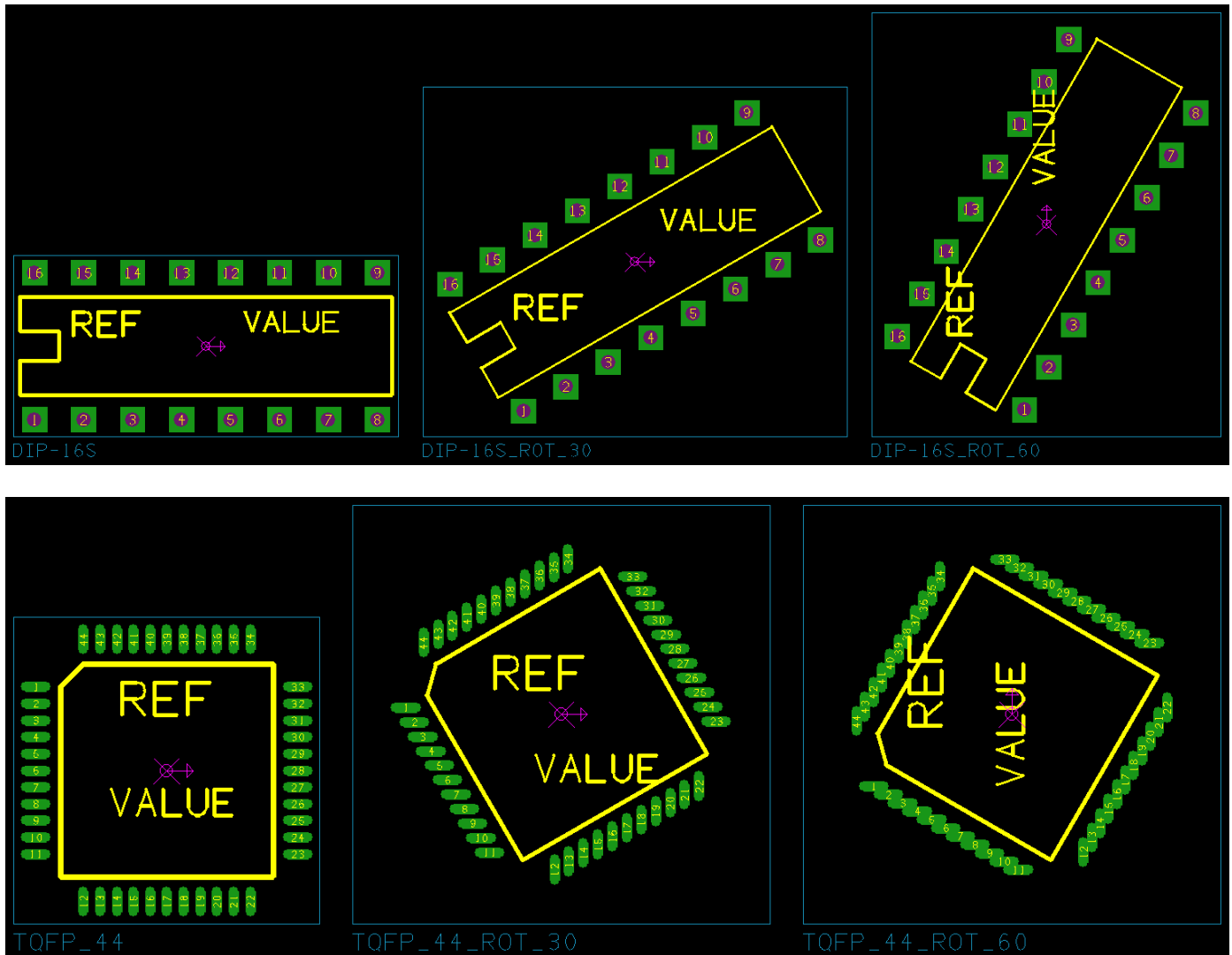
D:\Code\FpcLib\test>
```

As shown, parts that use only round pads and straight line segments will be accurately rotated.



Note that the courtyard bounding box is it's self rotated and it's new limits are used to form the new bounding box. This may lead to a slightly enlarged courtyard but the standard sized courtyard can be easily restored by loading the rotated footprint into the footprint editor and immediately saving it.

Sadly, the rotation process is not perfect. Although all features are correctly relocated during rotation, some **FreePCB** footprint items can not in themselves currently be rotated by an arbitrary angle; pins and text can, at best, only be rotated in 90° increments. Also polygon edges, that have been converted to arcs, behave poorly. The approach currently used is that pads *are not rotated at all* and that text items and the centroid angle snap to the nearest 90° increment while polygon edges are reverted to straight segments. In general, a footprint can be correctly rotated by any angle only if it uses round pads or, for the special 45° case, octagonal pads. After rotation, the footprint editor may need to be used to reposition text blocks and touch up polygons.



Save - Library Save / Save As

This command writes the current library to disk. An optional parameter can be used to specify a new library file name. If no name is entered, the original file is renamed by appending “**.bak**” and the current library is saved to the original name; for example **led.fpl** would be renamed **led.fpl.bak**. If a new name is entered, the current library is saved to that name and the original file is unchanged. The new name may not contain any wildcard characters but, if an extension is not present, “**.fpl**” will be appended.

If a library file is set to Read-Only, renaming or replacement will fail and an error message will be displayed.

Ex (assume **led.fpl** is currently loaded):

