
pykicadlib

Release 0.0.1

Oct 11, 2019

Contents

1	KiCAD symbols	3
1.1	Symbol types	3
1.2	Symbol elements	7
2	KiCAD footprints	19
3	Indices and tables	21
	Python Module Index	23
	Index	25

This package provides support to generate and manipulate schematic symbols and layout footprints for [KiCAD](#).

Note: Most of the documentation here is directly stolen from the [KiCAD](#) fileformat documetation.

This documentation is a work in progress. Feedback and revisions are most welcome!

KiCAD is an opensource EDA program.

Easiest way to install is to use [pip](#):

```
$ pip install canopen
```


1.1 Symbol types

1.1.1 Examples

Types used in symbol classes.

1.1.2 API

Generic

```
class pykicadlib.symbol.types.Representation  
    Symbol representation.
```

Specific

KiCAD symbol types.

```
class pykicadlib.symbol.types.Bold  
    Text element bold.  
  
    off = 0  
        Normal  
  
    on = 1  
        Bold  
  
class pykicadlib.symbol.types.Direction  
    Pin direction (flipped in opposition to KiCAD documentation).  
  
    down = 'U'  
        Down
```

```
    left = 'R'
        Left

    right = 'L'
        Right

    up = 'D'
        Up

class pykicadlib.symbol.types.Electric
    Electric pin type.

    bidirectional = 'B'
        Bidirectional

    input = 'I'
        Input

    not_connected = 'N'
        Not connected

    open_collector = 'C'
        Open collector

    open_emitter = 'E'
        Open emitter

    output = 'O'
        Output

    passive = 'P'
        Passive

    power_input = 'W'
        Power input

    power_output = 'w'
        Power output

    tristate = 'T'
        Tristate

    unspecified = 'U'
        Unspecified

class pykicadlib.symbol.types.Field
    Symbol field type.

    document = 3
        Document field

    footprint = 2
        Footprint field

    manufacturer = 4
        Manufacturer field

    model = 8
        Model field

    name = 1
        Name field
```



```
power = 10
    Power field

reference = 0
    Reference field

temperature = 7
    Temperature range field

tolerance = 6
    Tolerance field

value = 5
    Value field

voltage = 9
    Voltage field

class pykicadlib.symbol.types.Fill
    Element fill.

    background = 'f'
        Background

    foreground = 'F'
        Foreground

    none = 'N'
        None

class pykicadlib.symbol.types.Flag
    Flag normal or power symbol.

    normal = 'N'
        Normal symbol

    power = 'P'
        Power symbol

class pykicadlib.symbol.types.HJustify
    Field horizontal justify.

    center = 'C'
        Center

    left = 'L'
        Left

    right = 'R'
        Right

class pykicadlib.symbol.types.Italic
    Text element italic.

    off = 'Normal'
        Normal

    on = 'Italic'
        Italic

class pykicadlib.symbol.types.Orientation
    Field orientation.
```

horizontal = 'H'
Horizontal orientation

vertical = 'V'
Vertical orientation

class pykicadlib.symbol.types.**Representation**
Symbol representation.

both = 0
Both

morgan = 2
Morgan

normal = 1
Normal

class pykicadlib.symbol.types.**Shape**
Pin shape.

Note: Add 'N' before characters, to create an invisible pin.

clock = 'C'
Clock

clock_low = 'CL'
Clock low

falling_edge_clock = 'F'
Falling-edge clock

input_low = 'L'
Input low

inverted = 'I'
Inverted

inverted_clock = 'CI'
Inverted clock

invisible = 'N'
Invisible

line = ''
Line

non_logic = 'X'
Non logic

output_low = 'V'
Output low

class pykicadlib.symbol.types.**Style**
Field style.

bold = 'NB'
Bold

italic = 'IN'
Italic

italic_bold = 'IB'
Italic and Bold

```

    none = 'NN'
        None

class pykicadlib.symbol.types.Units
    Symbol units swappable or locked.

    locked = 'L'
        Locked

    swappable = 'F'
        Swappable

class pykicadlib.symbol.types.Visibility
    Field visibility.

    invisible = 'I'
        Invisible

    visible = 'V'
        Visible

class pykicadlib.symbol.types.Visible
    Symbol pin name/number visible.

    no = 'N'
        Not visible

    yes = 'Y'
        Visible

class pykicadlib.symbol.types.VJustify
    Field vertical justify.

    bottom = 'B'
        Bottom

    center = 'C'
        Center

    top = 'T'
        Top

```

1.2 Symbol elements

Symbols consist of multiple elements. These are well known elements like lines, circles and arcs. Additionally elements are added, the help construct symbols out of element primitives.

