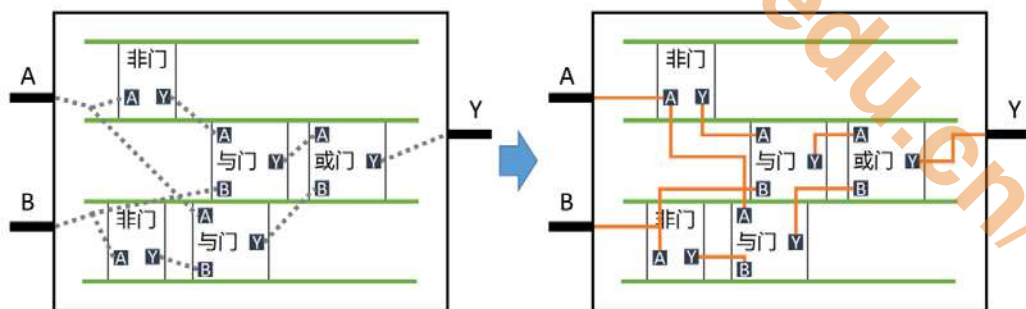

目录

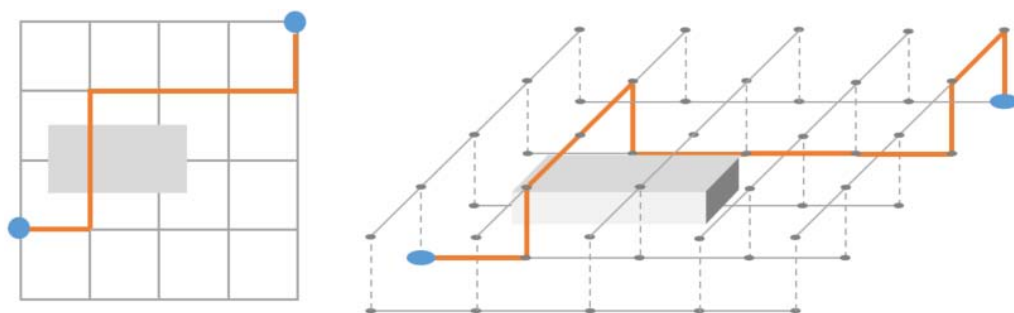
布线	2
实验目的	2
实验内容	2
布线基础概念	2
全局布线	4
详细布线	4
布线指标	5
实验步骤	6
生成布线结果	6
调整布线参数	12
小结	17

布线

布线是把已经布局好的标准单元，按照单元上的接口关系连接起来，这是布线的基本目标。由于制造方面的原因，所有的线能且只能通过横平竖直的方式连接。在连接的过程中还要保证线之间不能有重叠。



上图是俯视图，实际上，从三维上看，布线的结果是立体的，如下图所示。



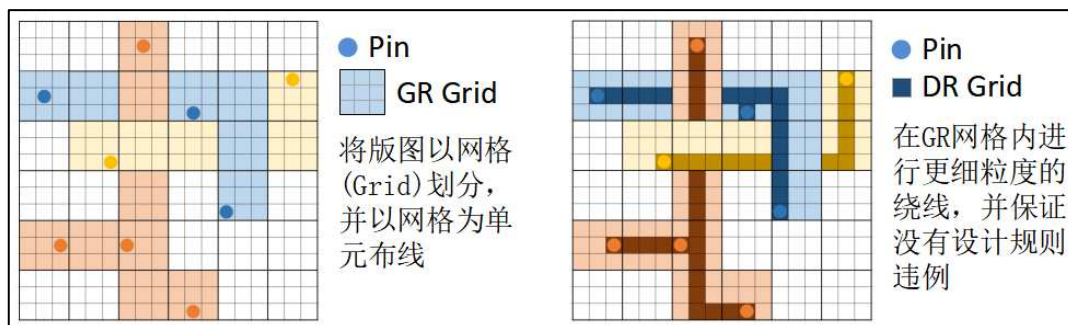
总的来说，布线的基本目标是将所有连接关系以空间直角线的形式连接起来，并且保证没有线与线之间的重叠。

实验目的

- (1) 了解全局布线与详细布线的基本概念
- (2) 掌握使用全局布线和详细布线工具

实验内容

布线基础概念



布线阶段分为全局布线（Global Routing, GR）和详细布线（Detail Routing, DR）。

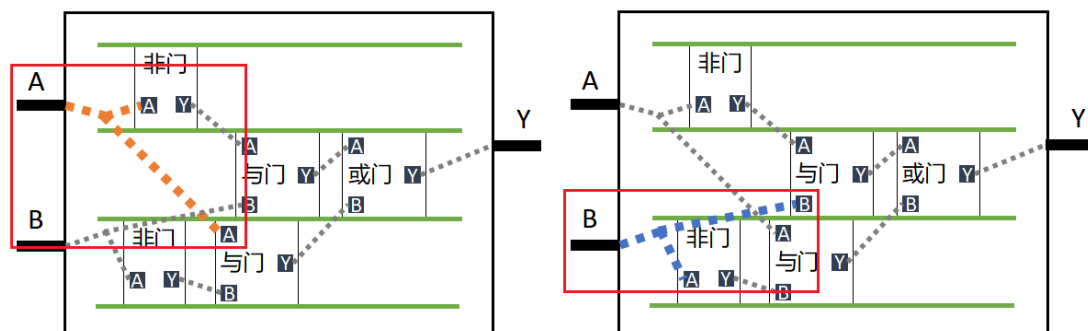
全局布线是将版图以方格（GR Grid）形式进行划分，然后以此方格为单位进行布线，下图中同等颜色点为需要将它们连接起来的点，对应颜色的方格为全局布线的布线结果。由于是三维结构，从俯视图上可以看出这些布线结果是有遮挡的。



详细布线是在全局布线的结果内进行更细致的布线。下图中在全局布线结果内的深颜色小方格（DR Grid），就是详细布线的布线单位。详细布线的结果同样是三维的，可以看到其在三维上也是有遮挡的。

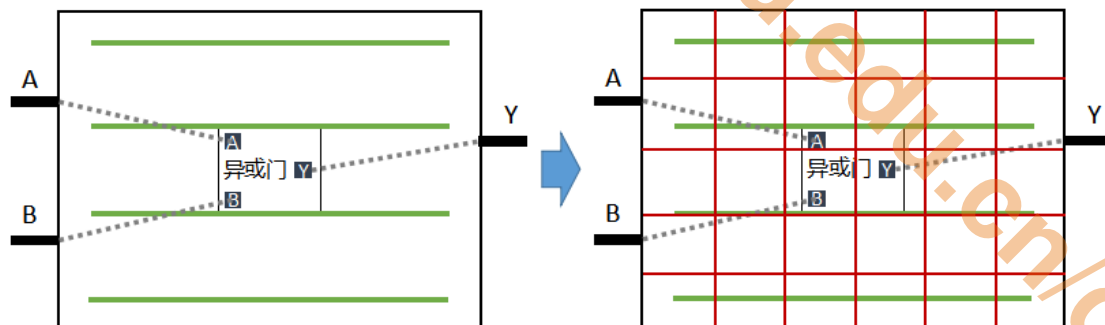


布线阶段的输入是带有多个连接关系的版图。以之前的双控开关电路为例子。如下图所示，左图中橙色的连接关系和右图中蓝色的连接关系都是多个连接关系中的一个。这样的连接关系被称之为线网（Net），布线就是将这些线网以直角线连接起来。