```
Cmd> save                ← Attempt to save an unmodified library

led.fpl unchanged, nothing done.

Cmd> sa led               ← Attempt to Save As to current file. Nothing done
Names match!

Cmd> sa bob               ← Save as an existing file.
(Y/N) bob.fpl exists, replace? Y

Cmd> sa bob2              ← Save as an existing RO file.
bob2.fpl exists and is Read-Only, can't replace
```

If the Read-Only library **bob2.fpl** is loaded and modifies:

```
Cmd> save                ← Attempt to save a Read-Only library

bob2.fpl is Read-Only, save to a new name
```

Sort - Sort footprints by name

This command sorts the current library footprints by name in ascending alphabetic order. An optional parameter, the minus sign "-", can be used to reverse the sort order.

The sorting process flags duplicate names and names that match but differ in case only. After deleting or renaming entries to correct these problems, resorting the library is necessary to update the dup status.

Duplicate names are an error condition that should be corrected by deleting or renaming footprints.

Footprints whose names differ in case only are also flagged with a warning. Although different names, the similarity could be a potential source error and/or confusion.

Ex:

Cmd> **sort** ← Sort in normal order

File: bob.fpl, 6 entries

```
1 "LED-3PIN"
2 "LED-T1"
3 "LED-T1.75"
4 "P500"
5 "p503"
6 "P507"
```

Cmd> **so -** ← Sort in reverse order

File: bob.fpl, 6 entries

```
1 "P507"
2 "p503"
3 "P500"
4 "LED-T1.75"
5 "LED-T1"
6 "LED-3PIN"
```

Cmd> **so** ← Sort and locate problems

File: bob.fpl, 6 entries

```
1 "LED-3PIN"
2 "LED-T1.75"
3 "P500" ** Error: Duplicate name **
4 "P500" ** Error: Duplicate name **
5 "P507" ** Warning: Name match, dif by case only **
6 "p507" ** Warning: Name match, dif by case only **
```

Cmd> **r P500 P500B 2** ← Rename second copy of P500

2nd "P500" renamed to "P500B"

Cmd> **so** ← Resort to update status

File: bob.fpl, 6 entries

```
1 "LED-3PIN"
2 "LED-T1.75"
3 "P500"
4 "P500B"
5 "P507" ** Warning: Name match, dif by case only **
6 "p507" ** Warning: Name match, dif by case only **
```

Test – Test a Library for Errors

The Test command reads an existing library, without loading it, and displays any data errors or warnings found. One parameter, without any wildcard characters, is used to specify the desired library file.

Ex:

