KiCad Documentation

Release 0.1

Thomas Pointhuber

Sep 09, 2018

User Documentation

This is the initial proposal for a high level KiCad. Python API.

Main goals are:

- easy to understand
- everything is documented and tested
- stable

Please note this in initial development. Everything can change, and this is at the moment an inofficial realisation of this idea (thus unstable 3rd-party software).

CHAPTER 1

kicad.pcbnew

1.1 Board

class kicad.pcbnew.**Board**(*board=None*)

Create a new Board object

Parameters **board** (pcbnew.BOARD) – already existing board object

Example

```
>>> from kicad.pcbnew import Board
>>> b = Board()
```

aux_origin

Aux origin of Board

Returns *kicad.util.Point2D*

Example

```
>>> from kicad.pcbnew import Board
>>> b = Board()
>>> b.aux_origin = [1, 2]
>>> b.aux_origin
kicad.util.point.Point2D(1.0, 2.0)
```

filepath

Filepath of the Board

Returns unicode

Example

```
>>> from kicad.pcbnew import Board
>>> b = Board()
>>> b.filepath = "path/to/board.kicad_mod"
```

(continues on next page)

(continued from previous page)

```
>>> print(b.filepath)
path/to/board.kicad_mod
```

static from_editor()
Get the current board visible in pcbnew

Returns *kicad.pcbnew.Board*

Example

```
>>> from kicad.pcbnew import Board
>>> b = Board.from_editor()
```

static from_file(*path*)
Load a board from a given filepath

Parameters **path** (str, unicode) – path to the ".kicad_mod" file

Returns *kicad.pcbnew.Board*

Example

```
>>> from kicad.pcbnew import Board
>>> b = Board.from_file("path/to/board.kicad_mod")
```

get_native()
Get native object from the low level API

Returns pcbnew.BOARD

grid_origin
Grid origin of Board

Returns *kicad.util.Point2D*

Example

```
>>> from kicad.pcbnew import Board
>>> b = Board()
>>> b.grid_origin = [1, 2]
>>> b.grid_origin
kicad.util.point.Point2D(1.0, 2.0)
```

is_highlighted
is highlighted?

Returns bool

is_locked
is locked?

Returns bool

is_selected
is selected?

Returns bool

layer
primary layer of the item

Returns kicad.pcbnew.Layer

layers
All layers where the item is present on

Returns kicad.pcbnew.LayerSet

modules
List of Modules present in the Board

Returns Iterator over *kicad.pcbnew.Module*

tracks
List of Tracks present in the Board

Returns Iterator over *kicad.pcbnew.Track*

vias
List of Vias present in the Board

Returns Iterator over *kicad.pcbnew.Via*

zones
List of Zones present in the Board

Returns Iterator over *kicad.pcbnew.Zone*

1.2 Dimension

class kicad.pcbnew.**Dimension**(*dimension*)

get_native()
Get native object from the low level API

Returns pcbnew.EDA_TEXT

is_highlighted
is highlighted?

Returns bool

is_locked
is locked?

Returns bool

is_selected
is selected?

Returns bool

layer
layer of the drawsegment

Returns *kicad.pcbnew.Layer*

layers
All layers where the item is present on

Returns kicad.pcbnew.LayerSet

text
Text

Returns unicode

value
value in mm

Returns float

1.3 Drawsegment

class kicad.pcbnew.**Drawsegment**(*drawsegment*)

get_native()
Get native object from the low level API

Returns pcbnew.DRAWSEGMENT

is_highlighted
is highlighted?

Returns bool

is_locked
is locked?

Returns bool

is_selected
is selected?

Returns bool

layer
layer of the drawsegment

Returns *kicad.pcbnew.Layer*

layers
All layers where the item is present on

Returns kicad.pcbnew.LayerSet

width
Width of line in mm

Returns float

1.4 Arc

class kicad.pcbnew.**Arc**(*arc*)

angle
angle of arc in degree

Returns float

center
Center point of arc

Returns *kicad.util.Point2D*

get_native()
Get native object from the low level API

Returns pcbnew.DRAWSEGMENT

is_highlighted
is highlighted?

Returns bool

is_locked
is locked?

Returns bool

is_selected
is selected?

