
PCB Tools Documentation

Release 0.1

Hamilton Kibbe

December 01, 2017

1	About PCB Tools	3
1.1	Image Rendering	3
1.2	Future Plans	4
2	Feature Support	5
3	PCB Tools Reference	7
3.1	rs274x — RS-274X file handling	7
3.2	excellon — Excellon file handling	9
3.3	operations — Cam File operations	11
3.4	render — Gerber file Rendering	11
4	Indices and tables	13
	Python Module Index	15

Contents:

About PCB Tools

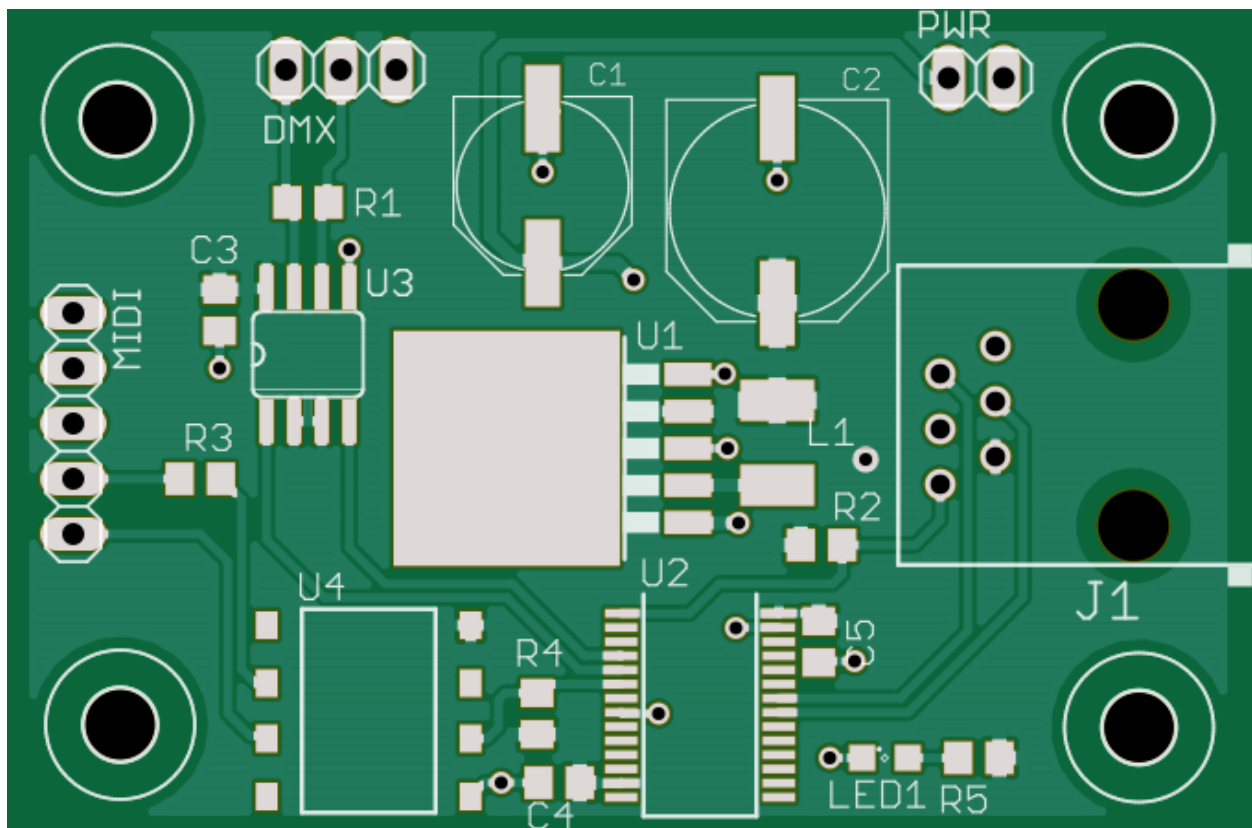
PCB Tools provides a set of utilities for visualizing and working with PCB design files in a variety of formats. The design files are generally referred to as Gerber files. This is a generic term that may refer to *RS-274X* (Gerber), *ODB++*, or *Excellon* files. These file formats are used by the CNC equipment used to manufacture PCBs.

PCB Tools currently supports the following file formats:

- Gerber (RS-274X)
- Excellon

with planned support for IPC-2581, ODB++ and more.

Image Rendering



The PCB Tools module provides tools to visualize PCBs and export images in a variety of formats, including SVG and PNG.

Future Plans

We are working on adding the following features to PCB Tools:

- Design Rules Checking
- Editing
- Panelization

Feature Support

Currently supported features are as follows:

File Format	Parsing	Rendering	Unit Conversion	Scale	Offset	Rotate
RS274-X	Yes	Yes	Yes	No	Yes	No
Excellon	Yes	Yes	Yes	No	Yes	No
ODB++	No	No	No	No	No	No

PCB Tools Reference

rs274x — RS-274X file handling

The RS-274X (Gerber) format is the most common format for exporting PCB artwork. The Specification is published by Ucamco and is available [here](#). The `rs274x` submodule implements classes to read and write RS-274X files without having to know the precise details of the format.

The `rs274x` submodule's `read()` function serves as a simple interface for parsing gerber files. The `GerberFile` class stores all the information contained in a gerber file allowing the file to be analyzed, modified, and updated. The `GerberParser` class is used in the background for parsing RS-274X files.

Functions

The `rs274x` module defines the following functions:

read (*filename*)

Read data from filename and return a `GerberFile`

Parameters filename : string

Filename of file to parse

Returns file : `gerber.rs274x.GerberFile`

A `GerberFile` created from the specified file.