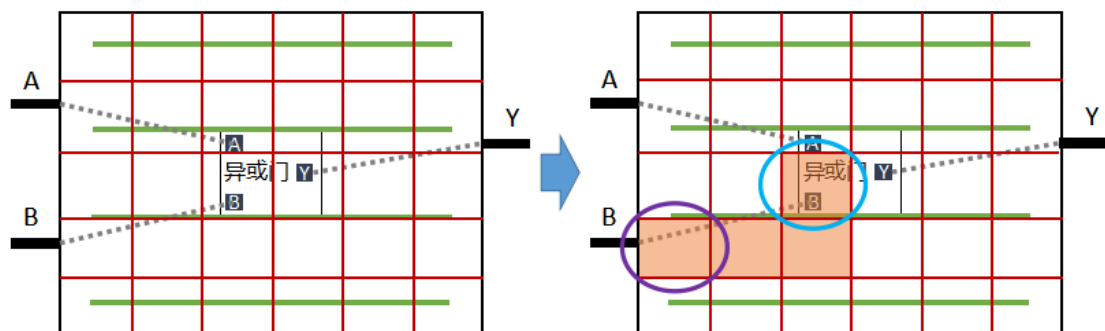


全局布线

以简单电路双控开关为例子。全局布线阶段首先将版图划分为多个方格。如下图所示，红色方框为划分的方格。

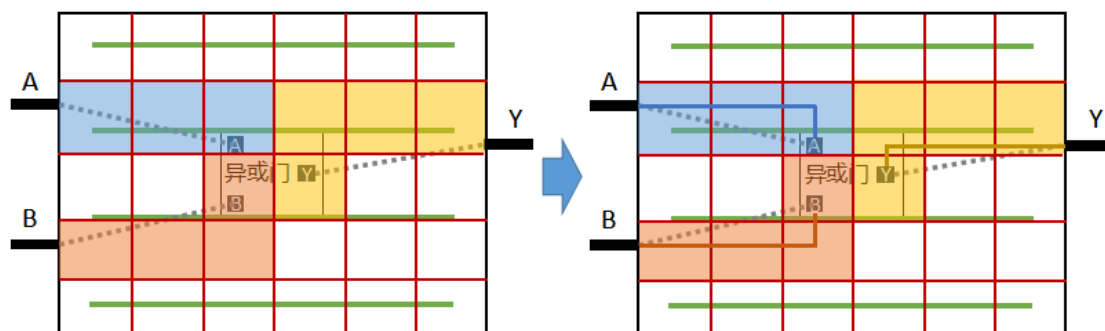


以方格为单位进行布线。将连接关系以直角线的形式连起来。如下图所示，芯片接口 B 与异或门的接口 B 连接，芯片接口 B 处于一个方格（紫色原环）内，异或门的接口 B（蓝色原环）内，下图将这两个方格连接了起来，连接的单位也是方格（图中橙色部分）。

