GDS Processing for Cadence in Matlab

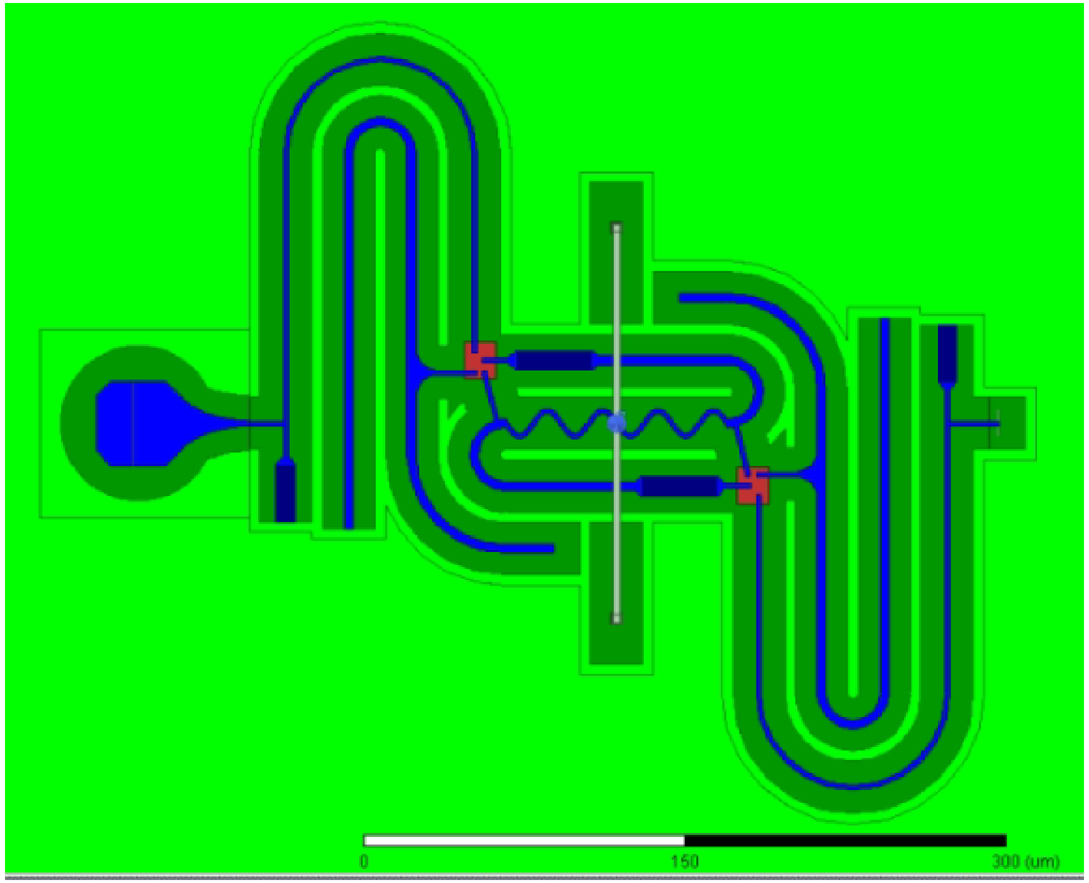
By Zainulabideen Khalifa

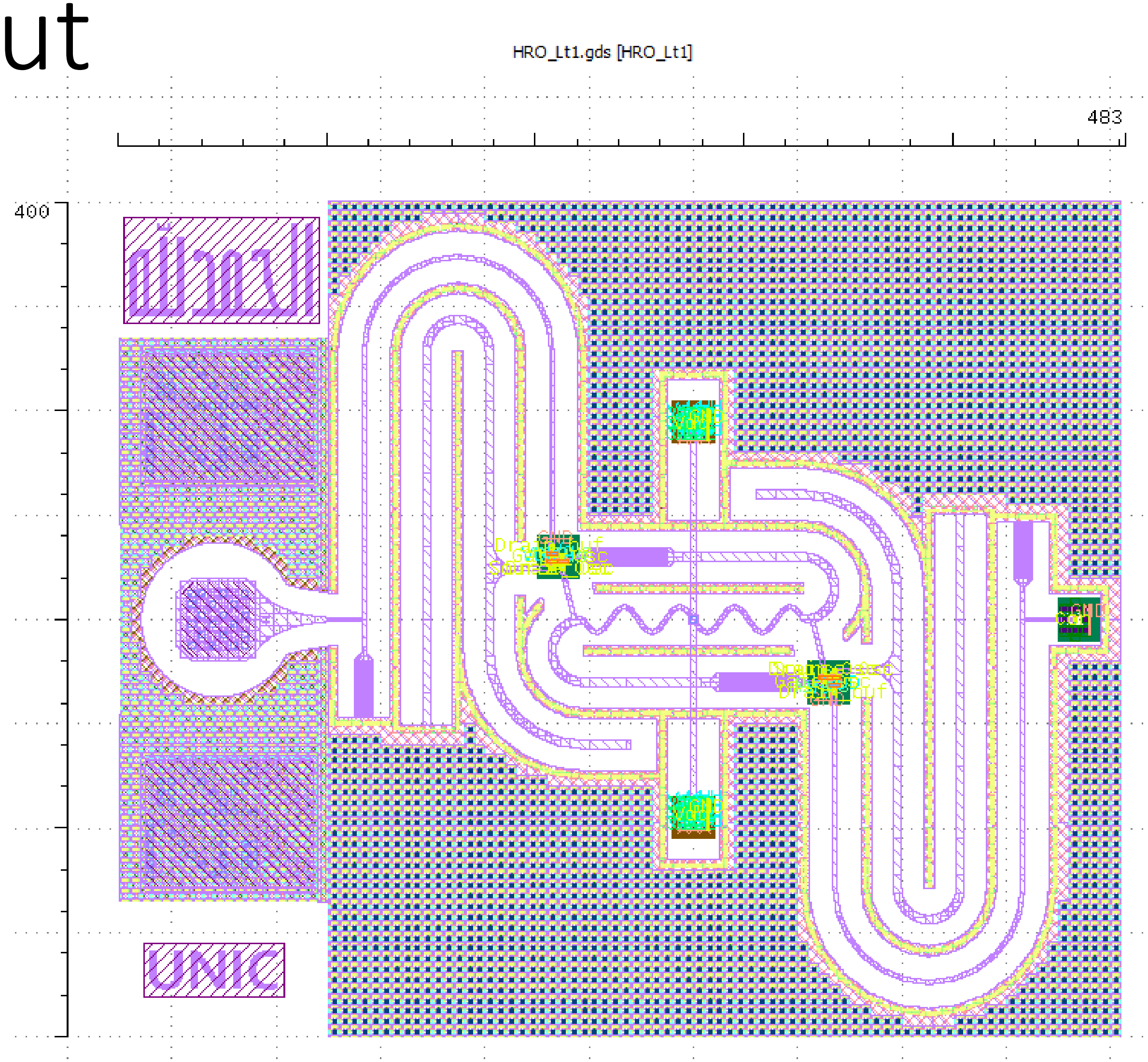
Ph.D. Student at the University of Michigan, Ann Arbor

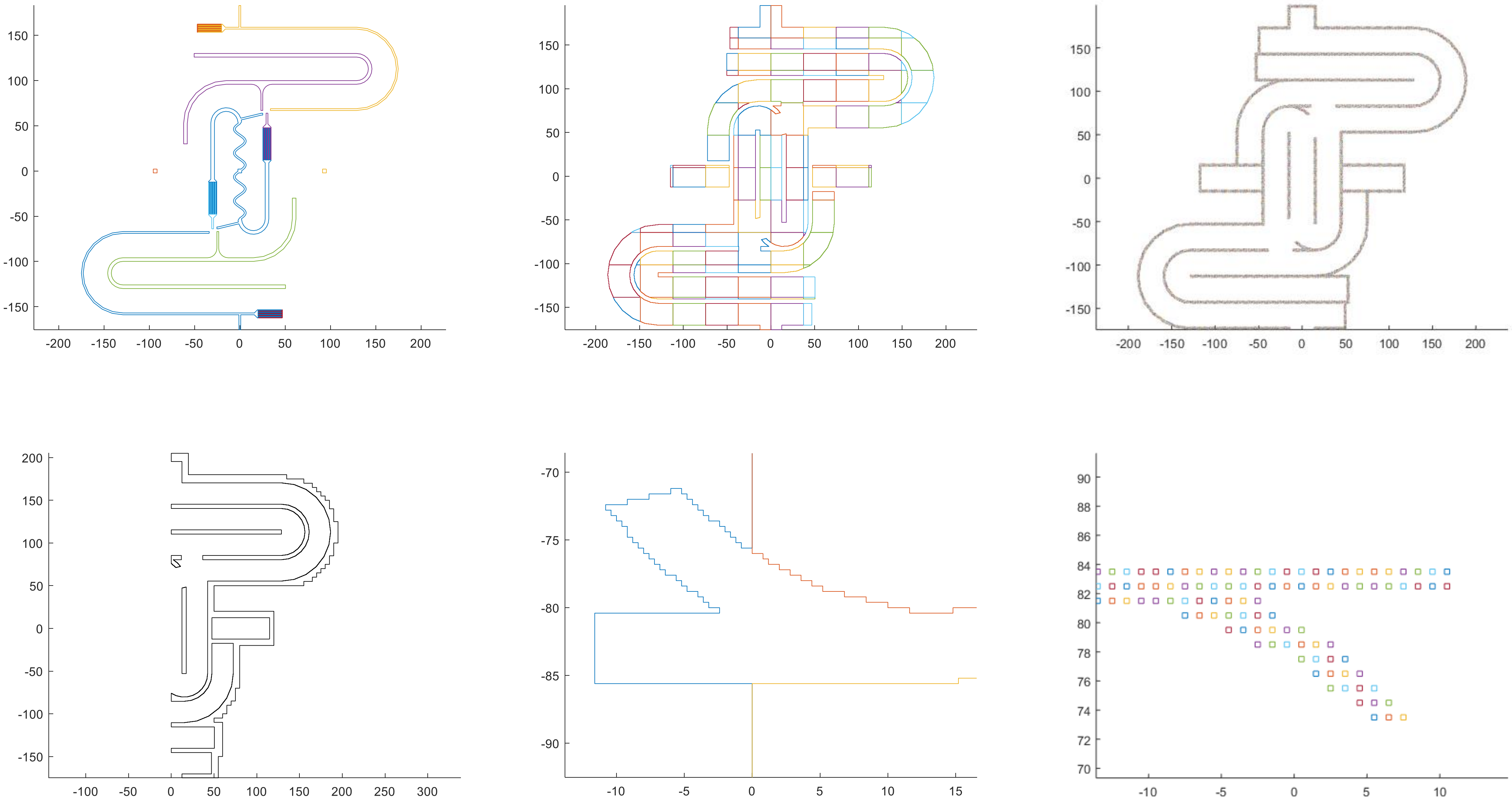
Contact: zainkh@umich.edu

9/9/2020

HFSS → Cadence Layout






150
100
50
0
-50
-100
-150
-200
-150
-100
-50
0
50
100
150
200
200
150
100
50
0
-50
-100
-150
-100
-50
0
50
100
150
200
250
300
-70
-75
-80
-85
-90
-10
-5
0
5
10
15
90
88
86
84
82
80
78
76
74
72
70
-10
-5
0
5
10

Overview

Each code will have the following general structure:

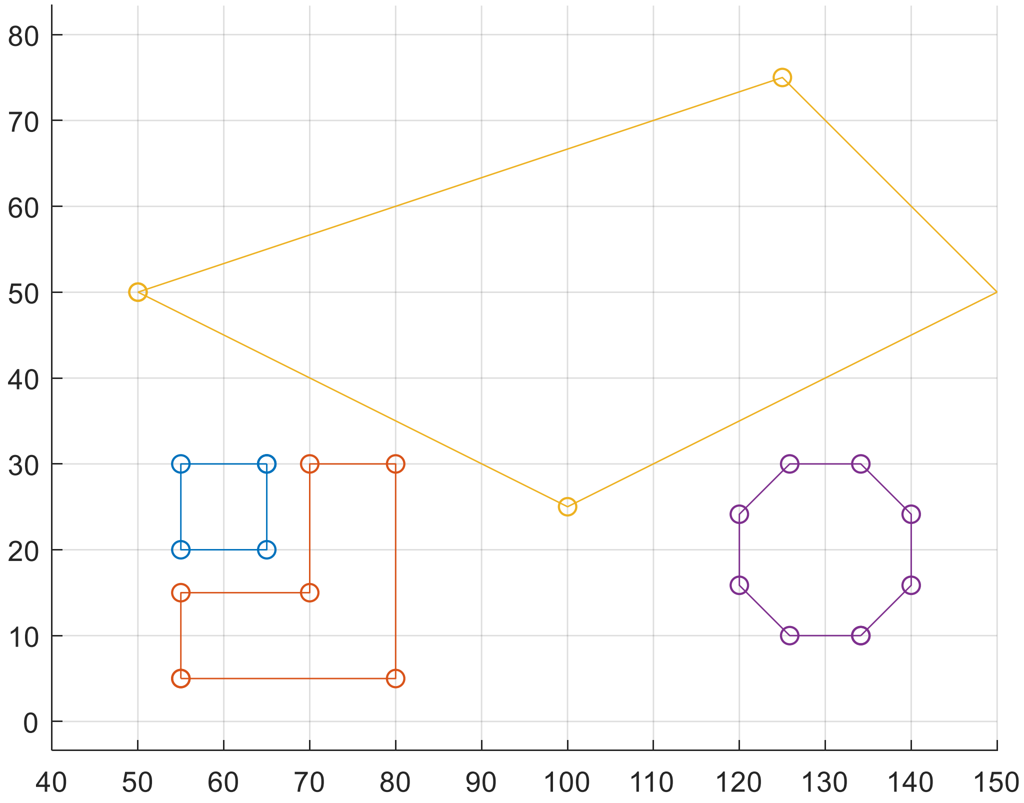
1. Import GDS libraries from HFSS or Cadence
2. Perform the needed operations
3. Assign layer and data type numbers (mapping).
4. Export the GDS library.

