Digital Microelectronics

Politecnico di Torino

Dipartimento di Elettronica e Telecomunicazioni

Final Project: Standard Cell layout of an inverter and a half-adder

Group 5:
Marco Andorno (247222)
Michele Caon (253027)
Matteo Perotti (251453)

Date: June 13, 2018

1 Introduction

The goal of this final project is to design and characterize two logic gates of a standard cell library. These are in particular an inverter whose transistors have width four times the minimum (`INV_X4` in the following), and a half adder (`HA_X1` in the following).

The design starts with the schematic drawing, using an available `.png` image as a reference, which has to be simulated as an initial condition. Then, the layout of the cell must be drawn and the parasitic elements must be extracted from it. Finally, those parasitics must be used for the final characterization in order to fill in the Liberty file of the standard cell library.

The files for these two gates are inside the following paths:

`/home/md18.5/project/INV_X4`

and

`/home/md18.5/project/HAX1`

2 Inverter

2.1 Schematic

Starting from the schematic of the available `.png` (figure 1), we copied it using Virtuoso Schematic Editor, obtaining the result shown in figure 2. Note that the four PMOS and NMOS transistors in parallel will be just one large transistor in the layout, with the source/drain diffusions fingered.

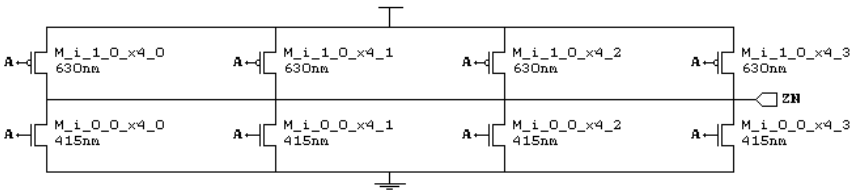

Figure 1: Reference schematic of the inverter

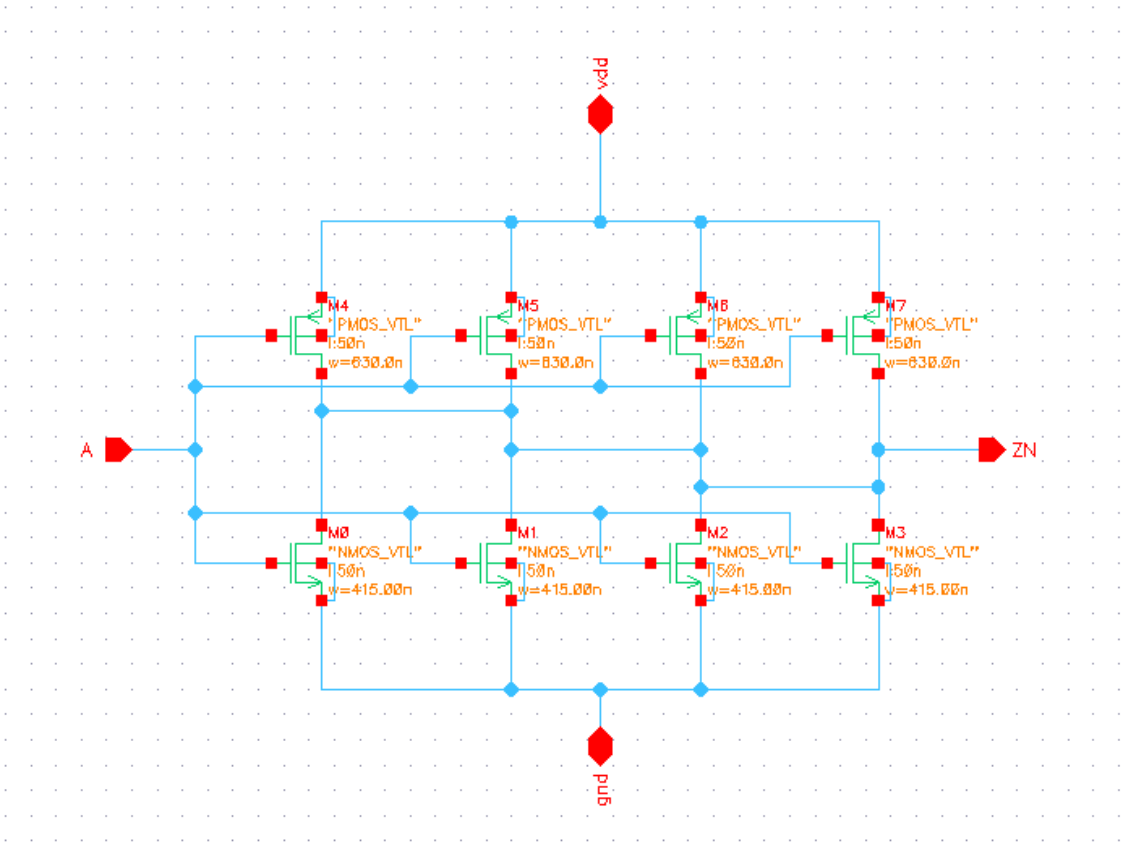

Figure 2: Final schematic of `INV_X4`

We then carried out some preliminary simulations using a test bench, to confirm that the circuit works correctly. We measured rise and fall times, as well as propagation delays, that will be compared with the actual characterization later on. We also measured the transfer characteristic, shown in figure 3.

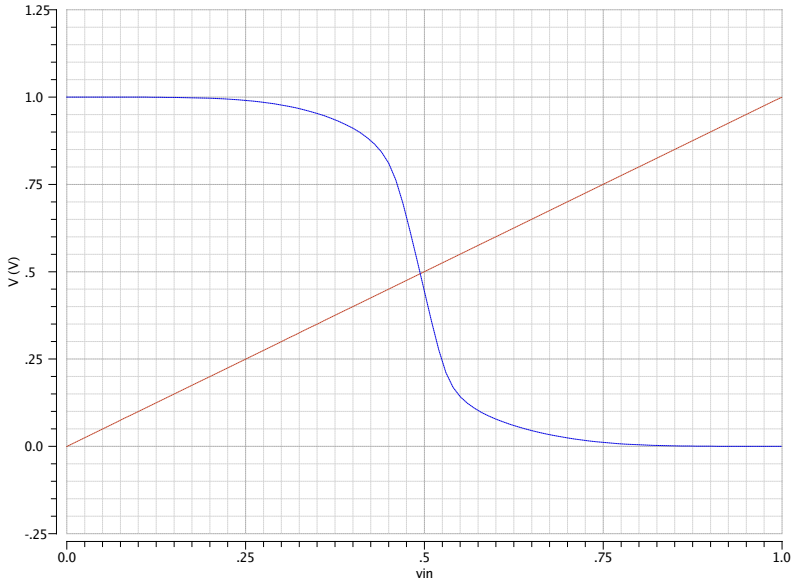

Figure 3: Inverter transfer characteristic

2.2 Layout

The final layout of `INV_X4` is shown in figure 4. The main trick to note here is how we used transistor fingering extensively, by sharing as much as possible source and drain diffusions. This is generally done in order to reduce the total area of the standard cell, but in this case it was actually mandatory, because a single transistor large 4 times the minimum width would not fit inside the well height.