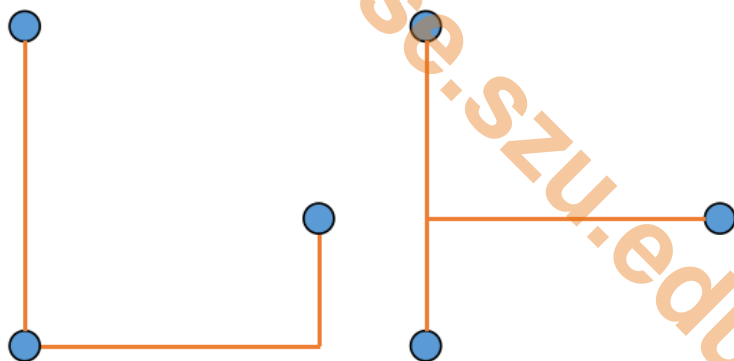


详细布线

将这些方格连接好后，最后在方格内以更细的方式进行布线。



将这些连接关系以直角线连起来很简单，但是要连起来的同时做到连接的线长**最短**却是一个很难的问题。

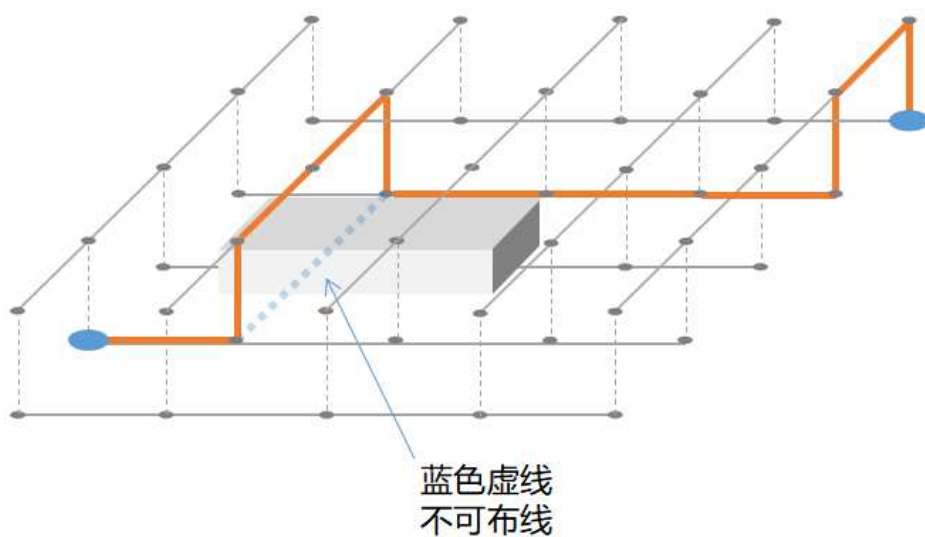


如上图所示，左图和右图都将三个蓝色的点连起来了，但是右图中的总线段长度明显比左图的更小。三个点的连线结果看起来很容易构建，但是点数多之后就非常难了。

想深入了解？

构造连接多个点的一棵树是一个很难的问题，感兴趣的同学可以搜索“斯坦纳树”学习。这里是“斯坦纳树”问题的变种——“直角斯坦纳树”。

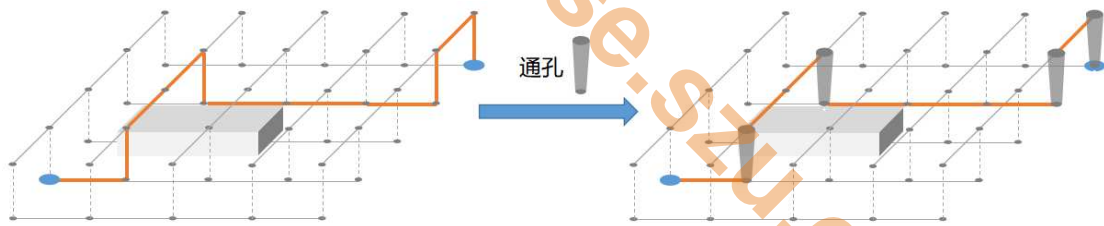
将这些连接关系一个接一个的以直角线的形式连起来，最后就是布线阶段的总结果了。



除此之外，版图中也会出现一些障碍，布线结果是不能从这些障碍中穿过去的，所以在布线的过程中还需要进行避障。

布线指标

基础的布线指标主要有总线长和总通孔数量两个指标。总线长很好理解，就是所有互连线长度之和。下面主要介绍通孔的基本概念。

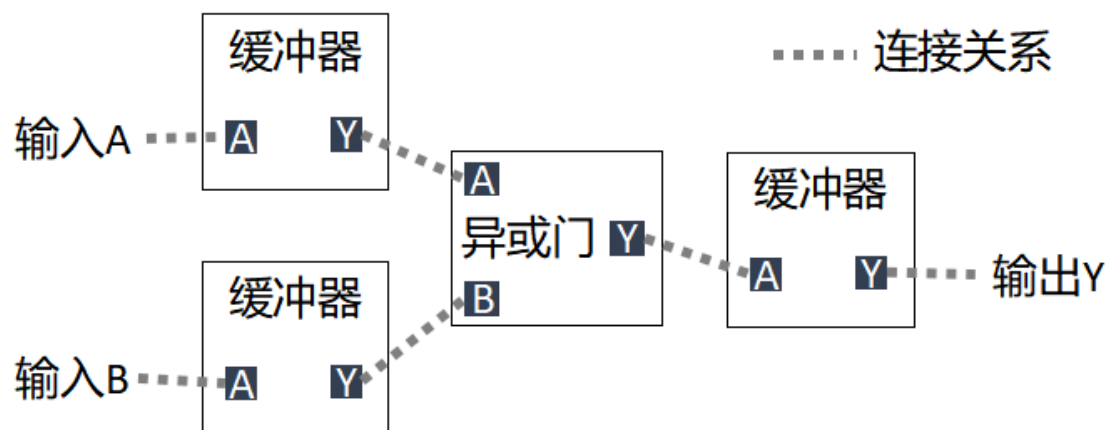


通孔是连接相邻两层互连线的一个柱状物体，如上图所示，在空间方向的线都由真实情况的通孔替代了。以上图的右图为例，总通孔数量有四个。总线长为右图中所有橙色线的总长度。

实验步骤

生成布线结果

这里我们使用 Switch_routing 设计作为例子。下图为 Switch_routing 的逻辑门电路。



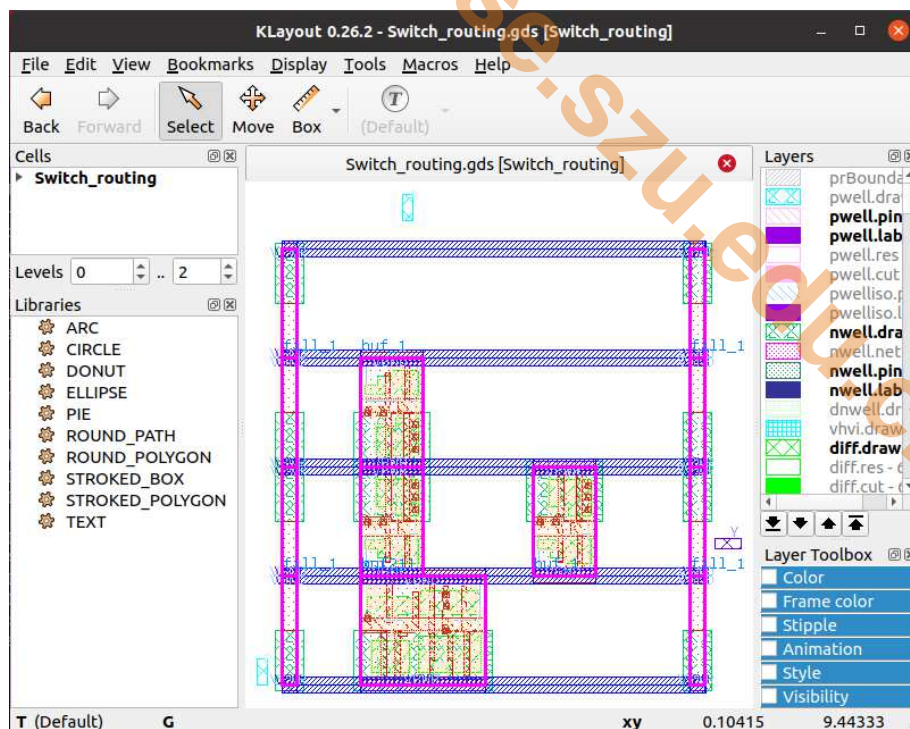
这个设计由三个缓冲器和一个异或门组成，在“实验二的工艺库介绍”中，我们知道缓冲器只是用于加强信号的区分度，并不会改变信号的状态。所以这个设计在逻辑上只是一个异或门，其实现与双控开关电路同样的功能。