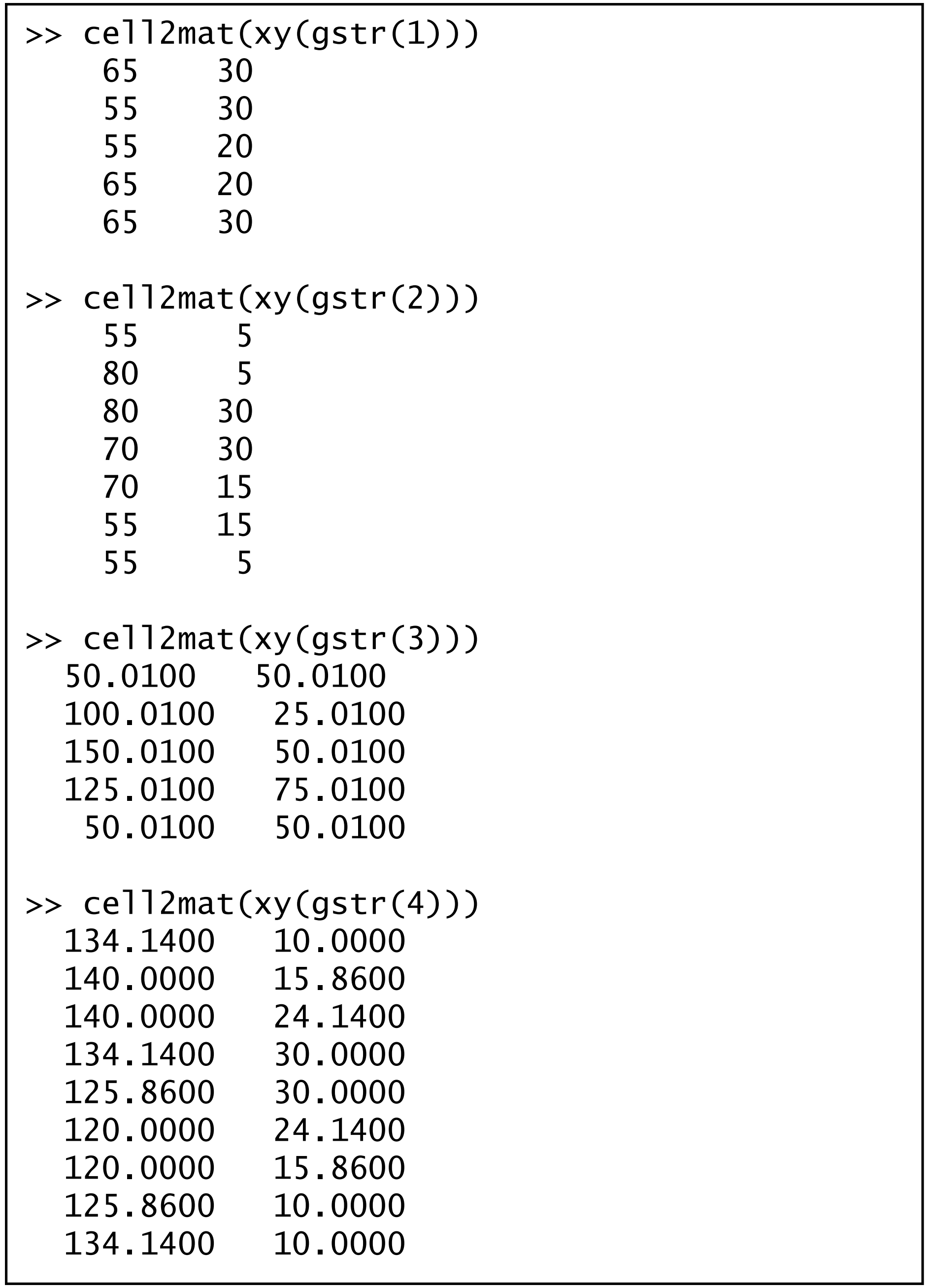
GDSII File Basics

- gds_library
  - gds_structures
    - gds_elements
      - Different types: “boundary” , “SREF” or … etc
      - Layer number
      - Data type number

```
>> glib = in_glib

glib is a GDSII library:

   Library name  :  HRO_M8_HFSS.gds
   Database unit :  1e-09 m
   User unit     :  1e-06 m
   Structures    :  1
        1 ... HRO_M8_HFSSstruct (11)

>> gstr = glib(1)

gstr is a GDSII structure with 11 elements:

   sname = HRO_M8_HFSSstruct
   cdate = 120-7-21, 13:31:41
   mdate = 120-7-21, 13:31:41

>> gelm = gstr(1)

gelm is a GDSII element:

   Type: boundary (1)
   layer = 1
   dtype = 0

>> XY = gelm(1)

ans =

    2.1000   -2.1000
    2.1000    0.2630
    2.5290    0.5770
    3.0150    0.9430
    ...
```

GDSII File Basics



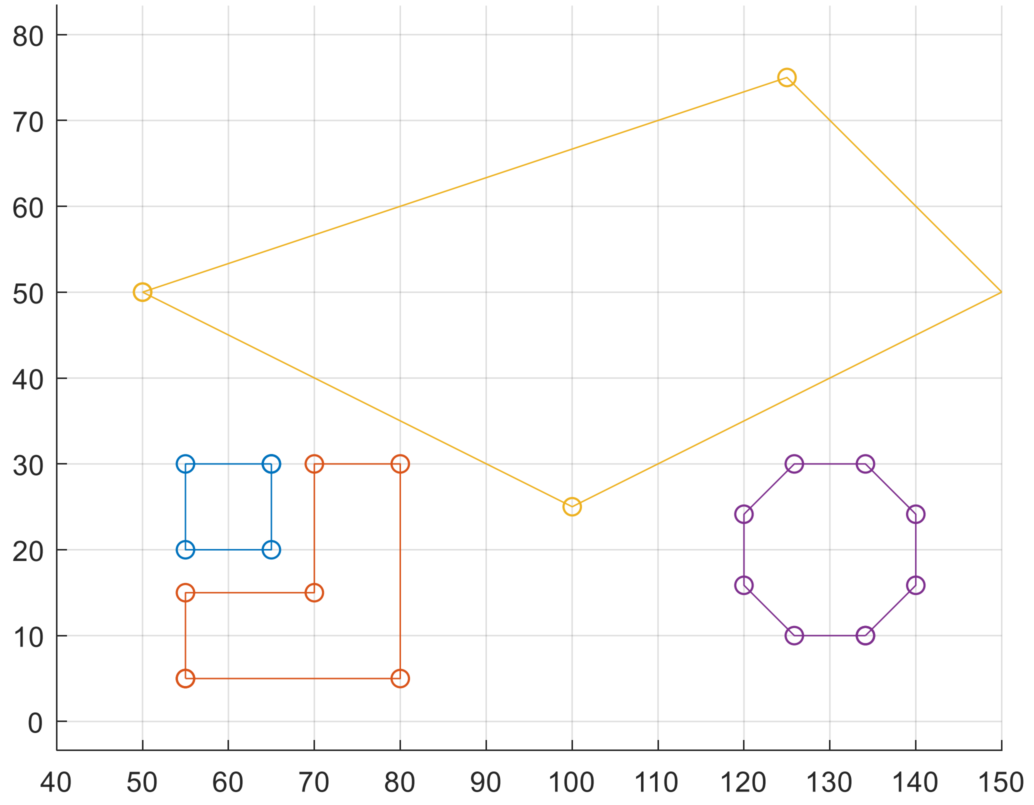


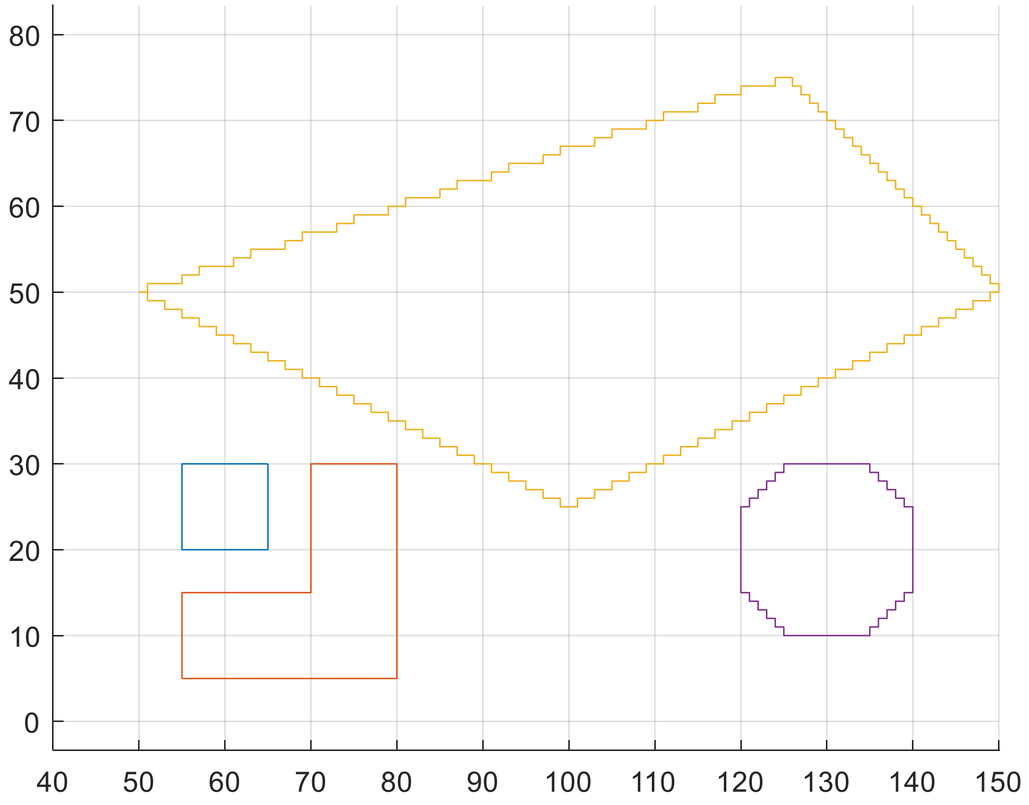
```
>> cell2mat(xy(gstr(1)))
    65    30
    55    30
    55    20
    65    20
    65    30

>> cell2mat(xy(gstr(2)))
    55     5
    80     5
    80    30
    70    30
    70    15
    55    15
    55     5

>> cell2mat(xy(gstr(3)))
   50.0100   50.0100
  100.0100   25.0100
  150.0100   50.0100
  125.0100   75.0100
   50.0100   50.0100

>> cell2mat(xy(gstr(4)))
  134.1400   10.0000
  140.0000   15.8600
  140.0000   24.1400
  134.1400   30.0000
  125.8600   30.0000
  120.0000   24.1400
  120.0000   15.8600
  125.8600   10.0000
  134.1400   10.0000
```

Main Operations

- The goad is to manipulate the layout to pass DRC and maintain your structure from EM point of view.
- Operations:
  - (Discretize): Discretize to correct for allowed angles and minimum grid.
  - (minWidth): Correct for minimum Width/Space.
  - Generate vias between metals
  - Fill metal block fillings like (Grid_Wall)
- With these operations, you can convert any design from HFSS to Cadence without any DRC errors.