Classes

The `rs274x` module defines the following classes:

class GerberFile (*statements, settings, primitives, apertures, filename=None*)

A class representing a single gerber file

The `GerberFile` class represents a single gerber file.

Parameters statements : list

list of gerber file statements

settings : dict

Dictionary of gerber file settings

filename : string

Filename of the source gerber file

Attributes

<p>comments: list of strings size</p>	<p>List of comments contained in the gerber file. (tuple, (<float>, <float>)) Size in [self.units] of the layer described by the gerber file.</p>
<p>bounds: tuple, ((<float>, <float>), (<float>, <float>))</p>	<p>boundaries of the layer described by the gerber file. <i>bounds</i> is stored as ((min x, max x), (min y, max y))</p>

Methods

```

offset
render
to_inch
to_metric
write
    
```

render — Gerber file Rendering

Render Module

write (*filename*, *settings=None*)
Write data out to a gerber file.

class GerberParser
GerberParser

Methods

```

dump_json
dump_str
evaluate
parse
parse_raw
    
```

evaluate (*stmt*)
Evaluate Gerber statement and update image accordingly.
This method is called once for each statement in the file as it is parsed.

Parameters statement : Statement

Gerber/Excellon statement to evaluate.

excellon — Excellon file handling

The Excellon format is the most common format for exporting PCB drill information. The Excellon format is used to program CNC drilling machines for drilling holes in PCBs. As such, excellon files are sometimes referred to as NC-drill files. The Excellon format reference is available [here](#). The `excellon` submodule implements classes to read and write excellon files without having to know the precise details of the format.

The `excellon` submodule's `read()` function serves as a simple interface for parsing excellon files. The `ExcellonFile` class stores all the information contained in an Excellon file allowing the file to be analyzed, modified, and updated. The `ExcellonParser` class is used in the background for parsing RS-274X files.

Functions

The `excellon` module defines the following functions:

read (*filename*)

Read data from filename and return an ExcellonFile Parameters

filename [string] Filename of file to parse

Returns file : `gerber.excellon.ExcellonFile`

An ExcellonFile created from the specified file.

Classes

The `excellon` module defines the following classes:

class ExcellonFile (*statements, tools, hits, settings, filename=None*)

A class representing a single excellon file

The ExcellonFile class represents a single excellon file.

<http://www.excellon.com/manuals/program.htm> (archived version at <https://web.archive.org/web/20150920001043/http://www.ex>)

Parameters tools : list

list of gerber file statements

hits : list of tuples

list of drill hits as (<Tool>, (x, y))

settings : dict

Dictionary of gerber file settings

filename : string

Filename of the source gerber file

Attributes

units	(string) either 'inch' or 'metric'.
-------	-------------------------------------

Methods

hit_count
offset
path_length
render
report
to_inch
to_metric
update_tool
write

render — Gerber file Rendering

Render Module

path_length (*tool_number=None*)

Return the path length for a given tool

primitives

Gets the primitives. Note that unlike Gerber, this generates new objects

report (*filename=None*)

Print or save drill report

to_inch ()

Convert units to inches

to_metric ()

Convert units to metric

update_tool (*tool_number, **kwargs*)

Change parameters of a tool

class ExcellonParser (*settings=None, ext_tools=None*)

Excellon File Parser

Parameters settings : FileSettings or dict-like

Excellon file settings to use when interpreting the excellon file.

Attributes

bounds
coordinates
hole_count
hole_sizes

Methods

parse
parse_raw

operations — Cam File operations

The `operations` module provides functions which modify `gerber.cam.CamFile` objects. All of the functions in this module return a modified copy of the supplied file.

Functions

The `operations` module defines the following functions:

to_inch (*cam_file*)

Convert Gerber or Excellon file units to imperial

Parameters **cam_file**: `gerber.cam.CamFile` subclass

Gerber or Excellon file to convert

Returns **cam_file**: `gerber.cam.CamFile` subclass

A deep copy of the source file with units converted to imperial.

to_metric (*cam_file*)

Convert Gerber or Excellon file units to metric

Parameters **cam_file**: `gerber.cam.CamFile` subclass

Gerber or Excellon file to convert

Returns **cam_file**: `gerber.cam.CamFile` subclass

A deep copy of the source file with units converted to metric.

offset (*cam_file*, *x_offset*, *y_offset*)

Offset a Cam file by a specified amount in the X and Y directions.

Parameters **cam_file**: `gerber.cam.CamFile` subclass

Gerber or Excellon file to offset

x_offset: float

Amount to offset the file in the X direction

y_offset: float

Amount to offset the file in the Y direction

Returns **cam_file**: `gerber.cam.CamFile` subclass

An offset deep copy of the source file.

render — Gerber file Rendering

Render Module

Indices and tables

- *genindex*
- *modindex*
- *search*

e

excellon, 9

o

operations, 11

r

render, 11

rs274x, 7

E

evaluate() (GerberParser method), 8
excellon (module), 9
ExcellonFile (class in gerber.excellon), 9
ExcellonParser (class in gerber.excellon), 10

G

GerberFile (class in gerber.rs274x), 7
GerberParser (class in gerber.rs274x), 8

O

offset() (in module gerber.operations), 11
operations (module), 11

P

path_length() (ExcellonFile method), 10
primitives (ExcellonFile attribute), 10

R

read() (in module gerber.excellon), 9
read() (in module gerber.rs274x), 7
render (module), 8, 10, 11
report() (ExcellonFile method), 10
rs274x (module), 7

T

to_inch() (ExcellonFile method), 10
to_inch() (in module gerber.operations), 11
to_metric() (ExcellonFile method), 10
to_metric() (in module gerber.operations), 11

U

update_tool() (ExcellonFile method), 10

W

write() (GerberFile method), 8