Returns bool

layer
layer of the drawsegment

Returns *kicad.pcbnew.Layer*

layers
All layers where the item is present on

Returns kicad.pcbnew.LayerSet

start
Start point of arc

Returns *kicad.util.Point2D*

width
Width of line in mm

Returns float

1.5 Circle

class kicad.pcbnew.**Circle**(*circle*)

center
Center point of circle

Returns *kicad.util.Point2D*

diameter
Diameter of circle

Returns float

get_native()
Get native object from the low level API

Returns pcbnew.DRAWSEGMENT

is_highlighted
is highlighted?

Returns bool

is_locked
is locked?

Returns bool

is_selected
is selected?

Returns bool

layer
layer of the drawsegment

Returns *kicad.pcbnew.Layer*

layers
All layers where the item is present on

Returns kicad.pcbnew.LayerSet

radius
Radius of circle

Returns float

width
Width of line in mm

Returns float

1.6 Line

class kicad.pcbnew.**Line**(*line*)

end
End point of line

Returns *kicad.util.Point2D*

get_native()
Get native object from the low level API

Returns pcbnew.DRAWSEGMENT

is_highlighted
is highlighted?

Returns bool

is_locked
is locked?

Returns bool

is_selected
is selected?

Returns bool

layer
layer of the drawsegment

Returns *kicad.pcbnew.Layer*

layers
All layers where the item is present on

Returns kicad.pcbnew.LayerSet

start
Start point of line

Returns *kicad.util.Point2D*

width
Width of line in mm

Returns float

1.7 Polygon

class kicad.pcbnew.**Polygon**(*polygon*)

get_native()
Get native object from the low level API

Returns pcbnew.DRAWSEGMENT

is_highlighted
is highlighted?

Returns bool

is_locked
is locked?

Returns bool

is_selected
is selected?

Returns bool

layer
layer of the drawsegment

Returns *kicad.pcbnew.Layer*

layers
All layers where the item is present on

Returns kicad.pcbnew.LayerSet

width
Width of line in mm

Returns float

1.8 Layer

class kicad.pcbnew.**Layer**(*id*)

static from_id(*id*)
Get Layer object from id

Parameters **id** (int) –

Returns *kicad.pcbnew.Layer*

static from_name(*name*)
Get Layer object from name

Parameters **name** –

Returns *kicad.pcbnew.Layer*

id
internal ID of the layer

Returns int

name
name of the layer

Returns unicode

class kicad.pcbnew.**LayerSet**(*layer_set*)

get_native()
Get native object from the low level API :return: pcbnew.LSET

1.9 Module

class kicad.pcbnew.**Module**(*module*)

description
Description of the Module

Returns unicode

static from_editor()
Get the current module

get_native()
Get native object from the low level API

Returns pcbnew.MODULE

is_highlighted
is highlighted?

Returns bool

is_locked
is locked?

Returns bool

is_selected
is selected?

Returns bool

keywords
Keywords of the Module

Returns unicode

layer
primary layer of the item

Returns kicad.pcbnew.Layer

layers
All layers where the item is present on

Returns kicad.pcbnew.LayerSet

pads
List of Pads present in the Module

Returns Iterator over *kicad.pcbnew.Pad*

position
Position of the Module

Returns *kicad.util.Point2D*

reference
Reference of the Module

Returns unicode

value
Value of the Module

Returns unicode

1.10 Net

class kicad.pcbnew.**Net**(*netinfo*)

get_native()
Get native object from the low level API

Returns pcbnew.NETINFO_ITEM

name
Name of Net

Returns unicode

1.11 Pad

class kicad.pcbnew.**Pad**(*pad*)

get_native()
Get native object from the low level API

Returns pcbnew.PAD

is_highlighted
is highlighted?

Returns bool

is_locked
is locked?

Returns bool

is_selected
is selected?

Returns bool

layer
primary layer of the item

Returns kicad.pcbnew.Layer

layers
All layers where the item is present on

Returns kicad.pcbnew.LayerSet

name
Name of the Pad

Returns unicode

net
Net of the Zone

Returns *kicad.pcbnew.Net*

1.12 PcbTarget

class kicad.pcbnew.**PcbTarget**(*target*)

get_native()
Get native object from the low level API

Returns pcbnew.EDA_TEXT

is_highlighted
is highlighted?

Returns bool

is_locked
is locked?

Returns bool

is_selected
is selected?