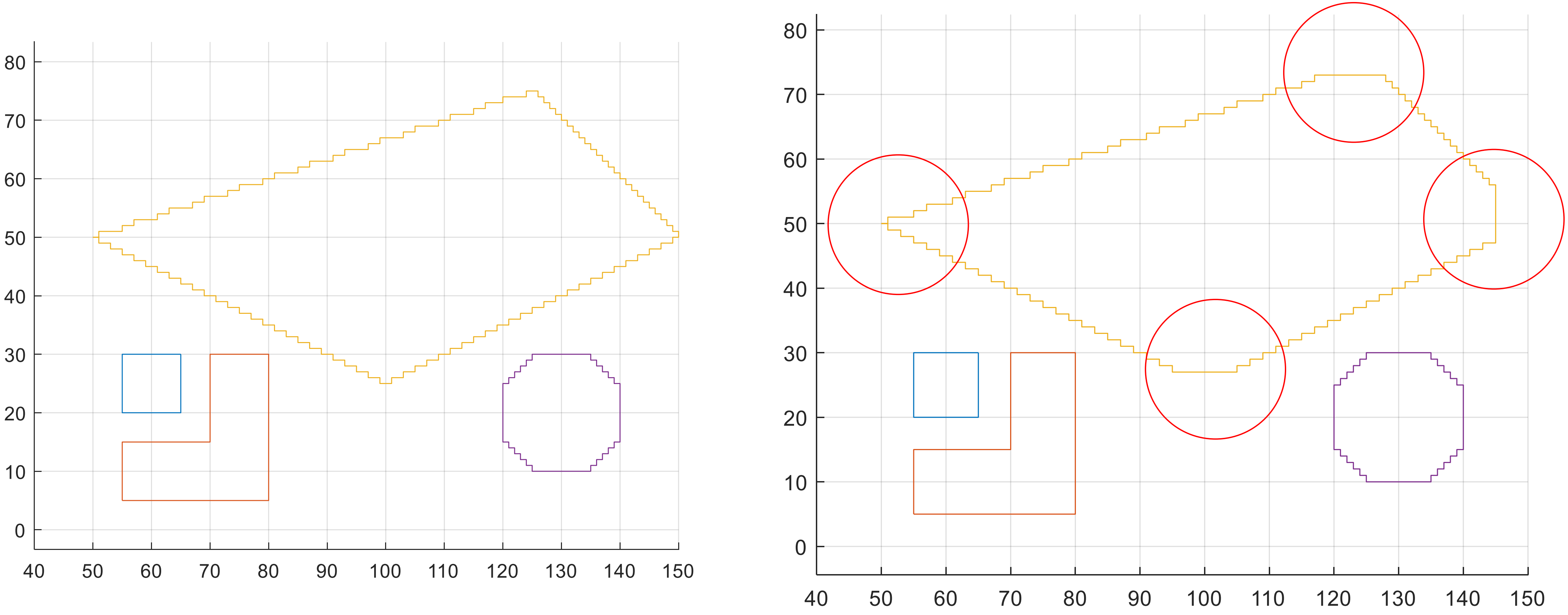
Operations - Discretize





The shapes after performing the function Discretize

Operations - minWidth

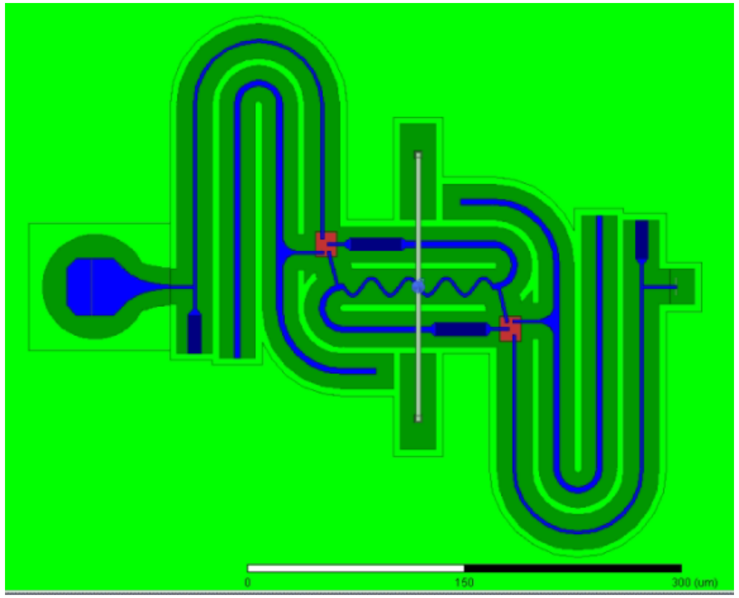


Appling minWidth on the discretized shapes will trim pointy head and might miss some. (why?!)

Functions

```
% % ----------------------General Functions--------------------------------
% function GDS_plot(igds,str)
% function [ogstr] = GDS_MATH(ig1,ig2,operation,units)
% function [ogstr] = GDS_Merge(igstr,units)
% function [ogstr] = GDS_combine_gstrcells(igstr)
% function [ogelm] = GDS_Create_box(d,c)
% function [ogelm] = GDS_Create_Octagonal(d_side, center, max45)
% function [ogstr] = GDS_Create_Grid(igstr,NxN)
% function [ogstr] = GDS_Split_gstr(igstr,NxN,units)
% function [RC,Center] = GDS_Mosaic_calc(igelm,Mosaic)
% function [ogstr] = GDS_Mosaic(igds,Mosaic,RC,Center)
% function [ogds] = GDS_Shift(igds,shift)
% function [ogds] = GDS_reset(igds,info)
% function [iglib] = GDS_auto_rename_glib(iglib,sname)

% % ----------------------Layout specific Functions------------------------
% function [ogstr] = GDS_checkvias(igstr,d)
% function [ogstr] = GDS_Mosaic_intersections(igds,Mosaic_gstr,units)
% function [ogstr] = GDS_Mosaic_imprint(block_gstr,bbox_block,igstr,bbox_gstr,units)
% function [ogstr]    = GDS_Discretize_gstr(igstr,minGrid,units)
% function [ogelm]    = GDS_minWidth_gelm(igelm,minWidth,minGrid,SmallestWireWidth,units)
% function [ogstr]    = GDS_minWidth_gstr(igstr,minWidth,minGrid,SmallestWireWidth,units)
% function [xo,yo]    = Discritize_2P(X,Y,minGrid)
% function [XY,count] = minWidth_corr(XY,minWidth,minGrid,SmallestWireWidth,units)

% % ----------------------ST55 Functions-----------------------------------
% function [info] = GDS_ST55(str)
% function [ogstr] = GDS_ST55_Generate_tileNot(igstr)
```

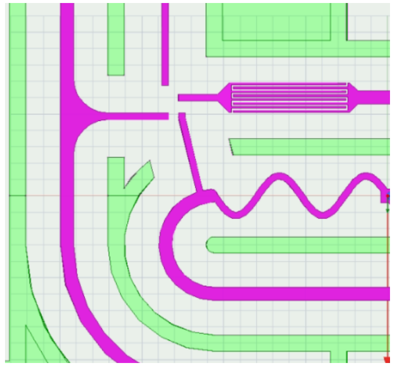
Needed layers from HFSS

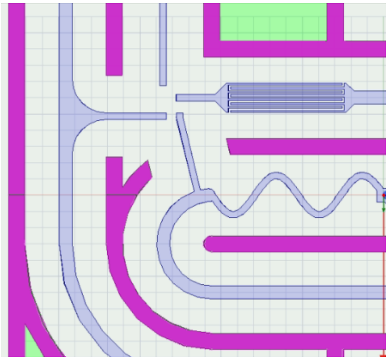


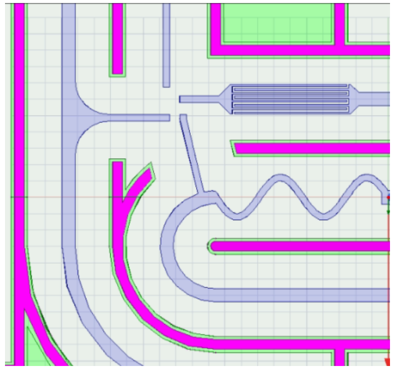
1. Top Metal or your line metal layer
2. Shielding Walls
3. Shrunk version of shielding walls for via filling
4. Grid_Wall filling layers (connected with shielding)
5. tileNot layer

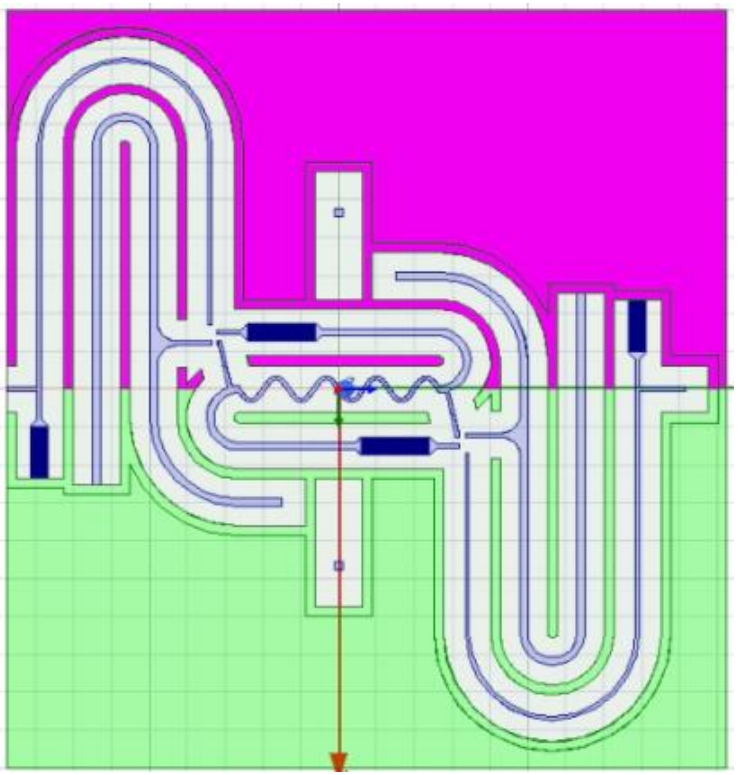
- Keep in mind that all layer must have the same reference point. This is so they will fit on each other in Cadence.