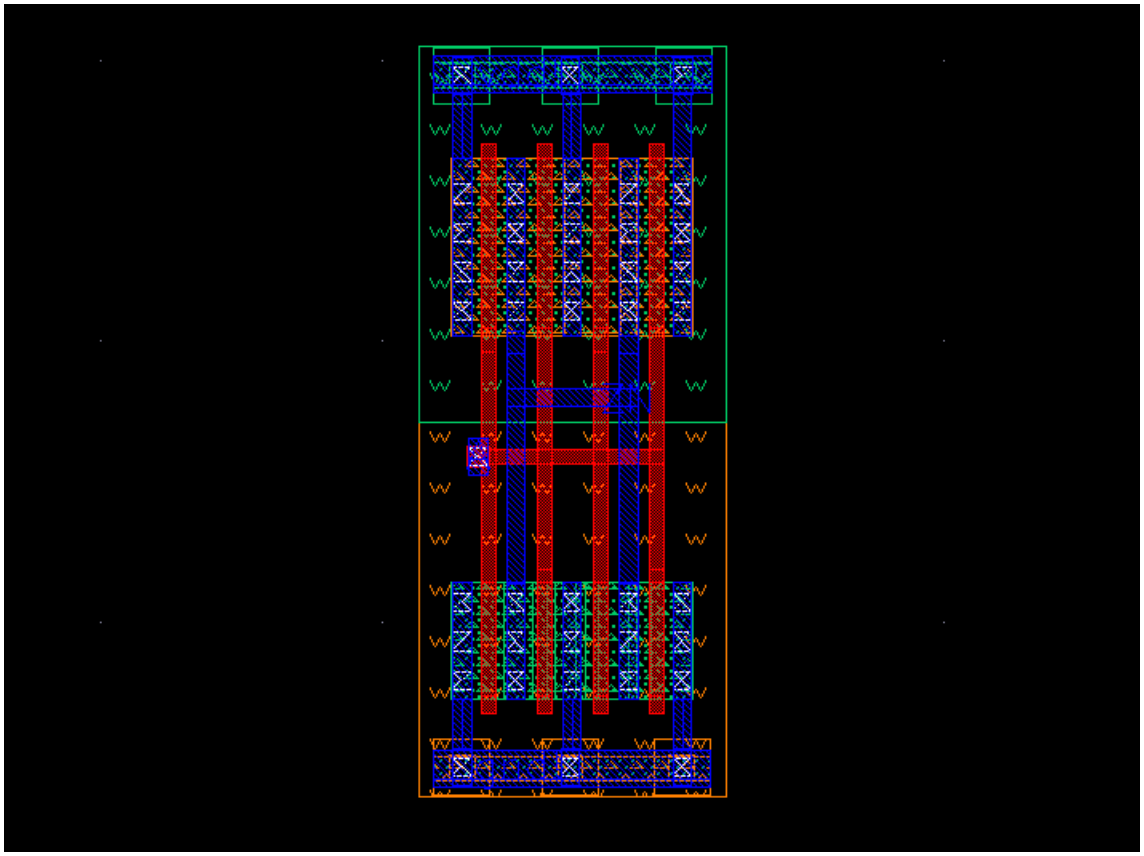

Figure 4: Layout of `INV_X4`

For PMOS transistors we were forced to finger four different transistor, while regarding the NMOS, probably, two fingered transistors with double width would have fit, but we decided to go with four single width transistors anyway to improve symmetry.

Regarding pins, the output `ZN` did not cause any troubles and was placed on the rightmost metal strip connecting the drains of the PMOS and NMOS. As for the input `A`, instead, our first idea had been to place its via in the middle of the horizontal polysilicon strip, but then we noted that this way the external connections to that pin would have needed to cross the metal 1 layer, thus requiring metal 2 to be used instead. So eventually we settled with placing the via for the input on the leftmost part of the polysilicon, that does not force any metal 1 crossing and also more accurately reflects the topology of the schematic.