1.2.1 Examples

Here is a simple example:

```

>>> element = pykicadlib.symbol.elements.Rectangle(10, 10, 20, 20, 5, pykicadlib.
↳symbol.types.Fill.none)
>>> print(element)
S 10 10 20 20 0 1 5 N

```

1.2.2 API

Alias

Field

class pykicadlib.symbol.elements.**Field**(*type_*, *x*, *y*, *value*)
Component field.

Parameters

- **type** (*types.Field*) – Type of *Field*
- **value** (*str*) – Value of *Field* text
- **x** (*int*) – X coordinate
- **y** (*int*) – Y coordinate
- **size** (*int*) – Text size
- **orientation** (*Orientation*) – Text orientation
- **visibility** (*Visibility*) – Text visibility
- **hjustify** (*HJustify*) – Horizontal text justify
- **vjustify** (*VJustify*) – Vertical text justify
- **style** (*Style*) – Text style

__str__()
Return *Field* in KiCAD format.

type = None
Type of *Field*

x = None
X coordinate

y = None
Y coordinate

value = None
Value of *Field* text

size = None
Text size

orientation = None
Text orientation

visibility = None
Text visibility

hjustify = None
Horizontal text justify

vjustify = None
Vertical text justify

style = None
Text style

```

class HJustify
    Field horizontal justify.

class Orientation
    Field orientation.

class Style
    Field style.

Type
    alias of pykicadlib.symbol.types.Field

class VJustify
    Field vertical justify.

class Visibility
    Field visibility.

```

Graphics

```

class pykicadlib.symbol.elements.Arc(x, y, radius, start_x, start_y, end_x, end_y, start_angle,
                                     end_angle)

```

Arc with center at x/y and radius.

Parameters

- **x**(*int*) – X coordinate
- **y**(*int*) – Y coordinate
- **start_x**(*int*) – Start X coordinate
- **start_y**(*int*) – Start Y coordinate
- **end_x**(*int*) – End X coordinate
- **end_y**(*int*) – End Y coordinate
- **start_angle**(*int*) – Start angle (?.?)
- **end_angle**(*int*) – End angle (?.?)
- **radius**(*int*) – Arc radius
- **thickness**(*int*) – Thickness of outline
- **fill**(*Fill*) – Fill type
- **unit**(*int*) – Unit index
- **representation**(*Representation*) – Representation type

```

__eq__(other)
    Compare Arc instances.

```

```

__str__()
    Return Arc in KiCAD format.

```

```

x = None
    X coordinate

```

```

y = None
    Y coordinate

```

radius = None
Arc radius

start_x = None
Start X coordinate

start_y = None
Start Y coordinate

end_x = None
End X coordinate

end_y = None
End Y coordinate

start_angle = None
Start angle

end_angle = None
End angle

thickness = None
Thickness of outline

fill = None
Fill type

bounds
Element boundary.

Type *Boundary*

class Boundary (*x1, y1, x2, y2*)
Element/symbol boundary class.

Parameters

- **x1** (*int*) – X1 coordinate
- **y1** (*int*) – Y1 coordinate
- **x2** (*int*) – X2 coordinate
- **y2** (*int*) – Y2 coordinate

class Fill
Element fill.

class Representation
Symbol representation.

priority
Element priority.

Type *int*

class pykicadlib.symbol.elements.**Circle** (*x, y, radius*)
Circle with center at x/y and radius.

Parameters

- **x** (*int*) – X coordinate
- **y** (*int*) – Y coordinate
- **radius** (*int*) – Circle radius

- **thickness** (*int*) – Thickness of outline
- **fill** (*Fill*) – Fill type
- **unit** (*int*) – Unit index
- **representation** (*Representation*) – Representation type

__eq__ (*other*)
Compare *Circle* instances.

__str__ ()
Return *Circle* in KiCAD format.

x = **None**
X coordinate

y = **None**
Y coordinate

radius = **None**
Circle radius

thickness = **None**
Thickness of outline

fill = **None**
Fill type

bounds
Element boundary.