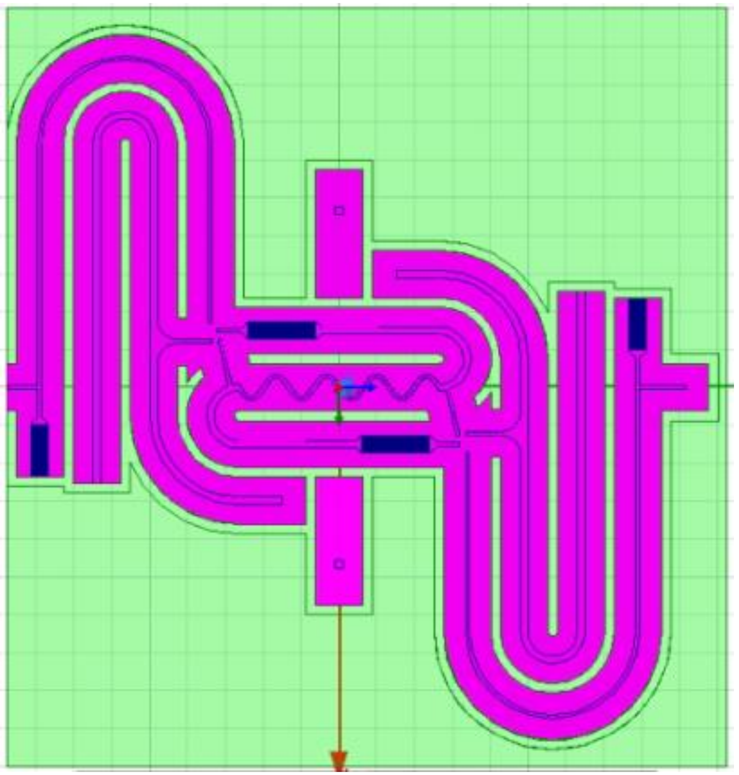
Needed layers from HFSS





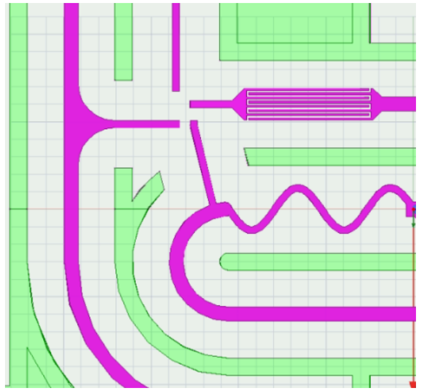


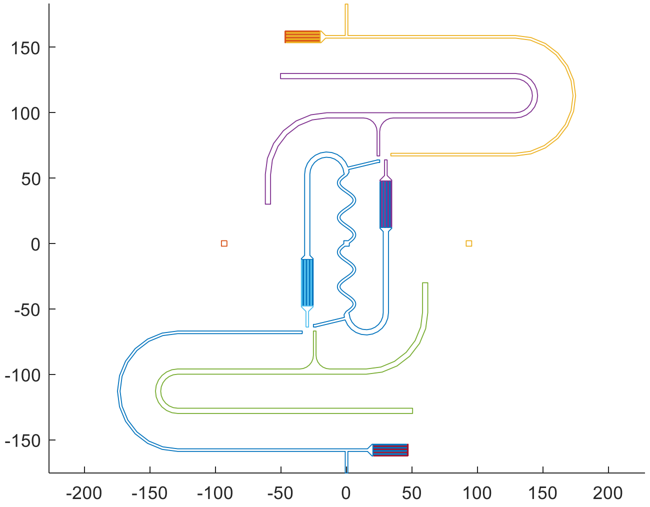


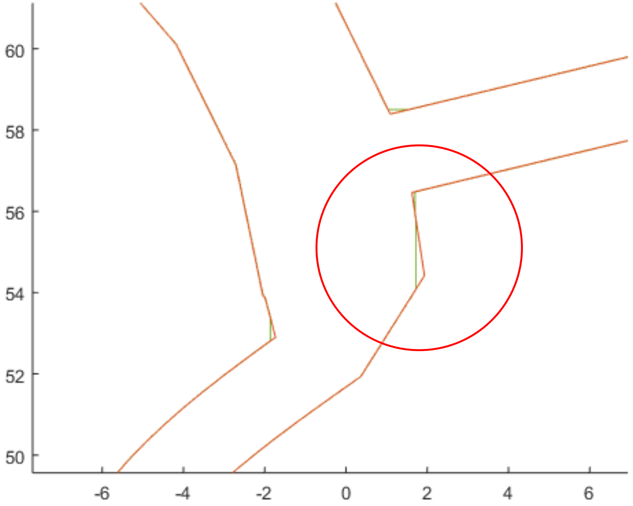


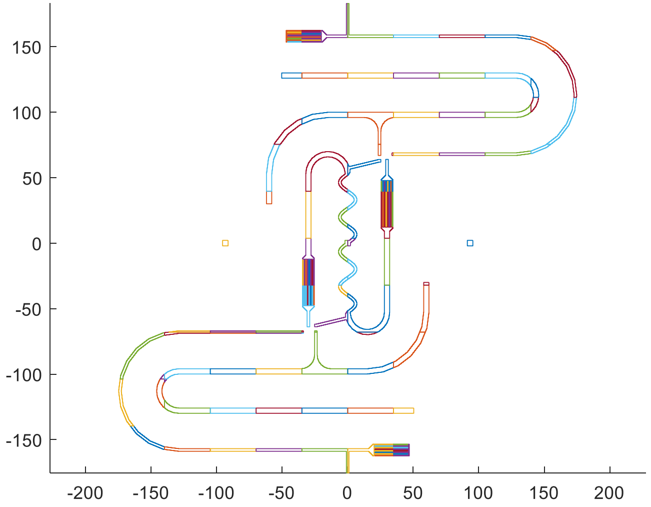
Processing M8

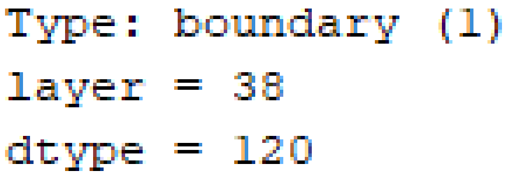
1. Export the GDS layer from HFSS
2. Import in Matlab
3. Discretize and minWidth
4. Split (Why ?!)
5. Reset layer and dtype
6. Export











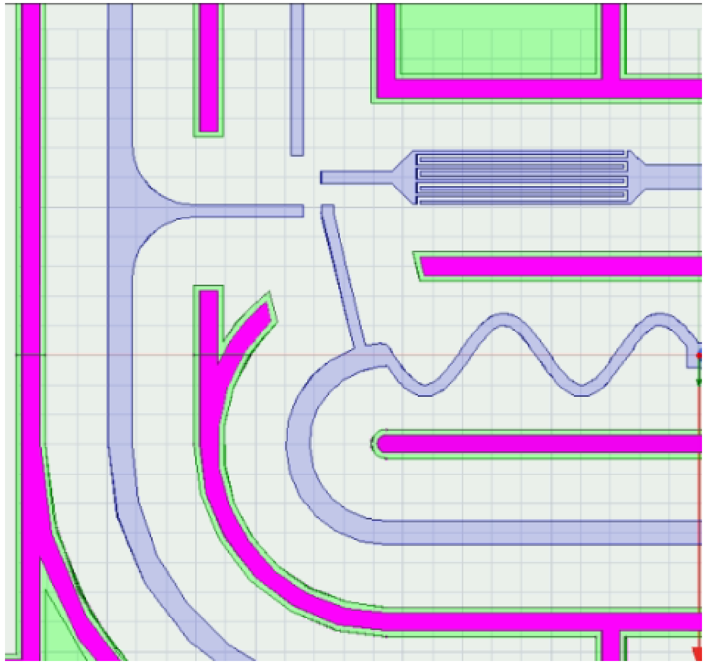
```
Type: boundary (1)
layer = 38
dtype = 120
```

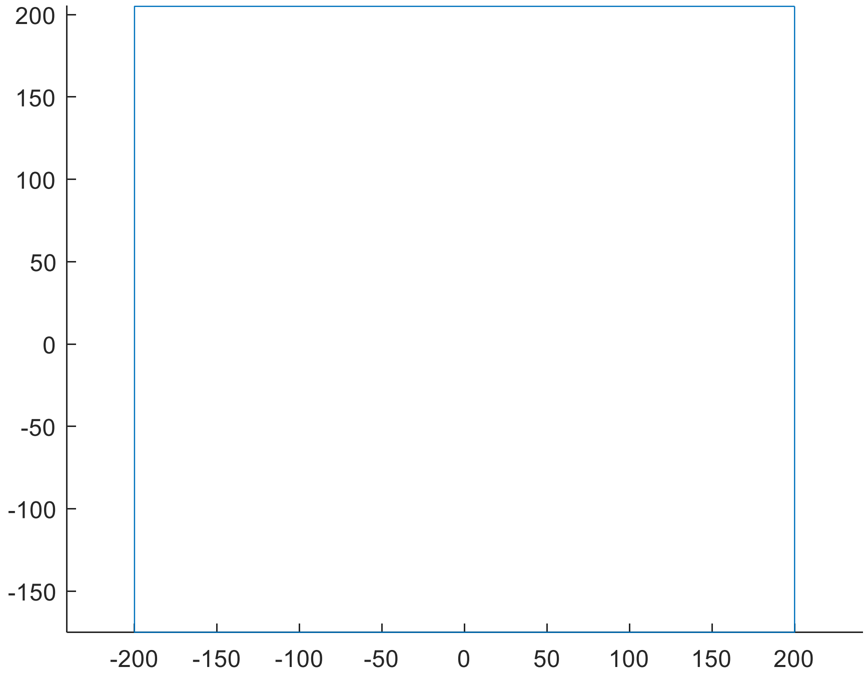
Processing Vias

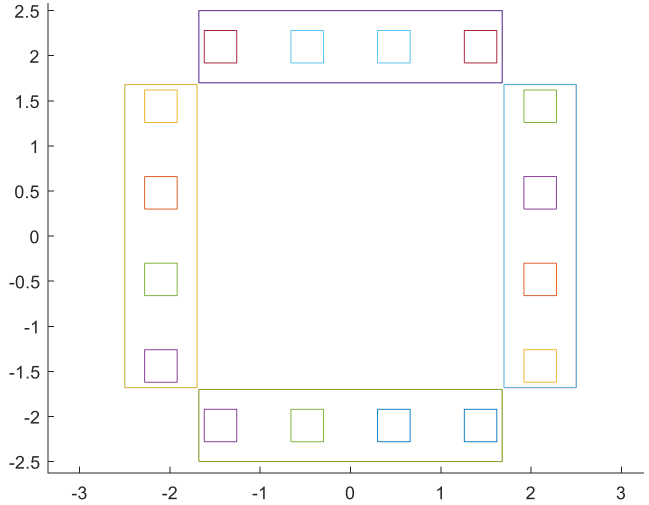
- There are two procedures for processing vias:

1. Processing vias between the shield walls and the ground layer.
2. Processing vias between the shield walls itself.

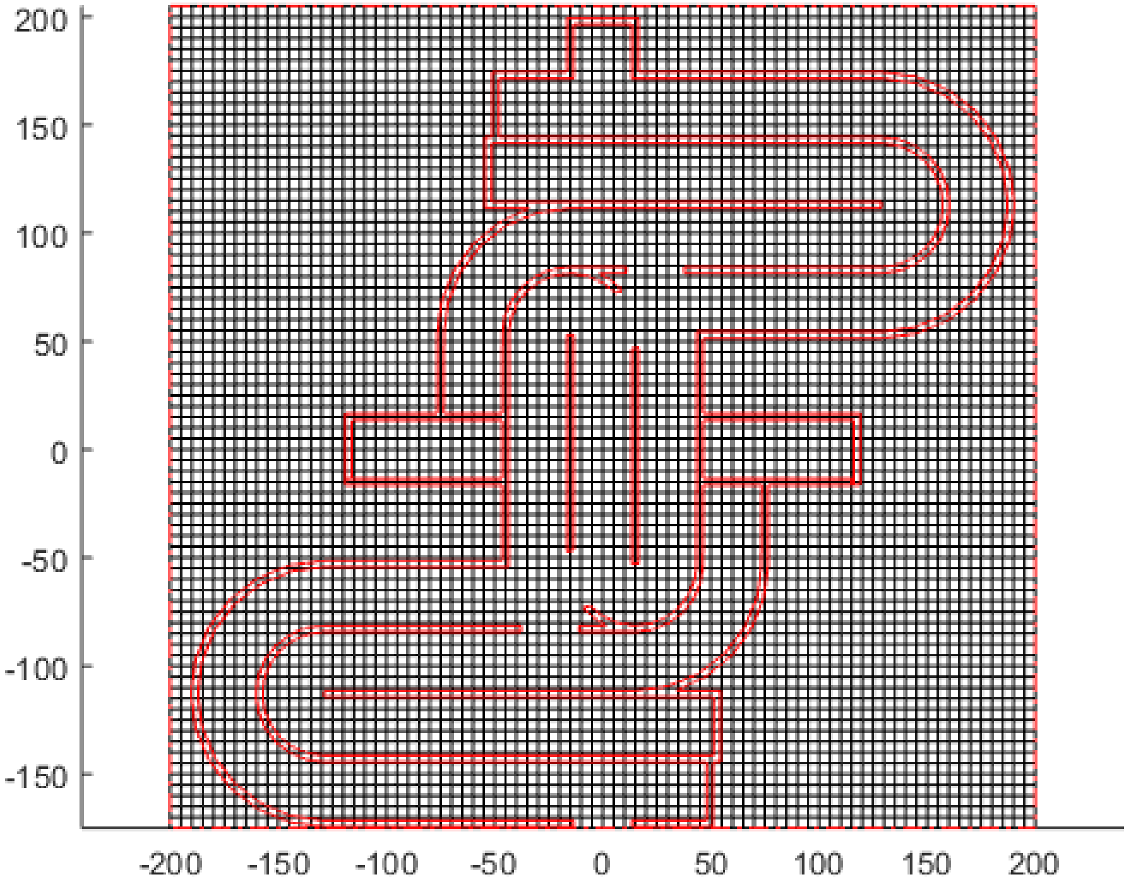
Processing Wall-Ground Vias

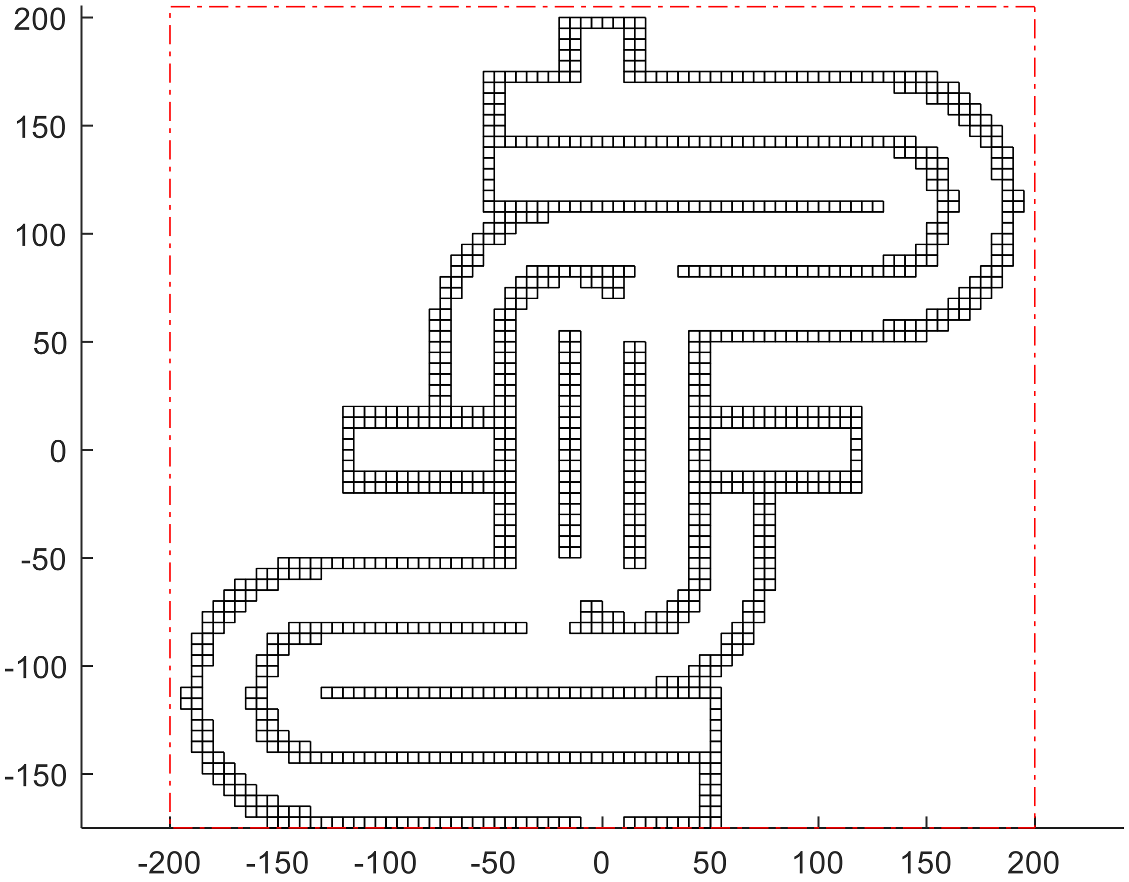




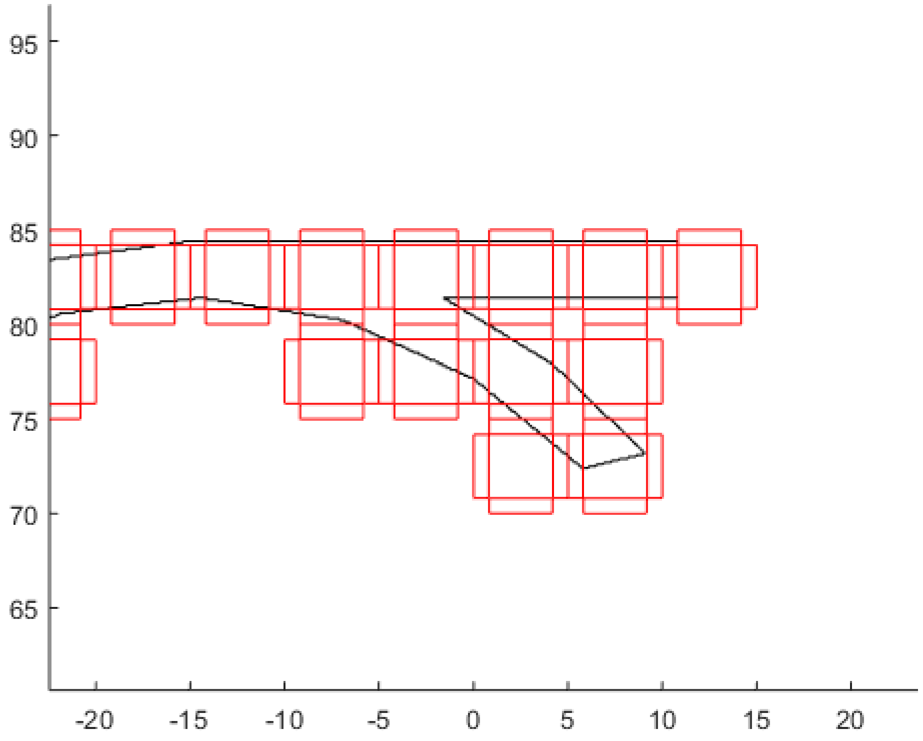


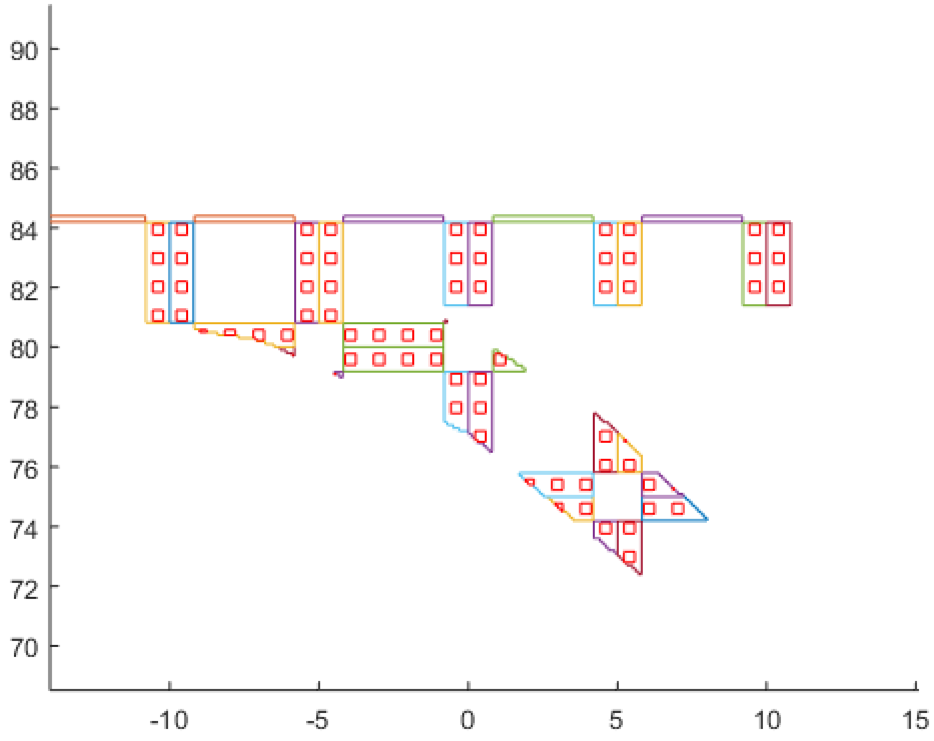
Processing Wall-Ground Vias



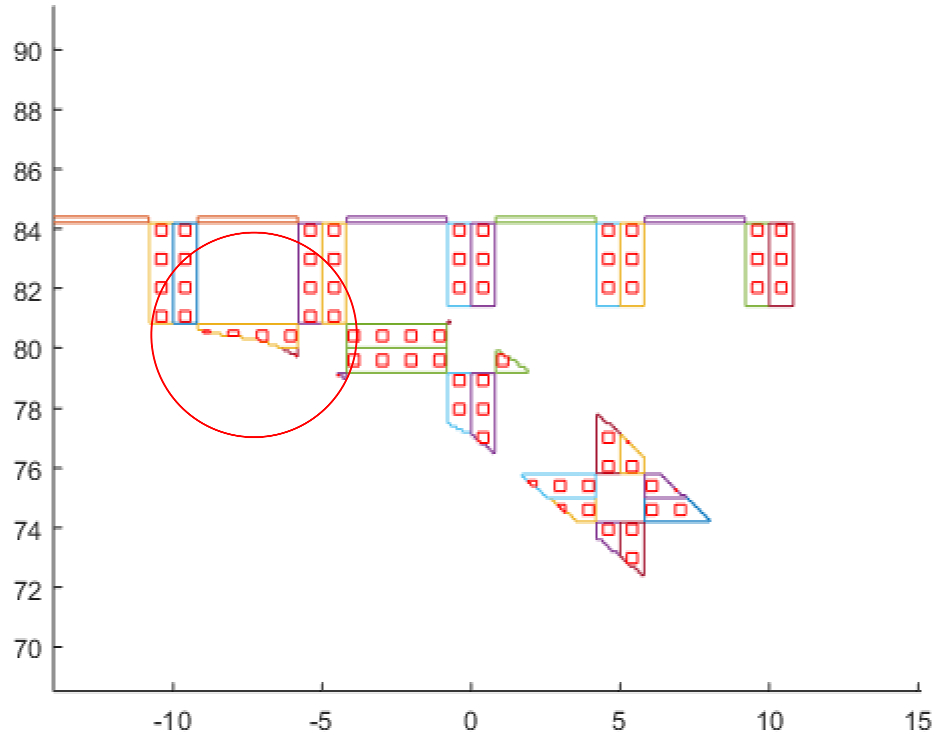


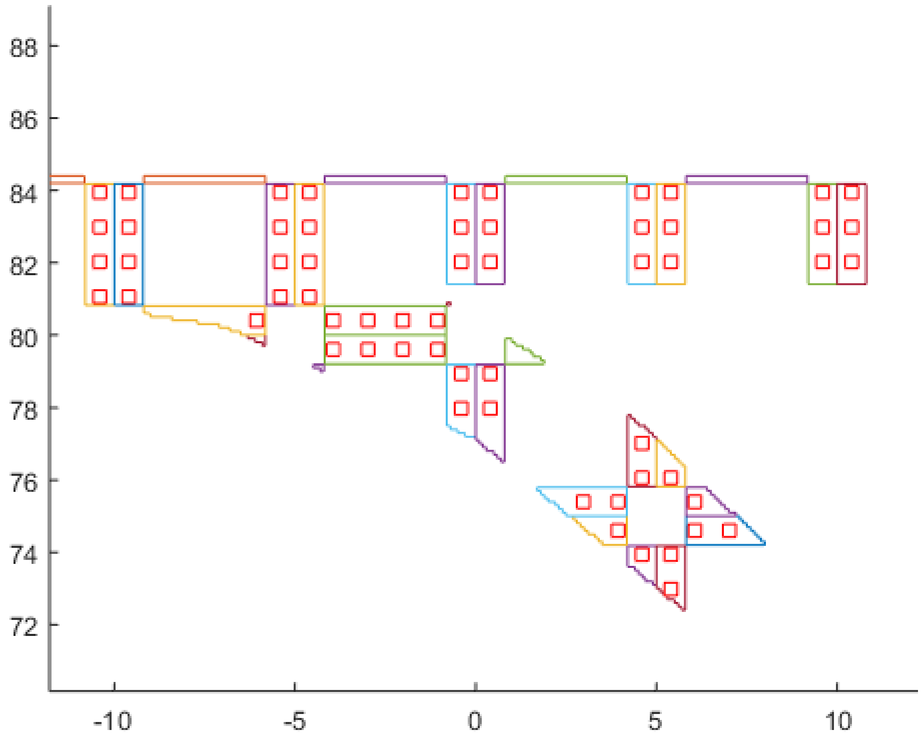