After having completed the layout, and actually many times during its development, we ran the DRC which reported the following final statement:

```
================================================================================
=== CALIBRE::DRC-H SUMMARY REPORT
===
Execution Date/Time:       Wed May 23 17:11:38 2018
Calibre Version:           v2011.4_14.13    Tue Nov 15 15:18:31 PST 2011
Rule File Pathname:        /home/md18.5/project/INV_X4/_calibreDRC.rul_
Rule File Title:
Layout System:             GDS
Layout Path(s):            INV_X4.calibre.db
Layout Primary Cell:       INV_X4
Current Directory:         /home/md18.5/project/INV_X4
User Name:                 md18.5
Maximum Results/RuleCheck: 1000
Maximum Result Vertices:   4096
DRC Results Database:      INV_X4.drc.results (ASCII)
Layout Depth:              ALL
Text Depth:                PRIMARY
Summary Report File:       INV_X4.drc.summary (REPLACE)
Geometry Flagging:         ACUTE = NO  SKEW = NO  ANGLED = NO  OFFGRID = NO
                           NONSIMPLE POLYGON = NO  NONSIMPLE PATH = NO
Excluded Cells:
CheckText Mapping:         COMMENT TEXT + RULE FILE INFORMATION
Layers:                    MEMORY-BASED
Keep Empty Checks:         YES
--------------------------------------------------------------------------------
--- RUNTIME WARNINGS
---
--------------------------------------------------------------------------------
--- ORIGINAL LAYER STATISTICS
---
LAYER pwell ...... TOTAL Original Geometry Count = 4  (4)
.
.
.
LAYER via9 ....... TOTAL Original Geometry Count = 0  (0)
--------------------------------------------------------------------------------
--- RULECHECK RESULTS STATISTICS
---
RULECHECK Well.1 ............ TOTAL Result Count = 0 (0)
.
.
.
RULECHECK Antenna.metal10 ... TOTAL Result Count = 0 (0)
--------------------------------------------------------------------------------
--- RULECHECK RESULTS STATISTICS (BY CELL)
---
--------------------------------------------------------------------------------
--- SUMMARY
---
TOTAL CPU Time:                  0
TOTAL REAL Time:                 0
TOTAL Original Layer Geometries: 171 (171)
TOTAL DRC RuleChecks Executed:   167
TOTAL DRC Results Generated:     0 (0)
```

Moreover, we compared layout and schematic netlists by using LVS, that succeeded on the first try with its reassuring smiley face:

```
                ################################################
                ##                                            ##
                ##        C A L I B R E   S Y S T E M         ##
                ##                                            ##
                ##            L V S   R E P O R T             ##
                ##                                            ##
                ################################################
REPORT FILE NAME:         INV_X4.lvs.report
LAYOUT NAME:              /home/md18.5/project/INV_X4/INV_X4.sp ('INV_X4')
SOURCE NAME:              /home/md18.5/project/INV_X4/INV_X4.src.net ('INV_X4')
RULE FILE:                /home/md18.5/project/INV_X4/_calibreLVS.rul_
RULE FILE TITLE:          LVS Rule File for FreePDK45
CREATION TIME:            Tue May 29 15:46:49 2018
CURRENT DIRECTORY:        /home/md18.5/project/INV_X4
USER NAME:                md18.5
CALIBRE VERSION:          v2011.4_14.13    Tue Nov 15 15:18:31 PST 2011
                               OVERALL COMPARISON RESULTS
                          #       ###################       _   _
                         #        #                 #       *   *
                    #   #         #     CORRECT     #         |
                     # #          #                 #       \___/
                      #           ###################
*****************************************************************************************************
                                      CELL  SUMMARY
*****************************************************************************************************
  Result         Layout                      Source
  -----------    -----------                 --------------
  CORRECT        INV_X4                      INV_X4
*****************************************************************************************************
                                      LVS PARAMETERS
*****************************************************************************************************
o LVS Setup:
.
.
.
                    CELL COMPARISON RESULTS ( TOP LEVEL )
                          #       ###################       _   _
                         #        #                 #       *   *
                    #   #         #     CORRECT     #         |
                     # #          #                 #       \___/
                      #           ###################
LAYOUT CELL NAME:         INV_X4
SOURCE CELL NAME:         INV_X4
-----------------------------------------------------------------------------------------------------
INITIAL NUMBERS OF OBJECTS
--------------------------
                 Layout    Source         Component Type
                 ------    ------         --------------
 Ports:               4         4
 Nets:                4         4
 Instances:           4         4         MN (4 pins)
                      4         4         MP (4 pins)
                 ------    ------
 Total Inst:          8         8
NUMBERS OF OBJECTS AFTER TRANSFORMATION
---------------------------------------
                 Layout    Source         Component Type
                 ------    ------         --------------
 Ports:               4         4
 Nets:                4         4
 Instances:           1         1         _invv (4 pins)
```

```
                ------    ------
 Total Inst:         1         1
*****************************************************************************************************
                              INFORMATION AND WARNINGS
*****************************************************************************************************
                  Matched    Matched    Unmatched    Unmatched    Component
                   Layout     Source       Layout       Source    Type
                  -------    -------    ---------    ---------    ---------
   Ports:               4          4            0            0
   Nets:                4          4            0            0
   Instances:           1          1            0            0    _invv
                  -------    -------    ---------    ---------
   Total Inst:          1          1            0            0
o Statistics:
   8 layout mos transistors were reduced to 2.
     6 mos transistors were deleted by parallel reduction.
   8 source mos transistors were reduced to 2.
     6 mos transistors were deleted by parallel reduction.
o Layout Names That Are Missing In The Source:
   Ports:        GND! VDD!
   Nets:         GND! VDD!
o Initial Correspondence Points:
   Ports:        A ZN
*****************************************************************************************************
                                        SUMMARY
*****************************************************************************************************
Total CPU Time:      0 sec
Total Elapsed Time:  0 sec
```