```
Cmd> test test_fpclib

192 footprints found in "test_fpclib.fpl"
** 3 duplicate names found **
** 3 Footprints have data error(s) **

Errors:
** (test_fpclib.fpl - line 6483) "feed_thru" sel_rect record missing 1 parameter **
** (test_fpclib.fpl - line 6502) "SMT Feed-Thru small" invalid bottom SMT pad shape: none **
** (test_fpclib.fpl - line 7069) "EIA-0603" degenerate Polyline record **
```

The test command can be used within a *ForEach* block to test an entire library folder for errors:

```
Cmd> dir "c:\program files\freepcb\lib\*" ← Get the file list of the folder to be tested
"c:\program files\freepcb\lib\*.fpl"

 1 led.fpl                               Mon May 31 17:00:36 2004      3674 A
 2 sm_capacitor.fpl                     Wed Jul 07 18:48:34 2004      8278 A
 3 sm_dip.fpl                           Wed Jun 30 15:47:00 2004     29320 A
.
.
.
18 th_resistor.fpl                      Wed Jun 30 16:39:00 2004     13071 A
19 th_sip.fpl                           Mon May 31 17:01:56 2004     16701 A
20 th_transistor.fpl                    Wed Jun 30 14:57:36 2004     32164 A

Cmd> f                                  ← Setup the ForEach block
Build> t $
Build> e
(Y/N) Run? y
For-> t "c:\program files\freepcb\lib\led.fpl"
 6 footprints found in "c:\program files\freepcb\lib\led.fpl"
For-> t "c:\program files\freepcb\lib\sm_capacitor.fpl"
18 footprints found in "c:\program files\freepcb\lib\sm_capacitor.fpl"
.
.
.
For-> t "c:\program files\freepcb\lib\th_sip.fpl"
10 footprints found in "c:\program files\freepcb\lib\th_sip.fpl"
For-> t "c:\program files\freepcb\lib\th_transistor.fpl"
28 footprints found in "c:\program files\freepcb\lib\th_transistor.fpl"
```

Command Scripts

FpcLib can optionally run a command script on startup. Script files are plain text with one command per line and may include blank lines and comments.

All commands may be used within a script but, while a script is executing, the **Yes/No** query mechanism is bypassed and execution continues as if a **Yes** were entered.

To start FpcLib with a script, use the -S option switch. The format is: `fpclib -Sscriptname libname` where *scriptname* is the script file and *libname* is, as before, the startup library.

Ex:

```
C:\Program Files\FreePCB\lib> fpclib -sTestScript.txt user_created
```

File TestScript.txt:

```
; a test script for FpcLib
dir                ; list and save all .fpl file names
; run a foreach block
f
o $                ; open each file in turn (lists contents)
e
del *              ; empty the current lib
; run another block to collect footprints
f
a $ DIP*
a $ Dip*
a $ dip*
e
l                  ; list the results
end                ; end script without terminating fpclib
quit              ; terminate fpclib
```

Temporary Files

During execution, **FpcLib** may create and use temporary files. The four files that may be created are *FpcLibDirList.txt*, file names from the last **Dir** command, *FpcLibCmdList.txt*, the command list from the last **ForEach** command, *FpcLibErrList.txt* the error list from the last load operation and *FpcLibTempLib.fpt* the content of the last Extract command. These files are deleted when **FpcLib** terminates.

Appendix A – Errors and Warnings

Fatal errors

Fatal errors fall into the two general categories of startup and dynamic memory allocation. These are caused by bad command line parameters or exhausting memory and will result in FpcLib terminating with a short error message and a usage reminder.

Ex:

```
Error 126: unable to open script file "test_script.txt"
```

```
usage:
```

```
FpcLib [switches] [libname[.fpl]]  
  libname defaults to "user_created" if not present  
valid switches:  
  -H or -?   Help (this list)  
  -S<name>   Run script <name>
```

Error Message	Cause
Invalid switch: "-X"	Bad (unsupported) command line switch.
unable to open script file "<filename>"	Bad script file name, file not found.
"<libname>" is empty	The startup library is empty.
Unable to open library "<libname>"	Bad startup library name, file not found.
allocating <memory type> (Where <memory type> is one of the following: footprint storage footprint text space footprint poly space footprint poly corner space FP pin storage)	System memory exhausted.

Library data error and warning messages

Type 1 error:

This error occurs when the footprint name can not be read because the name is missing or quotes are not balanced. The library file name and line number are displayed along with name record:

**** (<libname> line <nn>) bad name record: [name: xx] ****

Type 2 error or warning:

All other errors and warnings follow this second form where the library name, line number and footprint name are shown along with an descriptive message:

**** (<libname> line <nn>) "fpname": <msg> ****

Error Messages and causes:

unknown units: [units: xx]

Bad units record where xx is not MIL or NM

<nnn> record missing <r> parameters

A record type <nnn> found missing <r> numeric parameters

<u>Record Type</u>	<u>Parameters</u>
sel_rect	4
ref_text	5
text	7
polyline	3
next_corner	3
close_polyline	1
n_pins	1
pin	4
top_pad	4 or 5
inner_pad	4 or 5
bottom_pad	4 or 5

invalid <layer> SMT pad shape: none

Shape none is not valid for SMT pads.

invalid <layer> [SMT] pad shape: > 6

A pad record found with an invalid shape code.
Only shape codes 0-6 are defined.

invalid <layer> SMT pad size: 0.0

A SMT pad found with a zero dimension.

invalid <layer> [SMT] pad size: radius too large

A rounded rectangle pad was found with the radius greater than half the narrower dimension.

degenerate Polyline record

Polyline with no following next_corner record, a minimum of 1 is required.

premature EOF in footprint poly

Pass 1: hit EOF counting corners

premature EOF reading footprint poly

Pass 2: hit EOF while reading corners

premature EOF reading fp pins

Hit EOF before finding the required number of pins

premature EOF reading [SMT] pad record

Hit EOF before finding required pad records

footprint poly scan error

Pass 2: corners not all found

blank line while reading pins

Found blank line while looking for pin or pad record

bad pin record

Bad or missing pin record. Count should match n_pins.

bad pin name

Can't read pin name. Possible unbalanced quotes.

invalid SMT pad record: [rrr]

Found record rrr while looking for SMT pad.
Inner_pad records not valid for SMT.

invalid sel_rect values: min > max

A Min value found greater than Max. Possible text only footprint.

Warning messages and causes:

Warning: select rectangle very large (over 30 in)

May be hard or impossible to use in FreePCB.

Warning: Ref text size is over 10 in.

Very large text.

Warning: Ref text is over 30 in. from footprint origin

May be hard or impossible to work with in FreePCB.

Warning: Ref text stroke is over half the height

May be intentional, maybe not.

Warning: text size is over 10 in.

Very large text.

Warning: text is over 30 in. from footprint origin

May be hard or impossible to work with in FreePCB.

Warning: text stroke is over half the height

May be intentional, maybe not.

Warning: pin count over 2500 (nnn)

A very large part. Currently larger than the footprint wizard can create and may be too large for FreePCB.

Warning: pin name > 32 characters (nnn)

A very long pin name. May cause problems with some CAD tools.

Warning: pin hole diameter > 1.0 in. (n.nnn)

A very large drill size. Too large for many board houses.

Warning: pin located over 30 in. from origin

May be hard or impossible to work with in FreePCB.

Warning: pad dimension over 6.0 in.

A very large pad. May be intentional, maybe not.

Appendix B – Gerber Files and Viewers

The Gerber files produced by the *FpcLib* plot command are RS-274X compliant. When needed, parameterized compound macros are defined and used to flash some objects like rounded rectangle pads, the centroid icons and glue spots while all other pad shapes use standard aperture types.

