Manual of the GDS.py Library

Olivier Scholder

March 6, 2017

Contents

1 Installation

The installation is easy, just copy the script wherever you want and assign the environement variable PYTHONPATH to the directory path where you put the file.

ToDo

```
cd
mkdir scripts
cp PATH_TO_GDS/gds.py scripts/
echo 'export PYTHONPATH="~/scripts"' >> ~/.bashrc
```

2 General options

All the units are set in nm if not specified.

layer On which layer you want your line

width The line width. For Raith is normaly set to 0. It's the dose which will actually define your width

dose The dose set the DATATYPE variable which is interpreted by Raith as the dose.

loop Tell how many time in a row the object should be looped. If set to *None* (which is default), uses the internal *loops* variable.

This function works only for FIB (Elphy Multibeam). This tag will be probably discarded by Raith150 II or other system, meaning that only a single loop will be performed.

3 Basic Functions

3.1 First step

The very first step is to call the library:

```
#!/usr/bin/python

import gds
```

```
g=gds.GDSII()
```

This small piece of code will import you gds library and create a new GDSII object called g. This steps just prepare everything you need to write GDSII files. By calling `GDSII()` without argument, this will store all of your data in memory until you call the `save()` function. An alternative to save memory, as you are preparing large data, is to give in argument the file name of the gds you want to write into:

```
#!/usr/bin/python

import gds

g=gds.GDSII("mygds.gds")

# ...
# Add structures ...
# ...
g.close()
```

Notice, that you need to close that file with the command `close()` with this direct writing method.

The reader should also be aware that this script never check if file already exists an will overwrite already existing files.

3.2 Create new file

Headers The GDSII object provides you the `new()` function which creates automatically the header of your GDS file. As argument new take the name of your library, which has mainly no importance. Normally `new()` should be called just after the GDSII object creation.

Footers and writing file In order to save your structure to a GDS file you should write the footers which are done automatically by calling the `endLib()` function. Once this step is done you can save your GDS file with the `write()` function. You have to specifiy the path of your gds as argument to the `write()` function.

Simple Example The following example will write a valid GDSII file containing no structures. It should be the basic of all your scripts.

```
#!/usr/bin/python

import gds

g=gds.GDSII()
g.new('EMPTY')
g.endLib()
g.write('empty.gds')
```

4 Graphical Functions

Now let's see what we can do graphically with the GDSII files

4.1 Lines

`addLine(pts, layer=0, width=0, dose=1, loop=None)`

pts the points list. pts=[x1,y1,x2,y1,...,xn,yn]

4.2 Circles and Arcs

`addCircle(pos, radius, npts=10, layer=0, width=0, dose=1, A=0, B=360, loop=None)`

pos The position of the center of the circle. *pos*=[x,y]

radius The radius of the circle

npts The circle is actually a regular polygon and npts specify the number of submits. So if *npts*=4, you will get a square.

A The start angle

B The end angle. So if *A*=0 and *B*=90, you will get the quarter of a circle

4.3 Polygons

`addPoly(pos, layer=0, dose=1, loop=None)`

pos list of points. *pos*=[x1,y1,x2,y2,...,xn,yn]

4.4 Rectangles

`addRect(pos, layer=0, dose=1, CCW=False, loop=None)`

pos *pos*=[xll,yll,width,height]

CCW draw ClowckWise or CounterClockWise=?

4.5 Text

`addText(pos, txt, height=None, mag=22.22222, layer=0, width=0, angle=0, loop=None)`

pos position of the text. *pos*=[x,y]. Normaly it's the middle point of the text.

txt The text as string.

height Set the text height. This option will compute automatically the corresponding magnification and will override the option mag.

mag The magnification. A *mag* of 22.22222 will give a text of 1 µm height. It's maybe better to use the height option.

angle The rotation angle of the text.

5 Dose and Loops

5.1 Dose

6 Examples

Examples are always much better than words.

```
#!/usr/bin/python

import gds

g=gds.GDSII()
g.new('DEMO')
g.newStr('Dipant')
space=(1000,1000)
for y in range(5):
        for x in range(5):
                g.addLine((x*space[0],y*space[1],x*space[0]+width,y*space[1]),
                    dose=y*0.5+x*0.1)
for y in range(5):
        g.addText((-space[0],y*space[1]),str(y*0.5),height=500)
for x in range(5):
        g.addText((x*space[0],-space[1]),"+"+str(x*0.1),height=500)
g.endStr()
g.endLib()
g.write('demo.gds')
```