在桌面上打开终端，进入 iFlow/scripts 目录。

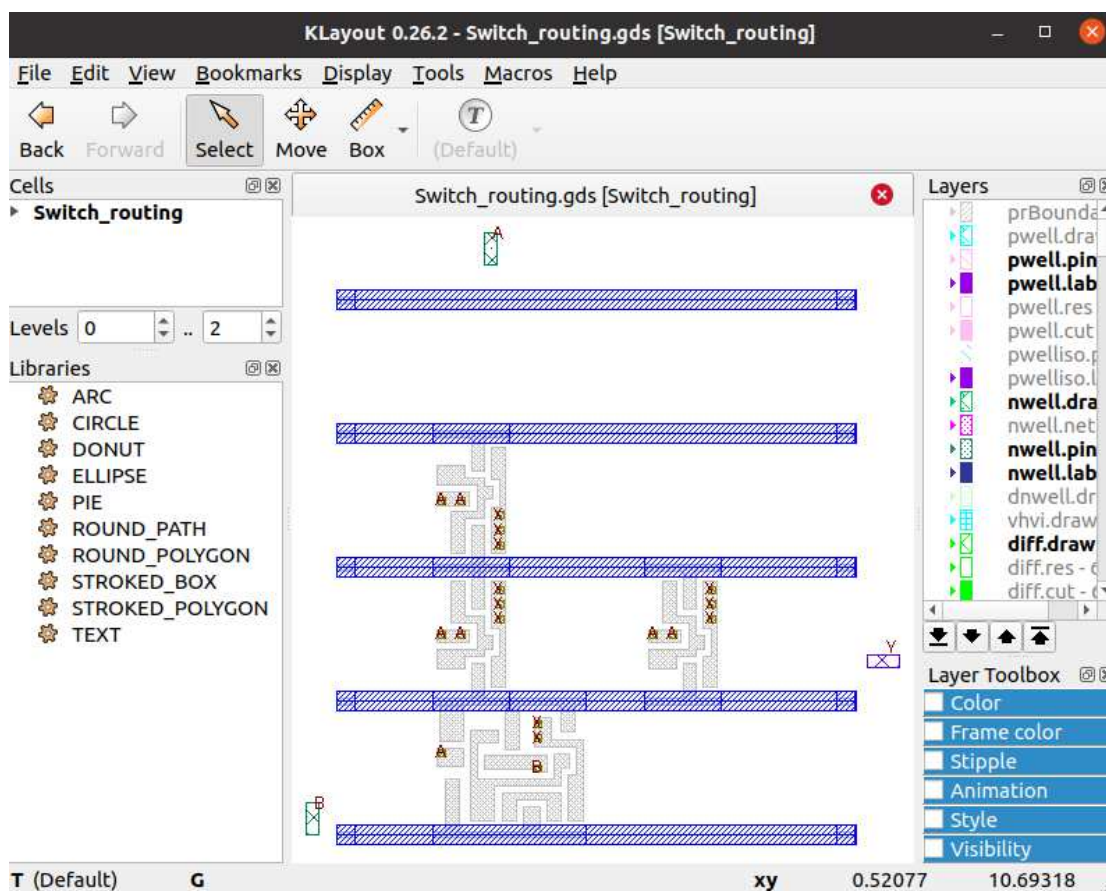
```
lc3@lc3-virtual-machine:~/Desktop$ cd iFlow/scripts/
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$ pwd
/home/lc3/Desktop/iFlow/scripts
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$
```

输入以下命令生成 Switch_routing 设计的布局结果并查看。

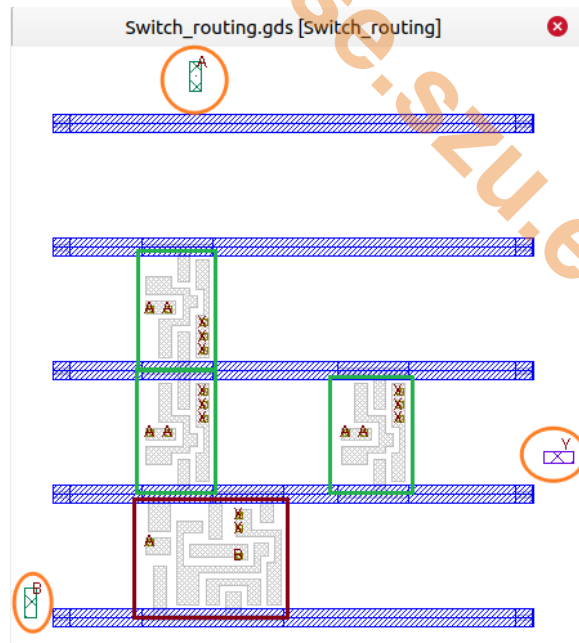
```
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$ ./run_flow.py -d Switch_routing -s
synth,floorplan,tapcell,pdn,gplace,resize,dplace,layout -f sky130 -t HS -c TYP
```

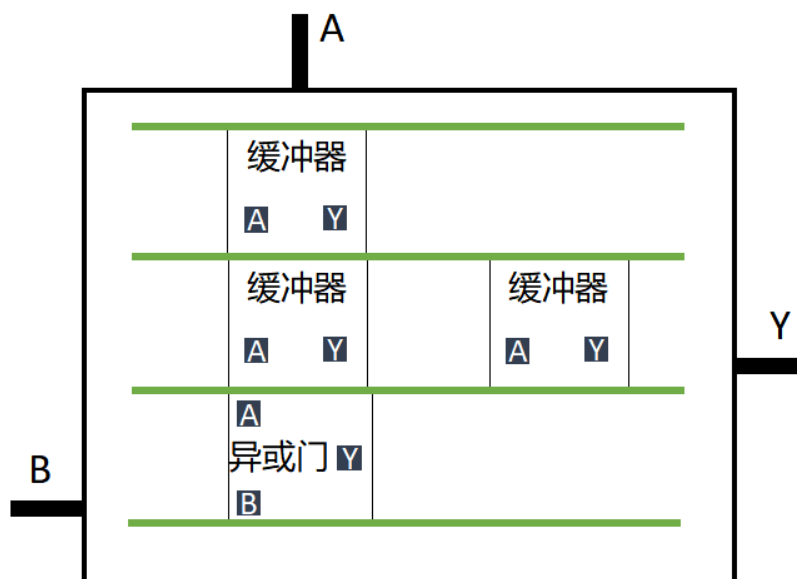
通过“附录：更换图层样式”更换后，可以更明显地看到布局结果如下。



这个设计一共由 4 个标准单元组成，其中红框的为异或门标准单元，绿框的为缓冲器，而周围的橙色椭圆框为三个芯片接口（A，B，Y）。



这个设计以逻辑门的电路示意图如下。



关闭 klayout，输入以下命令生成全局布线结果。

```
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$ ./run_flow.py -d Switch_routing -s
groute -f sky130 -t HS -c TYP
```

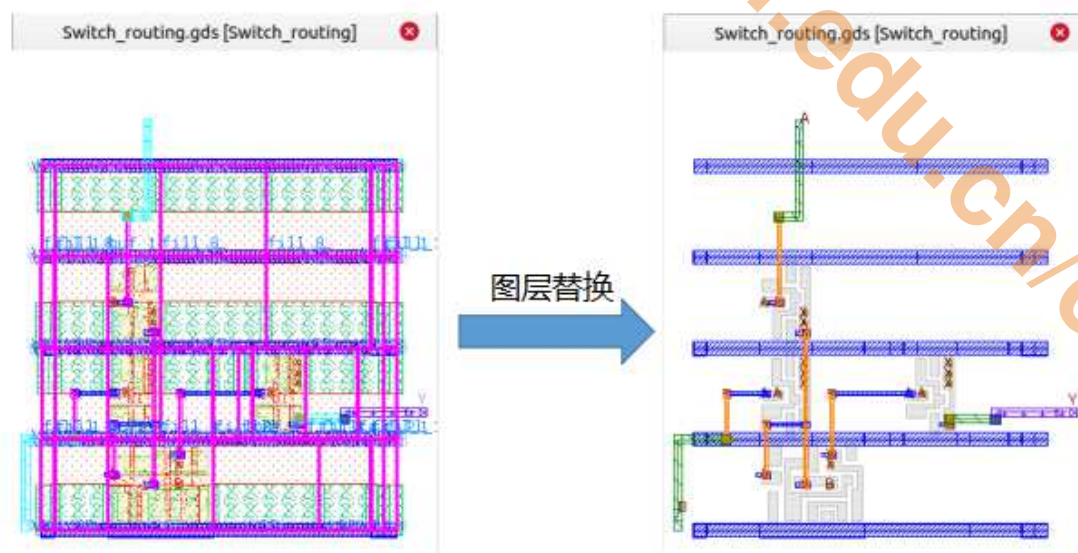
通过 show_guide 工具生成全局布线的可视化结果。

```
> Running FastRoute... Done!
> Writing guides...
> Num routed nets: 6
> Writing guides... Done!
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$ ../tools/guide_viewer/show_guid
e ../result/Switch_routing.groute.openroad_1.1.0.HS.TYP.default/route.guide ../r
esult/Switch_routing.groute.openroad_1.1.0.HS.TYP.default/guide.gds
successful!
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$
```