Processing Wall-Ground Vias



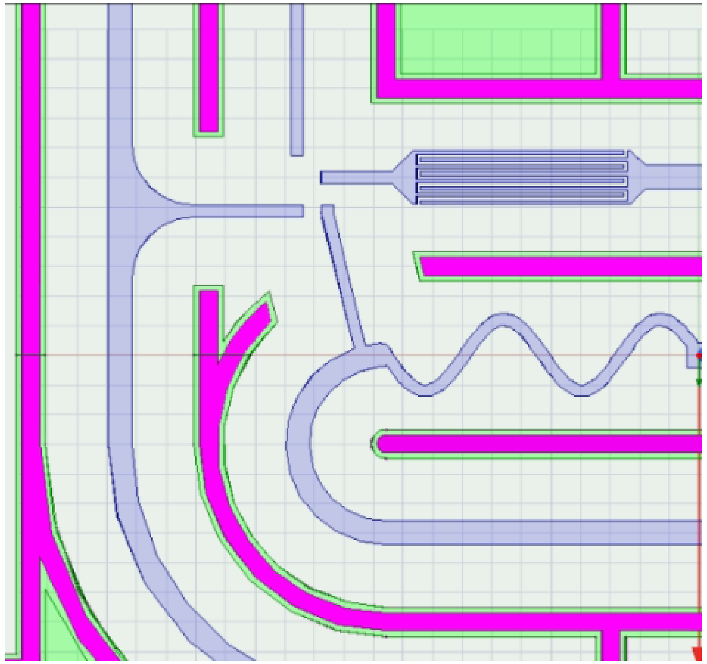


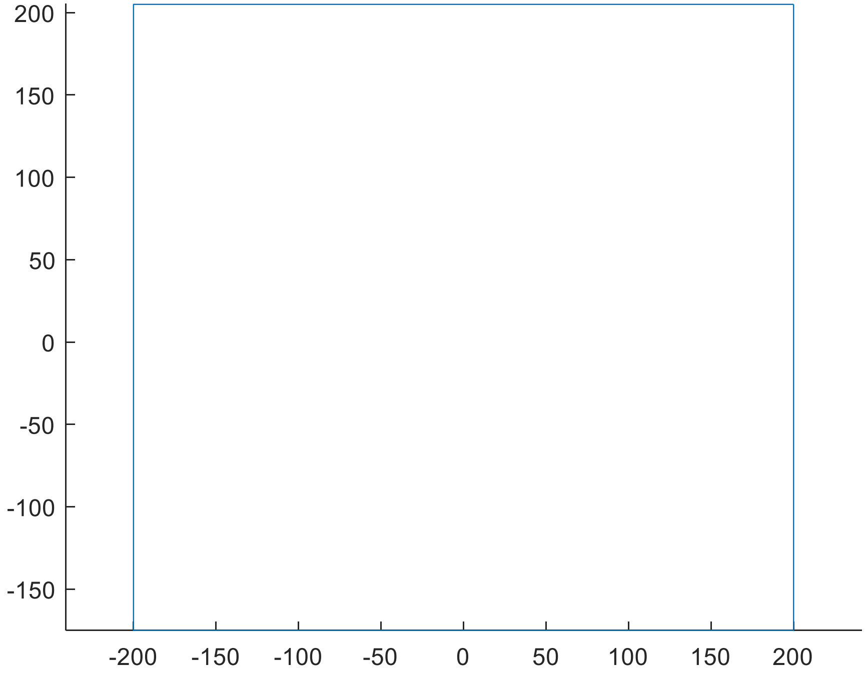
Processing Wall-Ground Vias

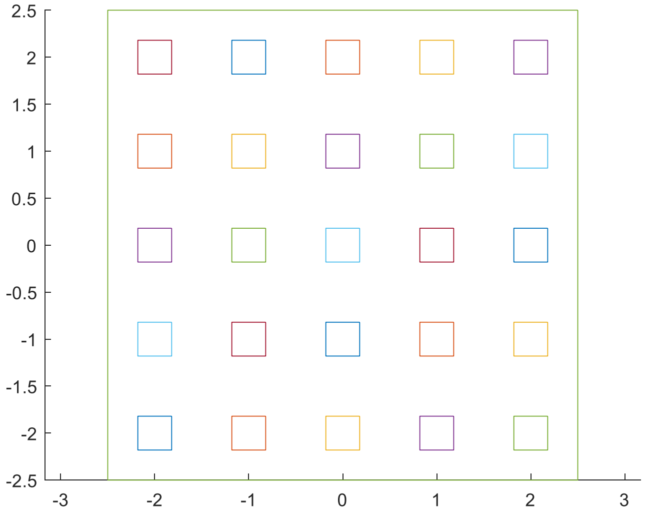




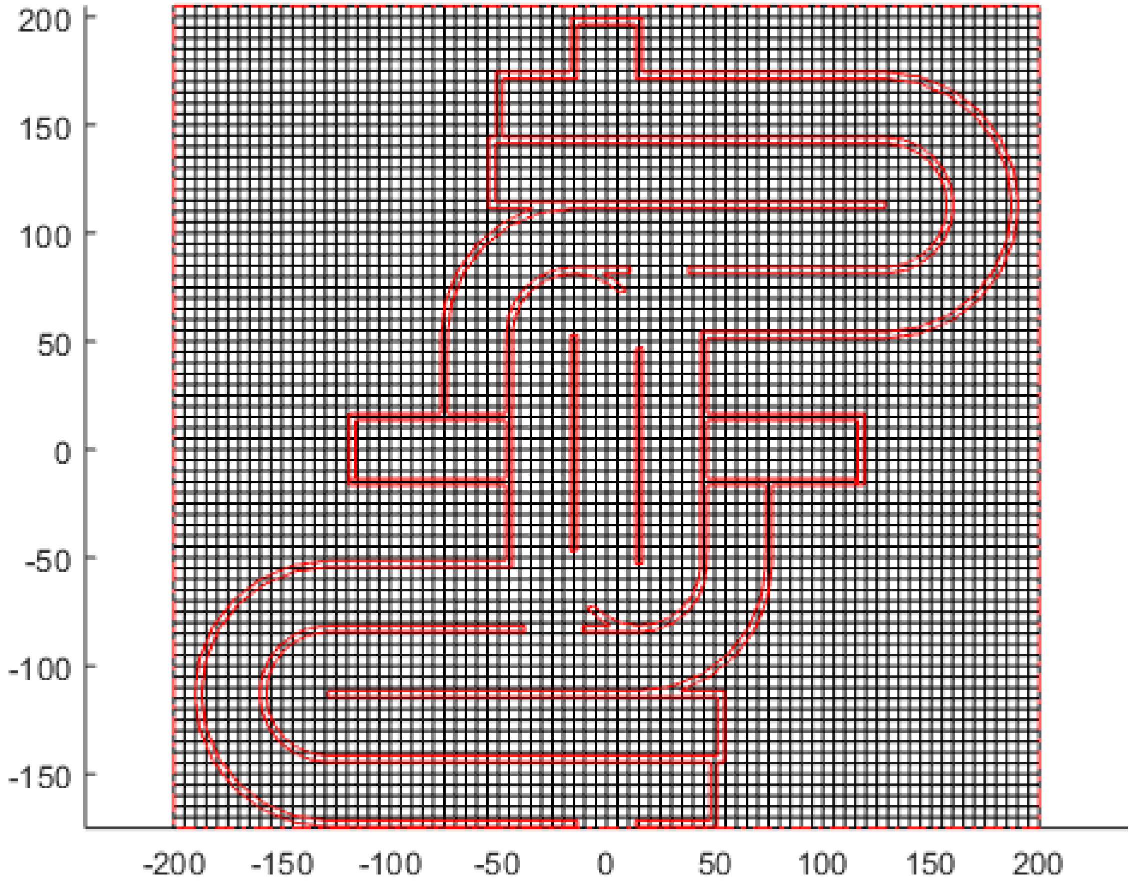
Processing Wall-Wall Vias

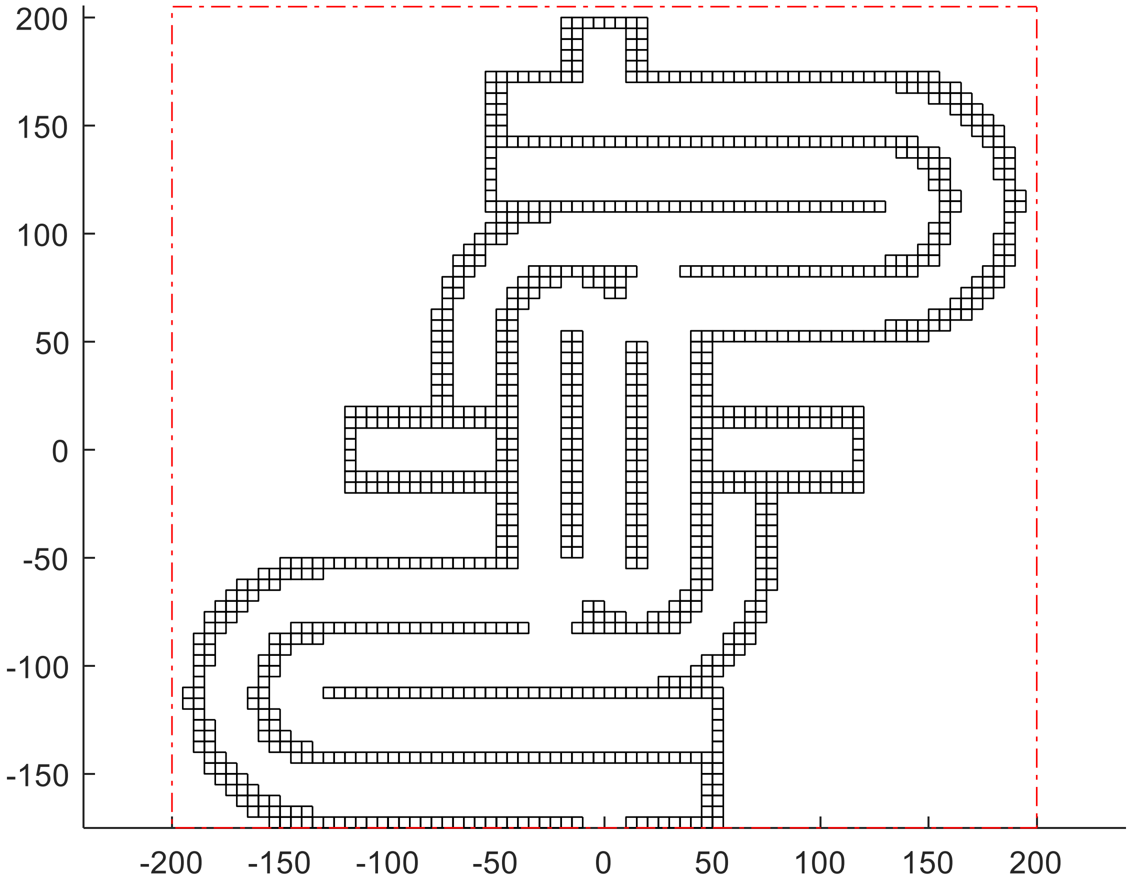




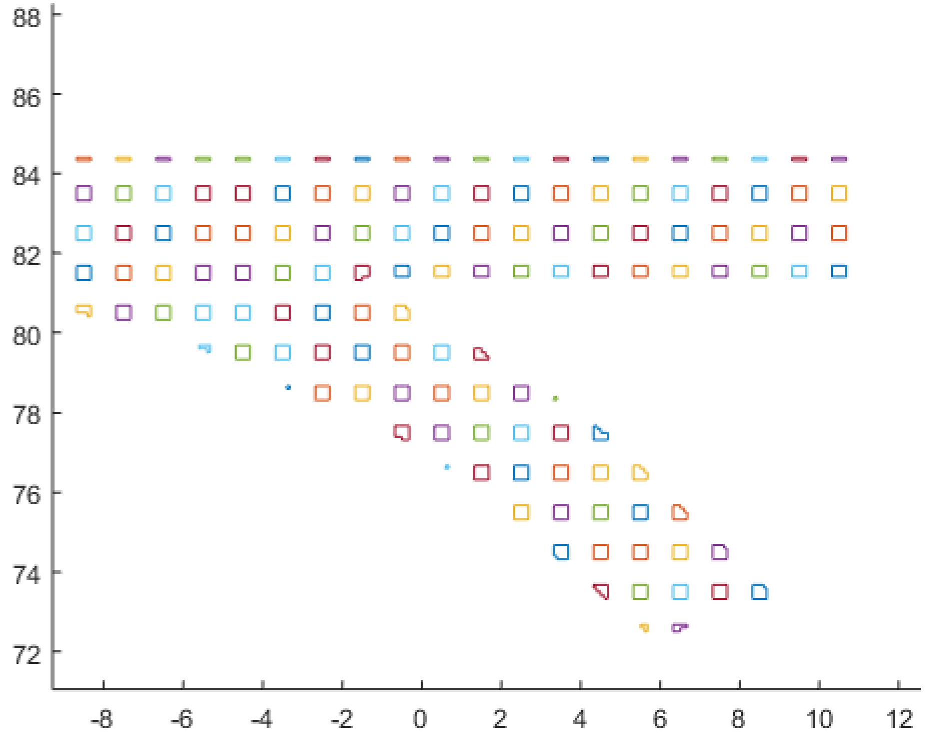


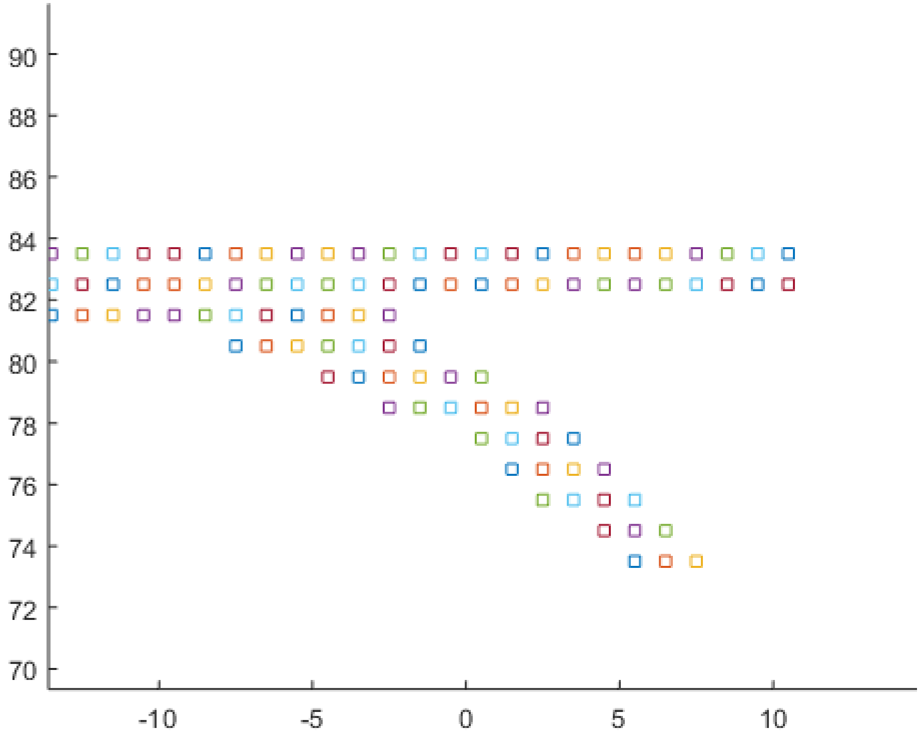
Processing Wall-Wall Vias



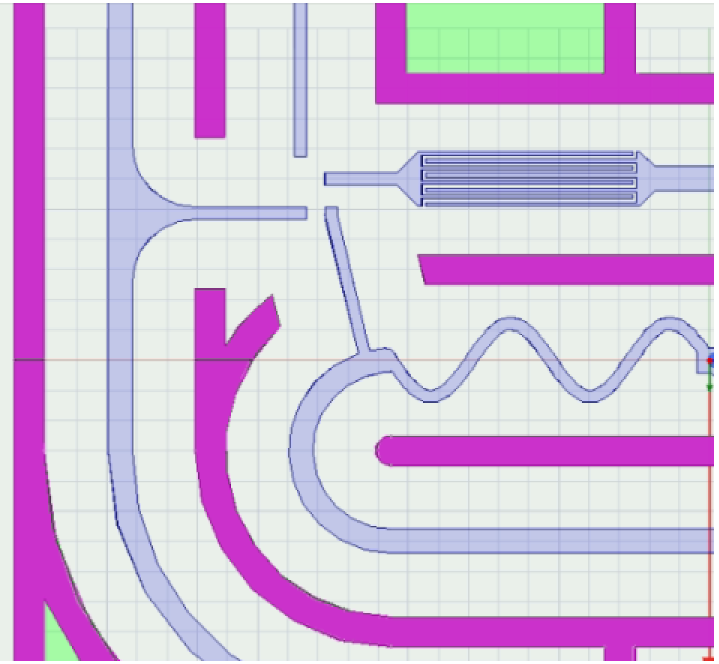