Returns bool

layer
primary layer of the item

Returns kicad.pcbnew.Layer

layers
All layers where the item is present on

Returns kicad.pcbnew.LayerSet

position
position of the PcbTarget

Returns *kicad.util.Point2D*

width
Width of line in mm

Returns float

1.13 Text

class kicad.pcbnew.**Text**(*text*)

get_native()
Get native object from the low level API

Returns pcbnew.EDA_TEXT

is_highlighted
is highlighted?

Returns bool

is_locked
is locked?

Returns bool

is_selected
is selected?

Returns bool

layer
layer of the drawsegment

Returns *kicad.pcbnew.Layer*

layers
All layers where the item is present on

Returns kicad.pcbnew.LayerSet

position
Position of the Text

Returns *kicad.util.Point2D*

text
Text

Returns unicode

text_size
Text Size

Returns *kicad.util.Point2D*

thickness
Thickness

Returns float

1.14 Track

class kicad.pcbnew.**Track**(*track*)

end
End of the Track

Returns *kicad.util.Point2D*

get_native()
Get native object from the low level API

Returns pcbnew.TRACK

is_highlighted
is highlighted?

Returns bool

is_locked
is locked?

Returns bool

is_selected
is selected?

Returns bool

layer
primary layer of the item

Returns kicad.pcbnew.Layer

layers
All layers where the item is present on

Returns kicad.pcbnew.LayerSet

net
Net of the Track

Returns *kicad.pcbnew.Net*

start
Start of the Track

Returns *kicad.util.Point2D*

width
Width of Track in mm

Returns float

1.15 Via

class kicad.pcbnew.**Via**(*via*)

drill
Drill size of Via in mm

Returns float

get_native()
Get native object from the low level API

Returns pcbnew.VIA

is_highlighted
is highlighted?

Returns bool

is_locked
is locked?

Returns bool

is_selected
is selected?

Returns bool

layer
primary layer of the item

Returns kicad.pcbnew.Layer

layers
All layers where the item is present on

Returns kicad.pcbnew.LayerSet

net
Net of the Via

Returns *kicad.pcbnew.Net*

position
Position of the Via

Returns *kicad.util.Point2D*

width
Width of Via in mm

Returns float

1.16 Zone

class kicad.pcbnew.**Zone**(*zone*)

get_native()
Get native object from the low level API

Returns `pcbnew.ZONE`

is_highlighted
is highlighted?

Returns `bool`

is_locked
is locked?

Returns `bool`

is_selected
is selected?

Returns `bool`

layer
primary layer of the item

Returns `kicad.pcbnew.Layer`

layers
All layers where the item is present on

Returns `kicad.pcbnew.LayerSet`

net
Net of the Zone

Returns *kicad.pcbnew.Net*

priority
Priority of the Zone

Returns `int`

CHAPTER 2

kicad.primitives

2.1 Polygon

class kicad.primitives.**Polygon**

class kicad.primitives.**PolygonSet**(*poly_set=None*)

difference(*other*)
Performs boolean PolygonSet difference

Parameters **other** (*kicad.primitives.PolygonSet*) – second operand of difference operation

fracture()
Converts a set of polygons with holes to a singe outline with slits/fractures connecting the outer ring to the inner holes

get_native()
Get native object from the low level API :return: pcbnew.SHAPE_POLY_SET

intersection(*other*)
Performs boolean PolygonSet intersection

Parameters **other** (*kicad.primitives.PolygonSet*) – second operand of intersection operation

unfracture()
Converts a single outline slitted ('fractured') polygon into a set of outlines with holes

union(*other*)
Performs boolean PolygonSet union

Parameters **other** (*kicad.primitives.PolygonSet*) – second operand of union operation

CHAPTER 3

kicad.util

3.1 Point2D

class kicad.util.**Point2D**(*coordinates=None*, *y=None*)

Representation of a 2D Point in space

Example

```
>>> from kicad.util.point import Point2D
>>> Point2D(0, 1)
kicad.util.point.Point2D(0.0, 1.0)
>>> Point2D([2, 3])
kicad.util.point.Point2D(2.0, 3.0)
>>> Point2D((4, 5))
kicad.util.point.Point2D(4.0, 5.0)
>>> Point2D({'x': 6, 'y':7})
kicad.util.point.Point2D(6.0, 7.0)
>>> Point2D(Point2D(8, 9))
kicad.util.point.Point2D(8.0, 9.0)
```

static from_wxPoint(*wxobj*)