Type *Boundary*

class Boundary (*x1, y1, x2, y2*)
Element/symbol boundary class.

Parameters

- **x1** (*int*) – X1 coordinate
- **y1** (*int*) – Y1 coordinate
- **x2** (*int*) – X2 coordinate
- **y2** (*int*) – Y2 coordinate

class Fill
Element fill.

class Representation
Symbol representation.

priority
Element priority.

Type *int*

class pykicadlib.symbol.elements.Pin (*x, y, name, number*)
Pin at x/y with name/number.

Parameters

- **x** (*int*) – X coordinate
- **y** (*int*) – Y coordinate

- **name** (*str*) – Pin name
- **number** (*str*) – Pin number
- **length** (*int*) – Pin length
- **direction** (*Direction*) – Pin direction
- **name_size** (*int*) – Pin name size
- **number_size** (*int*) – Pin number size
- **electric** (*Electric*) – Electric type
- **shape** (*Shape*) – Shape type
- **visible** (*bool*) – Visibility
- **unit** (*int*) – Unit index
- **representation** (*Representation*) – Representation type

__eq__ (*other*)
Compare *Pin* instances.

__str__ ()
Return *Pin* in KiCAD format.

x = **None**
X coordinate

y = **None**
Y coordinate

name = **None**
Pin name

number = **None**
Pin number

length = **None**
Pin length

direction = **None**
Pin direction

name_size = **None**
Pin name size

number_size = **None**
Pin number size

electric = **None**
Electric type

shape = **None**
Shape type

visible = **None**
Visibility

priority
Element priority.
Type int

bounds

Element boundary.

Type *Boundary*

class Boundary (*x1*, *y1*, *x2*, *y2*)

Element/symbol boundary class.

Parameters

- **x1** (*int*) – X1 coordinate
- **y1** (*int*) – Y1 coordinate
- **x2** (*int*) – X2 coordinate
- **y2** (*int*) – Y2 coordinate

class Direction

Pin direction (flipped in opposition to KiCAD documentation).

class Electric

Electric pin type.

class Fill

Element fill.

class Representation

Symbol representation.

class Shape

Pin shape.

Note: Add 'N' before characters, to create an invisible pin.

class pykicadlib.symbol.elements.**Polygon**

Polygon.

Parameters

- **thickness** (*int*) – Thickness of outline
- **fill** (*Fill*) – Fill type
- **unit** (*int*) – Unit index
- **representation** (*Representation*) – Representation type

__eq__ (*other*)

Compare *Polygon* instances.

__str__ ()

Return *Polygon* in KiCAD format.

thickness = None

Thickness of outline

fill = None

Fill type

points = None

Outline points

priority

Element priority.

Type *int*

bounds

Element boundary.

Type *Boundary***add** (*point*)

Add point to polygon.

Parameters **point** (*Point*) – Point to add**remove** (*index*)

Remove element from polygon.

Parameters **index** (*int*) – Index of point to remove**class Boundary** (*x1, y1, x2, y2*)

Element/symbol boundary class.

Parameters

- **x1** (*int*) – X1 coordinate
- **y1** (*int*) – Y1 coordinate
- **x2** (*int*) – X2 coordinate
- **y2** (*int*) – Y2 coordinate

class Fill

Element fill.

class Point (*x, y*)

Point helper.

Parameters

- **x** (*int*) – X coordinate
- **y** (*int*) – Y coordinate

class Representation

Symbol representation.

class pykicadlib.symbol.elements.Rectangle (*x1, y1, x2, y2*)

Rectangle from x1/y1 to x2/y2.

Parameters

- **x1** (*int*) – X1 coordinate
- **y1** (*int*) – Y1 coordinate
- **x2** (*int*) – X2 coordinate
- **y2** (*int*) – Y2 coordinate
- **thickness** (*int*) – Thickness of outline
- **fill** (*Fill*) – Fill type
- **unit** (*int*) – Unit index
- **representation** (*Representation*) – Representation type