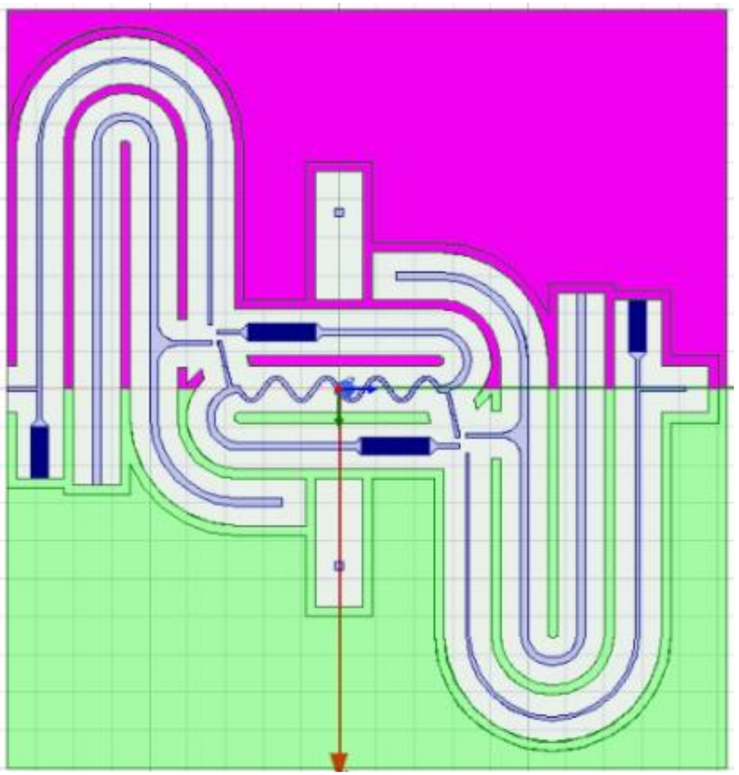
Processing Wall-Wall Vias

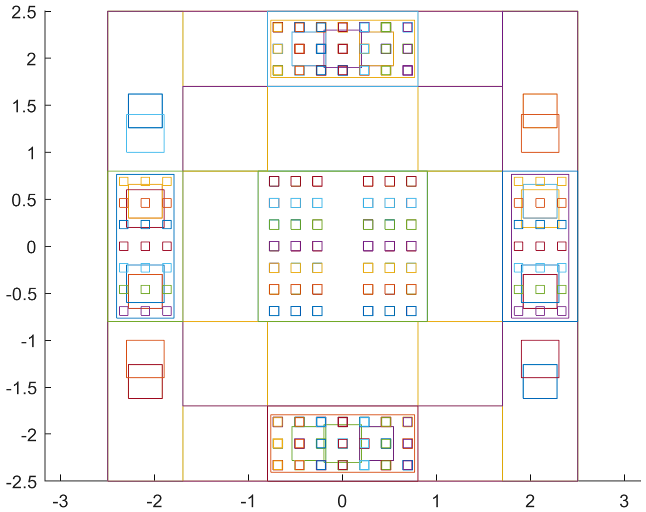




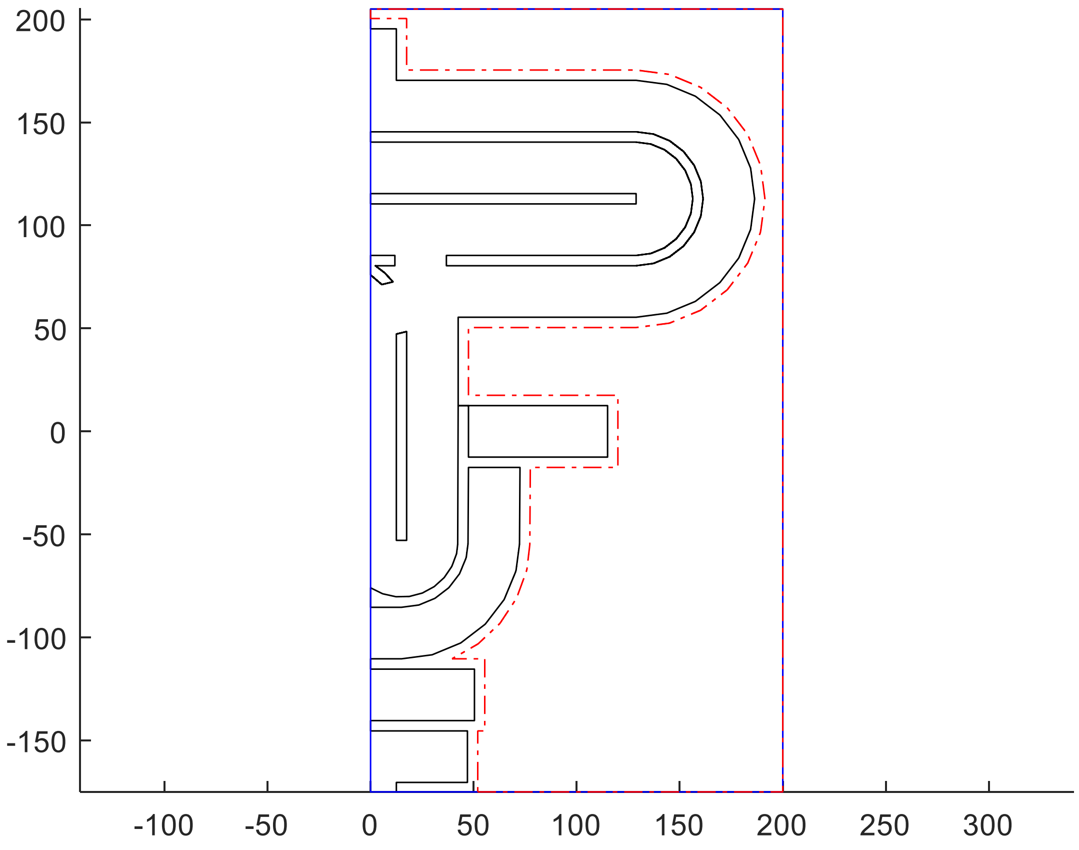
Processing Fillings and Wall Metals



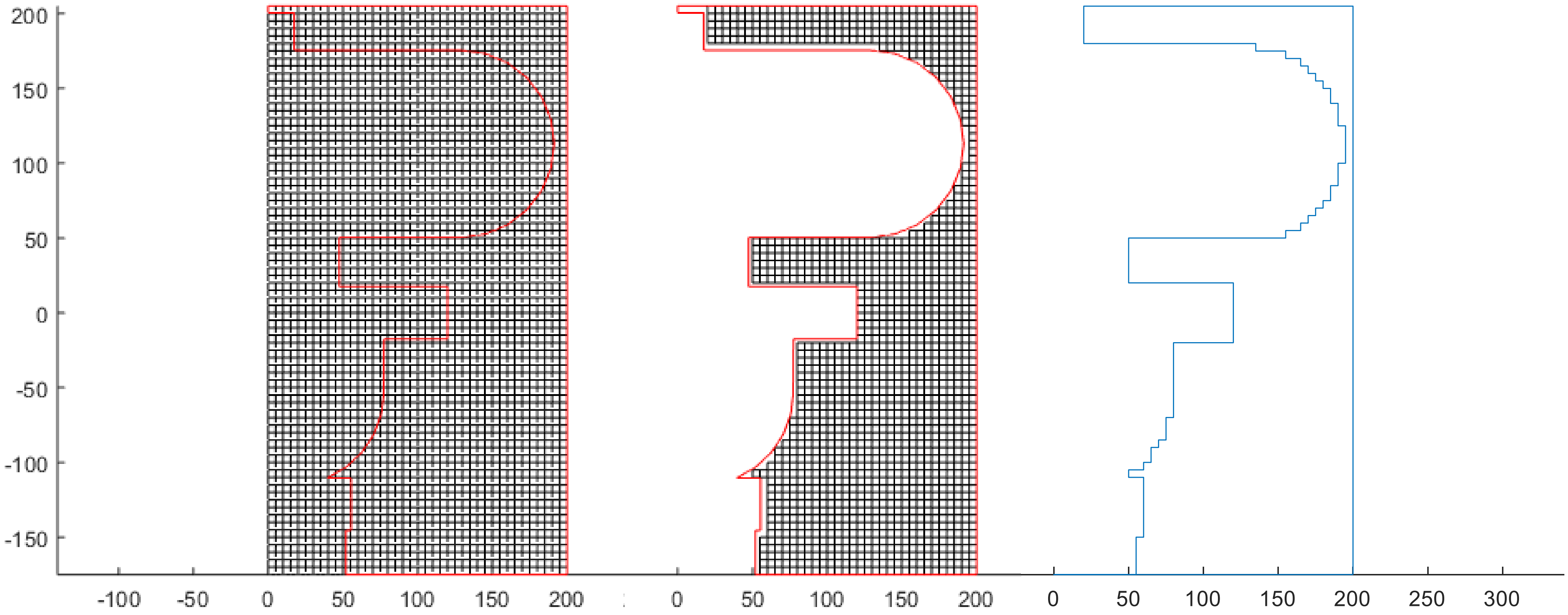




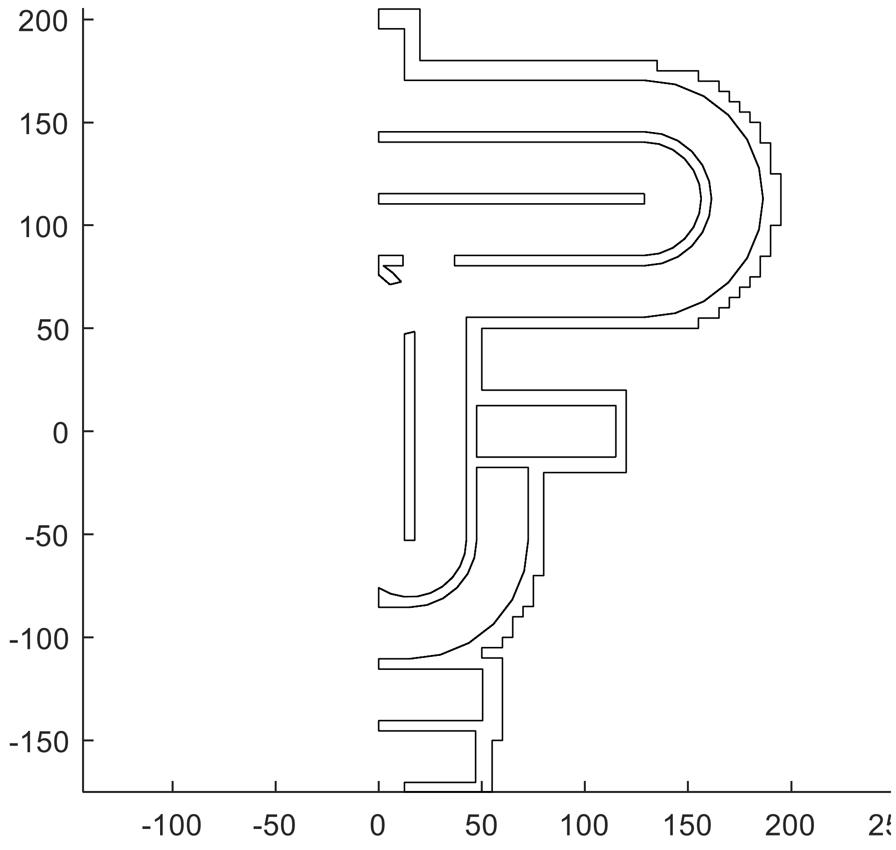
Processing Fillings and Wall Metals

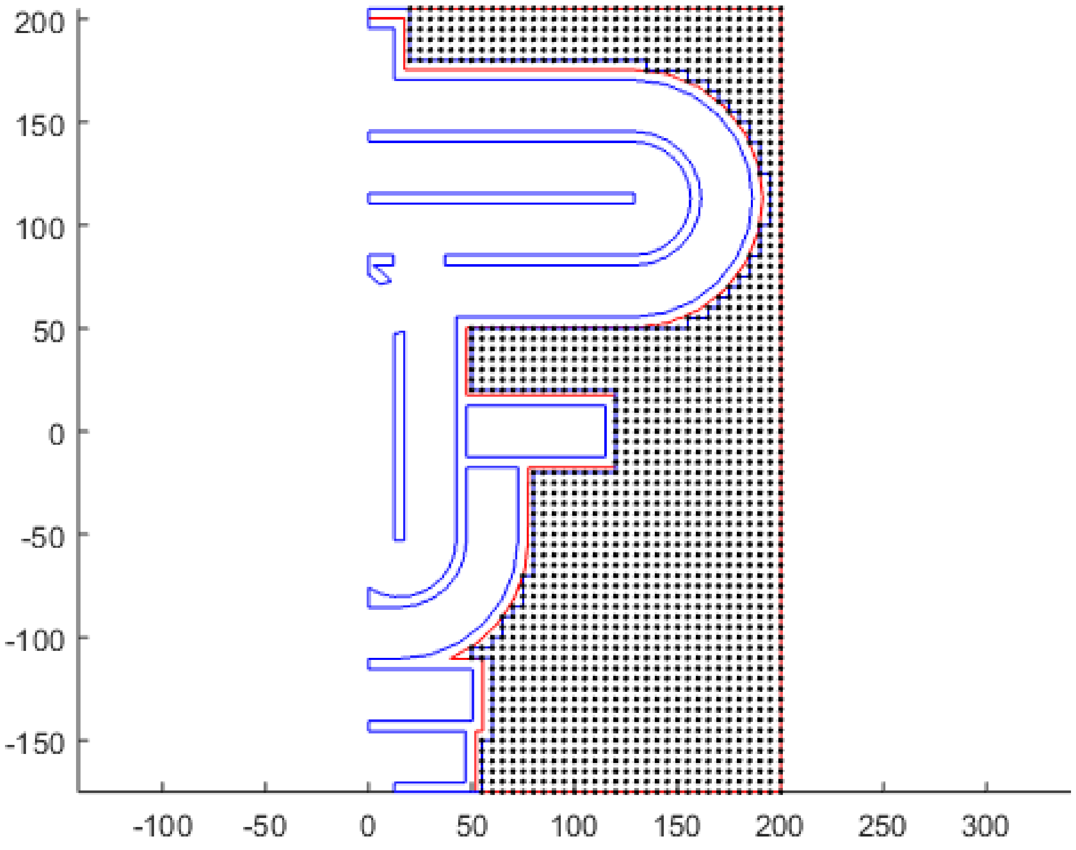


Processing Fillings and Wall Metals



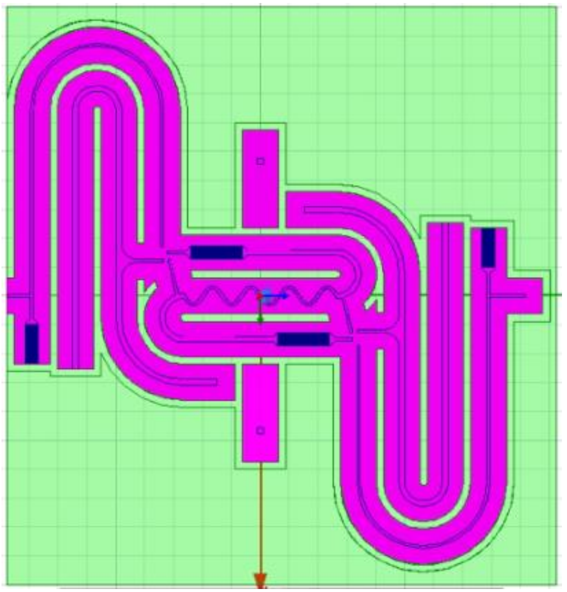