```
G04 Library plot for pad_chk.fpl - Top Layer *
G04 Generated by FpcLib Ver 1.12R at 6:34:26pm on Sun Oct 28, 2007 *

%ASAXBY*FSLAX26Y26*MOIN*MIA0B0*OFA0B0*SFA1.0B1.0*%      ← Format and Mode along with others

G04 Rounded Rectangle Macro, params: W/2, H/2, R *          ← Compound macro definition
%AMRNDREC*
21,1,$1+$1,$2+$2-$3-$3,0,0,0*
21,1,$1+$1-$3-$3,$2+$2,0,0,0*
1,1,$3+$3,$1-$3,$2-$3*
1,1,$3+$3,$3-$1,$2-$3*
1,1,$3+$3,$1-$3,$3-$2*
1,1,$3+$3,$3-$1,$3-$2*%

%ADD10C,0.00100*%      ← Define apertures
%ADD11C,0.05000*%      ← Circular aperture
%ADD12R,0.05000X0.05000*%      ← Rectangular aperture for square & rectangle
%ADD17R,0.07500X0.05000*%
%ADD13R,0.05000X0.07500*%
%ADD18RNDREC,0.03750X0.02500X0.01000*%      ← Use the macro for rounded rectangles
%ADD14RNDREC,0.02500X0.03750X0.01000*%
%ADD19O,0.07500X0.05000*%      ← Oblong aperture for obround
%ADD15O,0.05000X0.07500*%
%ADD16P,0.05412X8X22.5*%      ← Regular 8 sided poly rotated 22.5° for octagon

G04 Pad_Chk *          ← Footprint name in comment
G54D10*

X0Y0D02*
X1025000D01*          ← Drawn box, text and graphics
Y1025000D01*
X0D01*                ← Data is modal
Y0D01*
X0Y-15000D02*
X11375D01*
X0Y-27500D02*
X11375D01*
.
.
.
G54D16*

X912500Y912500D03*    ← Flashed pads
M02*
```

Many Gerber viewer packages do not fully support the RS-274X specification and may not correctly display the files created by *FpcLib*. *GC-Prevue* and *ViewMate*, two popular free viewers, have been tested and both display the files correctly. If you are not using one (or both) of these, please consider upgrading.

GC-Prevue is available, free of charge, from GraphiCode at <http://www.graphicode.com/>
ViewMate is available, also for free, from PentaLogix LLC at <http://www.pentalogix.com/>

Revision History

- Ver 1.00 8 Sep 07** Initial release to FreePCB forum and Yahoo group.
- Ver 1.01 10 Sep 07** Changed **ForEach** file name quoting to allow prefixes and/or suffixes to be added. Updated User Guide.
- Ver 1.10 30 Sep 07** Added path information to directory listings to allow Open/Save operations to any folder. Added footprint plot to Gerber function. User Guide updated.
- Ver 1.11 7 Oct 07** Fixed a plot bug that caused the frame on some plots to be too tall. Changed the startup to attempt to load "user_created.fpl" if no library specified. Changed plot row positions so that rows are added downward. Added generalized switch parsing to the plot command. Changed the plot dup's switch from "+" to "/D". Added plot width switch **"/Wn.nn"** to override the default 8.0 inch plot width. Updated User Guide.
- Ver 1.12 28 Oct 07** Added data checking and error listing to library load operations. Added *Clean* and *Test* commands for additional error handling. Added dynamic pin number sizing to plots. Fixed a memory leak bug. Updated User Guide.
- Ver 1.20 27 Jul 08** Updated file support for FreePCB Ver 1.344 format files. Added **@** and **#** meta characters to **ForEach** command. Updated User Guide.
- Note that there is no graphical (Plot) support for the new features Solder Mask, Paste Mask, Centroid, Glue Spots or Footprint Angle. These *will* be added in a future version.
- Ver 1.30 6 Mar 09** Added plot support for footprint Centroid, Glue Spots and Footprint Angle. Solder Mask and Paste Mask still not supported. Added footprint rotation.