__eq__ (*other*)Compare *Rectangle* instances.

```

__str__()
    Return Rectangle in KiCAD format.

x1 = None
    X1 coordinate

x2 = None
    X2 coordinate

y1 = None
    Y1 coordinate

y2 = None
    Y2 coordinate

thickness = None
    Thickness of outline

class Boundary(x1, y1, x2, y2)
    Element/symbol boundary class.

    Parameters
        • x1 (int) – X1 coordinate
        • y1 (int) – Y1 coordinate
        • x2 (int) – X2 coordinate
        • y2 (int) – Y2 coordinate

class Fill
    Element fill.

class Representation
    Symbol representation.

fill = None
    Fill type

priority
    Element priority.

    Type int

bounds
    Element boundary.

    Type Boundary

class pykicadlib.symbol.elements.Text(x, y, value, size)
    Text at x/y with value, size, angle and multiple style options.

    New format since 2.4?

    Parameters
        • x (int) – X coordinate
        • y (int) – Y coordinate
        • value (str) – Text value
        • size (int) – Text size
        • angle (int) – Text angle

```

- **italic** (*Italic*) – Text italic style
- **bold** (*Bold*) – Text bold style
- **hjustify** (*HJustify*) – Horizontal text justify
- **vjustify** (*VJustify*) – Vertical text justify
- **unit** (*int*) – Unit index
- **representation** (*Representation*) – Representation type

__eq__ (*other*)

Compare *Text* instances.

__str__ ()

Return *Text* in KiCAD format.

x = **None**

X coordinate

y = **None**

Y coordinate

value = **None**

Text value

size = **None**

Text size

angle = **None**

Text angle

italic = **None**

Text italic style

bold = **None**

Text bold style

hjustify = **None**

Horizontal text justify

vjustify = **None**

Vertical text justify

bounds

Element boundary.

Type *Boundary*

class Bold

Text element bold.

class Boundary (*x1, y1, x2, y2*)

Element/symbol boundary class.

Parameters

- **x1** (*int*) – X1 coordinate
- **y1** (*int*) – Y1 coordinate
- **x2** (*int*) – X2 coordinate
- **y2** (*int*) – Y2 coordinate

```
class Fill  
    Element fill.  
  
class HJustify  
    Field horizontal justify.  
  
class Italic  
    Text element italic.  
  
class Representation  
    Symbol representation.  
  
class VJustify  
    Field vertical justify.  
  
priority  
    Element priority.  
    Type int
```

Helper

CHAPTER 2

KiCAD footprints

CHAPTER 3

Indices and tables

- `genindex`
- `modindex`
- `search`

p

`pykicadlib.symbol.types`, 3

Symbols

`__eq__()` (*pykicadlib.symbol.elements.Arc* method), 9
`__eq__()` (*pykicadlib.symbol.elements.Circle* method), 11
`__eq__()` (*pykicadlib.symbol.elements.Pin* method), 12
`__eq__()` (*pykicadlib.symbol.elements.Polygon* method), 13
`__eq__()` (*pykicadlib.symbol.elements.Rectangle* method), 14
`__eq__()` (*pykicadlib.symbol.elements.Text* method), 16
`__str__()` (*pykicadlib.symbol.elements.Arc* method), 9
`__str__()` (*pykicadlib.symbol.elements.Circle* method), 11
`__str__()` (*pykicadlib.symbol.elements.Field* method), 8
`__str__()` (*pykicadlib.symbol.elements.Pin* method), 12
`__str__()` (*pykicadlib.symbol.elements.Polygon* method), 13
`__str__()` (*pykicadlib.symbol.elements.Rectangle* method), 14
`__str__()` (*pykicadlib.symbol.elements.Text* method), 16

A

`add()` (*pykicadlib.symbol.elements.Polygon* method), 14
`angle` (*pykicadlib.symbol.elements.Text* attribute), 16
`Arc` (class in *pykicadlib.symbol.elements*), 9
`Arc.Boundary` (class in *pykicadlib.symbol.elements*), 10
`Arc.Fill` (class in *pykicadlib.symbol.elements*), 10
`Arc.Representation` (class in *pykicadlib.symbol.elements*), 10

B

`background` (*pykicadlib.symbol.types.Fill* attribute), 5
`bidirectional` (*pykicadlib.symbol.types.Electric* attribute), 4