Just a note on the warning that says that nets `GND!` and `VDD!` are present in the layout but missing in the source (schematic). We tried to follow the steps of the laboratory sessions where the supply nets had the exclamation mark at the end in the layout view but not in the schematic one, and in fact we had no problems with this method for the `INV_X4`. However, we had a lot of issues for `HA_X1` when running LVS as it complained about missing power nets. We finally came to the conclusion that schematic and layout must have the exact same names for supply nets and that the exclamation mark can be avoided if the schematic does not exploit global nets to make the drawing cleaner and clearer. Even then, to this day it remains unclear why the inverter cell worked just as well with different names for power nets (disregarding this warning), while the half adder with the same setup caused so much trouble. We can only accept this and give in to the mighty secrets of Virtuoso.

2.3 Characterization

3 Half adder

3.1 Schematic

The schematic was drawn using Virtuoso schematic editor.

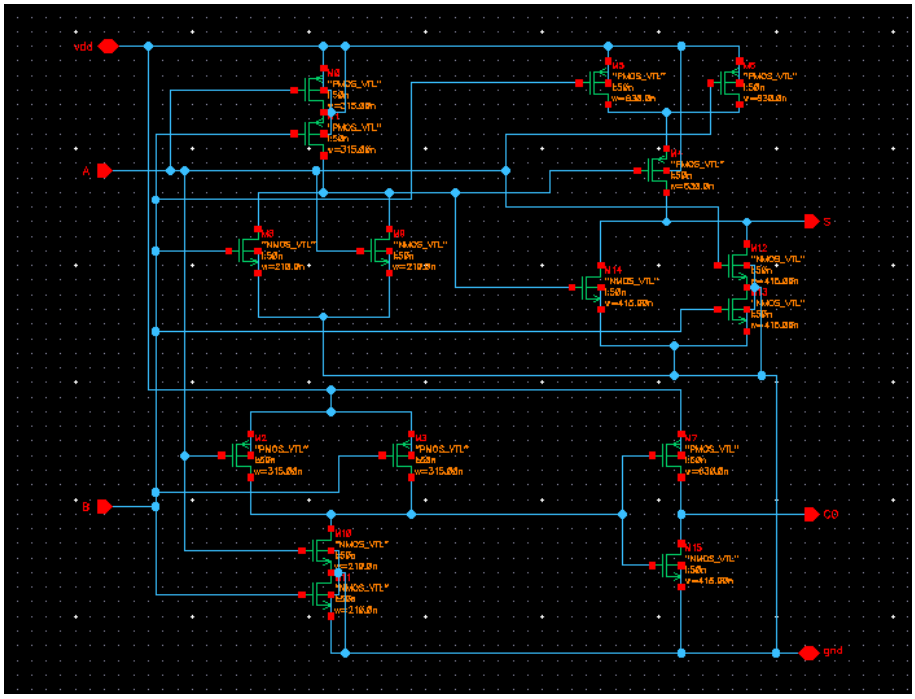

Figure 5: HAX1 schematic

3.2 Layout

Given that the layout of the half adder is quite more complex than the one of the inverter, we started by deeply analyzing the possible approaches using the pen-and-paper method. The final solution that we found consisted in sharing source/drain diffusions as much as possible, in order to minimize the area in the horizontal direction.