Processing Fillings and Wall Metals

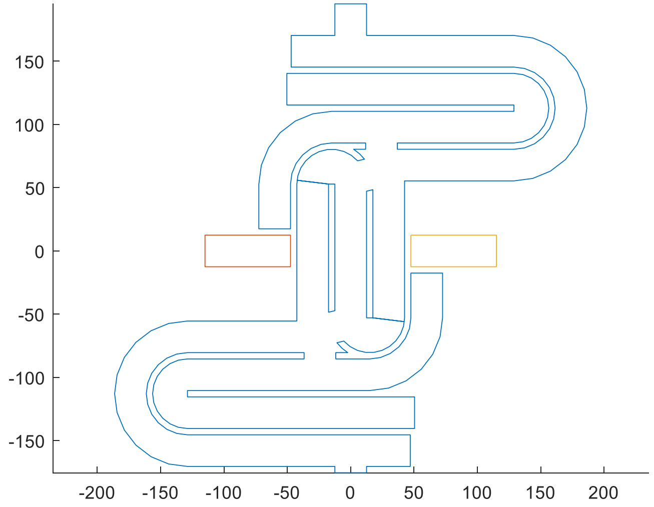


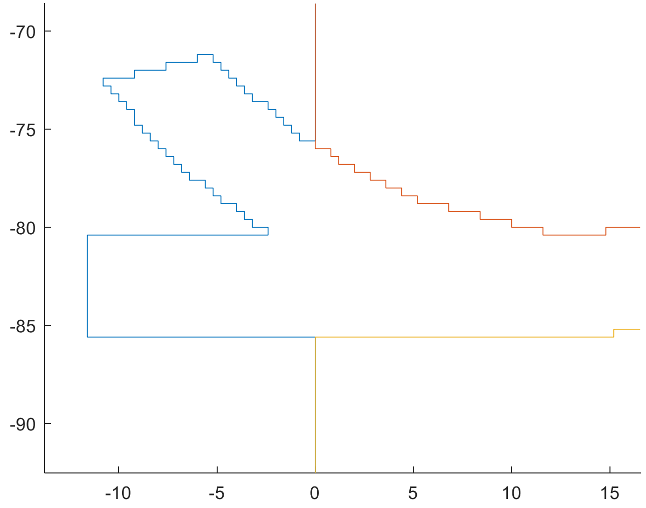


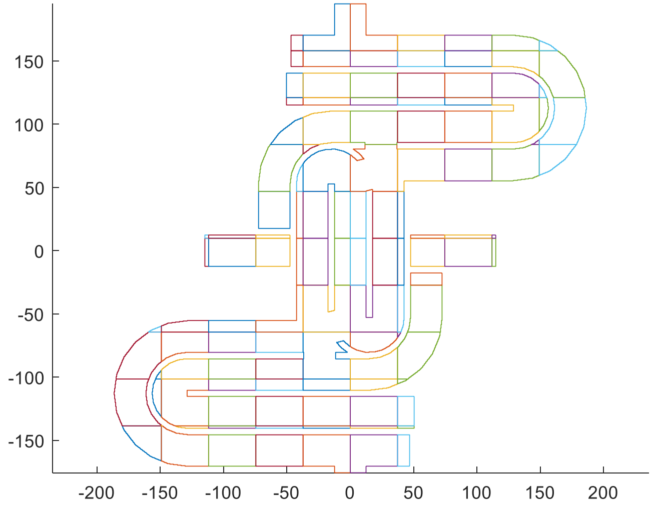
Processing tileNOT filling

1. Export the GDS layer from HFSS
2. Import in Matlab
3. Discretize
4. Split
5. Reset layer and dtype
6. Export

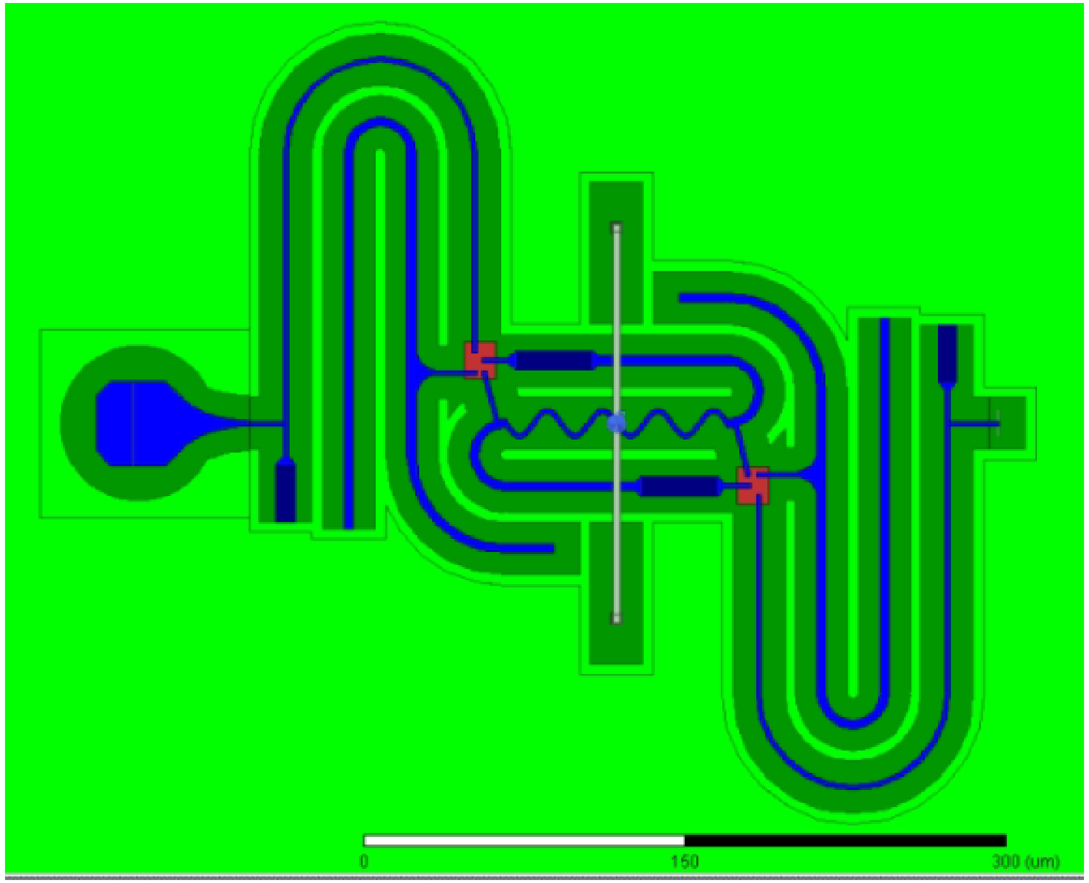


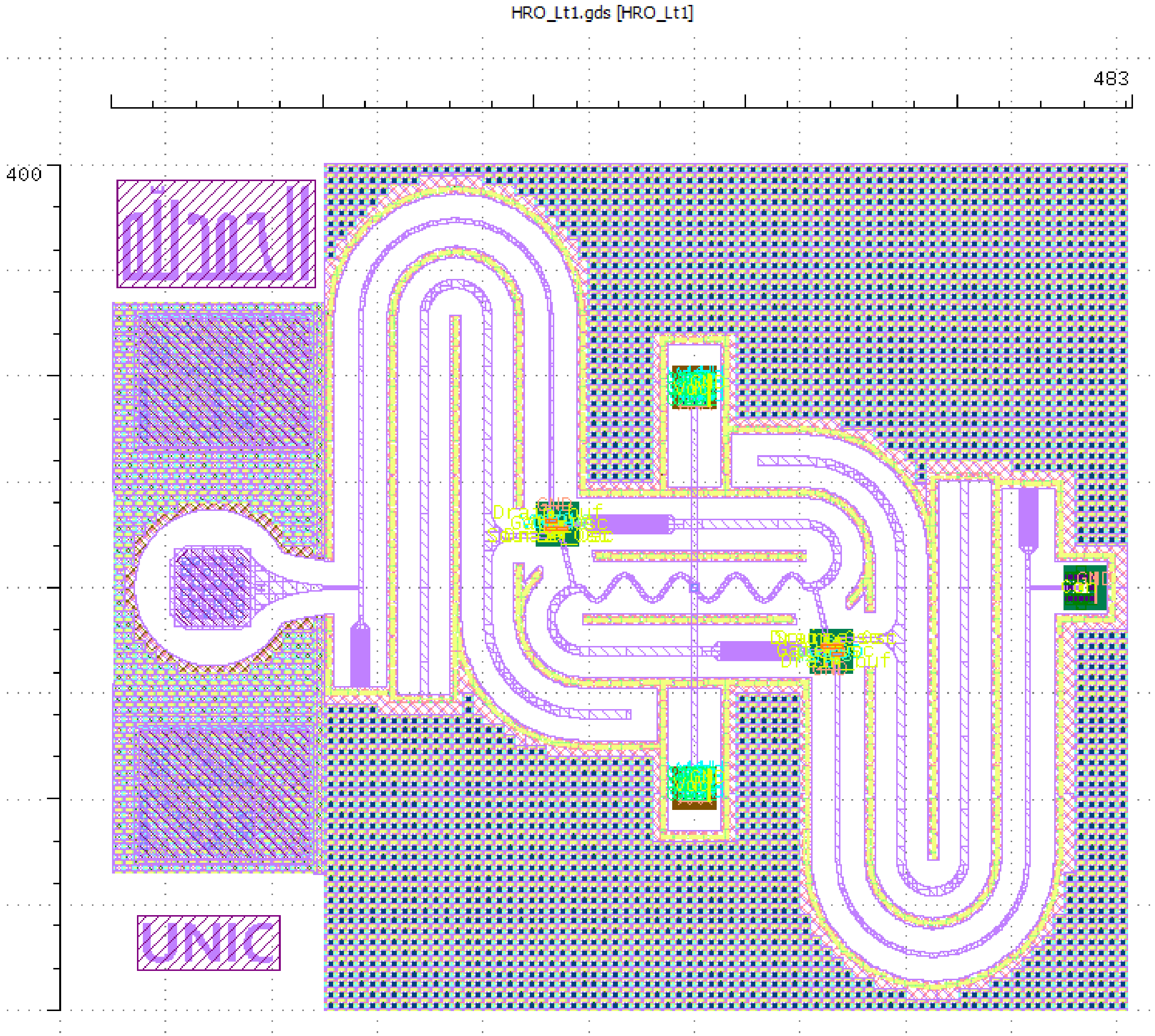






Conclusion





THE END !