`Bold` (class in *pykicadlib.symbol.types*), 3
`bold` (*pykicadlib.symbol.elements.Text* attribute), 16
`bold` (*pykicadlib.symbol.types.Style* attribute), 6
`both` (*pykicadlib.symbol.types.Representation* attribute), 6
`bottom` (*pykicadlib.symbol.types.VJustify* attribute), 7
`bounds` (*pykicadlib.symbol.elements.Arc* attribute), 10
`bounds` (*pykicadlib.symbol.elements.Circle* attribute), 11
`bounds` (*pykicadlib.symbol.elements.Pin* attribute), 12
`bounds` (*pykicadlib.symbol.elements.Polygon* attribute), 13
`bounds` (*pykicadlib.symbol.elements.Rectangle* attribute), 15
`bounds` (*pykicadlib.symbol.elements.Text* attribute), 16

C

`center` (*pykicadlib.symbol.types.HJustify* attribute), 5
`center` (*pykicadlib.symbol.types.VJustify* attribute), 7
`Circle` (class in *pykicadlib.symbol.elements*), 10
`Circle.Boundary` (class in *pykicadlib.symbol.elements*), 11
`Circle.Fill` (class in *pykicadlib.symbol.elements*), 11
`Circle.Representation` (class in *pykicadlib.symbol.elements*), 11
`clock` (*pykicadlib.symbol.types.Shape* attribute), 6
`clock_low` (*pykicadlib.symbol.types.Shape* attribute), 6

D

`Direction` (class in *pykicadlib.symbol.types*), 3
`direction` (*pykicadlib.symbol.elements.Pin* attribute), 12
`document` (*pykicadlib.symbol.types.Field* attribute), 4
`down` (*pykicadlib.symbol.types.Direction* attribute), 3

E

`Electric` (class in *pykicadlib.symbol.types*), 4

- electric (pykicadlib.symbol.elements.Pin attribute), 12
- end_angle (pykicadlib.symbol.elements.Arc attribute), 10
- end_x (pykicadlib.symbol.elements.Arc attribute), 10
- end_y (pykicadlib.symbol.elements.Arc attribute), 10
- ## F
- falling_edge_clock (pykicadlib.symbol.types.Shape attribute), 6
- Field (class in pykicadlib.symbol.elements), 8
- Field (class in pykicadlib.symbol.types), 4
- Field.HJustify (class in pykicadlib.symbol.elements), 8
- Field.Orientation (class in pykicadlib.symbol.elements), 9
- Field.Style (class in pykicadlib.symbol.elements), 9
- Field.Visibility (class in pykicadlib.symbol.elements), 9
- Field.VJustify (class in pykicadlib.symbol.elements), 9
- Fill (class in pykicadlib.symbol.types), 5
- fill (pykicadlib.symbol.elements.Arc attribute), 10
- fill (pykicadlib.symbol.elements.Circle attribute), 11
- fill (pykicadlib.symbol.elements.Polygon attribute), 13
- fill (pykicadlib.symbol.elements.Rectangle attribute), 15
- Flag (class in pykicadlib.symbol.types), 5
- footprint (pykicadlib.symbol.types.Field attribute), 4
- foreground (pykicadlib.symbol.types.Fill attribute), 5
- ## H
- HJustify (class in pykicadlib.symbol.types), 5
- hjustify (pykicadlib.symbol.elements.Field attribute), 8
- hjustify (pykicadlib.symbol.elements.Text attribute), 16
- horizontal (pykicadlib.symbol.types.Orientation attribute), 5
- ## I
- input (pykicadlib.symbol.types.Electric attribute), 4
- input_low (pykicadlib.symbol.types.Shape attribute), 6
- inverted (pykicadlib.symbol.types.Shape attribute), 6
- inverted_clock (pykicadlib.symbol.types.Shape attribute), 6
- invisible (pykicadlib.symbol.types.Shape attribute), 6
- invisible (pykicadlib.symbol.types.Visibility attribute), 7
- Italic (class in pykicadlib.symbol.types), 5
- italic (pykicadlib.symbol.elements.Text attribute), 16
- italic (pykicadlib.symbol.types.Style attribute), 6
- italic_bold (pykicadlib.symbol.types.Style attribute), 6
- ## L
- left (pykicadlib.symbol.types.Direction attribute), 3
- left (pykicadlib.symbol.types.HJustify attribute), 5
- length (pykicadlib.symbol.elements.Pin attribute), 12
- line (pykicadlib.symbol.types.Shape attribute), 6
- locked (pykicadlib.symbol.types.Units attribute), 7
- ## M
- manufacturer (pykicadlib.symbol.types.Field attribute), 4
- model (pykicadlib.symbol.types.Field attribute), 4
- morgan (pykicadlib.symbol.types.Representation attribute), 6
- ## N
- name (pykicadlib.symbol.elements.Pin attribute), 12
- name (pykicadlib.symbol.types.Field attribute), 4
- name_size (pykicadlib.symbol.elements.Pin attribute), 12
- no (pykicadlib.symbol.types.Visible attribute), 7
- non_logic (pykicadlib.symbol.types.Shape attribute), 6
- none (pykicadlib.symbol.types.Fill attribute), 5
- none (pykicadlib.symbol.types.Style attribute), 6
- normal (pykicadlib.symbol.types.Flag attribute), 5
- normal (pykicadlib.symbol.types.Representation attribute), 6
- not_connected (pykicadlib.symbol.types.Electric attribute), 4
- number (pykicadlib.symbol.elements.Pin attribute), 12
- number_size (pykicadlib.symbol.elements.Pin attribute), 12
- ## O
- off (pykicadlib.symbol.types.Bold attribute), 3
- off (pykicadlib.symbol.types.Italic attribute), 5
- on (pykicadlib.symbol.types.Bold attribute), 3
- on (pykicadlib.symbol.types.Italic attribute), 5
- open_collector (pykicadlib.symbol.types.Electric attribute), 4
- open_emitter (pykicadlib.symbol.types.Electric attribute), 4
- Orientation (class in pykicadlib.symbol.types), 5
- orientation (pykicadlib.symbol.elements.Field attribute), 8
- output (pykicadlib.symbol.types.Electric attribute), 4
- output_low (pykicadlib.symbol.types.Shape attribute), 6
- ## P
- passive (pykicadlib.symbol.types.Electric attribute), 4