When actually drawing the design in Virtuoso, we made sure of running the Design Rule Checker very often, in order to avoid problems in advance.

Layout design started looking at the schematic. First of all we tried to put the two outputs on the opposite sides of the cell, because this way, if the input are brought over it in the middle with the `metal 2` and linked with vias, the input-output paths are more probably balanced in length. Thus we started drawing on a paper the first nMos of the inverter whose output was CO. With a view to share the sources/drains as much as possible we began putting all the transistors one near the other drawing only the strictly necessary metal. We used different colours to have an easier readability and go on drawing only the `metal` for the sources/drains and the `polysilicon` for the gates until the end of the port and left the routing last. Then we completed the layout linking together the various elements coherently with the schematic view.

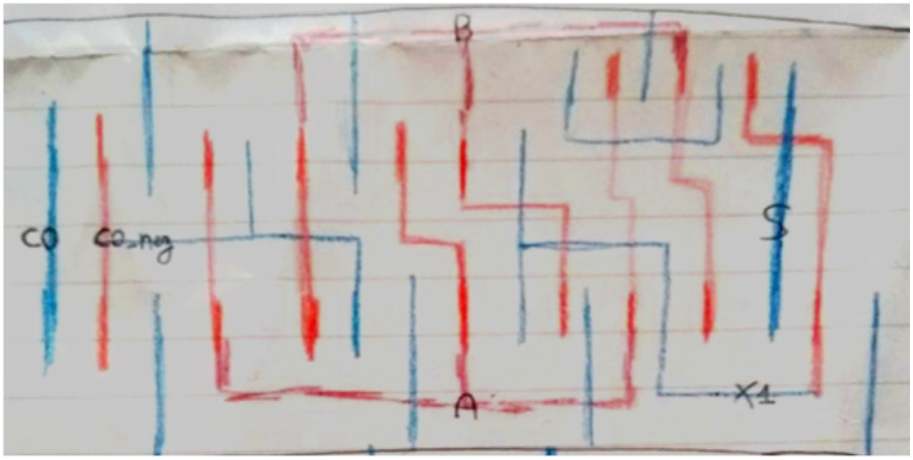

Figure 6: First sketch of the layout. Blue: metal1, red: polysilicon. Labels are not related with the position of the corresponding via.

Then we converted this sketch in a real layout design using the Virtuoso embedded layout editor. During our sessions we paid attention to reduce every distance to the minimum with the aid of the meter tool and in accordance with the Design Rules. We made a lot of effort to have a cell with the minimum area because the area is directly linked to the yield of the process and to logic ports density, capacitance and therefore delay and power cost per commutation. The less is the area, the more ports can stay in a chip or the more the chip will be small, and it is harder to find defects on a single chip if it is small, because the density of the defects per unit area is about constant (higher yield for the process). The delay and the power costs rise with the area because if more material is used, more resistance/capacitance is added. For this reason we also tried to reduce the length of every wire.

The height of the cell is fixed because this is a standard cell. The top and the ground of it are made of metal which has to feed the transistors. Many entity like this one can be put one next to the other to form a line of cells packed together and fed with global Val-GND voltages. This way it is simpler to reach every port with its supply.

When we drew the layout with the editor we changed a little bit the connections. Instead of putting a single wire of `polysilicon` to connect each gate with the related input A/B, we used vias and `metal1`. It would have been interesting to compare the performances of two implementations.