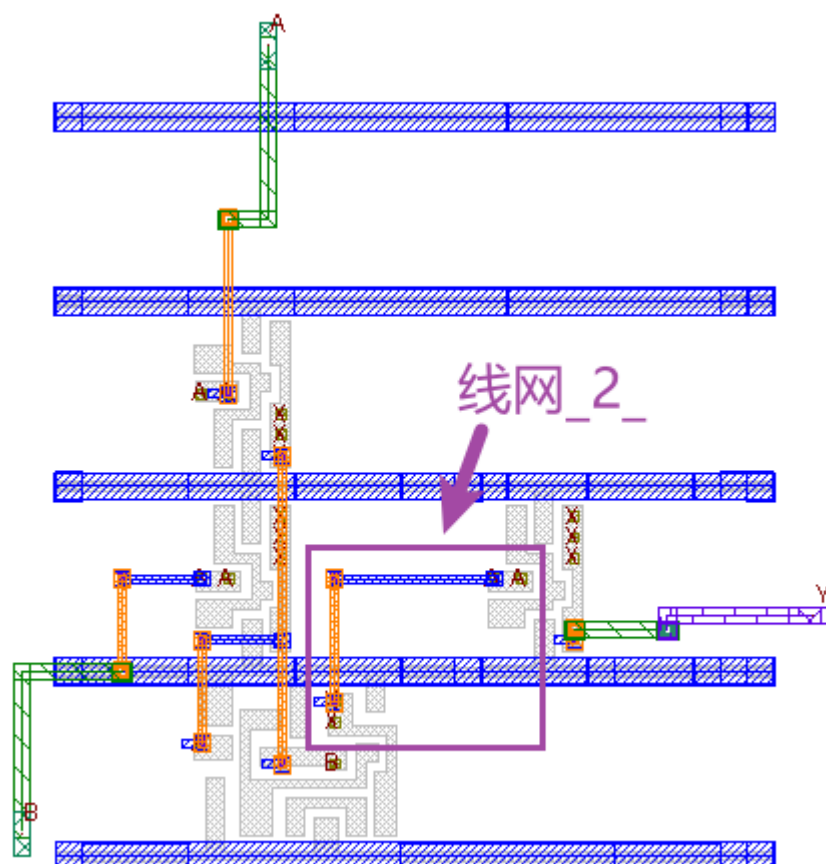

生成详细布线结果并查看。

```
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$ ./run_flow.py -d Switch_routing  
-s droute,layout -f sky130 -t HS -c TYP
```