Pin (class in *pykicadlib.symbol.elements*), 11
 Pin.Boundary (class in *pykicadlib.symbol.elements*), 13
 Pin.Direction (class in *pykicadlib.symbol.elements*), 13
 Pin.Electric (class in *pykicadlib.symbol.elements*), 13
 Pin.Fill (class in *pykicadlib.symbol.elements*), 13
 Pin.Representation (class in *pykicadlib.symbol.elements*), 13
 Pin.Shape (class in *pykicadlib.symbol.elements*), 13
 points (*pykicadlib.symbol.elements.Polygon* attribute), 13
 Polygon (class in *pykicadlib.symbol.elements*), 13
 Polygon.Boundary (class in *pykicadlib.symbol.elements*), 14
 Polygon.Fill (class in *pykicadlib.symbol.elements*), 14
 Polygon.Point (class in *pykicadlib.symbol.elements*), 14
 Polygon.Representation (class in *pykicadlib.symbol.elements*), 14
 power (*pykicadlib.symbol.types.Field* attribute), 4
 power (*pykicadlib.symbol.types.Flag* attribute), 5
 power_input (*pykicadlib.symbol.types.Electric* attribute), 4
 power_output (*pykicadlib.symbol.types.Electric* attribute), 4
 priority (*pykicadlib.symbol.elements.Arc* attribute), 10
 priority (*pykicadlib.symbol.elements.Circle* attribute), 11
 priority (*pykicadlib.symbol.elements.Pin* attribute), 12
 priority (*pykicadlib.symbol.elements.Polygon* attribute), 13
 priority (*pykicadlib.symbol.elements.Rectangle* attribute), 15
 priority (*pykicadlib.symbol.elements.Text* attribute), 17
 pykicadlib.symbol.types (module), 3

R

radius (*pykicadlib.symbol.elements.Arc* attribute), 9
 radius (*pykicadlib.symbol.elements.Circle* attribute), 11
 Rectangle (class in *pykicadlib.symbol.elements*), 14
 Rectangle.Boundary (class in *pykicadlib.symbol.elements*), 15
 Rectangle.Fill (class in *pykicadlib.symbol.elements*), 15
 Rectangle.Representation (class in *pykicadlib.symbol.elements*), 15
 reference (*pykicadlib.symbol.types.Field* attribute), 5

remove () (*pykicadlib.symbol.elements.Polygon* method), 14
 Representation (class in *pykicadlib.symbol.types*), 3, 6
 right (*pykicadlib.symbol.types.Direction* attribute), 4
 right (*pykicadlib.symbol.types.HJustify* attribute), 5

