Lab 3

Monday 14th February, 2022

1. **DRC**

   i. I count 24 rule violations. These are, specifically:

      1. Active.1: Minimum width of active
      2. Active.2: Minimum spacing of active areas
      3. Active.4: Active must be inside well
      4. Metal5.1: Minimum width of metal5
      5. Metal5.2: Minimum spacing of metal5
      6. Poly.2: Minimum spacing of gate poly
      7. Poly.3: Minimum extension of poly past active
      8. Poly.4: Minimum active enclosure of gate
      9. Poly.5: Minimum spacing of poly to active
      10. Poly.6: Minimum spacing of field poly
      11. Via6.1: Minimum width of via6
      12. Via6.2: Minimum spacing of via6
      13. Via6.3: Via6 must be inside metal6
      14. Via6.4: Via6 must be inside metal7
      15. Via7.1: Minimum width of via7
      16. Via7.2: Minimum spacing of via7
      17. Via7.3: Via7 must be inside metal7
      18. Via7.4: Via7 must be inside metal8
      19. Via9.1: Minimum width of via9
      20. Via9.2: Minimum spacing of via9
      21. Via9.3: Via9 must be inside metal9
      22. Via9.4: Via9 must be inside metal10
      23. Well.1: Nwell and Pwell must not overlap
      24. Well.4: Minimum width of well

      From Lab 2, the Poly layer was abysmal to print. It may be due to the minimum width violations that it was unable to print well (i.e. because it was so thin, it physically couldn't print). Active also has some violations, but what ended up printing was a lot thicker and "blob"ier than it should have been. This is most likely due to the violation of rule Active.2.

   ii. Metal 1 had no design violations, but from my Lab 2 submission, I noticed a lot of shorts pre-OPC. This could be soothed by adding a (or making stricter the already preexisting) minimum spacing between metal1 areas rule.

   iii. Rule Poly.1 had no violations, "Minimum width of poly". This could *probably* be shrunk a little bit, but looking at the poly design from Lab 2, I think it's good as it is. (But then again, I'm no chip designer, and maybe they could plausibly be brought just a smidge closer together.)

2. **Corner analysis**

   i. Running HSPICE for a PMOS shows:

```
**** mosfets


subckt
element  0:m_i_1
model    0:pmos_vtl
region     Saturati
 id      -593.1669u
 ibs     -1.683e-20
 ibd       1.1100p
 vgs      -1.1000
 vds      -1.1000
 vbs        0.
 vth     -203.8231m
 vdsat   -458.1642m
 vod     -896.1769m
 beta       4.6817m
 gam eff  400.0000m
 gm       713.1313u
 gds      128.5568u
 gmb      158.1467u
 cdtot    489.2191a
 cgtot    949.2317a
 cstot      1.0310f
 cbtot    808.4611a
 cgs      722.2381a
 cgd      224.4162a
```

Figure 1: PMOS data

And for an NMOS shows:

```
**** mosfets


subckt
element  0:m_i_1
model    0:nmos_vtl
region       Cutoff
 id        -2.6274u
 ibs      -35.4532f
 ibd       37.7770m
 vgs       -1.1000
 vds       -1.1000
 vbs        0.
 vth      109.8581m
 vdsat     41.5967m
 vod     -109.8581m
 beta      23.1047m
 gam eff  400.0000m
 gm        55.3641u
 gds        4.7119u
 gmb        5.3077u
 cdtot    651.3168a
 cgtot    540.5287a
 cstot    661.3101a
 cbtot    920.9305a
 cgs      163.9417a
 cgd      348.0212a
```

Figure 2: NMOS data

A simple CMOS inverter is shown in figure 3.

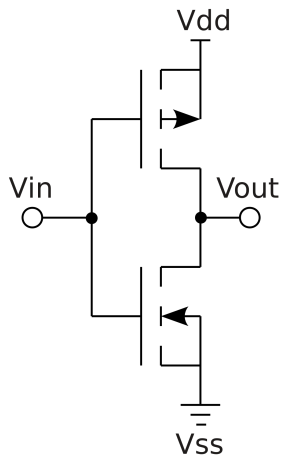

Figure 3: A CMOS inverter. (Source: Wikipedia)

Therefore, the I_{DSat} for an inverter is the added saturation current for both a PMOS and an NMOS.