Convert a wxPoint to a Point2D

Parameters **wxobj** (pcbnew.wxPoint) – point to convert

Returns *kicad.util.Point2D*

static from_wxSize(*wxobj*)

Convert a wxSize to a Point2D

Parameters **wxobj** (pcbnew.wxSize) – point to convert

Returns *kicad.util.Point2D*

round_to(*base*, *prec=10*)

Round to a specific base (like it's required for a grid)

Parameters

- **base** (float) – base we want to round to
- **prec** (int) – precision of rounding operation

Returns *kicad.util.Point2D*

Example

```
>>> from kicad.util.point import Point2D
>>> Point2D(0.1234, 0.5678).round_to(0.01)
kicad.util.point.Point2D(0.12, 0.57)
```

to_wxPoint()
Convert coordinate to internal coordinate

Returns pcbnew.wxPoint

to_wxSize()
Convert size given as Point2D to internal size

Returns pcbnew.wxSize

CHAPTER 4

kicad.plotter

class kicad.plotter.**Plotter**(*board*, *layer=None*, *color_mode=None*)

close()
Close a plotfile after writing

Example

```
>>> from kicad.pcbnew import Board, Layer
>>> from kicad.plotter import Plotter
>>> b = Board.from_editor()
>>> p = Plotter(b, layer=Layer.from_id(0))
>>> p.open('test', Plotter.PLOT_FORMAT_SVG)
kicad.plotter.Plotter(board="")
>>> p.plot_layer()
>>> p.close()
```

color_mode
is color mode enabled?

Returns bool

is_open
is plotfile open?

Returns bool

layer
layer to plot on

Returns *kicad.pcbnew.Layer*

open(*filename*, *format*, *sheet_description=None*)
Open a new plotfile for writing

Parameters

- **filename** (str) – Name of the file to plot

- **format** – format of the output file
- **sheet_description** – some description

Returns *kicad.plotter.Plotter*

Example

```
>>> from kicad.pcbnew import Board, Layer
>>> from kicad.plotter import Plotter
>>> b = Board.from_editor()
>>> p = Plotter(b, layer=Layer.from_id(0))
>>> with p.open('test', Plotter.PLOT_FORMAT_SVG):
...     p.plot_layer()
...
```

plot_layer()
plot layer to opened file

CHAPTER 5

List PCB Entities

This examples is a reimplementation of the listPcb.py script found in the official KiCad repository. It basically loads a board and then prints a short representation of all vias, tracks, drawings modules and zones.

```
#!/usr/bin/env python

from __future__ import print_function

import argparse

from kicad.pcbnew import Board
from kicad.pcbnew import Text


def list_pcb(board):
    print()
    print("LIST VIAS:")
    for via in board.vias:
        print(" * Via:   {} - {}/{}".format(via.position, via.drill, via.width))

    print()
    print("LIST TRACKS:")
    for track in board.tracks:
        print(" * Track: {} to {}, width {}".format(track.start, track.end, track.
↪width))

    print()
    print("LIST DRAWINGS:")
    for drawing in board.drawings:
        if type(drawing) is Text:
            print("* Text:    '{}' at {}".format(drawing.text, drawing.position))
        else:
            print("* Drawing: {}".format(drawing))

    print()
```

(continues on next page)

(continued from previous page)

```
    print("LIST MODULES:")
    for module in board.modules:
        print("* Module: {} at {}".format(module.reference, module.position))

    print()
    print("LIST ZONES:")
    for zone in board.zones:
        print("* Zone: '{}' with priority {}".format(zone.net.name, zone.priority))


# reimplementation of pcbnew/python/examples/listPcb.py script using our abstraction␣
↪layer
if __name__ == "__main__":
    parser = argparse.ArgumentParser()

    parser.add_argument('board', help='board file to list elements', action='store')

    args = parser.parse_args()

    board = Board.from_file(args.board)

    list_pcb(board)
```

This script can now simply be executed from the commandline, and outputs some nice informations about the board file:

```
$ python ./examples/list_pcb.py ./tests/pcbnew/testproject/testproject.kicad_pcb
```

CHAPTER 6

Indices and tables

- genindex
- modindex
- search

Python Module Index

k

Index

K

L

M

N

O

P

R

S

T

U

V

W

Z