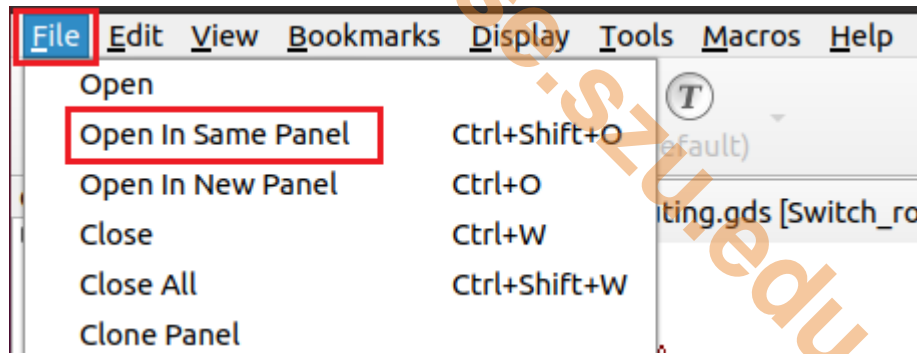
布线版图打开后，通过“附录：更换图层样式”更换图层样式。



下面我们以单个线网举例，这个线网的名字为“_2_”，由下图红框圈出，



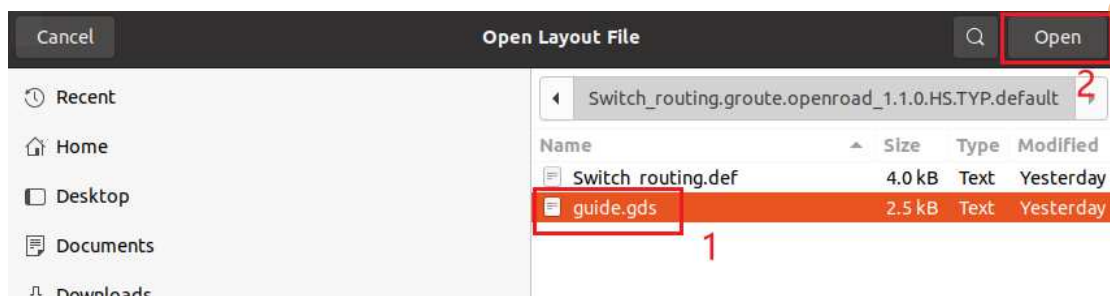
随后将全局布线结果导入图层中。



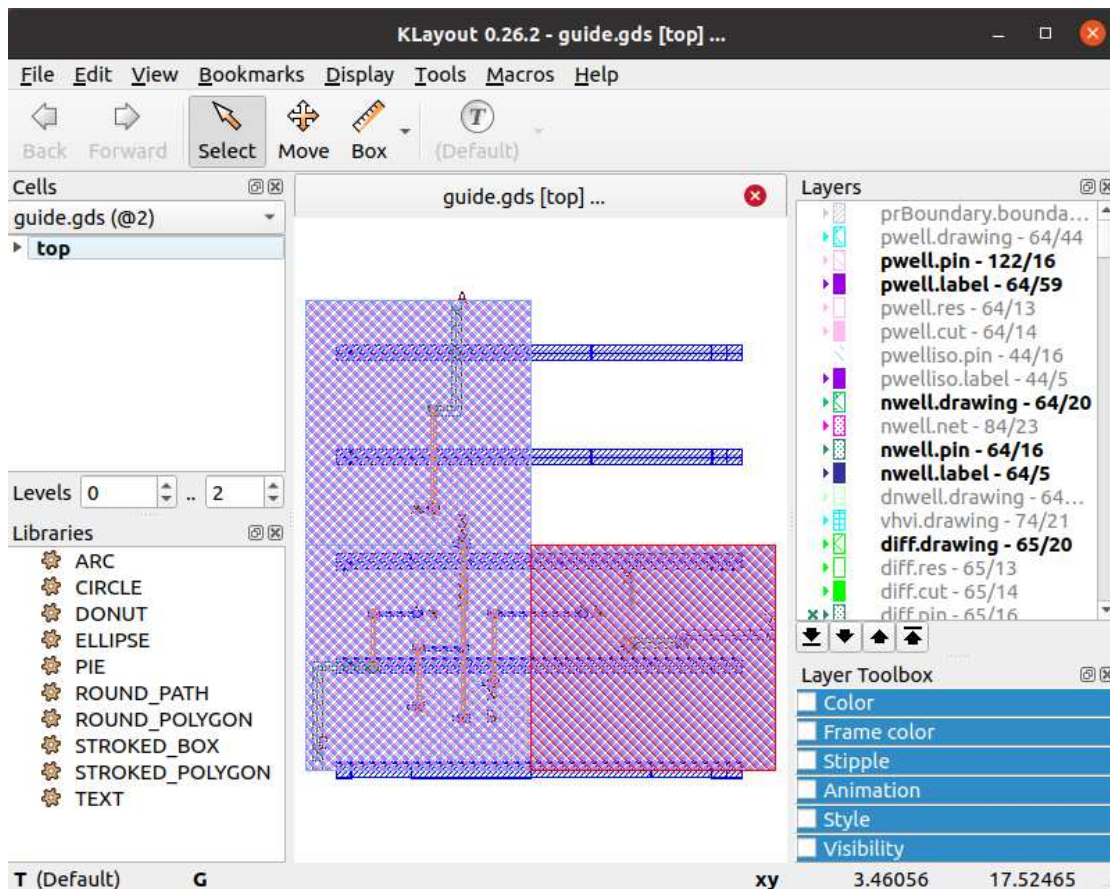
找到文件位置并打开，文件位置在

/home/lc3/Desktop/iFlow/result/Switch_routing.groute.openroad_1.1.0.HS.TYP.

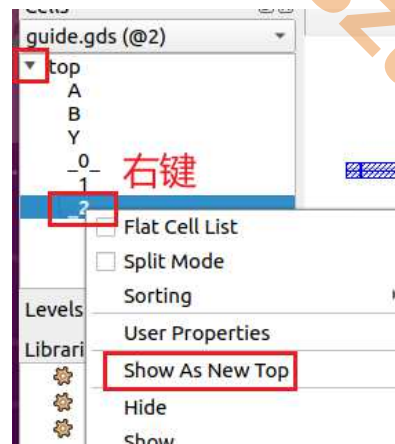
default



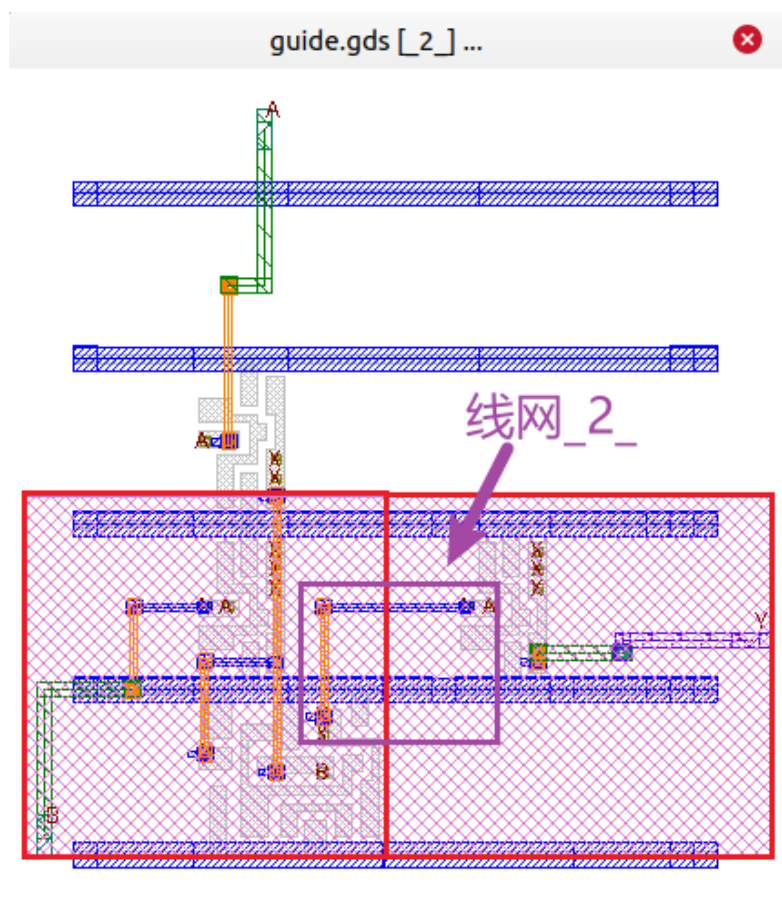
文件打开后如下显示。



将左侧栏的 top 展开，对着 “_2_” 右键，并选择 “Show As New Top”。将线网_2_的全局布线结果展示出来。



这是线网_2_的全局布线结果，可以看到在全局布线阶段是由两个大方格组成。



关闭可视化软件 klayout。

除了有可视化的布线结果，在终端上也输出了对应的线长与通孔数量。


```
lc3@lc3-virtual-machine: ~/Desktop/iFlow/scripts

[INFO DRT-0198] complete detail routing
total wire length = 33 um
total wire length on LAYER li1 = 0 um
total wire length on LAYER met1 = 5 um
total wire length on LAYER met2 = 14 um
total wire length on LAYER met3 = 10 um
total wire length on LAYER met4 = 2 um
total wire length on LAYER met5 = 0 um
total number of vias = 22
up-via summary (total 22):

-----
FR_MASTERSLICE    0
      li1         9
      met1        9
      met2        3
      met3        1
      met4        0
-----
                        22

[INFO DRT-0172] cpu time = 00:00:00, elapsed time = 00:00:00, memory = 1800.81 (
MB), peak = 1800.81 (MB)

[INFO DRT-0180] post processing ...
Current full name : Switch_routing.layout.klayout_0.26.2.HS.TYP.default
Previous full name : Switch_routing.drout.openroad_1.2.0.HS.TYP.default

klayoutInsertLef.py : Insert lefs into lyt file of klayout
klayoutInsertLef.py : Finished
[INFO] Clearing cells...
[INFO] Merging GDS files...
      /home/lc3/Desktop/iFlow/foundry/sky130/gds/sky130_fd_sc_hs.gds
[INFO] Copying toplevel cell 'Switch_routing'
WARNING: no fill config file specified
[INFO] Checking for missing GDS...
[INFO] All LEF cells have matching GDS cells
[INFO] Writing out GDS '/home/lc3/Desktop/iFlow/result/Switch_routing.layout.kla
yout_0.26.2.HS.TYP.default/Switch_routing.gds'
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$
```