Unfortuantely, I was unable to figure out how to get HSPICE working in time to submit this lab report.

3. **SADP compliant design**

   f. The task is to design masks to create this shape:

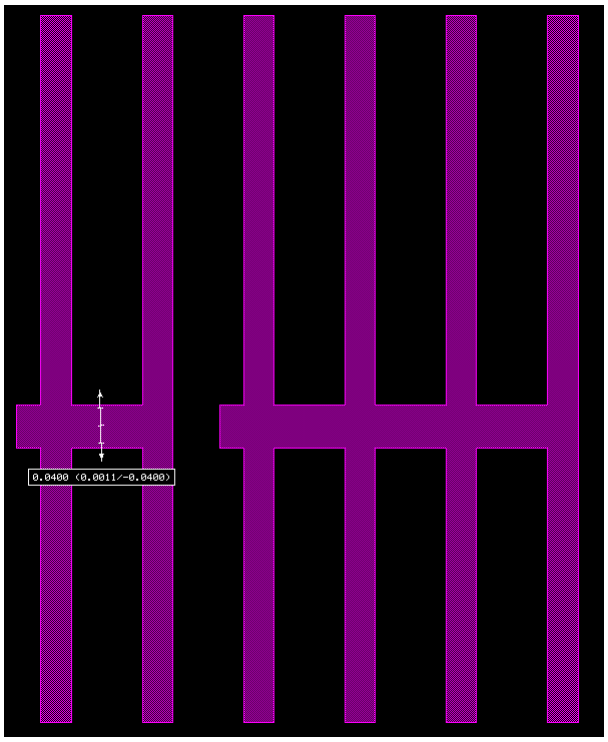

Figure 4: The layout for Layer 9, the gate.

My design is shown in figure 5.

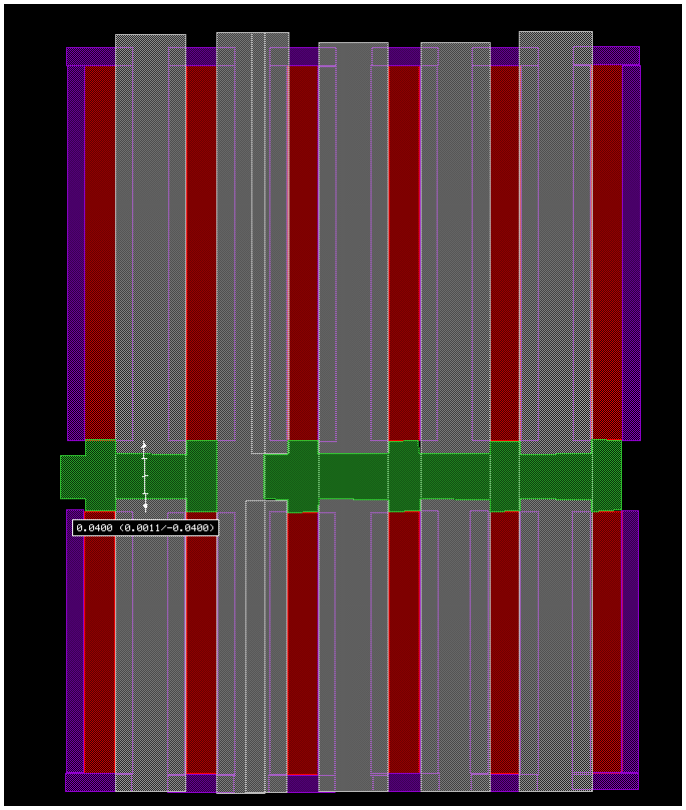

Figure 5: My masks.