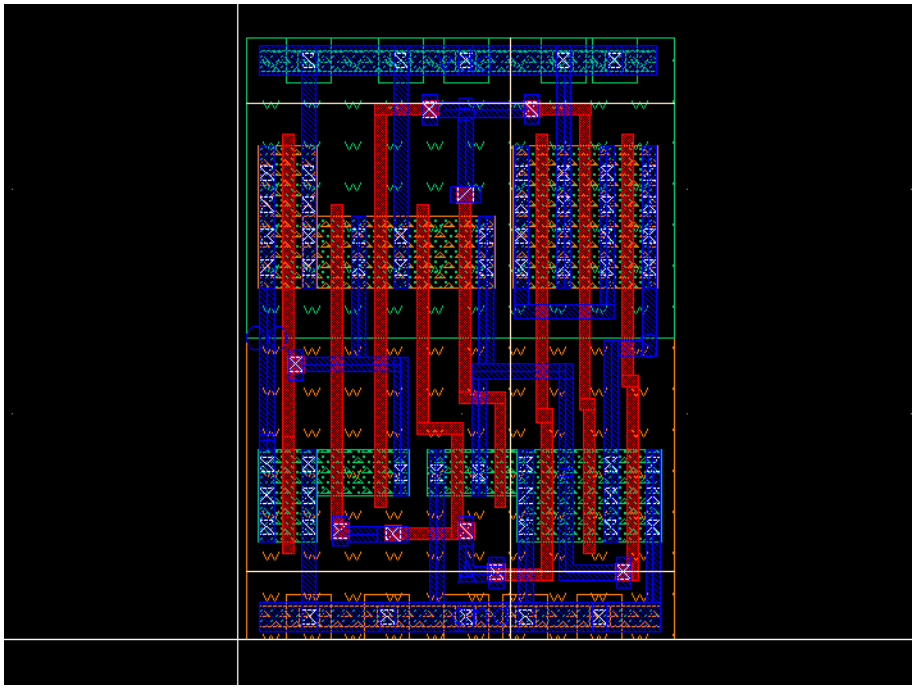

Figure 7: HAX1 layout, more metal1 and vias on pull-down connections

In the pull up network the advantage of putting `metal1` seems to be huge, whereas in the pull down network it's not so clear. We present here another possible implementation. For time reasons we have characterized only the first one, though.

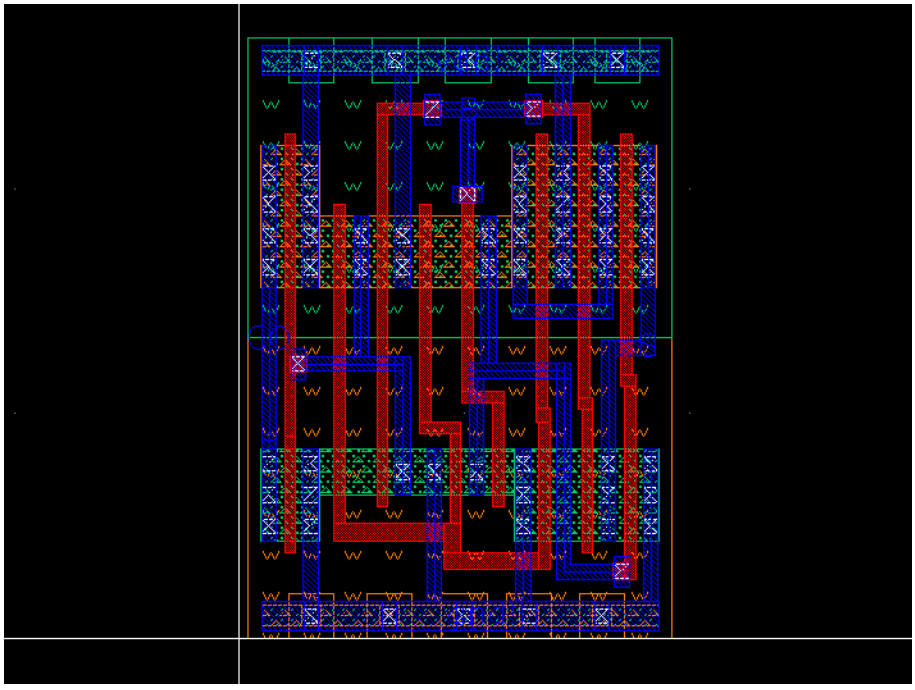

Figure 8: HAX1 layout, more polysilicon on pull-down connections

3.3 Characterization