S

Shape (class in *pykicadlib.symbol.types*), 6
 shape (*pykicadlib.symbol.elements.Pin* attribute), 12
 size (*pykicadlib.symbol.elements.Field* attribute), 8
 size (*pykicadlib.symbol.elements.Text* attribute), 16
 start_angle (*pykicadlib.symbol.elements.Arc* attribute), 10
 start_x (*pykicadlib.symbol.elements.Arc* attribute), 10
 start_y (*pykicadlib.symbol.elements.Arc* attribute), 10
 Style (class in *pykicadlib.symbol.types*), 6
 style (*pykicadlib.symbol.elements.Field* attribute), 8
 swappable (*pykicadlib.symbol.types.Units* attribute), 7

T

temperature (*pykicadlib.symbol.types.Field* attribute), 5
 Text (class in *pykicadlib.symbol.elements*), 15
 Text.Bold (class in *pykicadlib.symbol.elements*), 16
 Text.Boundary (class in *pykicadlib.symbol.elements*), 16
 Text.Fill (class in *pykicadlib.symbol.elements*), 16
 Text.HJustify (class in *pykicadlib.symbol.elements*), 17
 Text.Italic (class in *pykicadlib.symbol.elements*), 17
 Text.Representation (class in *pykicadlib.symbol.elements*), 17
 Text.VJustify (class in *pykicadlib.symbol.elements*), 17
 thickness (*pykicadlib.symbol.elements.Arc* attribute), 10
 thickness (*pykicadlib.symbol.elements.Circle* attribute), 11
 thickness (*pykicadlib.symbol.elements.Polygon* attribute), 13
 thickness (*pykicadlib.symbol.elements.Rectangle* attribute), 15
 tolerance (*pykicadlib.symbol.types.Field* attribute), 5
 top (*pykicadlib.symbol.types.VJustify* attribute), 7
 tristate (*pykicadlib.symbol.types.Electric* attribute), 4
 Type (*pykicadlib.symbol.elements.Field* attribute), 9
 type (*pykicadlib.symbol.elements.Field* attribute), 8

U

Units (class in *pykicadlib.symbol.types*), 7

unspecified (*pykicadlib.symbol.types.Electric attribute*), 4
up (*pykicadlib.symbol.types.Direction attribute*), 4

V

value (*pykicadlib.symbol.elements.Field attribute*), 8
value (*pykicadlib.symbol.elements.Text attribute*), 16
value (*pykicadlib.symbol.types.Field attribute*), 5
vertical (*pykicadlib.symbol.types.Orientation attribute*), 6
Visibility (*class in pykicadlib.symbol.types*), 7
visibility (*pykicadlib.symbol.elements.Field attribute*), 8
Visible (*class in pykicadlib.symbol.types*), 7
visible (*pykicadlib.symbol.elements.Pin attribute*), 12
visible (*pykicadlib.symbol.types.Visibility attribute*), 7
VJustify (*class in pykicadlib.symbol.types*), 7
vjustify (*pykicadlib.symbol.elements.Field attribute*), 8
vjustify (*pykicadlib.symbol.elements.Text attribute*), 16
voltage (*pykicadlib.symbol.types.Field attribute*), 5

X

x (*pykicadlib.symbol.elements.Arc attribute*), 9
x (*pykicadlib.symbol.elements.Circle attribute*), 11
x (*pykicadlib.symbol.elements.Field attribute*), 8
x (*pykicadlib.symbol.elements.Pin attribute*), 12
x (*pykicadlib.symbol.elements.Text attribute*), 16
x1 (*pykicadlib.symbol.elements.Rectangle attribute*), 15
x2 (*pykicadlib.symbol.elements.Rectangle attribute*), 15

Y

y (*pykicadlib.symbol.elements.Arc attribute*), 9
y (*pykicadlib.symbol.elements.Circle attribute*), 11
y (*pykicadlib.symbol.elements.Field attribute*), 8
y (*pykicadlib.symbol.elements.Pin attribute*), 12
y (*pykicadlib.symbol.elements.Text attribute*), 16
y1 (*pykicadlib.symbol.elements.Rectangle attribute*), 15
y2 (*pykicadlib.symbol.elements.Rectangle attribute*), 15
yes (*pykicadlib.symbol.types.Visible attribute*), 7