The shapes in red are the mandrels. The green shapes are the nonmandrels. The purple shapes are the spacers (deposited around the mandrel* at 0.01 µm), and the grey shapes are the trim. When the mandrels and nonmandrels are removed, the original shape should be printed. I included the vertical arrow showing that the mandrels are indeed 0.04 µm apart.
*I did not draw the spacers below the top row of mandrels and above the bottom row of mandrels because these would simply be covered by the nonmandrel. The nonmandrel prevents stitching.

g. I modified the given script, shown in figure 6.

```
 GNU nano 2.3.1                              File: sample.s

LAYOUT SYSTEM GDSII
// INPUT THE GDS
LAYOUT PATH "BUF_X4_7nm.gds"
// TOP CELL IN THE DESIGN
LAYOUT PRIMARY "BUF_X4"

PRECISION 2000
RESOLUTION 1
LAYOUT ERROR ON INPUT NO
FLAG SKEW YES
FLAG ACUTE   YES
FLAG OFFGRID YES
DRC MAXIMUM RESULTS ALL
DRC RESULTS DATABASE "CA.gds" GDSII
DRC SUMMARY REPORT "CA_CALC.rep"
DRC MAXIMUM VERTEX 199
DRC KEEP EMPTY YES
// Define necessary layers from your SADP decomposition
LAYER poly   9
LAYER mandrel 237
LAYER nonmandrel 238
LAYER spacer 241
LAYER trim 240

// Create a layer for the SADP printed layer
// And compare it agains the original gate layer
// FINISH COMMAND BELOW.
//METAL1 = 15 OR 16
//METAL2 = 18 OR 22 OR 23
//METAL3 = 43
temp1 = spacer OR trim
temp2 = mandrel NOT temp1
caa1 = temp2 OR nonmandrel

// STORE THE LAYER INTO LAYER 109
caa1 {copy caa1} DRC CHECK MAP caa1 109

// COPY THE POLY LAYER FOR REFERENCE
poly {copy poly} DRC CHECK MAP poly 9
```

Figure 6: SVRF Script

Running the script leads to a layout file shown in figure 7. It's not perfect, but it's about close enough for the sake of demonstration. Success!

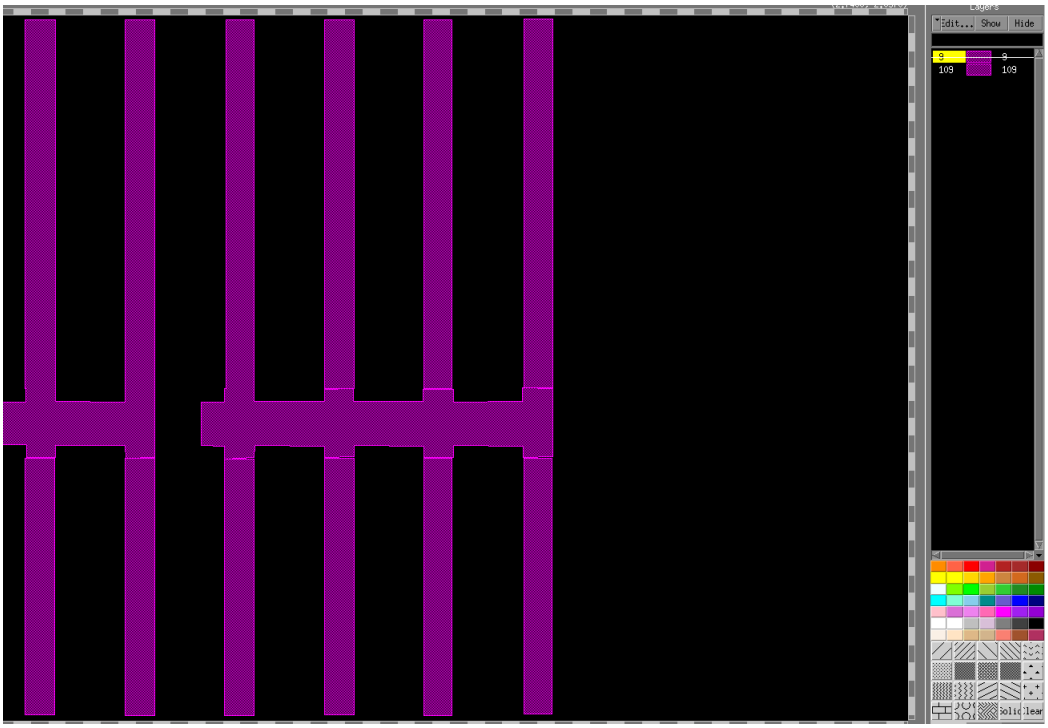

Figure 7: The generated layout from the SVRF script in Figure 6

.