线长

通孔

调整布线参数

由于详细布线基本没有可操作的参数，下面主要介绍全局布线时的布线参数。

```
fastroute -output_file $RESULT_PATH/route.guide \
-max_routing_layer $MAX_ROUTING_LAYER \
-unidirectional_routing true \
-capacity_adjustment 0.1 \
-overflow_iterations 200 \
-layers_adjustments {{1 0.1} {2 0.1}} \
-verbose 2
```

全局布线使用的工具是“fastroute”，这是一款开源全局布线器。

-output_file: 设置全局布线结果的输出路径

-max_routing_layer: 设置最大可以布线的层数，由于芯片是三维结构，在三维立体上有很多布线层，层与层之间的互连线通过“通孔”连接。

-unidirectional_routing: 设置布线的优先方向，每一层布线层都有对应的优先方向，一般来说相邻两层的布线方向是垂直的（如第一层为横方向，那么第二层则为竖方向）

-capacity_adjustment: 设置布线能力调整（值越小能布下去的线越多）

-overflow_iterations: 布线算法的迭代次数（次数越多，可以布通的概率越大）

-layers_adjustments: 每一层的布线能力调整，可以特殊设定每一层的布线能力，值越小能布下去的线越多

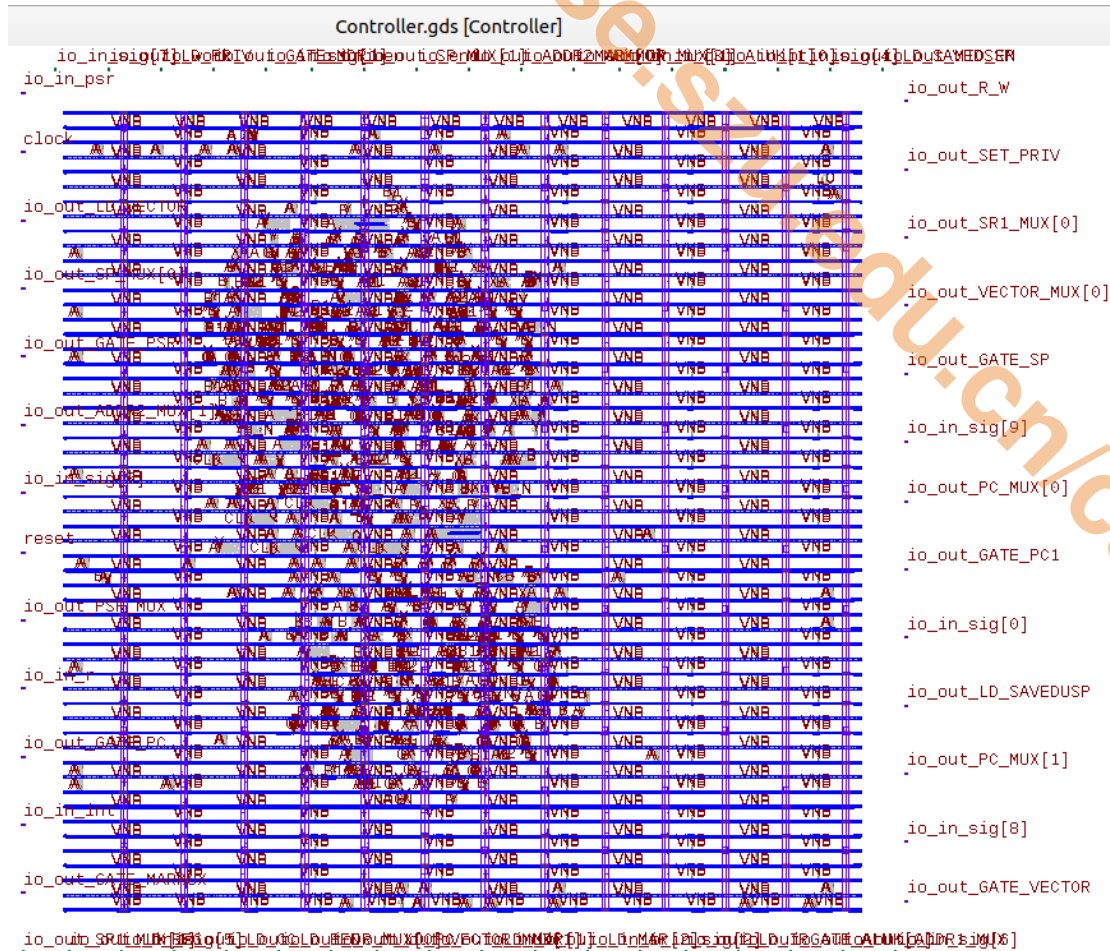
-verbose: log 的输出等级（数值越高输出的越多，分为 0, 1, 2）

下面我们主要来更改 **layers_adjustments** 参数来查看其对布线结果的影响。这里我们使用 LC3 里的控制器模块（Controller）作为实验设计，这相对于双控开关来说有更多线网需要布线，对于调整的参数也能更明显。

进入 iFlow/scripts 目录，输入以下命令，先生成控制器模块的布线前结果。

```
lc3@lc3-virtual-machine:~/Desktop$ cd iFlow/scripts/
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$ pwd
/home/lc3/Desktop/iFlow/scripts
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$ ./run_flow.py -d Controller -s
synth,floorplan,tapcell,pdn,gplace,resize,dplace,cts,filler,layout -f sky130 -t
HS -c TYP
```

布线版图打开后，通过“附录：更换图层样式”更换图层样式后，可以看到结果如下。



关闭版图查看器 layout，查看当前设置的全局布线参数。

```
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$ gedit Controller/groute.openroad_1.1.0.tcl
```

可以看到第 83 行开始为全局布线参数。

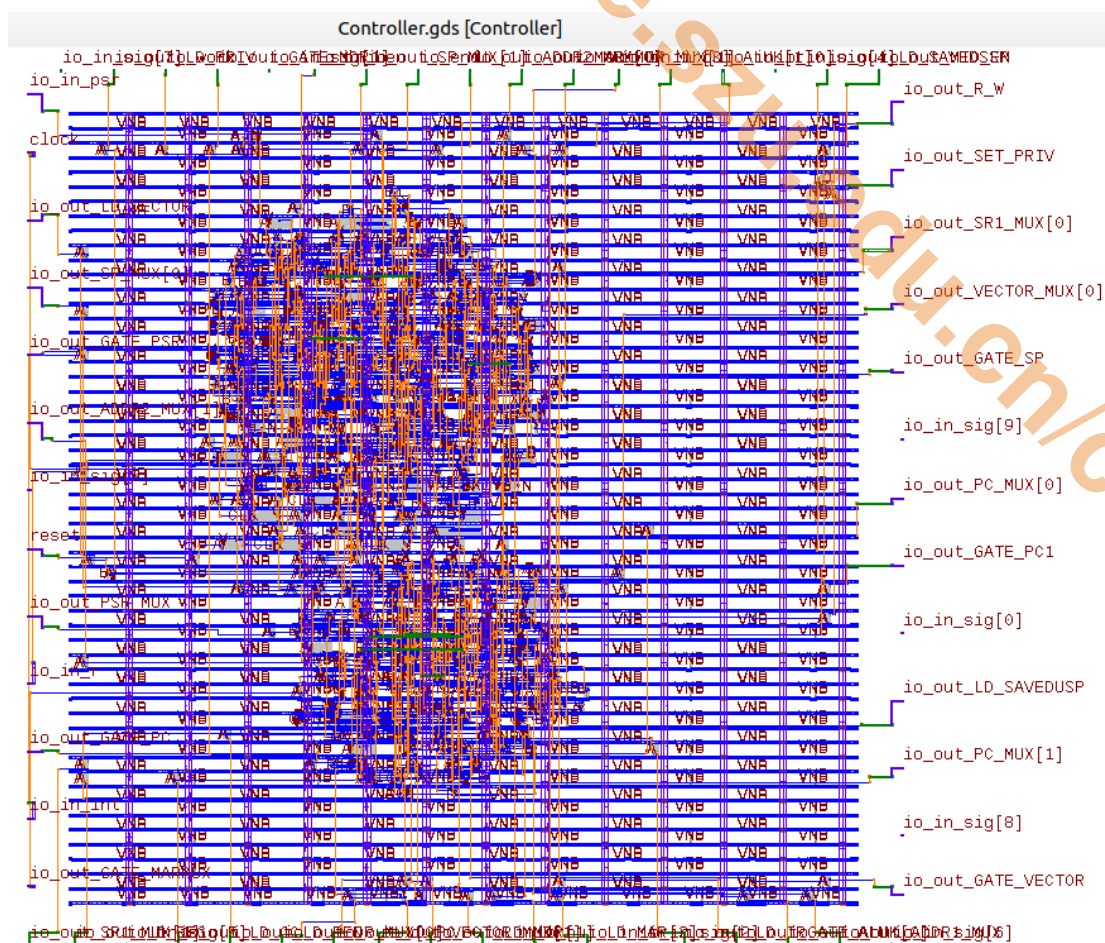
```
82
83 fastroute -output_file $RESULT_PATH/route.guide \
84           -max_routing_layer $MAX_ROUTING_LAYER \
85           -unidirectional_routing true \
86           -capacity_adjustment 0.1 \
87           -overflow iterations 200 \
88           -layers_adjustments {{1 0.1} {2 0.1}} \
89           -verbose 2
90
```

当前的参数表示，层 1、2 的布线能力为 0.1，0.1，表示大部分的线都会布线在层 1、2 上。

关闭文件浏览器，输入以下命令生成布线结果。

```
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$ ./run_flow.py -d Controller -s groute,droute,layout -f sky130 -t HS -c TYP
```

通过“附录：更换图层样式”更换图层样式后，可以看到布线结果如下。



在图的右侧，可以看到图层工具栏，在工具栏内右键并选择“Hide All”，随后找到“met3.drawing”双击打开这个图层。



“met3.drawing”为第三层，打开后可以看到在第三层的布线结果。



相应的层 1 为 “met1.drawing”，层 2 为 “met2.drawing”。

关闭 klayout，打开编辑器，将全局布线参数设置如下。这个参数表示，尽量不要在层 1，2 上布线。设置完后保存并关闭编辑器。

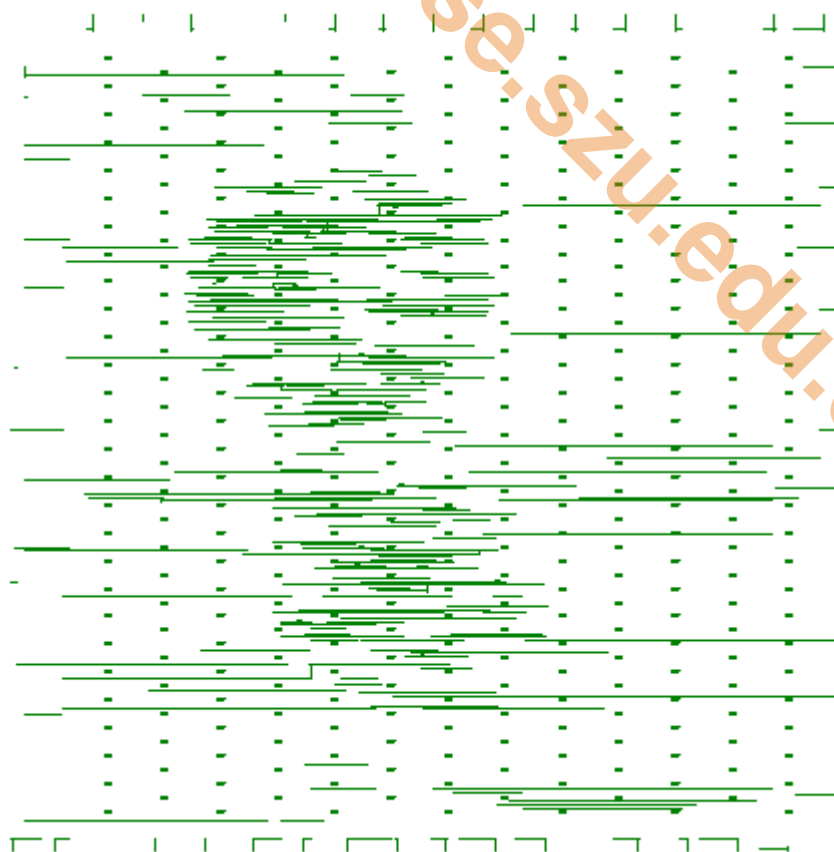
```
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$ gedit Controller/groute.openroad_1.1.0.tcl
```

```
fastroute -output_file $RESULT_PATH/route.guide \  
-max_routing_layer $MAX_ROUTING_LAYER \  
-unidirectional_routing true \  
-capacity_adjustment 0.1 \  
-overflow_iterations 200 \  
-layers_adjustments {{1 1} {2 1}} \  
-verbose 2
```

重新运行布线命令，并且更换图层后打开层 3。

```
lc3@lc3-virtual-machine:~/Desktop/iFlow/scripts$ ./run_flow.py -d Controller -s  
groute,droute,layout -f sky130 -t HS -c TYP
```

调整参数后，第三层的布线结果如下。



可以看到，相对于原参数，现在在第三层有更多的互连线了。

小结

布线的主要过程分为两部分，首先是将版图划分为多个大网格，以网格为单位布线，这一阶段为全局布线。然后再在全局布线的网格结果上进行更详细的布线，这一阶段称为详细布线。

在实验七布线中，我们学习了全局布线和详细布线的基本概念，了解了布线的基本指标。学会了通过工具生成一个设计的布线结果，并能知道布线结果的指标数据。通过调整全局布线参数，生成并对比对应的布线结果，大致了解了一些布线参数的作用。

效果。